Xièmes rencontres de la Société francophone de classification

- 2004 -

Bordeaux, France

Analyse de données probabilistes : Treillis de concepts et classification

Paula Brito * — Géraldine Polaillon ** — Francisco de A. T. de Carvalho ***

** Faculdade de Economia / LIACC , Universidade do Porto*
Rua Dr. Roberto Frias, 4200-464 Porto, PORTUGAL
mpbrito@fep.up.pt

*** SUPELEC - Service Informatique*
Plateau de Moulon, 91192 Gif-sur-Yvette Cedex, FRANCE
geraldine.polaillon@supelec.fr

**** Centro de Informatica - CIn/UFPE*
Av. Prof Luiz Freire, s/n, Cidade Universitaria, CEP 50.740-540, Recife - PE BRAZIL
fatc@cin.ufpe.br

RÉSUMÉ. *On s'intéresse aux données probabilistes, c'est-à-dire, quand chaque individu est décrit par des distributions de probabilités ou de fréquences sur les catégories de variables qualitatives. En définissant les opérateurs appropriés pour le calcul d'extension et d'intention, on obtient deux correspondances de Galois distinctes, qui permettent de définir deux treillis de concepts pour ce type de données. D'autre part, utilisant des mesures de généralité, adaptées aux données probabilistes, une méthode de classification ascendante hiérarchique /pyramidale est proposée, dont les classes formées sont des éléments du treillis de concepts correspondant.*

MOTS-CLÉS : *Analyse de données symboliques, Données probabilistes, Treillis de concepts, Classification conceptuelle, Hiérarchie, Pyramide*

1. Introduction

Le besoin de traiter des données ne pouvant être représentées dans une matrice n x p classique, combiné avec l'intérêt croissant pour des méthodes dont les résultats sont directement interprétés en termes des variables descriptives, ont conduit au développement de l'analyse de données symboliques. Les données symboliques étendent le modèle tabulaire classique, où chaque individu en ligne prend une et une seule valeur pour chaque variable en colonne, en permettant des valeurs multiples, éventuellement pondérées. De nouveaux types de variables ont été introduits - variables intervalles, catégoriques multi-valuées et modales - qui permettent de tenir compte de la variabilité et/ou l'incertitude inhérentes aux données. Une *variable modale* est une variable à valeurs multiples, où, pour chaque "individu", est donné un ensemble de modalités et, pour chaque modalité, une fréquence, probabilité ou un poids. Quand une distribution empirique est donnée, la variable est appelée *variable histogramme* ([BOC 00]). Des données décrites par des variables modales sont désignées par *donn ées modales*, si une distribution de probabilités ou de fréquences est donnée pour chaque variable, on les désigne par des *donn ées probabilistes*. On trouve souvent ce type de données dans des applications pratiques, par exemple, quand on désire exprimer une incertitude, ou quand on agrège les réponses d'une enquête.

Les treillis de Galois permettent d'organiser les observations dans des classes automatiquement interprétées, dont la structure ne dépend, ni de paramètres externes, ni de l'ordre d'observation ni de l'implémentation algo-

rithmique. Barbut and Monjardet ont été les premiers à s'intéresser aux correspondances de Galois, introduites par Birkhoff en 1940 ([BIR 40]), pour l'étude d'un tableau de données binaires ([BAR 70]). Depuis, beaucoup de développements théoriques et pratiques ont été accomplis, citons d'une part par l'école de Darmstadt en "Analyse Conceptuelle Formelle" ("Formal Concept Analysis") ([WIL 82, GAN 99]), et d'autre part Duquenne ([DUQ 86, DUQ 87]) et aussi ([GUé 93, MEP 93, GOD 95, GIR 99]), qui ont utilisé la théorie des treillis pour l'organisation et l'analyse des données.

L'extension des correspondances de Galois et des treillis de Galois aux données symboliques a été d'abord traité par Brito ([BRI 91, BRI 94a]) et développé ensuite par Polaillon ([POL 98a, POL 98b, POL 99]). Dans un papier récent, Brito et Polaillon ([BRI 04]) ont défini les outils permettant d'obtenir directement des treillis de Galois sur des données probabilistes, sans qu'une transformation préalable soit nécessaire, et proposé un algorithme qui construit le treillis de concepts.

Le treillis de concepts est en général assez complexe et contient un nombre important de classes, ce qui rend son interprétation difficile. Une alternative consiste à limiter le nombre de classes formées, en imposant une structure plus simple. Une méthode de classification hiérarchique ou pyramidale a été proposée ([BRI 91, BRI 94b]) où chaque classe formée est un concept, pour les opérateurs considérés : chaque classe est représentée par un objet symbolique dont l'extension doit coïncider avec la classe elle-même. Un critère numérique additionnel est défini, une mesure de "généralité", qui permet, à chaque étape, de choisir la "meilleure" agrégation parmi les agrégations possibles. Le cas des données décrites par des variables modales a été d'abord traité dans ([BRI 98]) ; Brito et De Carvalho ont étendu la méthode pour permettre de prendre en compte l'existence de règles hiérarchiques entre variables catégoriques multi-valuées ([BRI 99]) et entre variables modales ([BRI 02]), en définissant de façon adéquate les opérateurs de généralisation et les mesures de généralité. Les classes formées correspondent à des "concepts", décrits en extension, par la liste de ses membres, et en intention, par un objet symbolique qui exprime ses propriétés. La méthode rentre dans le cadre de la classification conceptuelle, puisque chaque classe formée est associée à une conjonction de propriétés portant sur les variables descriptives, qui constitue une condition nécessaire et suffisante d'appartenance à la classe.

Rappelons que les pyramides ([DID 84, DID 86, BER 85, BER 86]) étendent le modèle hiérarchique en permettant des classes recouvrantes qui ne sont pas emboîtées, mais le modèle pyramidal impose l'existence d'un ordre total sur l'ensemble des individus à classifier, tel que chaque classe formée soit un intervalle de cet ordre. La classification pyramidale produit une classification plus riche qu'une hiérarchie, en ce sens qu'elle permet la formation d'un plus grand nombre de classes, et elle fournit une sériation de l'ensemble donné.

Beaucoup d'autres méthodes de classification hiérarchique ont maintenant été proposées pour classifier des données symboliques, qui diffèrent selon le type des données qu'elles permettent de traiter et/ou le critère de formation des classes, citons [GOW 91, GOW 92, GOW 95b, GOW 95a, CHA 98, ELS 98].

Dans la Section 2, on commence par rappeler la définition de variable modale, et formaliser les notions d'événement modal et objet modal. On détaille ensuite, dans la Section 3, les résultats permettant de définir des treillis de Galois sur des données décrites par des variables modales. Dans la Section 4, on présente la méthode de classification hiérarchique / pyramidale pour ce type de données. Enfin, on illustre les méthodes présentées par un exemple.

2. Données Probabilistes

DÉFINITION 1 ([BOC 00])
Une *variable modale* Y définie sur un ensemble $E = \{\omega_1, \omega_2, \ldots\}$ de domaine $O = \{m_1, \ldots, m_k\}$ est une application $Y(\omega) = (U(\omega), \pi_\omega)$, pour $\omega \in E$, où π_ω est une distribution (fréquence ou probabilité) sur le domaine O des valeurs possibles (complétée par une σ-algèbre convenable), et $U(\omega) \subseteq O$ est le support de π_ω dans O.

En général, le support $U(\omega)$ peut être omis de la définition, et une variable modale considérée comme une application $Y : E \to M(O)$, de E dans la famille $M(O)$ des mesures non-négatives π sur O, à valeurs $Y(\omega) = \pi_\omega = \{m_1(p_1(\omega)), \ldots, m_k(p_k(\omega))\}$.

DÉFINITION 2

Un *événement modal* est une expression de la forme $e = [Y(\omega)R\{m_1(p_1), m_2(p_2), \ldots, m_k(p_k)\}]$ où $O = \{m_1, m_2, \ldots, m_k\}$ est le domaine de Y, et p_ℓ est la probabilité, fréquence ou poids de $m_\ell, \ell = 1, \ldots, k$. Il n'est pas imposé que $p_1 + p_2 + \ldots + p_k = 1$. R est une relation définie sur l'ensemble des distributions sur O. On considère les relations suivantes :

– $'\sim'$ tel que $[Y(\omega) \sim \{m_1(p_1), \ldots, m_k(p_k)\}]$ est vrai ssi $p_\ell(\omega) = p_\ell, \ell = 1, \ldots, k$
– $'\leq'$ tel que $[Y(\omega) \leq \{m_1(p_1), \ldots, m_k(p_k)\}]$ est vrai ssi $p_\ell(\omega) \leq p_\ell, \ell = 1, \ldots, k$
– $'\geq'$ tel que $[Y(\omega) \geq \{m_1(p_1), \ldots, m_k(p_k)\}]$ est vrai ssi $p_\ell(\omega) \geq p_\ell, \ell = 1, \ldots, k$

Un *objet modal* est formé d'une conjonction d'événements modaux.

Chaque individu $\omega \in E$ est associé à un objet modal :

$$s(\omega) = \bigwedge_{j=1}^{p} \left[Y_j(w) \sim \{m_1^j(p_1^j(\omega)), \ldots, m_{k_j}^j(p_{k_j}^j(\omega))\}\right]$$

Dans la description de chaque individu $w \in E$, on a toujours $p_1^j(\omega) + \ldots + p_{k_j}^j(\omega) = 1, j = 1, \ldots, p$. En fait, quand $p_1 + p_2 + \ldots + p_k = 1$, on a une distribution de probabilités ou de fréquences, alors que quand $p_1 + p_2 + \ldots + p_k \geq 1$ (resp. $p_1 + p_2 + \ldots + p_k \leq 1$) on a une enveloppe supérieure (resp. inférieure) d'une distribution de probabilités ou de fréquences.

DÉFINITION 3

On définit un ordre partiel sur l'ensemble des objets modaux définis sur le même ensemble de variables $\{Y_1, \ldots, Y_p\}$, comme suit :

Si $s_1 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(p_1^j), \ldots, m_{k_j}^j(p_{k_j}^j)\}\right]$ et $s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(q_1^j), \ldots, m_{k_j}^j(q_{k_j}^j)\}\right]$ alors $s_1 \leq s_2$ ssi $p_\ell^j \leq q_\ell^j, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

Si $s_1 \leq s_2$ on dira que s_1 est *plus spécifique* que s_2 et que s_2 est *plus général* que s_1.

3. Treillis de concepts sur des données probabilistes

Rappelons qu'une correspondance de Galois est une paire d'applications (f, g) entre deux ensembles ordonnés $(A, \leq_A)$ et $(B, \leq_B)$ qui sont antitones et dont les applications composées $h = g \circ f$ et $h' = f \circ g$ sont extensives.

Soit S l'ensemble des objets modaux, avec $0 \leq p_\ell^j \leq 1, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

THÉORÈME 1 :

Le couple d'applications

$$\begin{aligned} f_u : S &\rightarrow P(E) \\ s &\rightarrow ext_E s = \{\omega \in E : s(\omega) \leq s\} \end{aligned}$$

$$\begin{aligned} g_u : P(E) &\rightarrow S \\ C = \{w_1, \ldots, w_H\} &\rightarrow int(C) = s = \bigwedge_{j=1}^{p} \left[Y_j \leq \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right] \end{aligned}$$

avec $t_\ell^j = \text{Max}\ \{p_\ell^j(w_h), h = 1, \ldots, H\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ forment une correspondance de Galois entre $(P(E), \subseteq)$ et $(S, \geq)$.

Exemple :

Considérons le tableau de données présenté dans la section 5.

Alors, $g_u(\{\text{Ann, Bob}\}) = s_u =$
$[\text{ Tinnitus } \leq \{\text{ fréquent } (1),\text{ rare } (0.2)\}] \wedge [\text{ Maux de Tête } \leq \{\text{ fréquents } (0.9),\text{ rares } (1)\}] \wedge$
$[\text{ Pression Sanguine } \leq \{\text{ haute } (0.8),\text{ normale } (0.4),\text{ basse } (0.0)\}]$
et $f_u(s_u) = \{\text{Ann, Bob}\}$

THÉORÈME 2 :
Le couple d'applications

$$\begin{aligned} f_i : S &\rightarrow P(E) \\ s &\rightarrow ext_E s = \{\omega \in E : s(\omega) \geq s\} \end{aligned}$$

$$\begin{aligned} g_i : P(E) &\rightarrow S \\ C = \{w_1, \ldots, w_H\} &\rightarrow int(C) = s = \bigwedge_{j=1}^{p} \left[Y_j \geq \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right] \end{aligned}$$

avec $t_\ell^j =$ Min $\{p_\ell^j(w_h), h = 1, \ldots, H\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ forment une correspondance de Galois entre $(P(E), \subseteq)$ et $(S, \leq)$.

Exemple :
Considérons à nouveau le tableau de données présenté dans la section 5.
Alors, $g_i(\{\text{Ann, Bob}\}) = s_i =$
$[\text{ Tinnitus } \geq \{\text{ fréquent } (0.8),\text{ rare } (0.0)\}] \wedge [\text{ Maux de Tête } \geq \{\text{ fréquents } (0.0),\text{ rares } (0.1)\}] \wedge$
$[\text{ Pression Sanguine } \geq \{\text{ haute } (0.6),\text{ normale } (0.2),\text{ basse } (0.0)\}]$,
et $f_i(s_i) = \{\text{Ann, Bob}\}$

Pour les démonstrations des théorèmes 1 et 2, voir [BRI 04].

DÉFINITION 4
Un objet probabiliste s est *complet* si $h(s) = g(f(s) = s$.

DÉFINITION 5
Un *concept* est une paire (A, s), où $A \subseteq E$, $s \in S$, s est complet et $A = f(s)$.

Exemple :
D'après les exemples précédents, on peut conclure que s_u est un objet modal complet et que $(\{\text{Ann, Bob}\}, s_u)$ est un concept, pour la correspondance de Galois du théorème 1. De même, s_i est un objet modal complet et $(\{\text{Ann, Bob}\}, s_i)$ est un concept pour la correspondance de Galois du théorème 2.

Les théorèmes 1 et 2 établissent des correspondances de Galois entre deux treillis ; en conséquence, on obtient les théorèmes 3 et 4 ci-dessous ([BIR 40], [BAR 70], [BRI 04]).

THÉORÈME 3
Soit (f_u, g_u) la correspondance de Galois du théorème 1.
Si $s_1 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(r_1^j), \ldots, m_{k_j}^j(r_{k_j}^j)\}\right]$ et $s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(q_1^j), \ldots, m_{k_j}^j(q_{k_j}^j)\}\right]$ on définit
$s_1 \cup s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right]$, avec $t_\ell^j =$ Max $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ et
$s_1 \cap s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(z_1^j), \ldots, m_{k_j}^j(z_{k_j}^j)\}\right]$, avec $z_\ell^j =$ Min $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

Alors, l'ensemble des concepts, ordonnés par $(A_1, s_1) \leq (A_2, s_2) \Leftrightarrow A_1 \subseteq A_2$ forme un treillis, où le supremum et l'infimum de chaque paire d'éléments sont donnés par :

$\inf\left((A_1, s_1), (A_2, s_2)\right) = (A_1 \cap A_2, (g_u \circ f_u)(s_1 \cap s_2))$
$\sup\left((A_1, s_1), (A_2, s_2)\right) = ((f_u \circ g_u)(A_1 \cup A_2), s_1 \cup s_2)$

Ce treillis sera appelé “treillis de l’union”.

THÉORÈME 4 :
Soit (f_i, g_i) la correspondance de Galois du théorème 2.
Si $s_1 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(r_1^j), \ldots, m_{k_j}^j(r_{k_j}^j)\}\right]$ et $s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(q_1^j), \ldots, m_{k_j}^j(q_{k_j}^j)\}\right]$ on définit

$s_1 \cup s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right]$, avec $t_\ell^j =$ Min $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ et

$s_1 \cap s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(z_1^j), \ldots, m_{k_j}^j(z_{k_j}^j)\}\right]$, avec $z_\ell^j =$ Max $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

Alors, l’ensemble des concepts, ordonnés par $(A_1, s_1) \leq (A_2, s_2) \Leftrightarrow A_1 \subseteq A_2$ forme un treillis, où le supremum et l’infimum de chaque paire d’éléments sont donnés par :

$\inf\left((A_1, s_1), (A_2, s_2)\right) = (A_1 \cap A_2, (g_i \circ f_i)(s_1 \cap s_2))$
$\sup\left((A_1, s_1), (A_2, s_2)\right) = ((f_i \circ g_i)(A_1 \cup A_2), s_1 \cup s_2)$

Ce treillis sera appelé “treillis de l’intersection”.

Un algorithme permettant de cosntruire ces deux treillis est ptrsenté dans [BRI 04] ; il utilise l’algorithme de ([GAN 99]) pour déterminer la liste des concepts.

4. Classification hiérarchique / pyramidale

Les résultats présentés dans la Section 3 permettent d’obtenir tous les concepts associés à un tableau de données (étant donnés les fonctions f et g) et d’en construire le treillis de concepts. Mais ce treillis est en général trop complexe et contient trop de classes (noeuds) pour être aisément interprété. Une alternative consiste à limiter le nombre de classes formées, en leur imposant une structure plus simple. Dans cette section, on propose d’utiliser les modèles de classification hiérarchique ou pyramidale, en définissant un algorithme qui permette de construire une hiérarchie ou une pyramide dont les classes soient des éléments du treillis de concepts.

L’objectif général d’une méthode de classification est de grouper les éléments d’un ensemble E en classes homogènes. Dans le cas de la classification hiérarchique ou pyramidale, les classes formées sont organisées dans une structure arborescente. Suivant une approche ascendante, les éléments qui se ressemblent le plus sont d’abord réunis, après les classes similaires sont réunies, jusqu’à ce qu’une seule classe, réunissant tous les éléments de E soit formée. Dans le cas d’une hiérarchie, chaque niveau correspond à une partition ; dans le cas d’une pyramide on a, à chaque niveau, une famille de classes recouvrantes, mais toutes les classes sont des intervalles d’un ordre linéaire sur E ([DID 84, DID 86, BER 85, BER 86]).

La méthode de classification symbolique hiérarchique /pyramidale proposée dans ([BRI 91, BRI 94b]) construit une hiérarchie ou une pyramide en imposant que chaque classe soit un concept pour les opérateurs considérés. Le cas des données décrites par des variables modales a été traité dans ([BRI 98]) et puis dans ([BRI 02]).

Le critère qui guide la formation des classes est la dualité intention-extension : chaque classe de la hiérarchie ou de la pyramide doit correspondre à un concept, c’est-à-dire, chaque classe, qui est un sous-ensemble de E, est représentée par un objet modal complet dont l’extension est la classe elle-même. Ce qui veut dire que chaque classe est par construction associée à un objet qui généralise ses membres, et qu’aucun élément extérieur à la classe n’appartient à son extension.

Les classes, et les concepts correspondants , sont formées de façon récursive. L’ensemble initial des concepts est $\{(\omega_1, s_1), \ldots, (\omega_n, s_n)\}, s_i = s(\omega_i), i = 1, \ldots, n$; on suppose que tous les $(w_i, s_i),\ i = 1, \ldots, n$ sont des

concepts. À chaque étape, un nouveau concept (C, s) est formé, par l'union de concepts existants (C_α, s_α) et (C_β, s_β), avec $C = C_\alpha \cup C_\beta$, $s = s_\alpha \cup s_\beta = g(C) = int(C)$ et en imposant que $f(s) = ext_E s = C$. Selon que les fonctions f et g sont choisies comme dans le théorème 1 ou comme dans le théorème 2, on obtient une hiérarchie ou pyramide qui est un sous ensemble du treillis de l'union ou du treillis de l'intersection, respectivement.

4.1. *Degré de généralité*

Un critère additionnel doit être considéré, qui permette de choisir entre les agrégations possibles à une étape donnée. Le principe sera que les classes associées à des objets plus spécifiques doivent être d'abord formées. Comme la relation d'ordre (voir définition 3) est seulement un ordre partiel, un critère numérique a été défini, le "degré de généralité", qui permet d'évaluer la généralité d'un objet. Ainsi, à chaque étape, parmi les classes qui peuvent être formées, on choisira de former celle dont l'objet modal associé présente un moindre degré de généralité.

Pour des variables modales Y_j avec k_j modalités, $m_1^j, \ldots, m_{k_j}^j$, sur lesquelles on a une distribution de probabilités ou de fréquences $\{m_1^j(p_1^j), \ldots, m_{k_j}^j(p_{k_j}^j)\}$, et $s = \bigwedge_{j=1}^{p} e_j = \bigwedge_{j=1}^{p} [Y_j R_j \{m_1^j(p_1^j), \ldots, m_{k_j}^j(p_{k_j}^j)\}]$ avec $R_j \in \{\sim, \leq, \geq\}, j = 1, \ldots, p$, deux mesures ont été proposées, selon l'opérateur de généralisation utilisé :

– Si $R_j \in \{\sim, \leq\}, j = 1, \ldots, p$, et que la généralisation est effectuée comme indiqué dans le théorème 1 (c'est-à-dire, en prenant le maximum des probabilités ou fréquences associées aux différentes modalités), alors on considère

$$G_1(s) = \prod_{j=1}^{p} G_1(e_j) = \prod_{j=1}^{p} \frac{\sum_{\ell=1}^{k_j} \sqrt{p_\ell^j}}{\sqrt{k_j}} \qquad [1]$$

– Si $R_k \in \{\sim, \geq\}, j = 1, \ldots, p$, et que la généralisation est effectuée comme indiqué dans le théorème 2 (c'est-à-dire, en prenant le minimum des probabilités ou fréquences associées aux différentes modalités) alors on considère

$$G_2(s) = \prod_{j=1}^{p} G_2(e_j) = \prod_{j=1}^{p} \frac{\sum_{\ell=1}^{k_j} \sqrt{(1 - p_\ell^j)}}{\sqrt{k_j(k_j - 1)}} \qquad [2]$$

Ces mesures évaluent, dans chaque cas, la similarité entre la distribution donnée et la distribution uniforme. En fait, $G_1(e_j)$ est le coefficient d'affinité ([MAT 51]) entre $(p_1^j, \ldots, p_{k_j}^j)$ et la distribution uniforme $(\frac{1}{k_j}, \ldots, \frac{1}{k_j})$, $G_2(e_j)$ est le coefficient d'affinité entre $(1 - p_1^j, \ldots, 1 - p_{k_j}^j)$ et$(\frac{1}{k_j}\frac{1}{k_j - 1}, \ldots, \frac{1}{k_j}\frac{1}{k_j - 1})$, respectivement. Cela signifie que l'on considère un objet modal d'autant plus général que les distributions associées sont proches de la distribution uniforme. Pour les distributions où $p_1^j + \ldots + p_{k_j}^j = 1$, $G_1(e_j)$ et $G_2(e_j)$ sont maximaux, $G_1(e_j) = G_2(e_j) = 1$, quand la distribution associée à e_j est la distribution uniforme : $p_\ell^j = \frac{1}{k_j}, \ell = 1, \ldots, k_j$.

4.2. *Algorithme*

L'algorithme suivant construit une hiérarchie indicée (H, I) ou une pyramide indicée au sens large (P, I), où I est la fonction d'indexation $I : H \to \mathbb{R}_0^+$ (respec. $I : P \to \mathbb{R}_0^+$), telle que chaque classe formée correspond à un concept. Soit P_t l'ensemble des classes formées après l'étape t, Q_t le correspondant ensemble de concepts et $S_t \subseteq P_t \times P_t$ l'ensemble de paires d'éléments de P_t qui peuvent être agrégées à l'étape t+1, selon le modèle choisi. Pour simplifier, on supposera que $S_t \neq \emptyset$ à chaque étape.

– Initialisation :
$P_0 = E$, $Q_0 = \{(\omega_1, s_1), \ldots, (\omega_n, s_n)\}$, $S_0 = P_0 \times P_0$, $C_i = \{\omega_i\}$, $I(C_i) = 0, i = 1, \ldots, n$.
– Agrégation/Généralisation :
Après l'étape t : $P_t = \{C_h, h = 1, \ldots, m\}$, $Q_t = \{(C_h, s_h), h = 1, \ldots, m\}$, $S_t = \{(C_h, C_{h'}) \subseteq P_t \times P_t :$
C_h peut être agrégé avec $C_{h'}\}$
Tant que $E \notin P_t$:
1 Soit $(\alpha, \beta) : G(s_\alpha \cup s_\beta) = Min\{G(s_h \cup s_{h'})$ pour $(C_h, C_{h'}) \in S_t\}$
Si $ext_E(s_\alpha \cup s_\beta) = C_\alpha \cup C_\beta$
Alors
$C_{m+1} = C_\alpha \cup C_\beta$
$s_{m+1} = s_\alpha \cup s_\beta$
$I(C_{m+1}) = G(s_\alpha \cup s_\beta)$
$P_{t+1} = P_t \cup \{C_{m+1}\}$
$Q_{t+1} = Q_t \cup \{(C_{m+1}, s_{m+1})\}$
Sinon
$S_t = S_t \setminus (C_\alpha, C_\beta)$
Aller à **1**

5. Application

On a appliqué les méthodes décrites dans les sections précédentes au tableau de données présenté dans [HER 96], qui décrit 5 individus par 3 variables modales, exprimant la fréquence de certains paramètres pertinents pour l'étude de l'hypertension. Les variables sont les suivantes : *Tinnitus - fréquent ou rare, Maux de Tête - fréquents ou rares, Pression sanguine - haute, normale ou basse*.

Le tableau de données probabilistes est le suivant :

	Tinnitus		Maux de Tête		Pression Sanguine		
Nom	Fréquent	Rare	Fréquents	Rares	Haute	Normale	Basse
Anne	0.8	0.2	0.9	0.1	0.8	0.2	0.0
Bob	1.0	0.0	0.0	1.0	0.6	0.4	0.0
Chris	1.0	0.0	0.1	0.9	0.9	0.1	0.0
Doug	0.3	0.7	0.7	0.3	0.0	0.6	0.4
Eve	0.6	0.4	0.7	0.3	0.0	0.8	0.2

Sur ce tableau de données, on a construit le treillis de l'union (see Fig. 1), ainsi que la hiérarchie (see Fig. 2) et la pyramide (see Fig. 3) symboliques construites avec la fonction de généralisation f_u correspondante. Dans le treillis, les concepts sont indiqués par leur extension.

Comme prévu, la hiérarchie est contenue dans la pyramide, qui est elle-même une partie du treillis. Le nombre de concepts formés a été réduit de 31, dans le treillis, à 15 dans la pyramide et seulement 9 dans la hiérarchie. Le treillis organise tous les concepts, mais, sous cette forme, il ne permet pas de décider quels sont les concepts pertinents.

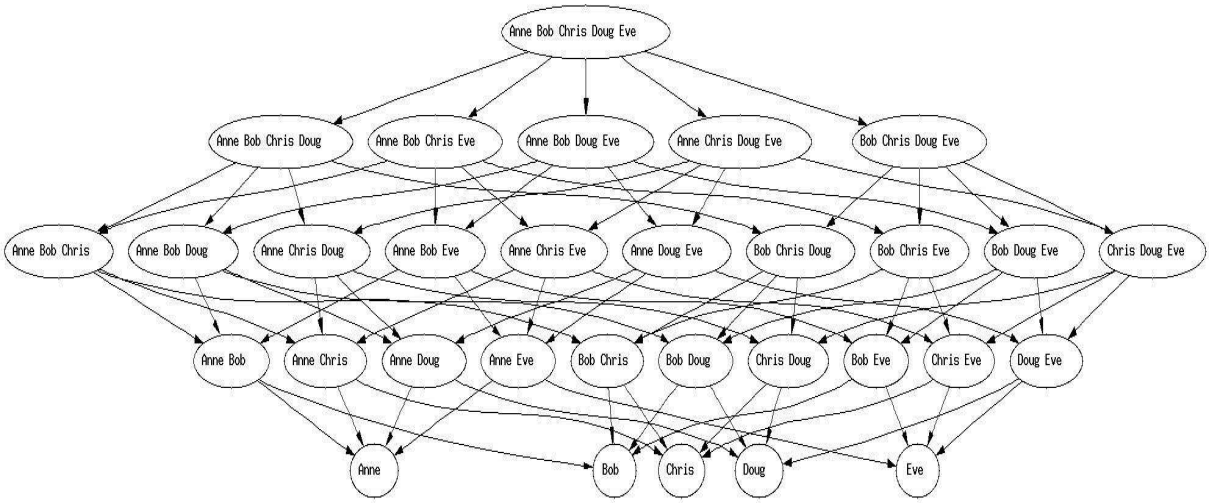


FIG. 1. *Données Herrman, Treillis de l'Union*

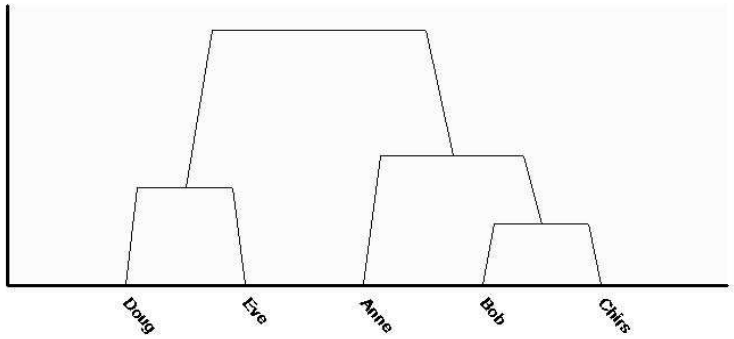


FIG. 2. *Données Herrman, Hiérarchie, Généralisation par le Maximum*

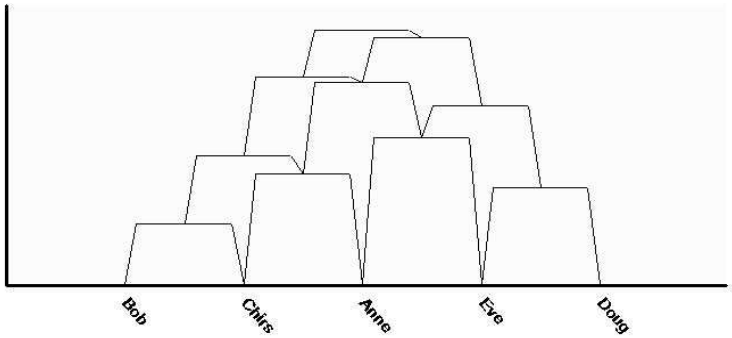


FIG. 3. *Données Herrman, Pyramide, Généralisation par le Maximum*

Dans la hiérarchie, on retient les concepts ((Anne, Bob, Chris), s1) et ((Doug, Eve), s2), avec
$s1 =$ [Tinnitus $\leq$ { fréquent (1), rare (0.2)}] $\wedge$ [Maux de Tête $\leq$ { fréquents (0.9), rares (1)}]$\wedge$
[Pression Sanguine $\leq$ { haute (0.9), normale (0.4), basse (0.0)}];
$s2 =$ [Tinnitus $\leq$ { fréquent (0.6), rare (0.7)}] $\wedge$ [Maux de Tête $\leq$ { fréquents (0.7), rares (0.3)}]$\wedge$
[Pression Sanguine $\leq$ { haute (0.0), normale (0.8), basse (0.4)}]

La pyramide permet la formation de concepts qui ne sont pas présents dans la hiérarchie, retenons par exemple le concept ((Anne, Eve), s3), avec
$s3 =$ [Tinnitus $\leq$ { fréquent (0.8), rare (0.4)}] $\wedge$ [Maux de Tête $\leq$ { fréquents (0.9), rares (0.3)}]$\wedge$
[Pression Sanguine $\leq$ { haute (0.8), normale (0.8), basse (0.2)}]
L'ordre induit par par pyramide, Bob-Chris-Anne-Eve-Doug, paraît traduire une importance décroissante du symptome *Tinnitus - fréquent*.

6. Conclusion

Dans ce papier, on a présenté des résultats permettant de construire deux treillis de Galois sur des données modales. L'avantage principal de l'approche proposée réside dans le fait qu'elle permet d'organiser les données modales directement, sans qu'aucune transformation préalable ne soit nécessaire. L'application pratique de la méthode reste cependant limitée par la taille des treillis obtenus, puisque le nombre de concepts tend à augmenter exponentiellement avec le nombre d'individus et de variables.

Une alternative consiste à limiter le nombre de concepts formés, en imposant un modèle de classification plus simple. Dans ce sens, une méthode de classification est proposée, qui utilise les modèles hiérarchique ou pyramidale. La méthode construit une hiérarchie ou une pyramide dont chaque classe est un concept du treillis correspondant, et permet d'ordonner les concepts par le *degr é de généralité* de leurs intentions, mesuré par l'affinité des distributions associées avec la distribution uniforme.

Le pas suivant consistera à prendre en compte l'ordre entre les modalités, quand il existe, dans la formation des concepts et dans l'évaluation de la généralité.

7. Bibliographie

[BAR 70] BARBUT M., MONJARDET B., *Ordre et Classification, Algèbre et Combinatoire*, vol. I et II, Hachette, Paris, 1970.

[BER 85] BERTRAND P., DIDAY E., A Visual Representation of the Compatibility between an Order and a Dissimilarity Index : The Pyramids, *Computational Statistics Quarterly*, vol. 2, n^o 1, 1985, p. 31-42.

[BER 86] BERTRAND P., Étude de la Représentation Pyramidale, PhD thesis, Université Paris IX Dauphine, Paris, France, 1986.

[BIR 40] BIRKHOFF G., *Lattice theory*, vol. XXV, 1st edition (3rd edition, 1967), American Mathematical Society Colloquium Publications, 1940.

[BOC 00] BOCK H. H., DIDAY E., *Analysis of Symbolic Data - Exploratory Methods for Extracting Statistical Information from Complex Data*, Springer Verlag, Berlin-Heidelberg, 2000.

[BRI 91] BRITO P., Analyse de données symboliques. Pyramides d'héritage, PhD thesis, Université Paris IX Dauphine, Paris, France, 1991.

[BRI 94a] BRITO P., Order Structure of Symbolic Assertion Objects, *IEEE Transactions on Knowledge and Data Engineering*, vol. 6, n^o 5, 1994, p. 830-835.

[BRI 94b] BRITO P., Use of Pyramids in Symbolic Data Analysis, DIDAY E., *et al*., Eds., *New Approaches in Classification and Data Analysis*, Springer-Verlag, Berlin-Heidelberg, 1994, p. 378-386.

[BRI 98] BRITO P., Symbolic Clustering of Probabilistic Data, RIZZI A., VICHI M., BOCK H.-H., Eds., *Advances in Data Science and Classification*, Springer-Verlag, Berlin-Heidelberg, 1998, p. 385-390.

[BRI 99] BRITO P., DE CARVALHO F., Symbolic Clustering in the Presence of Hierarchical Rules, *Proceedings of the Conference on Knowledge Extraction and Symbolic Data Analysis (KESDA'98)*, Office for Official Publications of the European Communities, Luxembourg, 1999, p. 119-128.

[BRI 02] BRITO P., DE CARVALHO F. A. T., Symbolic Clustering of Constrained Probabilistic Data, OPITZ O., SCHWAIGER M., Eds., *Exploratory Data Analysis in Empirical Research*, Springer Verlag, Heidelberg, 2002, p. 12-21.

[BRI 04] BRITO P., POLAILLON G., Structuring probabilistic data by Galois lattices, *Mathématiques, Informatique et Sciences Humaines*, vol. à paraître, 2004.

[CHA 98] CHAVENT M., A monothetic clustering method, *Pattern Recognition Letters*, vol. 19, 1998, p. 989-996.

[DID 84] DIDAY E., Une Représentation Visuelle des Classes Empiétantes : Les Pyramides, rapport n^{o}291, 1984, INRIA, Rocquencourt, Le Chesnay.

[DID 86] DIDAY E., Orders and Overlapping Clusters by Pyramids, LEEUW J. D., *et al.*, Eds., *Multidimensional Data Analysis*, DSWO Press, Leiden, 1986, p. 201-234.

[DUQ 86] DUQUENNE V., GUIGUES J., Familles minimales d'implication informatives résultant d'un tableau de données binaires, *Mathématiques, Informatique et Sciences Humaines*, vol. 95, 1986, p. 5-18.

[DUQ 87] DUQUENNE V., Contextual implications between attributes and some representation properties for finite lattices, GANTER B., WILLE R., WOLFF K. E., Eds., *Beitrage zur Begriffsanalyse*, Darmstadt, 1987, p. 149-172.

[ELS 98] EL-SONBATY, Y. ISMAIL M. A., On-line hierarchical clustering, *Pattern Recognition Letters*, vol. 19, 1998, p. 1285-1291.

[GAN 99] GANTER B., WILLE R., *Formal Concept Analysis - Mathematical Foundations*, Springer Verlag, New York, 1999.

[GIR 99] GIRARD R., RALAMBONDRAINY H., Recherche de concepts à partir de données arborescentes et imprécises, *Mathématiques, Informatique et Sciences Humaines*, vol. 147, 1999, p. 87-111.

[GOD 95] GODIN R., MINEAU G., MISSAOUI R., MILI H., Méthodes de classification conceptuelle basées sur les treilllis de Galois et applications, *Revue d'intelligence artificielle*, vol. 9(2), 1995, p. 105-137.

[GOW 91] GOWDA K. C., DIDAY E., Symbolic clustering using a new dissimilarity measure, *Pattern Recognition*, vol. 24, n^o 6, 1991, p. 567-578.

[GOW 92] GOWDA K. C., DIDAY E., Symbolic clustering using a new similarity measure, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, n^o 2, 1992, p. 368-378.

[GOW 95a] GOWDA K. C., RAVI T. R., Agglomerative clustering of symbolic objects using the concepts of both similarity and dissimilarity, *Pattern Recognition Letters*, vol. 16, 1995, p. 647-652.

[GOW 95b] GOWDA K. C., RAVI T. R., Divisive clustering of symbolic objects using the concepts of both similarity and dissimilarity, *Pattern Recognition*, vol. 28, n^o 8, 1995, p. 1277-1282.

[GUé 93] GUÉNOCHE A., Hiérarchies conceptuelles de données binaires, *Mathématiques, Informatique et Sciences Humaines*, vol. 121, 1993, p. 23-34.

[HER 96] HERRMANN C. S., HÖLLDOBLER S., STROHMAIER A., Fuzzy conceptual knowledge processing, *Proceedings of the ACM Symposium on Applied Computing, Philadelphia*, ACM Press, New York, 1996, p. 628-632.

[MAT 51] MATUSITA K., Decision rules based on distance for problems of fit, two samples and estimation, *Ann. Math. Stat.*, vol. 3, 1951, p. 1-30.

[MEP 93] MEPHU NGUIFO E., Une nouvelle approche basée sur le treillis de Galois, pour l'apprentissage de concepts, *Mathématiques, Informatique et Sciences Humaines*, vol. 124, 1993, p. 19-38.

[POL 98a] POLAILLON G., Interpretation and reduction of Galois lattices of complex data, RIZZI A., VICHI M., BOCK H.-H., Eds., *Advances in Data Science and Classification*, Springer-Verlag, Berlin-Heidelberg, 1998, p. 433-440.

[POL 98b] POLAILLON G., Organisation et interprétation par les treillis de Galois de données de type multivalué, intervalle ou histogramme, PhD thesis, Université Paris IX Dauphine, Paris, France, 1998.

[POL 99] POLAILLON G., DIDAY E., Reduction of symbolic Galois lattices via hierarchies, *Proceedings of the Conference on Knowledge Extraction and Symbolic Data Analysis (KESDA'98)*, Office for Official Publications of the European Communities, Luxembourg, 1999, p. 137-143.

[WIL 82] WILLE R., Restructuring lattice theory : an approach based on hierarchies of concepts, REIDEL D., Ed., *Ordered Sets*, Dordrecht-Boston, 1982, p. 445-470.

Reconnaissance des Formes et Analyse d'Images à Tours

Hubert CARDOT

LI, Université de Tours
Polytech'Tours - DI
64 avenue Jean Portalis
37200 Tours
hubert.cardot@univ-tours.fr

RÉSUMÉ. Plusieurs études sont présentées successivement : les SVM appliqués à de grandes bases de données, les réseaux de neurones récurrents pour la prévision, la fusion d'information en utilisant la théorie de l'évidence, une adaptation des contours actifs en segmentation d'images, une évolution des liens hypertextes afin de mieux représenter la connaissance, une analyse de vidéos pour aider à l'évaluation de l'autisme chez les enfants, une analyse de signaux vidéos et sonores dans le domaine médical, et enfin, l'authentification de personnes à partir de signatures manuscrites ou de l'utilisation de périphériques simples.

MOTS-CLÉS : Reconnaissance des formes, analyse d'images, classification, authentification.

1. Introduction

Cet article est une description sommaire de plusieurs études menées principalement au sein de l'équipe RFAI (Reconnaissance des Formes et Analyse d'Images) du LI (Laboratoire d'Informatique) de l'Université de Tours. Cette équipe est constituée d'une vingtaine de chercheurs en comptant les doctorants. L'objectif est de montrer à la communauté une partie significative de nos activités actuelles sachant que des descriptions plus approfondies pourront être trouvées dans la bibliographie.

La première partie va s'intéresser aux outils et méthodes développés pour la reconnaissance des formes. En particulier, nous verrons des études sur la classification de données, la prévision dans des séries temporelles, la fusion d'information, la segmentation d'images, et la structuration des connaissances.

Dans la deuxième partie, c'est plutôt l'aspect applicatif qui est décrit : analyse de vidéos pour aider à l'évaluation de l'autisme chez les enfants, analyse de signaux vidéos et sonores dans le domaine médical, et authentification de personnes à partir de signatures manuscrites ou de l'utilisation de périphériques simples.

2. Reconnaissance des Formes

2.1. *SVM*

Les Machines à Vecteurs Supports (SVM) [VAP 98] sont une approche très efficace pour résoudre les problèmes de classification. Nos travaux sur les SVM se font en collaboration avec des chercheurs du LUSAC de l'Université de Caen.

L'utilisation des SVM devient problématique quand on l'applique sur des grandes bases de données, même si on se sert d'algorithmes dédiés à ce type de problème (comme SMO [PLA 99]). En effet, la complexité, en temps de calcul, augmente plus que proportionnellement avec la taille de la base d'apprentissage. De plus, le

nombre de vecteurs supports augmente aussi avec la taille de la base d'apprentissage surtout si les données sont bruitées.

Notre idée principale pour résoudre ce problème repose sur l'utilisation de la Quantification Vectorielle (QV) [GER 91] pour construire une base d'apprentissage de taille réduite en produisant un ensemble de prototypes exemples. Les résultats expérimentaux obtenus (tableau 1) montrent que la méthode proposée réduit fortement le temps d'apprentissage avec une très faible détérioration des taux de classification [LEB 04]. De plus cette méthode produit une fonction de décision de complexité réduite lorsque les données sont bruitées.

	notre méthode		méthode classique	
bases	temps	erreur	temps	erreur
satimage	8.5	10.7	29.9	10.1
letter	19	6.34	827	6.34
shuttle	3.8	0.09	412	0.09

Tableau 1 : Taux d'erreur et temps d'apprentissage (en heures)

2.2. *Réseaux de neurones récurrents*

Les réseaux de neurones récurrents ou bouclés sont par leur structure adaptés au traitement de données séquentielles [BON 03]. En particulier, la prévision de séries temporelles est une application indispensable dans de nombreux domaines tels que la météo, la finance ou le marketing.

Le plus souvent le modèle mathématique qui a généré la série est inconnu. Les relations entre les valeurs passées de la série et les valeurs futures doivent alors être déduites par apprentissage à partir des valeurs passées. Cette relation peut être décrite par une fonction f telle que : $x_t = f(x_{t-1}, x_{t-2}, \ldots)$ avec x_t la valeur future à estimer et $(x_{t-1}, x_{t-2}, \ldots)$ les valeurs récentes de la série temporelle.

Le nombre de valeurs récentes à prendre en compte dans la fonction f dépend du contexte. L'algorithme BPTT (Back Propagation Through Time) [WER 90] [WIL 90] permet d'entraîner des réseaux de neurones récurrents en s'adaptant à ce contexte.

Nous développons plusieurs approches pour en améliorer les performances [ASS 03]. Une de ces approches est connue sous le nom de *Boosting* [SCH 90] [FRE 90]. Le principe est de réaliser l'apprentissage plusieurs fois sur tout ou partie de la base d'apprentissage en favorisant à chaque étape les exemples qui ont été précédemment estimés difficiles. Ensuite, il faut combiner les modèles issus de ces apprentissages.

Les résultats actuels, obtenus sur deux séries temporelles (taches solaires, Mackey-Glass 17), montrent une amélioration des performances grâce à notre algorithme de *Boosting* comparées à l'utilisation d'un seul réseau de neurones récurrent déjà très performant.

Nous continuons la recherche d'améliorations à appliquer au problème de l'apprentissage des dépendances temporelles pour la prévision de valeurs futures. Une application en cours de développement est la reconnaissance de phonèmes à partir de séquences vidéo.

2.3. *Fusion d'information*

Nous travaillons sur la fusion de l'information et sur son application au problème de l'authentification de signatures graphiques hors-ligne.

La base de données consiste en 525 images réelles de signatures hors-ligne manuscrites obtenues auprès de 35 personnes qui ont signé chacune une quinzaine fois. Deux classifieurs de distance à base de distance euclidienne sont employés pour la classification de ces images de signatures après l'extraction de leurs caractéristiques. Le premier classifieur représente les caractéristiques globales et générales de la forme de signature, extraites avec des primitives basées sur l'histogramme. Le deuxième classifieur représente les caractéristiques globales et locales, et aussi la lisibilité et la complexité d'une signature, extraites avec un ensemble de primitives comme la dimension fractale, la dimension de masse, la pente de la signature et la fraction du contour d'une image de la signature, etc. La performance de ces deux classifieurs a été testée avec la méthode du « Leave-one-out ». Les taux de reconnaissance de ces deux classifieurs sont de 69,52 % et 69,33 % respectivement. Ici, la décision repose sur un classifieur des 5 plus-proches-voisins à la majorité simple.

Le taux de reconnaissance des classifieurs a été amélioré en utilisant la théorie de l'évidence de Dempster-Shafer [SHA 76] avec notre approche floue non paramétrique pour la modélisation des fonctions de croyance. Le résultat obtenu [ARI 04] est montré dans le tableau 2. La fusion a été réalisée en prenant les 5 premières classes

d'après leur rang (k = 5) pour chaque classifieur. Les éléments focaux possibles ont été choisis automatiquement par notre algorithme et leurs masses de croyance ont été calculées à l'aide de la fonction du degré d'appartenance. La décision de la fusion a été prise avec un argument de maximum de la valeur de masse de croyance. En cas de non fusion ou d'ambiguïté, notre algorithme continue son travail en prenant k = 10, ou 15, jusqu'à 20. Si l'ambiguïté persiste avec k = 20, le résultat initial avec k = 5 est retenu.

La validation et l'exécution de notre méthodologie ont été vérifiées en comparant les résultats avec ceux obtenus avec une méthode de fusion se fondant sur la matrice de confusion des classifieurs et l'intégration de croyance basée sur la formule bayésienne qui est expliquée dans [XU 92]. Le résultat obtenu avec cette méthode est aussi montré dans le tableau 2.

Méthode	Taux de Reconnaissance
Classifieur 1	69,52%
Classifieur 2	69,33%
Théorie de l'évidence (notre approche)	91,85 %
Matrice de confusion	91,15 %

Tableau 2 : Performance des méthodes

Ce travail se poursuit dans le sens d'une généralisation de notre fonction de degré d'appartenance floue en vue de l'application à la combinaison de différents types de classifieurs.

2.4. *Analyse d'images*

Nos travaux de reconnaissance des formes portent principalement sur des formes extraites à partir d'images. Dans ce cas, une étape importante est la segmentation qui permet de partitionner l'image en régions dont l'analyse pourra aboutir à des formes connues.

Plusieurs approches de la segmentation co-existent avec chacune leur domaine de prédilection. Celle que nous privilégions est basée sur les contours actifs (*snakes*) [KAS 87] car elle est adaptée au suivi d'objets déformables dans des séquences d'images.

Le principe est d'associer au contour une fonctionnelle d'énergie qu'ensuite l'algorithme va tendre à minimiser [ROU 03]. Il existe plusieurs possibilités pour cet algorithme ; nous privilégions l'algorithme glouton (*greedy*) qui est très rapide pour une qualité satisfaisante.

La fonctionnelle d'énergie du contour est composée de trois termes d'énergie : l'énergie interne, externe, et de contexte. Ces trois termes d'énergie, qui peuvent eux-mêmes être subdivisés, vont être pondérés par un coefficient proportionnel à l'importance de ce terme d'énergie dans le résultat final. Ces coefficients dépendent des images et des objets à retrouver : contours anguleux ou arrondis, région contrastée, … Le réglage de ces coefficients peut se faire par essais-erreurs par un expert du traitement d'images ou même par un expert du domaine d'application. Cette étape est fastidieuse, c'est pourquoi nous recherchons des méthodes pour les déterminer automatiquement. Nos essais actuels portent notamment sur une méthode basée sur l'algorithme Tabou. Cette méthode est appliquée aussi bien pour l'optimisation de la fonctionnelle d'énergie que pour les coefficients (poids) qu'elle contient.

Pour répondre à des besoins applicatifs imminents pour le suivi d'objets 3D dans des séquences d'images 3D, nous allons généraliser nos contours actifs à des surfaces actives.

2.5. *Liens intelligents*

La reconnaissance des formes s'applique naturellement à des objets dans des images mais elle peut aussi s'étendre à des formes moins tangibles comme le contexte d'un lien hypertexte dans une page web.

Cette étude part de la constatation que les auteurs qui publient des connaissances sous la forme de documents électroniques lisibles sur un écran utilisent de plus en plus la technologie des liens hypertextes pour améliorer la

présentation et la lisibilité de leur travail [VER 00]. Il est alors logique de se demander si ces liens ne représenteraient pas une structure formelle intéressante pour intégrer les outils conceptuels et technologiques favorisant la structuration des connaissances et leur partage par une communauté.

Pour répondre à cette question, nous réalisons une typologie des besoins auxquels répondent les liens que les auteurs placent dans les liens hypertextes électroniques sur le web. En d'autres termes, nous explorons toute la richesse sémantique explicite ou implicite contenue dans les liens hypertextes présents dans un domaine précis, puis nous vérifions la présence de cette richesse sémantique dans les outils technologiques existants avec l'objectif de confirmer que l'exploitation de cette richesse sémantique peut faciliter le partage des connaissances sur le web.

3. Applications

3.1. *Autisme*

Cette étude a pour but de quantifier de façon précise les différences de préhension existant entre les enfants sains et les enfants atteints d'autisme [MAR 03]. Elle se déroule en collaboration avec l'Equipe 1 « Autisme et troubles du développement : psychopathologie, physiopathologie et thérapeutique » de l'Unité Inserm 619 « Dynamique et pathologie du développement cérébral » et dans le cadre de l'IFR 135 « Imagerie fonctionnelle ».

Pour réaliser ces analyses, l'enfant est assis dans une chaise adaptée à sa taille et fait reposer son avant-bras sur une table spécifiquement ajustée en fonction de sa taille et de la hauteur de la chaise. La consigne donnée à l'enfant est de saisir l'objet lors du signal de départ et de le reposer à un endroit cible de la table. Plusieurs formes d'objets sont testées (cube, cylindre, sphère) de différentes tailles. Les objets peuvent être composés d'une forme simple ou de plusieurs formes (cube, cylindre, fourchette en plastique…).

L'enfant est filmé pendant ses mouvements par 3 caméras permettant d'obtenir une vision en 3 dimensions de la main lors de la saisie (vue de dessus, vue de face, vue de profil). A partir de ces séquences, des opérateurs de traitement d'images (contours actifs) permettent de suivre la main et de calculer des caractéristiques (angle global, angle des phalanges, écartement des doigts autour de l'objet manipulé, vitesse de déplacement…).

Quand cette étape d'acquisition et d'extraction de caractéristiques sera au point, des analyses des données obtenues [LEZ 01] avec des enfants sains et des enfants atteints d'autisme permettront de rechercher des caractéristiques discriminantes pour l'évaluation fine de cette pathologie.

3.2. *Analyse de signaux vidéos et sonores : application à l'étude de signaux médicaux*

La problématique considérée concerne l'étude de séquences multimédia constituées d'images et de sons dont il s'agit d'étudier les corrélations de manière à aider à la compréhension de l'origine des bruits.

L'analyse des séquences d'images consiste à suivre les objets en mouvement de manière à permettre leur étude. Une méthode générique, reposant sur une combinaison de suivi de régions et de contours, et une méthode adaptée aux objets homogènes, reposant sur la théorie des ensembles de niveaux, sont proposées [DEL 03].

L'analyse des données sonores consiste en l'élaboration d'un système d'identification reposant sur l'étude de la structure des signaux grâce à des codages adaptés et à leur modélisation par les lois de Zipf.

Ces méthodes ont été évaluées sur des séquences acoustico-radiologiques dans le cadre de l'étude de la pathologie du reflux gastro-oesophagien, en collaboration avec l'équipe Acoustique et Motricité Digestive de l'Université de Tours.

3.3. *Authentification basée sur l'écriture manuscrite*

L'authentification basée sur l'écriture manuscrite et en particulier sur la signature manuscrite est la manière la mieux acceptée par les utilisateurs parmi les différentes méthodes d'authentification biométrique. En effet, c'est un moyen simple et encore très utilisé d'authentifier les documents ou les chèques. Nous parlons dans ces cas de signatures hors-ligne, c'est-à-dire que seule l'image est disponible, par opposition aux signatures en ligne où l'information dynamique a pu être conservée.

La difficulté dans de tels systèmes est la détermination de caractéristiques stables et discriminantes. C'est pourquoi nous proposons de nouvelles caractéristiques basées sur la dimension fractale [HUA 00], [WIR 04]. Nos résultats actuels montrent que nos caractéristiques associées à des caractéristiques classiques permettent d'obtenir de meilleures performances qu'avec les caractéristiques classiques seules. L'application d'une méthode de sélection de caractéristiques basée sur les algorithmes génétiques a permis de confirmer la pertinence de nos nouvelles caractéristiques.

3.4. *Authentification basée sur le clavier et la souris*

Aujourd'hui, les exigences de sécurité pour l'accès à des ressources informatiques, ne permettent plus de se limiter au traditionnel couple login et mot de passe pour authentifier un utilisateur. En effet les utilisateurs ne prêtent souvent pas assez attention à sa sécurité, ils choisissent ainsi des mots de passe trop courts ou trop simples, de plus ils n'accordent pas assez d'importance à la confidentialité de ceux-ci entraînant souvent leurs divulgations de façon intensionnel ou non. Pour remédier à ce problème une solution prometteuse est la biométrie. Mais celle-ci nécessite l'ajout de capteurs coûteux et est souvent mal acceptée par les utilisateurs. Il semble donc intéressant d'étudier la possibilité de se limiter aux données pouvant être extraites à l'aide de périphériques se trouvant sur tout ordinateur, c'est-à-dire dans un premier temps uniquement à l'aide du clavier et de la souris.

Dans le cas du clavier, l'authentification va utiliser l'étude de la dynamique de frappe des utilisateurs [DOW 01] [GUV 03]. Cette dynamique, en fait le style de l'utilisateur au clavier, va être étudiée en s'intéressant à la frappe de touches successives qui vont être regroupées par deux. Pour chacun de ces couples, nous allons extraire des donnés temporelles (temps entre touches, temps de pression d'une touche) afin de caractériser le comportement de l'utilisateur. A ces données quantitatives, il va être possible de rajouter des données qualitatives comme par exemple l'ordre de relâchement des touches lors de la réalisation d'une majuscule.

Pour la souris, l'étude va se concentrer dans un premier temps sur des séquences d'interactions prédéfinies. Ces séquences pourront être une signature [SYU 98], un mot de passe graphique ou encore le comportement au cours d'un mini-jeu. Au cours ce ces séquences, on extrait des caractéristiques permettant de définir un utilisateur, ces caractéristiques pourront être spatiales (longueur parcourue, courbure…) ou dynamiques (vitesse, accélération, vitesse angulaire…). Les problèmes principaux de cette étude sont, d'une part d'identifier les caractéristiques qui vont nous permettre de séparer un utilisateur authentique d'un imposteur et, d'autre part la mise en place des outils de comparaison.

Les premiers résultats obtenus à l'aide du clavier nous donne 7 % pour les deux taux d'erreur (faux acceptés et vrais rejetés) sur une petite phrase connue. A partir de 10 lignes de texte tapées, les taux d'erreur tombent à 0 %. Ces résultats encourageants restent bien sûr à valider et à améliorer. Pour la souris, les résultats ne sont pas encore significatifs.

4. Conclusion

Nous avons eu, dans cet article, l'occasion de découvrir un nombre important d'études menées par des membres de l'équipe RFAI du LI de Tours. Je n'ai d'ailleurs pas mentionné leur nom pour ne pas alourdir le texte mais il est évident que ces travaux sont le résultat de nombreux chercheurs et d'étudiants encadrés par eux.

Toutefois, cette description est loin d'être exhaustive, ce qui n'était pas l'objectif ; en particulier, nous travaillons aussi sur les chaînes de Markov cachées [LEF 03] et sur l'hybridation de modèles.

Ainsi, nous couvrons un large spectre de méthodes du domaine de la reconnaissance des formes que nous avons l'intention de faire évoluer et d'appliquer dans les années à venir à des objets obtenus à partir d'une ou plusieurs sources vidéos et à l'authentification de personnes.

5. Bibliographie

[VAP 98] V. N. Vapnik. *Statistical Learning Theory*. New York, Wiley edition, 1998.

[PLA 99] J. Platt. *Fast Training of Support Vector Machines using Sequential Minimal Optimization, Advances in Kernel Methods-Support Vector Learning*. MIT Press, pp. 185-208, 1999.

[GER 91] A. Gersho et R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic, 1991.

[LEB 04] Gilles LEBRUN Christophe CHARRIER Olivier LEZORAY, *Réduction du temps d'apprentissage des SVM par Quantification Vectorielle* , CORESA, mai 2004.

[BON 03] Boné R. (2003), *Les dépendances temporelles dans les réseaux de neurones. Application à la prévision à un pas et multipas* , conférence plénière, 10ème rencontre internationale Approches Connexionnistes en Sciences Economiques et de Gestion (ACSEG 2003), Nantes, France, Novembre 2003, pp 117-129.

[WER 90] P.J. Werbos. *Backpropagation through time: what it does and how to do it*. Proceedings of IEEE, Special issue on neural networks, vol. 78, No. 10, pp.1550-1560, October 1990.

[WIL 90] R.J. Williams, J. Peng. *An efficient gradient-based algorithm for on line training of recurrent network trajectories*. Neural Computation 2: 490-501, 1990.

[ASS 03] Assaad M., Boné R. (2003), *Apprentissage itératif de réseaux de neurones récurrents pour la prévision de séries temporelles*, 10ème rencontre internationale Approches Connexionnistes en Sciences Economiques et de Gestion (ACSEG 2003), Nantes, France, Novembre 2003, pp. 63-74.

[SCH 90] Schapire R. E., *The strenght of weak learnability*, Machine Learning, 5, 197-227, 1990.

[FRE 90] Freund Y., *Boosting a weak learning algorithm by majority*, 3rd annual workshop on Computational Learning Theory, 202-216, 1990.

[SHA 76] G. Shafer. *A mathematical Theory of Evidence*. Princeton Univ Press., Princeton New Jersey, 1976.

[ARI 04] M. Arif, T. Brouard, N. Vincent, *Amélioration de la reconnaissance des formes par la fusion de l'information*, SETIT 2004, Sousse, Tunisie, 15-20 mars 2004.

[XU 92] L Xu, A Krzyżak, C. Y Suen, *Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition*, *IEEE Trans. Sys., Man & Cyber.*, vol. 22(3), pp. 418-435, 1992.

[KAS 87] M. Kass, A. Witkin, D. Terzopoulos, *Snakes : Active Contour Models*, Proceedings of the first International Conference on Computer Vision, juin 1987, pp. 259-268.

[ROU 03] J-J. ROUSSELLE. *Les contours actifs, une méthode de segmentation - Application à l'imagerie médicale*. Thèse de doctorat en Informatique. Laboratoire Informatique - Université de Tours. Juillet 2003. p. 162.

[VER 00] G. Verley, J-J. Rousselle, *An Evolved Link-specification Language for Creating and Sharing Documents on the Web*, CRIS 2000, 25-27 mai 2000.

[MAR 03] MARTINEAU J, COCHIN S. *Visual perception in children : human, animal and virtual movement activates different cortical areas*. International Journal of Psychophysiology, 2003, 51(1):37-44.

[LEZ 01] LEZORAY O, CARDOT H. *A Neural Network Architecture for Data Classification, International Journal of Neural Systems*, Vol. 11, Numéro 1, pp. 33-42, février 2001.

[DEL 03] E. DELLANDREA. *Analyse de signaux vidéos et sonores : application à l'étude de signaux médicaux*. Thèse de Doctorat en Informatique. Université de Tours. Octobre 2003.

[HUA 00] K. Huang and H. Yan, *Signature Verification using Fractal Transformation*, International Conference on Pattern Recognition (ICPR'00), pp. 2851-2854, Barcelona, Spain, September 2000.

[WIR 04] M. Wirotius, J.Y. Ramel, N. Vincent, *New features for authentication by on-line handwritten signatures*, International Conference on Biometric Authentication (ICBA), Japan, July 2004.

[DOW 01] Dowland, P.S., H. Singh, and S.M. Furnell. *A Preliminary Investigation of User Authentication Using Continuous Keystroke Analysis,* IFIP 8th Annual Working Conference on Information Security Management & Small Systems Security. 2001. Las Vegas.

[GUV 03] Guven, A. and I. Sogukpinar, *Understanding users'keystroke patterns for computer access security*. Computers & Security, 2003. 22(8).

[SYU 98] Syukri, A.F., E. Okamoto, and M. Mambo. *A User Identification System Using Signature Written with Mouse*. Third Australasian Conference on Information Security and Privacy. 1998.

[LEF 03] S. LEFEVRE, E. BOUTON, T. BROUARD, N. VINCENT, *A new way to use hidden Markov Models for Object Tracking in Video Sequences*, International Conference on Image Processing (ICIP2003), 14-17 septembre 2003, Barcelone (Espagne).

Classifying Classification Problems

J. C. Gower

The Open University
Department of Statistics
Walton Hall
Milton Keynes, MK7 6AA, U.K.

RÉSUMÉ. Classification problems, both for assignment and class construction, are specified either in probabilistic form or not. Underlying issues are (i) the types of sampling unit under consideration: in particular, are they differentiated into previously determined classes (possibly with identical members) or are they undifferentiated and (ii) considerations of the types of variable used; are they quantitative or categorical? Rather than a simple data-matrix, the fundamental form of data is taken to be the between and within-group structure. These considerations lead to a simple cross-classification of familiar, and some novel, classification problems.

MOTS-CLÉS: Probabilistic Classification, Non-probabilistic Classification, Classes, Groups, Assignment, Class Construction, Approximation.

1. Introduction

In the following, I shall attempt a simple classification of a range of classification problems. A major, and well-recognised, division is into problems of forming classes and problems of assigning to previously recognised classes. The other elements of the classification proposed here are the role of probabilistic formulations and the structure of the data. These are discussed first in a general context and in subsequent sections in more detail.

When forming classes, we must define just what kind of thing it is that we are classifying. In the early days of numerical taxonomy, Sneath and Sokal (1973) coined the term Operational Taxonomic Unit (OTU) to describe the units under study. This term is little used nowadays, perhaps because taxonomy is now subsumed into a wider classification discipline. Nevertheless, it is useful to have some such term, so I shall use Operational Unit (OU).

Perhaps, it is not fully appreciated that very often we form classes from OUs that are already recognized as classes. For example, Linnaeus classified animal and plant life but life-forms such as “cats”, and “lilies” and so on, had long been universally recognised. These are the relevant OUs. Just by giving names to groups recognizes them as classes with associated widely accepted properties; nouns are very often the names of classes. Probabilistic notions play little or no part in defining the class of cats, nor of classifying them further into genera, families, etc. Thus, at the outset the OUs may be differentiated into recognised classes. Even though these initial classes are recognised, there remains the goal of further classification to organise what may be a very large body of information.

On other occasions, the OUs are undifferentiated but the possibility is recognised that there may be some heterogeneity that may be useful as a basis for differentiating the OUs into two,

or more classes. This is the position with mixture problems which naturally, though not essentially, are given probabilistic formulations. In its purest form, instead of forming classes of previously named OUs, mixture problems are concerned with the possibility of deriving totally new classes from undifferentiated units – which may then be named, if we wish. Historically, Karl Pearson (1898) had a sample of ancient bones that were almost certainly a mixture of male and female bones which he wanted to separate; in this case he knew the names and number of classes involved. Nowadays, his problem might have been treated by discriminant analysis, assigning each bone to a population either of known female or known male bones. Nevertheless, there remains the possibility that modern known populations might not reflect the variability of ancient times, so justifying the mixture formulation.

Fundamentally, we have a division into where the OUs to be classified are (i) differentiated into named classes and (ii) where they are undifferentiated.

Probabilistic and non-probabilistic approaches have been mentioned. An influence on the more appropriate formulation is associated with the types of variable used to describe the OUs. With quantitative variables, there is certainly going to be variation within classes. When variation is large within OUs compared to the variation between OUs, overlap between the OUs is likely and probabilistic ideas become relevant. With categorical, including binary, variables there may still be overlapping variation but often it is possible to choose characters that do not vary within OUs (e.g. all cats have retractable claws) and then probability is less irrelevant. Clearly, when possible, it is better to base classifications on non-variable characteristics of the OUs.

2. Data structure

The starting point of most algorithms is a data matrix $\boldsymbol{X}$ with N rows (often referring to the OUs) and P columns of variables. The above remarks on structured and unstructured OUs suggest that a data matrix may be of several types that at first sight may look very similar. Nevertheless the differences can be fundamental. I believe that the failure to distinguish between superficially similar types of data matrix is at the root of many misuses of classification software. The situation may be clarified by considering the usual between and within groups sample-structure that is familiar in simple analysis of variance and in canonical variate analysis. Thus, the N rows of a data matrix $\boldsymbol{X}$ refer to objects drawn from K groups the kth of which has n_k members and $N = \sum_{k=1}^{K} n_k$. The K groups are the OUs whereas the n_k objects are undifferentiated samples pertaining to the kth OU.

When $K = 1$, so that $N = n_1$, we have the classical data-matrix of statistical multivariate analysis. Typically, its rows represent a random sample of size $N = n_1$. The samples have no *a priori* structure, so are said to be *unstructured*. Note that although the *data* are unstructured, this does not preclude the possibility that they might be fitted to a highly structured model. Indeed, that is the approach of the multivariate mixture problem which seeks a representation of the underlying probability distribution as a mixture of M more simple distributions. After analysis, the previously unstructured samples can be assigned to the M newly-found distribution-groups which may become named structured OUs of interest.

By contrast, when $n_k = 1$ for $k = 1,2,...,K$ we have $N = K$ OUs at the outset, where each OU is represented by *one* object, or row, of $\boldsymbol{X}$. Normally, the OUs will bear names indicating an initial classification. Botanists recognise species like daisies and dandelions, linguists recognise languages, geographers recognise rivers and mountains, librarians recognise book-titles and so on. The setting $n_k = 1$ need not imply that the kth OU class has only one member; merely that all its members are indistinguishable on the basis of the P chosen variables. This initial classification does not inhibit a desire for further classification; rather it encourages classification to organise what can be very large bodies of information.

This structure with $N = K$, $n_k = 1$ is quite different from the familiar statistical data-matrix with $K = 1$, $N = n_1$ and would be an inadequate representation of reality when variability within

groups is substantial. However, there are many classification problems where within-group variability may be ignored. For example, the *k*th OU may have a unique member (there is only one Bordeaux) or we may be able to choose variables which do not vary within groups (all cats have claws) but do vary between groups. As we have seen, this is a very common situation and those interested in classifying groups will seek to describe the groups by variables which are constant within groups, or essentially so (see below for "typical" OUs in taxonomy); indeed, they would be foolish not to do this when it is a possibility. When constant within-group variables are unavailable, the only remaining possibility is to base classifications on variables that do vary within groups, in which case probabilistic methods become relevant and the choice $n_k = 1$ becomes untenable; then replication of samples within groups is essential to capture the within-group variability. Within groups variation is a familiar component of the assignment problem of classical discriminant analysis but we show below, in section 3, how it can also be important when constructing classes.

To recapitulate, the objects to be classified are said to be structured when they fall into named OUs, which implies that a preliminary classification is recognised. Otherwise they are said to be unstructured, which usually means that the rows of $\boldsymbol{X}$ are undifferentiated as with a random sample assumed to be drawn from some notional population or mixture of populations. All non-probabilistic classification problems are concerned with structured data. In probabilistic classification problems we meet with both structured (e.g. canonical variate analysis) and unstructured (e.g. multivariate mixtures) sets of OUs.

When there is replication within groups, it is important to know whether or not the n_k replicates are chosen by some random process. It is not obvious that simple random samples are necessarily a desirable basis for forming classifications. For example, the number of speakers of a language is unlikely to be relevant for classifying languages thus indicating that some better representation of speakers than is given by a random sample. Indeed, it is the *distribution* of variables within the groups which is important, not the relative frequencies of speakers. Each language might be represented by a single set of characteristics as in taxonomy where within-group variation is often handled by representing each group by a single invented object with *typical* values; this is acceptable when within group variation is small relative to between group variation. Choosing $n_k = 1$ gives an extreme form of non-random sampling but nevertheless may provide a better representation of the language OU.

We have considered the most simple between-within structure where the K groups represent the totality of objects to be classified. This does not preclude the possibility of additional groups being added at a later stage. Also the groups might be regarded as a random sample of some larger set, and then probabilistic methods might have a role to play; this seems an artificial set-up. Rather than a basic K groups structure, the groups might have an elaborate *a priori* imposed structure of the crossed and nested kinds, indicating an equally elaborate *a priori* classification. It seems unlikely that one would wish to use this classification as the starting point for further classification. Rather, one might begin again by subsuming the complex structure into a K-group structure and comparing any new classification with the old one. These possibilities point the way to some areas of future research in classification theory.

The above has said little about the types of variable used in classification. Variables may be numerical or categorical. If numerical, they may be continuous on ratio or interval scales, a distinction which rarely affects classification. Categorical variables may be nominal, ordinal or dichotomous, a special binary categorical variable with one category merely defined to be not the other; all these types contain different kinds of information which may be exploited when classifying things. Numerical variables are less important for non-probabilistic classification than are categorical variables. This is because numerical variables are likely to vary within groups, so when non-varying categorical variables can be found, they are to be preferred. Just as objects may be structured, so may variables. Structure in variables has been long-recognised in survey design and Gower (1971) took it into account when designing a general similarity coefficient in which primary variables could be associated with sets of secondary variables, in turn associated with tertiary variables, and so on. More generally, we may have multiphase sampling, where the variables themselves have a nested structure, and multistage sampling,

with the sample units at each stage being described by their own variables. This level of sophistication is not known in classification work but perhaps it should be given some attention.

Many methods of classification are based on a measure of distance or dissimilarity between OUs. Overwhelmingly, the same definition of dissimilarity is assumed for all pairs of OUs but Friedman and Meulman (2004) recognise the possibility of using a different metric within, and by implication, between groups.

3. Probabilistic and non-probabilistic classification

Probabilistic methods such as discriminant analysis for assigning to classes and mixture problems for constructing classes are familiar in the statistical literature. Non-probabilistic methods are less familiar – at least to statisticians. They usually pertain to structured data with $n_k = 1$.

Identification keys constructed by *ad hoc* methods have been known to botanists for several centuries and foreshadow recent developments, such as Regression Trees and certain aspects of Expert Systems. To use a key a single object has to be identified, that is it has to be named. The simplest thing to do is to compare the object with each row of $\boldsymbol{X}$ until a match is found, when identification is achieved. This is inefficient because an enormous number of comparisons may have to be made, and it may be impracticable because variables included in $\boldsymbol{X}$ may be unavailable (e.g. a flowering plant may give no information on the characteristics of its seeds). Diagnostic keys overcome these difficulties. A key is essentially a tree with one binary variable associated with each node. The value of the binary variable for the object to be identified determines which of the two possible branches of the tree one traverses to reach the next node. In this way one answers a series of questions until one reaches an end-point of the tree, where the identification is given. Many interesting problems in constructing keys have been reviewed by Payne and Preece (1980). We may require the tree with fewest nodes or using fewest different among the binary variables or with minimum average numbers of steps to achieve identification. Costs may be associated with ascertaining the values of the binary variables, in which case we may require the key that is cheapest to use. When tests take time (as with some biochemical tests) it may be efficient to do a group of tests simultaneously and then it has to be decided how best to group the tests. Probability concepts are irrelevant for these interesting problems but may enter if we recognise that the OUs have different frequencies of occurrence, so affecting average numbers of steps to identification and average costs. However, any probabilistic distribution associated with the binary variables themselves is neither required nor material.

To construct classifications with binary variables, we can classify the K objects into $M<K$ groups in such a way that on being told that an object belongs to one of these groups, more correct statements about the likely value of its binary variables can be made than for any other classification. This is maximal predictive classification (Gower, 1975) which models the dictum of a distinguished botanical taxonomist, Gilmour (1937), that *a system of classification is the more natural the more propositions there are that can be made regarding its constituent classes*. No probability distribution is associated with the binary variables, although again the relative frequencies of the objects may be accommodated. The maximal predictive classes have optimal assignment properties. In general, optimal future assignment is a valid criterion for constructing classes.

We may apply the K-means algorithm to numerical data with $n_k = 1$, thus forming homogeneous groups among the $N = K$ OUs without any appeal to probability. In this approach, we would ignore the unknown within-group variation, which may or may not be a reasonable thing to do. When $\boldsymbol{X}$ is regarded as a multivariate data-matrix (with $K = 1$ and $n_1 = K$) then the K-means algorithm finds the maximum likelihood solution to the mixture problem modelled as a combination of multinormal populations. Thus, the same algorithm may be used to compute solutions to two different classification problems, one probabilistic and the other not. Of course, different mixture models have different maximum likelihood solutions whereas the K-means algorithm is only fully valid for multinormal mixtures. A similar distinction

occurs with principal components analysis (PCA). On the one hand, multivariate PCA is based on multinormal assumptions and is concerned with deriving significance tests for zero eigenvalues, thus identifying subspaces, perhaps characterised by reified latent variables, while on the other hand, a PCA of differentiated OUs is concerned with approximating the distances by configurations in few dimensions. Any kind of multidimensional scaling (MDS) of undifferentiated samples seems of little interest.

Many, mostly heuristic, algorithms applied to differentiated OUs give a hierarchical classification of the K OUs. Heuristic algorithms are acceptable when there is no practicable way of optimising an objective criterion (e.g. NP complete problems). Ultrametrics and additive trees give objective criteria for fitting trees to dissimilarity data by least squares, thus providing well-defined models. The use of least-squares does not necessarily imply an appeal to probability. One may note the eighteenth century work of mathematicians like Legendre and Laguerre who used L_1, L_2, or L_∞ norms to approximate complicated functions (e.g. Bessel functions) by polynomials. Polynomials give an acceptable approximation to the function - probability is irrelevant. Analogously, we may regard non-probabilistic classifications as giving similar approximations to $\boldsymbol{X}$ where the goodness of fit may be used to assess the adequacy of the tree approximation or of the class-predictors fitted to maximise prediction. Similarly, MDS approximates a dissimilarity matrix. Such measures of approximation are at least as useful as significance tests associated with probabilistic models; *significant* is not synonymous with *important* and *not significant* is not synonymous with *unimportant*.

Other non-probabilistic objectives for classification concern mixtures of hierarchical and non hierarchical organisation. We may wish to classify the K groups into $M<K$ classes arranged hierarchically with each class containing undifferentiated members. Indeed, it seems to us that this is more frequently required than full hierarchical classification but it rarely gets mentioned, except when pointing out that branches of a full tree may be amalgamated, perhaps governed by specifying some threshold level in a dendrogram. Again with a multivariate data-matrix we could formulate a variant of the mixture problem in which the M classes are to be arranged hierarchically. It is strange that the specification of M in a K means classification into M disjoint classes is fully accepted, the corresponding problem for nested classes is not. Gower (1975) gives a method for constraining maximal predictive classes to have hierarchical organisation which immediately extends to any classification criteria C_m into m classes. One only has to optimise $\sum_{m=1}^{M} W_m C_m$ where the classes are constrained to be hierarchically arranged and W_m is an optional weighting function, perhaps a function of m. Although the computational problems of optimising this criterion, as with other objective classification criteria, are formidable, it easily allows different hierarchical classifications to be compared.

4. The Classification

	OUs	*Assignment*	*Construction*
-probabilistic	*Structured*	Matching Diagnostic keys	Maximal predictive classes *Non* Cluster analysis: *M*-groups, hierarchic, other.
Probabilistic	*Unstructured*	(Null)	Mixture problems
	Structured	Discrimination	Undeveloped

Table 1: Types of classification problem depending on whether (i) the problem is probabilistic or non-probabilistic, (ii) is for assignment to classes or construction of

classes, or (iii) is concerned with structured or unstructured OUs. (Simplification of a similar table in Gower, 1998)

We are now in a position to show the classification of classification methods. We have a three-way cross classification as shown in Table 1. The classifying factors are (i) whether the method is probabilistic or not (ii) whether the problem is one of forming classes or one of assigning to classes and (iii) whether the data is structured or not. We have seen that non-probabilistic methods depend only on structured data so the *unstructured, non-probabilistic* classification does not exist. Similarly, the *probabilistic, unstructured, assignment* problem is null because there are no named classes to assign to. The statistical literature is mostly concerned with the two cells (i) *probabilistic, structured, assignment* and (ii) *probabilistic, unstructured, construction*. There is a vast classification literature on *non-probabilistic, structured, construction* problems.

There remains the cell labelled *probabilistic, structured, construction*. This problem seems not to be addressed in the literature, yet it is very interesting. Suppose we wish to classify K normal populations into $M<K$ groups. Following discrimination ideas, we could seek boundaries which minimise the overlap between the M groups, thus minimising future errors of misclassification. With limiting point-densities *any* set of boundaries give no overlap and no possibility of misclassification. Nevertheless, it remains reasonable to require a grouping of the populations into M classes, possibly nested, by grouping together pairs of point-populations that are closer than others, as judged by distances based on the non-probabilistic information contained in the values of the variables. When the point densities expand to conventional distributions, it seems that a combination of probabilistic and non-probabilistic information should be used for constructing classifications; assignment to these classes might be entirely probabilistic.

By focussing attention on different types of data structure, some of the issues that underlie probabilistic and non-probabilistic problems have been brought into focus. I hope to have shown that many non-probabilistic classification problems are relevant and interesting in their own right and should not be regarded, as they sometimes are, as heuristics for a more desirable fully stochastic formulation. Further, the non-probablilistic formulations support a renewed interest into eighteenth century ideas of approximation for assessing fit, in addition to statistical ideas based on stochastic variability.

The classification of Table 1 is oversimplified. For example, we have already mentioned that one purpose for forming classes is so that future assignment to these classes is optimal. Also, in discriminant analysis we may derive discriminant functions from a training sample and validate the process by assigning the remaining samples. Thus, the cells of Table 1 are not independent. Nevertheless, I hope that the classification will be of some use in focussing attention on and clarify major issues arising in classification n theory.

5. Bibliographie

[FRE 04] FRIEDMAN J., MEULMAN, J., Clustering objects on subsets of attributes (with discussion), *J. R. Statist. Soc. B.,* vol. 66, 2004, p. 1-25.

[GOW 71] GOWER J., A general coefficient of similarity and some of its properties, *Biometrics,* vol. 27, 1971, p. 857-871.

[GOW 75] GOWER J., Maximal predictive classification, *Biometrics,* vol. 30, 1975, p. 634-654.

[GOW 98a] GOWER J., Classification, overview, in: *Encyclopaedia of Biostatistics*, Armitage, P., Colton, T. (Eds.), Wiley, Chichester p. 656-667.

[GOW 98b] GOWER J., ROSS, G., Non-probabilistic classification, in: *Advances in Data science and Classification*, Rizzi, A. Vichi, M., Bock, H.-H. (Eds.), Springer, Berlin, p. 21-28.

[PAY 80] PAYNE J., PREECE, D., Identification keys and diagnostic tables: a review (with discussion), *J. R. Statist. Soc. A.,* vol. 143, 1980, p. 253-292.

[Pea 98] PEARSON K., Mathematical contributions to the theory of evolution, V. On the reconstruction of the stature of prehistoric races. *Philosophical Transactions of the Royal Society of London, Series A,* 192, 1898, 169-244.

[SNE 73] SNEATH P., SOKAL R., *Numerical Taxonomy*, Freeman, San Francisco, 1973.

Les méthodes de classification et de détermination du nombre de classes : du classique au symbolique

André HARDY

Unité de Statistique
Département de Mathématique
Université de Namur
8 Rempart de la Vierge
B - 5000 Namur Belgique
andre.hardy@fundp.ac.be

RÉSUMÉ. *Le but de cet exposé est de montrer comment certaines méthodes de classification et de détermination du nombre de classes classiques ont pu être étendues en des méthodes "symboliques". On insistera plus particulièrement sur les travaux effectués dans l'équipe de statistique de l'Université de Namur : les méthodes classiques et symboliques de classification basées sur les processus de Poisson homogène et non homogène, le module de détermination du nombre de classes symbolique NBCLUST. Des applications à des ensembles de données symboliques, artificielles et réelles illustrent le travail.*

MOTS-CLÉS : *Classification, Détermination du nombre de classes, Objet symbolique, Processus de Poisson, Enveloppe convexe, Critère des Hypervolumes*

1. Le problème de classification

Le problème de classification auquel nous nous intéressons est le suivant.

$E = \{x_1,\ x_2,\ ...,\ x_n\}$ est un ensemble de n objets sur lesquels on mesure la valeur de p variables $Y_1,\ Y_2,\ ...,\ Y_p$. Nous recherchons une partition $P = \{C_1,\ C_2,\ ...,\ C_k\}$ de l'ensemble E des objets en k classes.

2. Les méthodes de classification classiques

2.1. *Les méthodes basées sur une matrice de dissimilarité*

Nous considérerons en premier lieu quatre méthodes de classification hiérarchiques agglomératives classiques bien connues : les méthodes du lien simple, du lien complet, de la moyenne et de Ward, et une méthode de partitionnement : la méthode des Nuées dynamiques [CEL 89].

2.2. *Les méthodes basées sur les processus de Poisson*

2.2.1. *Introduction*

Pour éviter le choix (bien souvent arbitraire) d'une distance ou d'une dissimilarité en classification, nous utilisons des méthodes statistiques basées sur les processus de Poisson homogène et non homogène [RAS 96],

[KAR 91]. Le point de départ de ces approches est le problème suivant : "Etant donné la réalisation d'un processus de Poisson homogène dans un domaine convexe compact D, estimer D en utilisant des méthodes d'inférence statistique". La solution de ce problème fut trouvée par Rasson et Ripley [RIP 77]. L'estimateur du maximum de vraisemblance du domaine D, qui est également une statistique exhaustive pour D, est l'enveloppe convexe des points. L'estimateur non biaisé correspondant est une dilatation de l'enveloppe convexe à partir de son centre de gravité.

2.2.2. *La méthode de classification et le critère des Hypervolumes*

Sur base du résultat précédent, une première méthode de classification automatique fut élaborée [HAR 82], [HAR 83]. Elle suppose que les points observés sont générés par un processus de Poisson homogène dans un domaine D de R^p, où D est l'union de k domaines convexes compacts disjoints $D_1,\ D_2,\ ...,\ D_k$; $C_i \subset \{x_1,\ x_2,\ ...,\ x_n\}$ est le sous-ensemble des observations appartenant à $D_i\ (1 \leq i \leq k)$. Le problème revient à estimer les domaines inconnus D_i dans lesquels les points ont été générés. Les estimateurs du maximum de vraisemblance des k domaines inconnus $D_1,\ D_2,\ ...,\ D_k$ sont les k enveloppes convexes $H(C_i)$ des k sous-groupes C_i de points tels que la somme des mesures de Lebesgue des enveloppes convexes disjointes $H(C_i)$ est minimale. Le critère des Hypervolumes est alors défini par

$$W_k = \sum_{i=1}^{k} m(H(C_i))$$

où $m(H(C_i))$ est la mesure de Lebesgue multidimensionnelle de l'enveloppe convexe des points appartenant à C_i.

2.2.3. *Une méthode polythétique divisive basée sur le processus de Poisson homogène*

Un des inconvénients souvent cité pour des algorithmes divisifs de classification est qu'une partition obtenue à un niveau de la procédure n'est jamais remise en cause. L'algorithme divisif de classification polythétique basé sur le critère des Hypervolumes élaboré par A. Hardy [HAR 96a] , propose une solution à ce problème. La première partie de l'algorithme est une procédure hiérarchique divisive qui produit une partition de l'ensemble des données en k classes (k fixé). La solution obtenue à la fin de cette première partie ne correspond pas toujours à la structure "naturelle" des données. La seconde partie de l'algorithme est basé sur la propriété d'admissibilité par rapport à l'omission de classes de Fisher et Van Ness [FIS 71]. Elle consiste en une procédure de "recollement-division" qui améliore, lorsque c'est possible, la partition obtenue à la fin de la première étape de l'algorithme, et la valeur correspondante du critère à optimiser. Une approche comparable a été utilisée plus tard par Pirçon [PIR 04].

2.2.4. *Cinq nouvelles méthodes de classification monothétiques*

J.-Y. Pirçon [PIR 04] propose cinq nouvelles méthodes de classification monothétiques divisives. Elles sont toutes basées sur le processus de Poisson. Une première méthode (HOPP) est développée en faisant l'hypothèse que les points sont la réalisation d'un processus de Poisson homogène. Les quatre autres méthodes font l'hypothèse que les points sont la réalisation d'un processus de Poisson non homogène (UNHOPPHI et UNHOPPKI), ou d'une superposition de processus de Poisson non homogènes (SONHOPPHI et SONHOPPKI). Dans ces quatre derniers cas l'intensité du processus de Poisson doit être estimée. Deux estimations d'intensité non paramétriques sont utilisées : les histogrammes et les noyaux.

Un arbre est construit. Le critère de coupure est obtenu par la méthode du maximum de vraisemblance. Il s'agit de trouver la variable pour laquelle le "vide" dans les données selon cette variable est maximum. Pour trouver le meilleur sous-arbre de l'arbre construit, un processus d'élagage est nécessaire. Deux critères sont proposés. Un premier est basé sur l'inertie, et l'autre sur le Gap test [KUB 96], qui sera développé dans le paragraphe suivant.

Les cinq méthodes proposées sont monothétiques et les coupures sont faites perpendiculairement aux axes. Dans certains cas, une classe compacte est divisée en plusieurs parties. Une des originalités des méthodes proposées par Pirçon est d'inclure à la fin de la procédure une étape de "recollement" qui a pour objet de tester si deux

feuilles de l'arbre doivent être fusionnées. Une condition nécessaire de recollement est que les deux groupes soient connexes. Les critères utilisés pour le recollement sont à nouveau le critère de l'inertie et le Gap test.

La méthode UNHOPPKI a été étendue en une méthode de classification symbolique, appelée SCLASS. Elle sera présentée dans la deuxième partie de ce papier.

3. Les méthodes de détermination du nombre de classes

3.1. *Méthodes basées sur une matrice de dissimilarité*

De nombreuses méthodes de détermination du nombre de classes utilisent principalement une matrice de dissimilarité entre les objets. Nous considérerons tout d'abord cinq méthodes classiques de détermination du nombre de classes, les "meilleures" du classement de Milligan et Cooper [MIL 85] : la méthode de Caliński et Harabasz [CAL 74], l'index J [DUD 73], l'index C [HUB 76], l'index Γ [BAK 75] et le test de Beale [BEA 69].

3.2. *Méthodes basées sur les processus de Poisson*

Soit $x = (x_1,\ x_2,\ ...,\ x_n)$ un échantillon aléatoire généré par un processus de Poisson homogène dans k domaines convexes compacts disjoints $D_1,\ D_2,\ ...,\ D_k$ d'un espace Euclidien à p dimensions.

3.2.1. *Un test du quotient de vraisemblance*

Pour un entier $k \geq 2$, on teste l'hypothèse nulle d'une structure naturelle en k classes contre l'alternative d'une structure en $k-1$ classes. La statistique du test est déduite du modèle statistique par la méthode du quotient de vraisemblance. Elle est donnée par [HAR 96b]

$$S(x) = \frac{W_k}{W_{k-1}}.$$

Le test est réalisé de manière séquentielle. Si k_0 est la première valeur de $k \geq 2$ pour laquelle on rejette H_0, alors on considérera $k_0 - 1$ comme le nombre approprié de classes naturelles.

3.2.2. *Le Gap test*

Notons par C l'ensemble des points et par $P = \{C_1, C_2\}$ une partition de C en deux classes. On teste les hypothèses suivantes :

- H_0 : les $n = n_1 + n_2$ points sont une réalisation du processus de Poisson homogène dans le domaine D
- H_1 : n_1 points sont la réalisation d'un processus de Poisson homogène dans le domaine D_1 et n_2 points dans D_2 où $D_1 \cap D_2 = \emptyset$.

Nous développons le test dans le cas unidimensionnel. Une description dans le cas multidimensionnel est faite dans [KUB 96].

Un test du quotient de vraisemblance donne [RAS 94] :

$$Q(x) = \frac{max\ L_{H_0}(x)}{max\ L_{H_1}(x)} = \left(1 - \frac{m(\triangle)}{m(D)}\right)^n$$

où $\triangle$ est le plus grand "vide" dans les données et $m(D)$ la mesure de Lebesgue du domaine D.

Par conséquent, la région critique du Gap test, au niveau α, est donnée par

$$W_\alpha = \{x : \frac{m(\triangle)}{m(D)} \geq t_\alpha\}.$$

Le seuil t_α est obtenu par Kubushishi en utilisant des lois limites ; il est donné par $t_\alpha = -\log(-\log(1-\alpha))$.

Les méthodes de détermination du nombre de classes de Milligan et Cooper, le test des Hypervolumes et le Gap test ont été appliqués et évalués sur des ensembles de données artificielles et réelles [HAR 96b], [BEA 02].

4. Les données symboliques

Considérons un ensemble d'objets $E = \{x_1, ..., x_n\}$ sur lesquels on mesure un ensemble de p variables symboliques $Y_1, ..., Y_p$. Ces variables peuvent être des variables intervalles, multivaluées ou modales [BOC 00]. La plupart des méthodes de classification utilisent une matrice de dissimilarité, qui reflète la structure de l'ensemble des n objets symboliques. Un module implémentant des méthodes de détermination du nombre de classes pour des données symboliques, appelé NBCLUST, a été intégré dans la méthode de classification symbolique SCLUST. C'est pourquoi nous nous concentrerons tout d'abord sur les mesures de dissimilarité utilisées dans le logiciel SCLUST.

4.1. *Variables intervalles*

Soit $E = \{x_1, \ldots, x_n\}$ un ensemble de n objets décrits par p variables intervalles $Y_1, \ldots, Y_p$ de domaines $\mathcal{Y}_1, \ldots, \mathcal{Y}_p$ respectivement. La variable Y_j sera donnée par

$$Y_j : E \to \mathcal{B}_j : x_k \mapsto Y_j(x_k) = x_{kj} = [\alpha, \beta] \subset \mathcal{R}.$$

$\mathcal{B}_j$ est donc l'ensemble des intervalles bornés fermés de $\mathcal{R}$.

On associe p indices de dissimilarité δ_1, ..., δ_p définis sur les ensembles $\mathcal{B}_j$ de manière à obtenir une mesure de dissimilarité globale sur E. Si $x_{kj} = [\alpha_{kj}, \beta_{kj}]$ et $x_{\ell j} = [\alpha_{\ell j}, \beta_{\ell j}]$, on définit les trois distances suivantes pour des variables intervalles.

$$\begin{aligned}
\text{La distance de Hausdorff :} \quad \delta_j(x_{kj}, x_{\ell j}) &= \max\{\,|\,\alpha_{kj} - \alpha_{\ell j}\,|, |\,\beta_{kj} - \beta_{\ell j}\,|\,\} \\
\text{La distance } L_1 : \quad \delta_j(x_{kj}, x_{\ell j}) &= |\,\alpha_{kj} - \alpha_{\ell j}\,| + |\,\beta_{kj} - \beta_{\ell j}\,| \\
\text{La distance } L_2 : \quad \delta_j(x_{kj}, x_{\ell j}) &= (\alpha_{kj} - \alpha_{\ell j})^2 + (\beta_{kj} - \beta_{\ell j})^2.
\end{aligned}$$

On obtient ainsi une mesure de dissimilarité globale sur E

$$d : E \times E \longrightarrow R^+ : (x_k, x_\ell) \longmapsto d(x_k, x_\ell) = \left(\sum_{j=1}^{p} \delta_j^2(x_{kj}, x_{\ell j}) \right)^{1/2}.$$

4.2. *Variables multivaluées*

$E = \{x_1, \ldots, x_n\}$ est un ensemble de n objets décrits par p variables multivaluées $Y_1, \ldots, Y_p$ dont les domaines sont respectivement $\mathcal{Y}_1, \ldots, \mathcal{Y}_p$. $Y_j(x_k)$ est un ensemble de catégories et $\mathcal{B}_j = \mathcal{P}(\mathcal{Y}_j)$. Notons m_j

le nombre de catégories prises par Y_j. Nous transformons la matrice des données initiales en une matrice de fréquences. La fréquence $q_{j,x_k}(c_s)$ associée à la catégorie c_s $(s = 1, \ldots, m_j)$ de $Y_j(x_k)$ est donnée par

$$q_{j,x_k}(c_s) = \begin{cases} \frac{1}{|Y_j(x_k)|} & \text{si} \quad \mathrm{c_s} \in \mathrm{Y_j(x_k)} \\ 0 & \text{sinon.} \end{cases}$$

Les distances L_1 et L_2 sur $\mathcal{B}_j$ sont respectivement définies par :

$$\delta_j(x_{kj}, x_{\ell j}) = \sum_{i=1}^{|\mathcal{Y}_j|} |q_{j,x_k}(c_i) - q_{j,x_\ell}(c_i)| \quad \text{et} \quad \delta_{\mathrm{j}}(\mathrm{x_{kj}}, \mathrm{x_{\ell j}}) = \sum_{\mathrm{i}=1}^{|\mathcal{Y}_{\mathrm{j}}|} (\mathrm{q_{j,x_k}(c_i) - q_{j,x_\ell}(c_i)})^2$$

et la distance de De Carvalho par

$$\delta_j(x_{kj}, x_{\ell j}) = \sum_{i=1}^{|\mathcal{Y}_j|} (\gamma q_{j,x_k}(c_i) + \gamma' q_{j,x_\ell}(c_i))$$

où

* $\gamma = \begin{cases} 1 & \text{si} \quad \mathrm{c_i} \in \mathrm{Y_j(x_k)} \text{ et } \mathrm{c_i} \notin \mathrm{Y_j(x_\ell)} \\ 0 & \text{sinon} \end{cases}$

* $\gamma' = \begin{cases} 1 & \text{si} \quad \mathrm{c_i} \notin \mathrm{Y_j(x_k)} \text{ et } \mathrm{c_i} \in \mathrm{Y_j(x_\ell)} \\ 0 & \text{sinon.} \end{cases}$

On obtient une mesure de dissimilarité globale sur E.

$$d : E \times E \longrightarrow R^+ : (x_k, x_\ell) \longmapsto d(x_k, x_\ell) = \left(\sum_{j=1}^{p} \delta_j^2(x_{kj}, x_{\ell j}) \right)^{1/2}.$$

4.3. *Variables modales*

Le cas des variables modales est semblable à celui des variables multivaluées. Les fréquences $q_{j,x_k}(c_s)$ sont simplement remplacées par les valeurs de la distribution $\pi_{j,k}$ associées à chacune des catégories de $Y_j(x_k)$.

5. Les méthodes de classification symboliques

5.1. *Les méthodes basées sur une matrice de dissimilarité*

L'existence d'une matrice de dissimilarité pour des objets symboliques décrits par des variables intervalles, multivaluées et modales permet l'application directe des algorithmes de classification classiques basés sur des dissimilarités. Ce sera entre autres le cas des méthodes du lien simple, du lien complet, du centroïde et de Ward considérées dans ce travail.

Il en va de même pour la méthode de classification SCLUST [VER 00] disponible dans le logiciel SODAS2. SCLUST est une extension symbolique de la méthode des Nuées Dynamiques [CEL 89]. Elle détermine d'une manière itérative une série de partitions qui améliore, à chaque étape, la valeur d'un critère mathématique.

L'algorithme est basé sur la définition de prototypes pour représenter les classes et d'une fonction de proximité qui assigne les objets aux classes.

Dans le cas de variables intervalles, le prototype de la classe C_ℓ, noté $g^{(\ell)}$, est l'hyperrectangle de gravité de C_ℓ, défini par

$$g^{(\ell)} = \left(\left[\frac{1}{n_\ell} \sum_{x_i \in C_\ell} \alpha_{i1}, \frac{1}{n_\ell} \sum_{x_i \in C_\ell} \beta_{i1} \right], \ldots, \left[\frac{1}{n_\ell} \sum_{x_i \in C_\ell} \alpha_{ip}, \frac{1}{n_\ell} \sum_{x_i \in C_\ell} \beta_{ip} \right] \right).$$

Pour des variables multivaluées, le prototype $g^{(\ell)}$ est défini par

$$g^{(\ell)} = (\frac{1}{n_\ell} \sum_{x_i \in C_\ell} q_{1,x_i}(c_1), \ldots, \frac{1}{n_\ell} \sum_{x_i \in C_\ell} q_{1,x_i}(c_{m_1})), ..., (\frac{1}{n_\ell} \sum_{x_i \in C_l} q_{p,x_i}(c_1), \ldots, \frac{1}{n_l} \sum_{x_i \in C_\ell} q_{p,x_i}(c_{m_p}))).$$

Le cas des variables modales est similaire au cas des variables multivaluées. Les fréquences $q_{j,x_k}(c_s)$ sont simplement remplacées par les valeurs de la distribution $\pi_{j,k}$ associées à chacune des catégories de $Y_j(x_k)$. Les fonctions de proximité qui assignent les objets aux classes sont définies à partir des mesures de dissimilarité définies dans le paragraphe précédent.

5.2. *Une méthode basée sur le processus de Poisson non homogène : SCLASS*

La méthode de classification SCLASS suppose que les points observés sont générés par un processus de Poisson non homogène d'intensité $q(.)$ dans un ensemble D où D est l'union de k domaines convexes disjoints D_i $(i = 1, \cdots, k)$. La fonction de vraisemblance, pour les observations $x = (x_1, ..., x_n)$ avec $x_i \in R^p$ $(i = 1, \ldots, n)$, vaut

$$L_D(x) = \frac{1}{(\rho(D))^n} \prod_{i=1}^{n} I_D(x_i) q(x_i)$$

où

$$\rho(D) = \int_D q(x)dx$$

est l'intensité intégrée du processus et I_D la fonction indicatrice de l'ensemble D.

Si l'intensité du processus est connue, les estimateurs du maximum de vraisemblance des k domaines inconnus D_i seront les k enveloppes convexes $H(C_i)$ des k sous-groupes C_i de points tels que la somme des intensités intégrées est minimale.

L'intensité du processus de Poisson non homogène est estimée par la méthode des noyaux [SIL 81] en utilisant un noyau normal. Le paramètre de lissage choisi est celui pour lequel l'estimation change de l'unimodalité à la multimodalité [SIL 86]. Le critère de coupure est le suivant : pour chaque variable on partitionne C en deux classes C_1 et C_2 de telle manière que la somme des intensités intégrées soit minimale. On retient alors la meilleure variable et la partition correspondante. Un critère d'arrêt classique est utilisé.

Dans le cas de variables intervalles, chaque intervalle est représenté par ses coordonnées (Milieu, Longueur) dans l'espace $(M, L) \subset RXR^+$. La valeur de coupure est obtenue en minimisant

$$\int_{M_i}^{M_{i+1}} \rho_1(m)dm + \int_{min(L_i, L_{i+1})}^{max(L_i, L_{i+1})} \rho_2(l)dl$$

où ρ_1 est l'intensité sur l'axe M et ρ_2 l'intensité sur l'axe L.

SCLASS est donc une méthode de classification hiérarchique monothétique divisive. Le résultat est un arbre de décision où chaque classe correspond à un objet symbolique.

6. Les méthodes de détermination du nombre de classes symboliques

Différentes approches ont été analysées.

- Nous avons appliqué les cinq règles d'arrêt issues de l'analyse de Milligan et Cooper aux hiérarchies de partition fournies par les quatre méthodes de classification hiérarchiques, en utilisant la matrice de dissimilarité calculée dans le cadre de ces méthodes hiérarchiques [TRO 04].
- SCLUST n'est pas une méthode de classification hiérarchique. Parmi les cinq règles de Milligan et Cooper, seuls la méthode de Calinski et Harabasz, l'indice C et l'index Γ sont applicables à des ensembles de partitions non emboîtées en ℓ classes $(1 \leq \ell \leq K)$ où K est une constante fixée par l'utilisateur [HAR 04], [HAR 02b], [HAR 02a].
- Le test des Hypervolumes est basé sur le calcul d'enveloppes convexes de points ; il ne requiert pas la connaissance d'une matrice de dissimilarité, mais seulement la position des points. Ces positions sont connues dans chacune des p représentations Milieu-Longueur. Nous sélectionnons parmi les p variables intervalles celle qui contribue le plus à l'inertie de l'ensemble des objets symboliques. Nous retenons alors le nombre de classes donné par le test des Hypervolumes associé à cette variable.

7. Exemples

Les différentes méthodes de classification et de détermination du nombre de classes ont été appliquées à des ensembles de données symboliques simulées tests, mais également à des ensembles de données réelles : les huiles d'Ichino, les boucles mérovingiennes, les magasins "e-fashion stores", ...

8. Conclusion

Nous avons voulu montrer, dans cet exposé, comment certaines méthodes classiques de classification et de détermination du nombre de classes pouvaient être étendues en des méthodes symboliques, en insistant sur les recherches effectuées dans l'unité de statistique de l'Université de Namur. Parmi les autres contributions importantes dans le domaine, soulignons la méthode symbolique de classification hiérarchique monothétique divisive DIV [CHA 97], la méthode de classification hiérarchique et pyramidale HIPYR [BRI 00] ainsi que des extensions importantes de la méthode de classification dynamique [VER 00], [VER 04].

Le logiciel SODAS2 a été développé dans le cadre du projet européen ASSO (Analysis System of Symbolic Official Data). Il intègre un nombre important de méthodes d'analyse des données symboliques parmi lesquelles on trouve les méthodes de classification symboliques SCLUST (dans laquelle est intégrée le module de détermination du nombre de classes NBCLUST), DIV, HIPYR et SCLASS.

9. Bibliographie

[BAK 75] BAKER F., HUBERT L., Measuring the power of hierarchical cluster analysis, *Journal of the American Statistical Association*, vol. 70, 1975, p. 31-38.

[BEA 69] BEALE E., Euclidean cluster analysis, *Bulletin of the International Statistical Institute*, vol. 43, 2, 1969, p. 92-94.

[BEA 02] BEAUTHIER C., Comparaison entre le Gap test et le test des Hypervolumes en classification, 2002, Mémoire, Université de Namur.

[BOC 00] BOCK H.-H., DIDAY E., *Analysis of Symbolic Data. Exploratory Methods for Extracting Statistical Information from Complex Data.*, Springer Verlag, 2000.

[BRI 00] BRITO P., Hierarchical and Pyramidal Clustering with Complete Symbolic Objects, *Analysis of Symbolic Data Analysis, H.-H. Bock and E. Diday (Eds)*, Studies in Classification, Data Analysis, and Knowledge Organization, 2000, p. 312–323.

[CAL 74] CALINSKI T., HARABASZ J., A dendrite method for cluster analysis, *Communication in Statistics*, vol. 3, 1974, p. 1-27.

[CEL 89] CELEUX G., DIDAY E., GOVAERT G., LECHEVALLIER Y., RALAMBONDRAINY H., *Classification Automatique des Données*, Dunod, 1989.

[CHA 97] CHAVENT M., Analyse des Données Symboliques : Une méthode divisive de classification, PhD thesis, Université Paris IX-Dauphine, 1997.

[DUD 73] DUDA R., HART P., *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.

[FIS 71] FISHER L., VAN NESS J., Admissible Clustering Procedures, *Biometrika*, vol. 58, 1971, p. 91–104.

[HAR 82] HARDY A., RASSON J.-P., Une nouvelle approche des problèmes de classification automatique, *Statistique et Analyse des données*, , 1982, p. 41–56.

[HAR 83] HARDY A., Statistique et Classification Automatique : un modèle - un nouveau critère - des algorithmes - des applications, PhD thesis, Université de Namur, 1983.

[HAR 96a] HARDY A., A heuristic approach for the Hypervolumes method in cluster analysis, *Jorbel*, vol. 36, 1996, p. 43–55.

[HAR 96b] HARDY A., On the Number of Clusters, *Computational Statistics and Data Analysis*, vol. 23, 1996, p. 83–96.

[HAR 02a] HARDY A., LALLEMAND P., Determination of the number of clusters for symbolic objects described by interval variables, *Studies in Classification, Data Analysis and Knowledge Organisation*, , 2002, p. 311-318.

[HAR 02b] HARDY A., LALLEMAND P., LECHEVALLIER Y., La determination du nombre de classes pour la méthode de classification symbolique SCLUST, *Actes des huitièmes rencontres de la Société Francophone de Classification*, 2002, p. 27–31.

[HAR 04] HARDY A., LALLEMAND P., Clustering of Symbolic Objects described by multi-valued and modal variables, *Proceedings IFCS 2004*, , 2004.

[HUB 76] HUBERT L., LEVIN J., A general statistical framework for assessing categorical clustering in free recall, *Psychological Bulletin*, vol. 83, 1976, p. 1072-1080.

[KAR 91] KARR A. F., *Point Processes and their Statistical Inference*, Marcel Dekker, Inc., 1991.

[KUB 96] KUBUSHISHI T., On some Applications of the Point Process Theory in Cluster Analysis and Pattern Recognition, PhD thesis, Université de Namur, 1996.

[MIL 85] MILLIGAN G., COOPER M., An examination of procedures for determining the number of clusters in a data set, *Psychometrika*, vol. 50, 1985, p. 159-179.

[PIR 04] PIRÇON J.-Y., Le clustering et les processus de Poisson pour de nouvelles méthodes monothétiques, PhD thesis, Université de Namur, 2004.

[RAS 94] RASSON J.-P., KUBUSHISHI T., The Gap Test : an Optimal Method for Determining the Number of Natural Classes in Cluster Analysis, DIDAY E., LECHEVALLIER Y., SCHADER M., BERTRAND P., BURTSCHY B., Eds., *New Approaches in Classification and Data Analysis*, 1994, p. 186–193.

[RAS 96] RASSON J.-P., GRANVILLE V., Geometrical tools in classification, *Computational Statistic and Data Analysis*, vol. 23, 1996, p. 105–123.

[RIP 77] RIPLEY B., RASSON J.-P., Finding the edge of a Poisson Forest, *Journal of Applied Probability*, , 1977, p. 483–491.

[SIL 81] SILVERMAN B. W., Using Kernel Density Estimates to Investigate Multimodality, *Journal of Royal Statistical Society, B*, vol. 43, 1981, p. 97–99.

[SIL 86] SILVERMAN B., *Principal Component Analysis*, Springer-Verlag, 1986.

[TRO 04] TROCLET J., Méthodes de détermination du nombre de classes pour des objets symboliques, 2004, Mémoire, Université de Namur.

[VER 00] VERDE R., CARVALHO F. D., LECHEVALLIER Y., A Dynamical Clustering Algorithm for Multi-Nominal Data, *Data Analysis, Classification, and Related Methods, H.A.L. Kiers, J.-P. Rasson, P.J.F. Groenen and M. Schader (Eds.)*, Springer Verlag, Heidelberg, 2000, p. 387–394.

[VER 04] VERDE R., Clustering Methods in Symbolic Data Analysis, *Proceedings IFCS 2004* , 2004.

Transformation de longues séries temporelles en descriptions symboliques

Georges Hébrail

ENST Paris, LTCI-UMR 5141 CNRS
Département Informatique et Réseaux
46, Rue Barrault
75013 Paris, France

RÉSUMÉ. De nombreuses longues séries temporelles sont disponibles sous forme informatique (courbes de consommation d'électricité, d'eau, de gaz, courbes de ventes des grandes surfaces, courbes de trafic automobile, …). Leur forme brute (succession de données numériques) est peu adaptée aux traitements de fouille de données où l'on cherche à extraire une information de haut niveau. Cette communication décrit des méthodes de construction de descriptions symboliques de longues séries temporelles. Deux approches sont exposées : la segmentation en épisodes à niveaux constants, et la classification automatique d'épisodes de durée fixe. Dans une dernière partie, il est montré que les descriptions symboliques obtenues peuvent trouver de nombreuses applications.

MOTS-CLÉS : Séries temporelles, courbes, représentation symbolique, classification automatique.

1. Introduction

De nos jours, la plupart des activités humaines sont assistées par des systèmes informatiques qui permettent, au-delà de la fonction assurée, de disposer d'une trace chronologique de ces activités. Par exemple, les entreprises de distribution d'électricité, d'eau, de gaz disposent de courbes de consommation de leurs clients, courbes initialement générées pour les opérations de facturation. De même, les banques disposent des historiques des soldes des comptes de leurs clients, les opérateurs de télécommunications disposent de courbes de consommations téléphoniques, les grandes surfaces de distribution disposent de courbes de ventes magasin par magasin, produit par produit, au fil du temps. D'autres sources d'informations de nature chronologique peuvent être générées par des mesures réalisées périodiquement, comme par exemple les mesures de trafic automobile dans les grandes agglomérations ou sur les grands axes. La Figure 1 donne un exemple de courbe de consommation d'électricité d'une entreprise sur une durée de trois mois.

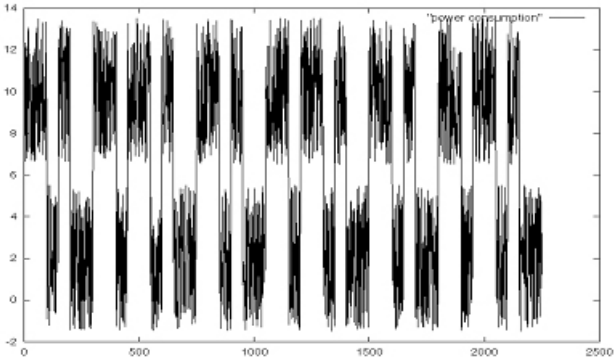


Figure 1 : Longue série temporelle

Dans cette communication, on s'intéresse à l'analyse exploratoire d'une ou plusieurs longues séries temporelles de type numérique, typiquement des séries d'une année, comportant un point par heure. L'objectif

est de synthétiser cet ensemble de séries temporelles, afin de mieux comprendre les comportements sous-jacents : le plus souvent il s'agit du comportement de clients d'entreprises, dans une optique marketing.

Nous proposons de transformer chaque série temporelle en une séquence de symboles d'un alphabet de faible taille (typiquement une dizaine de symboles), constituant ainsi une information résumée de la série, directement interprétable par l'humain ou pouvant être traitée par des techniques de fouilles de données. Deux approches complémentaires ont été étudiées dans la thèse de Bernard Hugueney [HUG 03] :

- La segmentation de la série en épisodes contigus de durée variable, chaque épisode étant alors représenté par une courbe constante à un niveau prenant ses valeurs dans un ensemble discret de valeurs. Les symboles correspondent dans ce cas aux différents niveaux discrets.
- Le découpage de la série en épisodes contigus de même durée (par exemple les jours), et l'association à chaque épisode d'une forme type obtenue par classification automatique des formes observées sur une ou plusieurs séries. Les symboles correspondent dans ce cas aux formes type.

La Section 2 présente l'approche par segmentation et la Section 3 présente l'approche par classification automatique. La Section 4 décrit les applications possibles des représentations symboliques des longues séries temporelles.

2. Représentation symbolique par segmentation

L'approche de base consiste à découper la série en *P* épisodes contigus (encore appelés segments) et à approcher la série par une fonction constante sur chacun des segments, où *P* est un paramètre de l'algorithme fixé à l'avance. L'écart entre la série initiale et son approximation par des segments à niveau constant est évalué par la somme des erreurs quadratiques en chacun des points de la série. Le problème posé est donc la détermination de *P-1* points de coupure de la série, minimisant la somme des erreurs quadratiques. A l'optimum, chaque segment est approché par une fonction constante dont le niveau est égal à la moyenne des points du segment. Ce problème trouve sa solution exacte par programmation dynamique ([BEL 61]) avec une complexité en $O(PN^2)$, où *N* est le nombre de points de la série à segmenter. Des algorithmes approchés ont été proposés ([KEO 01]), permettant de réduire la complexité à $O(ln(P)N)$ par exemple.

Des critères ont été proposés et évalués pour choisir le nombre d'épisodes *P* (voir [HUG 03]). Le choix de *P* est fortement lié à l'utilisation qui est faite du résultat de la segmentation. La Figure 2 montre le résultat d'une segmentation avec *P=6*.

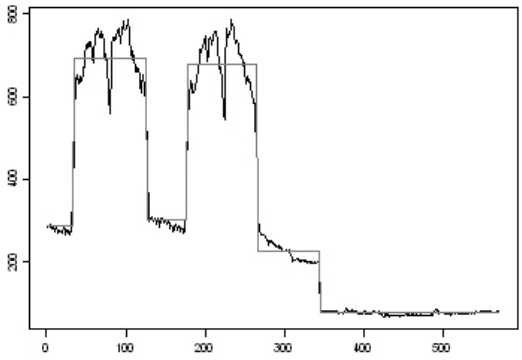


Figure 2 : Segmentation d'une série

La segmentation décrite ci-dessus définit, pour chaque épisode, un niveau constant égal à la moyenne des valeurs de l'épisode. Les différents épisodes ont donc des niveaux tous (ou presque tous) différents. Dans [HUG 02] et [HUG 03], il est proposé de rechercher les points de coupure de la série de telle sorte que le niveau des segments ne prenne sa valeur que dans une liste prédéfinie de niveaux discrets fixés à l'avance (environ une dizaine). On parle alors de segmentation *prototypique*. On peut donc considérer dans ce cas que la série est transformée en une séquence de symboles, chaque symbole correspondant à l'un des niveaux discrets définis à l'avance. La complexité de l'algorithme optimal est alors en $O(KPN^2)$, où *K* est le nombre de niveaux discrets de l'alphabet. Les niveaux discrets peuvent être définis par une classification automatique de l'ensemble des valeurs

prises par la ou les séries temporelles à traiter. La Figure 3 donne un exemple de segmentation prototypique de la série de la Figure 2, avec *K=3 et P=6*.

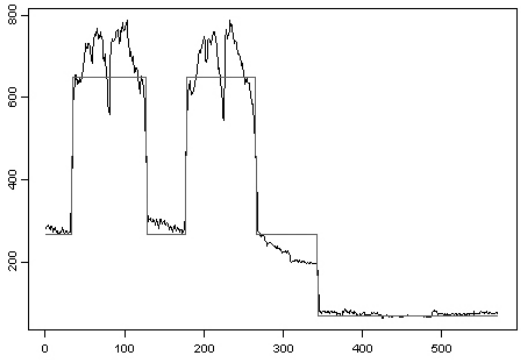


Figure 3 : Segmentation prototypique d'une série

3. Représentation symbolique par classification automatique

Dans l'approche par segmentation, la durée des épisodes est variable, mais la forme de la courbe au sein de chaque épisode est imposée. L'autre approche explorée dans [HUG 03] consiste à donner plus de richesse à la forme de la courbe au sein de chaque épisode, mais en imposant que la durée de tous les épisodes soit la même. Cette durée est supposée fixée par l'utilisateur (par exemple une journée ou une semaine dans la pratique), ainsi que le point de la série où commence le premier épisode (*offset*).

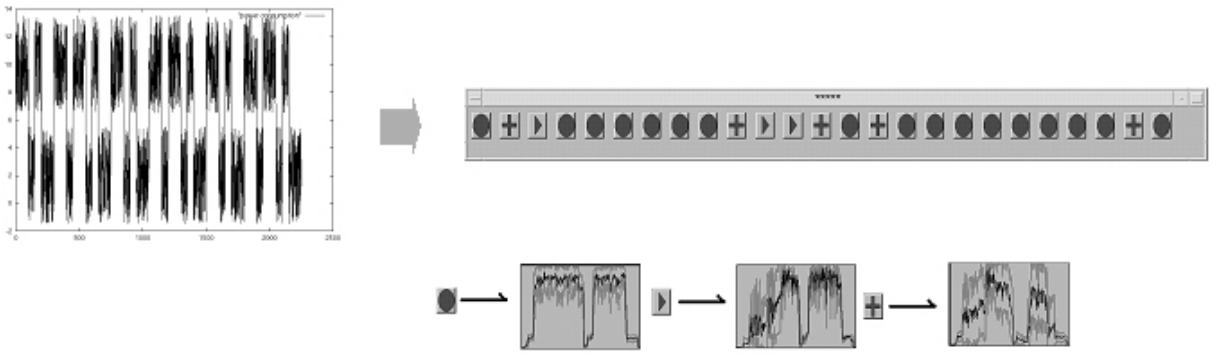


Figure 4: Représentation symbolique par classification automatique

Le principe de construction de la description symbolique par classification automatique est le suivant :

- La ou les séries à transformer sont découpées en épisodes de durée fixe, à partir de l'offset. On obtient donc un ensemble de « petites » courbes de même durée.
- Une classification automatique est réalisée sur l'ensemble des courbes des épisodes extraits dans l'étape précédente. Les classes obtenues définissent les symboles de l'alphabet, en leur associant la courbe moyenne de la classe.
- Les séries initiales sont transformées en une séquence de symboles (chaque épisode est représenté par le symbole de la classe à laquelle sa courbe appartient), comme l'illustre la Figure 4. Chaque épisode peut également être décrit par des attributs temporels (comme le jour, le mois, la saison, …).

Dans ses expérimentations, B.Hugueney a utilisé les cartes auto-organisatrices de Kohonen ([KOH 95]) comme méthode de classification automatique, mais d'autres méthodes peuvent être employées, comme les méthodes de la famille des nuées dynamiques.

Différents paramètres sont à régler pour construire les descriptions symboliques par classification automatique :

- La durée des épisodes : celle-ci peut être guidée par une analyse fréquentielle de la série par transformation de Fourier.
- L'offset depuis le début de la série : celui-ci peut être choisi arbitrairement en début de journée (0h) ou de semaine, ou peut être optimisé, par exemple en recherchant l'offset dont tous les points anniversaire sont de variance minimum.
- Le nombre de classes de la classification automatique (qui correspond au nombre de symboles) : il peut être guidé par les indicateurs standard en classification automatique, mais il est préférable de faire intervenir l'utilisateur pour valider les classes et supprimer éventuellement les classes correspondant à des individus atypiques. Dans tous les cas, le nombre de classes doit rester petit (en général une dizaine) afin que la description symbolique de la série reste intelligible.

4. Applications des descriptions symboliques

La représentation symbolique d'une série temporelle n'est intéressante que si le nombre de symboles différents reste faible et si chaque symbole porte une information de haut niveau conceptuel. Les niveaux prototypiques d'une segmentation et les formes type d'une représentation par classification sont des bons candidats pour définir des symboles de haut niveau conceptuel, mais il faut veiller à ce que chaque symbole retenu trouve une interprétation intuitive auprès de l'utilisateur. Par exemple, pour des séries de trafic automobile, trois niveaux prototypiques peuvent être définis, correspondant à des trafics 'fluide', 'ralenti', et 'congestionné'. Sous la condition que les symboles élaborés lors de la construction des descriptions symboliques aient un sens, de nombreuses applications sont possibles :

- Visualisation des descriptions symboliques : ainsi que le montre la Figure 4, les descriptions symboliques par classification automatique peuvent être facilement visualisées avec une lecture plus facile que la visualisation de la série initiale.

- Statistique descriptive : les techniques standard de statistique descriptive et de visualisation peuvent être appliquées aux descriptions symboliques. Par exemple, sur des séries de trafic automobile, il est immédiat (et informatif) de compter le nombre de jours dans l'année où un axe routier est congestionné.

- Détection d'aléas et correction de valeurs aberrantes : en comparant la série initiale avec sa représentation symbolique, il est possible de détecter des épisodes s'éloignant de leur forme type associée. Ces épisodes peuvent correspondre à des aléas pour lesquels une alarme est déclenchée, ou bien à des défauts de mesures, pour lesquels la forme type permet d'estimer de bonnes valeurs pour les valeurs aberrantes.

- Fouille de données exploratoire : la forme « symbolique » de la représentation symbolique permet d'utiliser facilement la plupart des méthodes de fouille de données exploratoire. Les résultats produits par ces méthodes sont facilement interprétables du fait de l'existence d'une interprétation des symboles. Les méthodes de recherche de séquences fréquentes ([AGR 95]) sont particulièrement adaptées au traitement de séquences de symboles que sont les descriptions symboliques de séries temporelles.

- Fouille de données décisionnelle : les descriptions symboliques de séries temporelles peuvent facilement être utilisées par les méthodes de fouilles de données décisionnelles (arbres de décision, réseaux de neurones, …). Leur caractère symbolique leur permet de pouvoir intervenir dans les modèles de prédiction soit comme prédicteur, soit comme variable à prédire (prévision de séries temporelles).

- Bases de données de séries temporelles : les descriptions symboliques de séries temporelles sont adaptées au stockage et à l'indexation de séries temporelles dans les bases de données. Par exemple, dans une base de données stockant une description symbolique de séries temporelles de trafic automobile, il est facile de répondre à la requête suivante : « trouver les grands axes où le trafic est toujours fluide le week-end ».

5. Conclusion

Cette communication a présenté rapidement les deux approches développées dans [HUG 03] de transformation de longues séries temporelles en descriptions symboliques. Le lecteur intéressé se reportera à [HUG 03] pour une définition précise du modèle de représentation symbolique de séries temporelles, ainsi que pour le détail des algorithmes de construction de ces représentations.

Dans [HUG 02] il est également proposé une définition de représentations symboliques *floues* de séries temporelles, avec des algorithmes permettant de les construire. L'introduction du flou permet d'améliorer la précision de la représentation par rapport à la série initiale, tout en conservant un petit nombre de symboles pour une interprétation aisée de la représentation.

Sur les nombreuses applications proposées dans la Section 4, seules quelques-unes ont été effectivement explorées : beaucoup de travail reste à faire dans la continuité de ce qui a été décrit.

6. Bibliographie

[AGR 95] AGRAWAL, R. SRIKANT, R., " Mining Sequential Patterns ", Proceedings of 11th Int'l Conf. on Data Engineering (DE'95), Taipei, Taiwan, 1995.

[BEL 61] BELLMAN R., " On the approximation of curves by line segments using dynamic programming ", *Communications of the ACM*, VOL.4, N°6, Juin 1961.

[HEB 01] HÉBRAIL G., HUGUENEY B., " Symbolic representation of long time series ", Conference on Applied Statistical Models and Data Analysis (ASMDA'2001), Compiègne, Juin 2001.

[HUG 02] HUGUENEY B., BOUCHON-MEUNIER B., HEBRAIL G., LE P., " Segmentation de séries temporelles en segments de niveaux prototypiques et de durées floues ", Rencontres francophones sur la logique floue et ses applications, Montpellier, France, Octobre 2002.

[HUG 03] HUGUENEY B., " Représentations symboliques de longues séries temporelles ", Thèse de doctorat de l'Université Paris VI, Janvier 2003.

[KEO 01] " An online algorithm for segmenting time series ", IEEE International Conference on Data Mining, 2001.

[KOH 95] KOHONEN T., *Self organizing maps*, Springer, Berlin, Heidelberg, 1995.

Algorithmes biomimétiques et classification

H. Azzag[*]**, C. Guinot**[**]**, G. Venturini**[*]

École Polytechnique de l'Université de tours, Laboratoire d'Informatique,
64, Avenue Jean Portalis, 37200 Tours, France
hanene.azzag@etu.univ-tours.fr, venturini@univ-tours.fr

[**]*CE.R.I.E.S., 20, rue Victor Noir, 92521 Neuilly-sur-Seine Cédex, France*
christiane.guinot@ceries-lab.com

RÉSUMÉ. *Nous présentons dans cet article un survol des algorithmes et méthodes biomimétiques pour résoudre le problème de la classification. Nous décrivons les approches utilisant les algorithmes génétiques et évolutionnaires avec les différents codages et représentations ayant été utilisés. Nous abordons l'approche à base de fourmis artificielles qui se trouve être une riche source d'inspiration pour la classification. Nous détaillons finalement d'autres approches à base d'agents avec notamment l'intelligence en essaim (nuages d'agents) et avec les systèmes immunitaires. Enfin, nous résumons les ressemblances et différences des travaux présentés et nous concluons sur les perspectives liées à l'approche biomimétique pour la classification.*

MOTS-CLÉS : *Classification, algorithmes évolutionaires, fourmis artificielles, systèmes immunitaires, nuages d'agents*

1. Introduction

Le problème de la classification de données est identifié comme une des problématiques majeures en extraction des connaissances à partir de données. Depuis des décennies, de nombreux sous-problèmes ont été identifiés, comme par exemple la sélection des données ou des descripteurs, la variété des espaces de représentation (numérique, symbolique, etc), l'incrémentalité, la nécessité de découvrir des concepts, d'obtenir une hiérarchie, etc. La popularité, la complexité et toutes ces variantes du problème de la classification de données ont donné naissance à une multitude de méthodes de résolution. Ces méthodes peuvent à la fois faire appel à des principes heuristiques ou encore mathématiques. Parmi celles-ci, il existe une branche qui s'inspire plus spécialement de principes issus de la biologie. Les motivations des chercheurs sont d'une part de tester de nouveaux algorithmes sur le problème de la classification et de connaître leurs apports. Mais elles sont aussi de proposer de nouvelles sources d'inspiration, car le problème de la classification se rencontre souvent chez les animaux et dans les systèmes biologiques.

Nous allons donc donner un aperçu de ces méthodes. Nous ne traiterons pas ici les approches neuronales mais plutôt les approches à base de population d'agents (algorithmes évolutionnaires, fourmis artificielles, intelligence en essaim, systèmes immunitaires). Nous allons considérer par la suite un ensemble de n données $d_1, ..., d_n$ à regrouper en classes. Nous ne ferons pas plus d'hypothèses à propos de la représentation des données ou de la forme de la classification désirée. Nous allons commencer dans la section 2 par détailler les approches génétiques qui manipulent une population de classifications candidates et qui les font évoluer en utilisant les principes de la sélection naturelle. Ensuite, nous abordons dans la section 3 la manière dont les fourmis artificielles sont appliquées à ce problème : chaque fourmi va intervenir sur une partie de la classification en cours de construction, avec des modèles très diversifiés qui dénotent une richesse importante de ce domaine. Dans la section 4, nous détaillons des approches moins connues mais qui apportent néanmoins leur potentiel à notre problématique : d'une part les déplacements sociaux d'une population d'agents permettent de créer des groupes, et d'autre part l'utilisation des systèmes immunitaires qui vont répondre aux stimulations d'antigènes (les données) en produisant des anticorps

(les éléments de la structure classificatoire). Enfin, nous donnons dans la section 5 une discussion sur les méthodes présentées ainsi que les perspectives que nous pouvons déduire des différents travaux en cours. Compte tenu du nombre de pages limité, nous ne citons pas d'articles fondateurs ou d'introduction à la classification ou aux méthodes biomimétiques, de même pour les travaux de biologie sous jacents aux modèles informatiques.

2. Approches évolutionnaires

2.1. *Quatre catégories d'algorithmes*

Dans les années 70, les premiers travaux sur l'évolution artificielle ont concerné les algorithmes génétiques (AG), les stratégies d'évolution (SE) et la programmation évolutive (PE). Ces trois types d'algorithmes ont utilisé des principes globalement communs car ils se sont tous inspirés des mêmes principes du neo-darwinisme : utilisation d'une population d'individus (dans notre cas chaque individu représente une classification des données), évaluation des individus par une fonction, sélection des meilleurs et génération d'une nouvelle population avec des opérateurs de croisement et de mutation. Cependant, des choix méthodologiques ont initialement opposé ces méthodes. Ainsi, les premiers AG utilisaient plutôt un codage binaire des individus alors que les SE utilisaient un codage en nombre réel. Ensuite, dans les années 90 est apparue la programmation génétique (PG) qui introduit notamment des représentations arborescentes.

Pour toutes ces approches, la représentation va également imposer des opérateurs particuliers pour engendrer de nouvelles solutions. Par exemple, l'un des principes fondamentaux des AG étant d'utiliser un opérateur de croisement combinant utilement les gènes de deux individus, le problème posé est alors de définir des opérateurs de croisement permettant l'échange de caractéristiques entre deux classifications. Les SE utilisent plutôt des mutations à base de lois gaussiennes qui vont modifier les paramètres réels d'un individu.

2.2. *Algorithmes génétiques*

Les premiers travaux proposant un AG (et plus généralement un algorithme évolutionnaire) pour le problème de la classification sont dus à [RAG 79]. Le nombre de classes est fixé à l'avance et la représentation de longueur n associe une classe à chaque donnée. Les opérateurs génétiques sont une adaptation directe des opérateurs génétiques binaires pour le cas d'un individu représenté par une chaîne n-aire (croisement avec un point de coupure). Par exemple, le croisement à un point échange des étiquettes de classe entre deux individus. Cet opérateur peut donc faire disparaître des classes. Seule la mutation peut faire apparaître de nouvelles classes. La fonction d'évaluation consiste à minimiser une erreur quadratique. Notons que ce codage est utilisé également par [HAN 00] dans le contexte du bipartitionnement : le premier vecteur d'entiers code la partition sur les données, le deuxième sur les variables.

De nouveaux codages utilisant des permutations ont été introduits par plusieurs auteurs. Dans [JON 91], un premier codage consiste à utiliser une permutation des n données représentées par leur indice en ajoutant en plus des symboles servant de séparateurs : par exemple pour $n = 6$, l'individu (2,4,-,5,1,3,-,6) représente un partionnement en 3 classes. Pour obtenir k classes, le nombre de séparateurs introduits est égal à $k - 1$. Un autre codage à base de permutation proposé dans le même travail consiste à utiliser deux parties dans un même individu. Les prototypes sont codés sur la première partie de la partition puis la suite de la partition représente un ordre sur la manière d'affecter les données restantes à ces prototypes. Dans [BHU 91], ce type de permutation est utilisé uniquement pour fixer un ordre sur les données : un algorithme heuristique décide alors comment construire la partition en "coupant" la permutation en des endroits judicieux. Pour ces codages sont utilisés des opérateurs génétiques de croisement définis dans le cadre du problème du voyageur de commerce (OX et PMX, voir [GOL 89]).

Un autre codage alternatif consiste non pas à représenter une classe pour chacun des objets dans un individu de longueur n, mais plutôt k prototypes de ces classes [LUC 93]. Ces k prototypes sont choisis parmi l'ensemble des n données. Ainsi, un individu devient un vecteur d'indices de prototype Ensuite, pour calculer la partition résultante, les données sont affectées à chaque prototype de classe sur la base d'un algorithme de type plus proche

voisin. Dans cette représentation, les opérateurs génétiques classiques peuvent poser des problèmes : à la suite d'un croisement, un même prototype peut se retrouver deux fois dans un individu.

Le codage introduit dans [BEZ 94] consiste à représenter le partitionnement à l'aide d'une matrice M booléenne de type classe $\times$ données. $M(i, j)$ prend la valeur 1 ($i \in [1, k]$, $j \in [1, n]$) si la donnée d_j appartient à la classe i, 0 sinon. Dans cette représentation, l'opérateur de croisement est défini cette fois en 2D. Un point important à noter dans cette représentation est la possibilité de la généraliser à des classifications recouvrantes ainsi que floues : dans le premier cas plusieurs 1 peuvent apparaître sur une même ligne, dans le deuxième les valeurs ne sont plus booléennes mais représentent des degrés d'appartenance.

Un codage permettant de manipuler directement des groupes a été proposé dans [FAL 94]. Une classification est constituée de n gènes représentant la classe de chaque donnée (comme dans le premier codage introduit dans cette section), suivis de la liste des groupes apparaissant dans l'individu (par exemple si les données appartiennent à trois classes, l'individu finit par (3, 2, 1)). Cette représentation utilise donc un opérateur de croisement permettant d'échanger directement des groupes. L'opérateur de mutation agit également au niveau des groupes (éclatement, regroupement, etc) avec des heuristiques locales (réaffectation des données isolées).

Dans [GRE 03] a été développé à notre connaissance le seul AG apprenant une classification hiérarchique présentée sous la forme d'un arbre de centroïdes. Cet algorithme est restreint aux données numériques mais ne fait pas d'hypothèses sur le nombre de classes.

2.3. *Autres approches évolutionnaires*

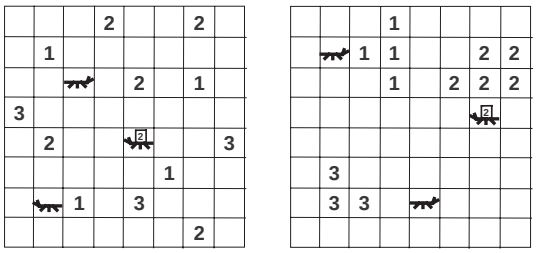
Les trois autres catégories d'algorithmes évolutionnaires ont été nettement moins développées que les AG. Par exemple, dans [BAB 94], les SE ont été utilisées avec un codage matriciel : chaque colonne du tableau représente un centre de classe de la même dimension que l'espace numérique de description des données. La PE a également été utilisée mais dans un seul travail à notre connaissance [FOG 93]. Également, nous n'avons pas trouvé d'articles traitant du problème général de la classification avec la PG.

Plusieurs approches hybrides ont cependant été proposées en utilisant conjointement les AG avec des approches plus classiques comme K-Means ou encore Fuzzy-C-Means. Ces heuristiques sont utilisées par exemple juste après l'AG qui sert donc à trouver un bonne partition initiale [BAB 93]. Elles peuvent également servir au même titre que les opérateurs génétiques dans la boucle de l'AG [KRI 99] : elles sont appliquées sur chaque individu. Cela permet d'accélérer la convergence des AG tout en conservant les avantages d'une méthode globale. Ces hybridations restent cependant liées aux données numériques.

3. Fourmis artificielles

3.1. *Quelques principes*

Les fourmis réelles ont inspiré les chercheurs en informatique dans de nombreux domaines. Cela se justifie particulièrement quand on connaît la richesse comportementale de ces animaux. L'un des modèles les plus connus (ACO pour Ant Colony Optimization) a été introduit par [COL 91] initialement dans le cadre du problème du voyageur de commerce. Les fourmis utilisent des phéromones pour marquer des arcs entre les villes. Ces phéromones représentent en fait une distribution de probabilités qui est mise à jour en fonction des résultats observés (longueur totale du chemin par exemple). Cette approche a été depuis largement développée et appliquée à de nombreux problèmes d'optimisation combinatoire et numérique. Un survol de ces articles ne rentre cependant pas dans le cadre de notre étude puisque ce modèle n'a pas été utilisé à notre connaissance pour résoudre le problème de la classification (voir cependant [ALE 00] mais apparemment aucune suite n'a été donnée à ces travaux). Pourtant, les sections suivantes vont montrer que le modèle des fourmis artificielles est très riche dans ce domaine. La raison vient du fait que d'autres comportements observés chez les fourmis peuvent être directement mis en relation avec le problème de la classification, à commencer par le tri du couvain.



(a) (b)

FIG. 1. *Principe de la classification de données par des fourmis articielles selon l'algorithme présenté par [Lumer et Faieta, 1994]. En (a) les objets sont répartis aléatoirement sur la grille. Les fourmis peuvent s'en saisir et les déposer dans des cases où la densité d'objets similaires est élevée. Il en résulte la formation de groupes comme en (b).*

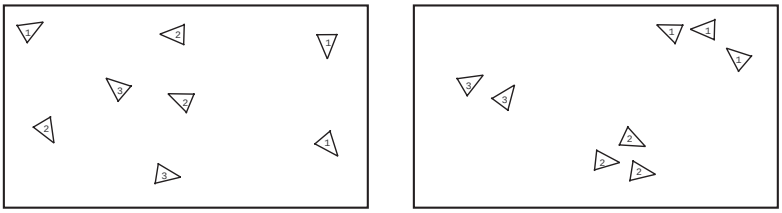
3.2. *Les travaux fondateurs*

Ces travaux datent des années 1990. Il s'agit d'abord des travaux de biologistes s'intéressant de près à la modélisation des fourmis en termes mathématiques et informatiques, et à l'utilisation concrète de ces modèles. Deneubourg apparaît donc comme un pionnier dans le domaine du tri d'objets par des fourmis artificielles. Dans [DEN 90], il propose avec ses collègues les principes suivants : des fourmis artificielles se déplacent sur un plan. Les objets à rassembler sont répartis sur ce plan. Une fourmi ne dispose que d'une perception locale de ces objets et ne communique pas avec les autres. Au lieu de cela, la configuration des objets sur le sol va influencer leurs actions. Lorsqu'une fourmi rencontre un objet, elle le ramasse avec une probabilité $\frac{c_1}{c_1+f}$, où f représente la fréquence de rencontre d'objets dans un passé récent. Autrement dit, plus une fourmi rencontre d'objets, moins elle a de chance d'en prendre un (elle se trouve dans une zone avec beaucoup d'objets). Ensuite, une fois un objet ramassé, la fourmi se déplace au hasard dans le plan, et elle dépose l'objet avec une probabilité $\frac{f}{c_2+f}$. Cette probabilité est d'autant plus grande que la fourmi a rencontré récemment des objets. Ces principes relativement simples font qu'il apparaît des regroupements d'objets. L'approche peut être généralisée à plusieurs types d'objets (les fréquences f sont spécifiques à chaque type d'objets) : cet algorithme permet alors de trier des objets.

Le pas qui sépare le tri d'objets de la classification a ensuite été franchi dans [LUM 94]. Ils ont adapté l'algorithme présenté précédemment (voir figure 1) : les données sont initialement réparties aléatoirement sur une grille 2D. Chaque fourmi est située dans une case de cette grille et ne perçoit que les données situées dans son voisinage (8 voisins par exemple). Ensuite la fréquence f utilisée dans l'algorithme de tri vu précédemment peut être remplacée par une moyenne des similarités entre une donnée d_i portée par une fourmi et les données d_j situées dans son voisinage. Une donnée d_i sur la grille est ramassée avec une probabilité d'autant plus grande qu'elle est peu similaire aux données voisines. De la même manière, une donnée d_i portée par une fourmi est plus facilement déposée dans une région comportant des données qui lui sont similaires. Cet algorithme a été depuis étendu à d'autres applications comme le partitionnement de graphes [KUN 97] ou la classification de sessions sur des sites Web [ABR 03].

3.3. *Approches récentes*

Une extension de l'algorithme de [LUM 94] a été présentée dans [MON 99]. D'une part, les fourmis peuvent empiler les objets les uns sur les autres dans une même case de la grille. Lorsqu'elles rencontrent un tas d'objets, elles peuvent ainsi se saisir de l'objet le plus dissimilaire. D'autre part, une hybridation a été effectuée avec l'algorithme des K-Means. Cette hybridation consiste à utiliser la séquence d'algorithmes suivante : AntClass, K-Means, AntClass, K-Means. AntClass fournit une partition initiale, les K-Means corrige des erreurs d'AntClass qui mettrait beaucoup plus de temps a être corrigée avec AntClass seul.



(a) (b)

FIG. 2. *Principes utilisés pour la classification par nuages d'agents. Les agents sont placés initialement avec des coordonnées et des vecteurs vitesse aléatoires (voir (a)). Les mouvements d'un agent dépendent des autres agents perçus dans son voisinage et des similarités entre les données qu'ils représentent. Le comportement local de chaque agent tend à former globalement des groupes d'agents similaires se déplaçant de manière cohérente (voir (b)).*

Dans [LAB 02] a été introduit un nouveau modèle à base de fourmis pour la classification utilisant le système d'identification chimique des fourmis. Celui-ci est fondé sur la construction d'une odeur coloniale qui est le fruit des apports génétiques, environnementaux et comportementaux. Cette odeur est construite par les individus pour identifier qui fait partie du groupe et qui doit être rejeté. A partir de ce modèle, un nouvel algorithme de classification a été proposé dans lequel chaque donnée est une fourmi dont l'odeur est déterminée par les valeurs prises par les attributs décrivant cette donnée. Les fourmis effectuent des rencontres aléatoires et décident d'appartenir au même groupe ou non. Il en résulte l'établissement d'une classification.

Enfin, dans [AZZ 03], a été introduit un nouveau modèle permettant d'effectuer rapidement une classification hiérarchique. Il s'agit de copier la manière dont les fourmis construisent des structures vivantes en s'accrochant les unes aux autres en fonction de critères locaux (la forme de la structure influençant le comportement d'accrochage ou de décrochage). Dans ce modèle, chaque fourmi artificielle représente une donnée. Les fourmis sont placées initialement à la racine de l'arbre et vont pouvoir se déplacer dans cet arbre et s'accrocher afin de construire une structure hiérarchique dont chaque noeud représente une donnée. L'objectif est de construire automatiquement un site portail (données textuelles) et d'obtenir la propriété suivante : chaque noeud o de l'arbre est une catégorie composée de toutes les données portées par les sous-arbres de o. Les sous catégories (représentées par les noeuds connectés à o) doivent être très similaires à leur mère dans l'arbre, mais également les plus dissimilaires entre elles. Les résultats obtenus sont très compétitifs par rapport à la classification ascendante hiérarchique notamment.

4. Autres approches

4.1. *Intelligence en essaim*

L'intelligence en essaim ("swarm intelligence") regroupe de nombreux algorithmes à base de population d'agents. Les fourmis artificielles en font partie mais nous nous intéressons dans cette section à des algorithmes plus spécifiques qui utilisent les déplacements d'un essaim d'agents pour résoudre un problème. A titre d'exemple, les algorithmes PSO ("particle swarm optimization") utilisent un ensemble de particules caractérisées par leur position et leur vitesse pour maximiser une fonction dans un espace de recherche. Des interactions ont lieu entre les particules afin d'obtenir des comportements globaux efficaces.

Dans la biologie, de nombreux chercheurs se sont intéressés à la manière dont les animaux se déplacent en groupe. Aucun individu ne contrôle les autres mais pourtant des formes et des comportements complexes peuvent apparaître lors de ces déplacements. [REY 87] a été probablement le premier à proposer une utilisation informatique de tels modèles, simulations qui sont utilisées notamment dans l'industrie du cinéma pour donner des mouvements réalistes à des groupes d'individus. Dans ces travaux, chaque individu évolue dans un espace 3D. Il est donc caractérisé par sa position et sa vitesse. Un individu perçoit les autres dans un voisinage donné. Des règles

comportementales généralement simples permettent aux individus de se déplacer en groupe, d'éviter des obstacles, etc.

En 1998, ces principes ont été appliqués pour la première fois à un problème de classification [PRO 98] (voir figure 2). Les agents représentent chacun une donnée. Un agent réagit aux autres agents présents dans son voisinage en tenant compte de la similarité des données. Un agent se déplacera plutôt vers des données qui lui sont similaires. Cette règle comportementale permet donc de former des groupes de données similaires.

Dans [MON 02], cet algorithme a été amélioré et évalué d'une manière plus systématique. Une distance idéale entre individus est définie, distance qui dépend de la similarité entre les données. Un critère d'arrêt est utilisé également en mesurant l'entropie spatiale du nuage d'agents. Cet algorithme a été intégré dans un système de fouille visuelle de données utilisant la réalité virtuelle.

4.2. *Systèmes immunitaires*

Les systèmes immunitaires (SI) sont un ensemble de modélisations du système immunitaire humain et animal appliqués à différents problèmes en informatique. Ils utilisent les principes suivants : des agents (lymphocytes) qui génèrent des anticorps vont apprendre à reconnaître le soi du non-soi (les antigènes). Pour cela, ces agents doivent d'abord être engendrés en utilisant un principe de composition de briques élémentaires. Ensuite, ils subissent un test de sélection (dit de sélection négative) : les agents rejetant le soi sont éliminés, et les autres, qui vont rejeter le non-soi, sont gardés. Chaque fois qu'il y a reconnaissance d'un antigène par un anticorps, la présence des lymphocytes générant ces anticorps est favorisée par un processus de sélection par clonage et par la disparition des lymphocytes non stimulés par les antigènes. Ce clonage donne donc lieu à des interactions entre les lymphocytes et peut mettre en oeuvre des mutations. Certains lymphocytes, lorsqu'ils sont souvent utilisés, prennent un rôle d'élément de mémorisation à long terme. Ces systèmes disposent de propriétés complexes car ils sont capable de générer des solutions et de les sélectionner en fonction de leur efficacité selon des heuristiques originales.

En ce qui concerne la classification, les principes des systèmes immunitaires sont, à un niveau général, les suivants (voir par exemple le système aiNet [CAS 00]) : les données $d_1, ..., d_n$ représentent les antigènes. Ces antigènes sont présentés itérativement au système jusqu'à l'obtention d'une condition d'arrêt. On suppose que les données sont numériques, et donc qu'un antigène est un vecteur de dimension n. A chaque itération, l'antigène présenté va activer des anticorps (assimilés dans cette modélisation à des lymphocytes-B). Un anticorps est également représenté par un vecteur de dimension n. Les anticorps suffisamment proches de l'antigène (au sens de la distance euclidienne) vont subir des clonages avec mutation (interaction anticorps/antigènes) afin d'amplifier et d'affiner la réponse du système. Egalement, ces anticorps vont subir une sélection (interaction anticorps/anticorps) : ceux qui sont trop proches les uns des autres seront diminués en nombre. Après ces itérations, le système converge en plaçant des anticorps (qui agissent comme des détecteurs) de manière judicieuse et en nombre adapté aux données.

D'autres modèles plus complexes existent. Ainsi dans [KNI 02] le système utilise plusieurs niveaux de cellules et d'interaction (anticorps, lymphocytes, cellules de mémorisation). Dans [NAS 02], ce même système est généralisé et amélioré en utilisant des fonctions d'appartenance floue plutôt qu'une distance euclidienne et un seuil.

5. Discussion et perspectives

Il ressort de cette étude des points saillants que l'on peut résumer comme suit. Il est certain que les AG pour la classification à eux seuls ont fait l'objet d'un volume de travaux plus important que toutes les autres méthodes réunies, cela étant certainement dû à leur popularité mais aussi aux succès rencontrés en tant que méthode globale d'optimisation. Cependant, au sein des méthodes biomimétiques, ces algorithmes n'ont pas nécessairement tous les avantages de leur côté. Le problème du choix des paramètres reste difficile (cela ne concerne pas le choix du nombre de classes mais plutôt des paramètres liés à la méthode comme la taille de la population, les opérateurs,

etc). La diversité des codages utilisés montre par ailleurs que les AG sont sensibles aux choix de la représentation et des opérateurs, et que le choix d'un opérateur de croisement pour optimiser des partitions n'est pas simple.

Il faut noter également que les principales différences entre les AG et les autres méthodes viennent du fait que dans les AG, un individu représente généralement une classification entière et la population est un ensemble de classifications, alors que dans les autres algorithmes, un individu représente une donnée et la population dans son ensemble représente la classification. Dans un AG, la solution au problème est le meilleur individu de la population, alors que dans les autres algorithmes, la solution est l'ensemble des individus. Cette différence est fondamentale puisque elle va obliger l'AG à centraliser son fonctionnement. Les autres algorithmes vont au contraire utiliser des principes heuristiques plutôt locaux et agissant en parallèle sur toute la classification. Sans doute que cela a des répercutions sur le temps d'apprentissage dans les AG, ce qui justifie l'étude d'approches génétiques hybrides.

Parmi les perspectives que l'on peut dégager, il faut noter qu'il existe encore peu de méthodes biomimétiques qui soient incrémentales, conceptuelles et/ou hiérarchiques. Des potentialités existent cependant : on peut ajouter des fourmis/données dans un algorithme tel [LUM 94], ou encore faire apparaître de nouveaux agents en déplacement dans un essaim. L'incrémentalité est possible dans de nombreux algorithmes, mais n'a pas encore été réellement testée, sans doute parce que les problèmes traités ne le requièrent pas. La classification conceptuelle semble un peu plus difficile car les méthodes représentent plutôt un partitionnement (au sens ensembliste du terme) plutôt que des regroupements selon des caractéristiques communes. Des pistes sont lancées notamment avec les systèmes immunitaires, mais en général les algorithmes n'utilisent pas, sauf dans le cas numérique avec des centroïdes, l'espace de description des données pour créer des groupes. Les approches hiérarchiques sont également globalement ignorées, pourtant leur intérêt est grand pour l'interprétation des résultats par un expert humain. On aurait pu penser pour les AG que l'apparition de représentations arborescentes (programmation génétique, etc) allait donner lieu à des études génétiques et hiérarchiques, mais cela n'a apparemment pas encore été le cas. Un autre axe encore inexploré mais qui devrait pouvoir l'être par ces méthodes est le traitement de grandes bases de données. D'une part ces méthodes biomimétiques peuvent être assez facilement parallélisées ce qui n'est pas nécessairement le cas des méthodes classiques en classification. D'autres part elles utilisent (fourmi, essaim et systèmes immunitaires notamment) des propriétés statistiques des données plutôt que des cas particuliers : on peut imaginer par exemple augmenter par des facteurs importants la taille d'une population de fourmis ou encore celle d'un nuage d'agents, ou recourir par exemple à de l'échantillonnage.

Un des axes extrêmement prometteur pour certaines de ces méthodes (fourmis, essaim principalement) vient de leur capacité à fournir une classification comme les autres méthodes, mais également une visualisation de ces classifications. Cette capacité est une grande force des algorithmes à base d'essaim et de l'algorithme de [LUM 94] notamment (à l'image des cartes de Kohonen pour les réseaux de neurones). Cela permet à l'expert du domaine d'interpréter directement et interactivement les résultats et de formuler graphiquement des requêtes sur ces données. Notons que d'autres approches biomimétiques, comme les automates cellulaires par exemple, n'ont pas encore été réellement exploités pour le domaine de la classification.

6. Bibliographie

[ABR 03] ABRAHAM A., RAMOS V., Web Usage Mining Using Artificial Ant Colony Clustering and Linear Genetic Programming, *The Congress on Evolutionary Computation*, Canberra, Australia, 08-12 December 2003, IEEE-Press, p. 1384-1391.

[ALE 00] ALEXANDROV D., Randomized Algorithms for the Minmax Diameter k-Clustering Problem, *Proceedings of ECCO 13*, Capri, Italy, May 2000, p. 193–194.

[AZZ 03] AZZAG H. MONMARCHÉ N., SLIMANE M., VENTURINI G., GUINOT C., AntTree : a New Model for Clustering with Artificial Ants, *IEEE Congress on Evolutionary Computation*, Canberra, Australia, 08-12 December 2003.

[BAB 93] BABU G., MURTY M., A near-optimal initial seed value selection in K-means algorithm using a genetic algorithm, vol. 14, 1993, p. 763-769.

[BAB 94] BABU G., MURTY M., Clustering with evolution strategies, vol. 27, n^o 2, 1994, p. 321-329.

[BEZ 94] BEZDEK J., BOGGAVARAPU S., HALL L., BENSAID A., Genetic algorithm guided clustering, *Proceedings of the First IEEE Conference on Evolutionary Computation*, 1994, p. 34–39.

[BHU 91] BHUYAN J., RAGHAVAN V., ELAYAVALLI V., Genetic algorithms for clustering with an ordered representation, BELEW R., BOOKER L., Eds., *Proceedings of the Fourth International Conference on Genetic Algorithms*, San Diego, CA, 1991, Morgan Kaufmann, p. 408–15.

[CAS 00] DE CASTRO L., VON ZUBEN F., An Evolutionary Immune Network for Data Clustering, *In Proceedings of the IEEE SBRN'00 (Brazilian Symposium on Artificial Neural Networks)*, 2000, p. 84–89.

[COL 91] COLORNI A., DORIGO M., MANIEZZO V., Distributed Optimization by Ant Colonies, *Proceedings of the First European Conference on Artificial Life*, 1991, p. 134–142.

[DEN 90] DENEUBOURG J.-L., GOSS S., FRANKS N., SENDOVA-FRANKS A., DETRAIN C., CHRETIEN L., The dynamics of collective sorting : robot-like ant and ant-like robots, *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, 1990, p. 356–365.

[FAL 94] FALKENAUER E., A new representation and operators for genetic algorithms applied to grouping problems, *Evolutionary Computation*, vol. 2, n^o 2, 1994, p. 123-144.

[FOG 93] FOGEL D., SIMPSON P., Evolving Fuzzy Clusters, *ICNN93*, San Francisco, 1993, p. 1829-1834.

[GOL 89] GOLDBERG D., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[GRE 03] GREENE W., Unsupervised Hierarchical Clustering via a Genetic Algorithm, PRESS I., Ed., *Proceedings of the 2003 Congress on Evolutionary Computation*, Canberra, Australia, 2003, p. 998-1005.

[HAN 00] HANSOHM J., Two-mode Clustering with Genetic Algorithms, *Classification, Automation, and New Media : Proceedings of the 24th Annual Conference of the Gesellschaft Fur Klassifikation E.V.*, 2000, p. 87-94.

[JON 91] JONES D., BELTRAMO M., Solving Partitioning Problems with Genetic Algorithms, BELEW R., BOOKER L., Eds., *Proceedings of the Fourth International Conference on Genetic Algorithms*, San Diego, CA, 1991, Morgan Kaufmann, p. 442-449.

[KNI 02] KNIGHT T., TIMMIS J., On data clustering with artificial ants, A. LOTFI J. G., JOHN R., Eds., *Proceedings of the 4th International Conference on Recent Advances in Soft Computing*, Nottingham, UK., December 2002, p. 266-271.

[KRI 99] KRISHNA K., MURTY M., Genetic K-means algorithm, *IEEE Transactions on Systems, Man and Cybernetics - Part B*, vol. 29, n^o 3, 1999, p. 433-439.

[KUN 97] KUNTZ P., LAYZELL P., SNYERS D., A Colony of Ant-like Agents for Partitioning in VLSI Technology, HUSBANDS P., HARVEY I., Eds., *Proceedings of the Fourth European Conference on Artificial Life*, 1997, p. 417–424.

[LAB 02] LABROCHE N., MONMARCHÉ N., VENTURINI G., A new clustering algorithm based on the chemical recognition system of ants, HARMELEN F., Ed., *Proceedings of the 15th European Conference on Artificial Intelligence*, Lyon, France, july 2002, IOS Press, p. 345–349.

[LUC 93] LUCASIUS C., DANE A., KATEMAN G., On K-medoid clustering of large data sets with the aid of a genetic algorithm : background, feasibility and comparison, *Analytica Chimica Acta*, vol. 282, 1993, p. 647-669.

[LUM 94] LUMER E., FAIETA B., Diversity and Adaptation in Populations of Clustering Ants, *Proceedings of the Third International Conference on Simulation of Adaptive Behaviour*, 1994, p. 501–508.

[MON 99] MONMARCHÉ N., SLIMANE M., VENTURINI G., On Improving Clustering in Numerical Databases with Artificial Ants, FLOREANO D., NICOUD J., MONDALA F., Eds., *5th European Conference on Artificial Life (ECAL'99), Lecture Notes in Artificial Intelligence*, vol. 1674, Swiss Federal Institute of Technology, Lausanne, Switzerland, 13-17 September 1999, Springer-Verlag, p. 626-635.

[MON 02] MONMARCHÉ N., GUINOT C., VENTURINI G., Fouille Visuelle et classification de données par nuage d'insectes volants, *RSTI-RIA-ECA : Méthodes d'optimisation pour l'extraction de connaissances et l'apprentissage*, , n^o 6, 2002, p. 729–752.

[NAS 02] NASAROUI O., DASGUPTA D., GONZALEZ. F., The Fuzzy Artificial Immune System : Motivations, Basic Concepts, and Application to Clustering and Web Profiling, *Proceedings of the IEEE International Conference on Fuzzy Systems at WCCI*, May 12-17 2002, p. 711-716.

[PRO 98] PROCTOR G., WINTER C., Information Flocking : Data Visualisation in Virtual Worlds using Emergent Behaviours, HEUDIN J.-C., Ed., *Proc. 1st Int. Conf. Virtual Worlds, VW*, vol. 1434, Springer-Verlag, 1998, p. 168–176.

[RAG 79] RAGHAVAN V., BIRCHARD K., A Clustering Strategy Based on a Formalism of the Reproductive Process in Natural Systems, *Information Implications into the Eighties, Proceedings of the Second International Conference on Information Storage and Retrieval*, ACM, 1979, p. 10-22.

[REY 87] REYNOLDS C. W., Flocks, Herds, and Schools : A Distributed Behavioral Model, *Computer Graphics (SIGGRAPH '87 Conference Proceedings)*, vol. 21, n^o 4, 1987, p. 25–34.

Analyse de données probabilistes : Treillis de concepts et classification

Paula Brito * — Géraldine Polaillon ** — Francisco de A. T. de Carvalho ***

** Faculdade de Economia / LIACC , Universidade do Porto*
Rua Dr. Roberto Frias, 4200-464 Porto, PORTUGAL
mpbrito@fep.up.pt

*** SUPELEC - Service Informatique*
Plateau de Moulon, 91192 Gif-sur-Yvette Cedex, FRANCE
geraldine.polaillon@supelec.fr

**** Centro de Informatica - CIn/UFPE*
Av. Prof Luiz Freire, s/n, Cidade Universitaria, CEP 50.740-540, Recife - PE BRAZIL
fatc@cin.ufpe.br

RÉSUMÉ. *On s'intéresse aux données probabilistes, c'est-à-dire, quand chaque individu est décrit par des distributions de probabilités ou de fréquences sur les catégories de variables qualitatives. En définissant les opérateurs appropriés pour le calcul d'extension et d'intention, on obtient deux correspondances de Galois distinctes, qui permettent de définir deux treillis de concepts pour ce type de données. D'autre part, utilisant des mesures de généralité, adaptées aux données probabilistes, une méthode de classification ascendante hiérarchique /pyramidale est proposée, dont les classes formées sont des éléments du treillis de concepts correspondant.*

MOTS-CLÉS : *Analyse de données symboliques, Données probabilistes, Treillis de concepts, Classification conceptuelle, Hiérarchie, Pyramide*

1. Introduction

Le besoin de traiter des données ne pouvant être représentées dans une matrice n x p classique, combiné avec l'intérêt croissant pour des méthodes dont les résultats sont directement interprétés en termes des variables descriptives, ont conduit au développement de l'analyse de données symboliques. Les données symboliques étendent le modèle tabulaire classique, où chaque individu en ligne prend une et une seule valeur pour chaque variable en colonne, en permettant des valeurs multiples, éventuellement pondérées. De nouveaux types de variables ont été introduits - variables intervalles, catégoriques multi-valuées et modales - qui permettent de tenir compte de la variabilité et/ou l'incertitude inhérentes aux données. Une *variable modale* est une variable à valeurs multiples, où, pour chaque "individu", est donné un ensemble de modalités et, pour chaque modalité, une fréquence, probabilité ou un poids. Quand une distribution empirique est donnée, la variable est appelée *variable histogramme* ([BOC 00]). Des données décrites par des variables modales sont désignées par *données modales*, si une distribution de probabilités ou de fréquences est donnée pour chaque variable, on les désigne par des *données probabilistes*. On trouve souvent ce type de données dans des applications pratiques, par exemple, quand on désire exprimer une incertitude, ou quand on agrège les réponses d'une enquête.

Les treillis de Galois permettent d'organiser les observations dans des classes automatiquement interprétées, dont la structure ne dépend, ni de paramètres externes, ni de l'ordre d'observation ni de l'implémentation algo-

rithmique. Barbut and Monjardet ont été les premiers à s'intéresser aux correspondances de Galois, introduites par Birkhoff en 1940 ([BIR 40]), pour l'étude d'un tableau de données binaires ([BAR 70]). Depuis, beaucoup de développements théoriques et pratiques ont été accomplis, citons d'une part par l'école de Darmstadt en "Analyse Conceptuelle Formelle" ("Formal Concept Analysis") ([WIL 82, GAN 99]), et d'autre part Duquenne ([DUQ 86, DUQ 87]) et aussi ([GUé 93, MEP 93, GOD 95, GIR 99]), qui ont utilisé la théorie des treillis pour l'organisation et l'analyse des données.

L'extension des correspondances de Galois et des treillis de Galois aux données symboliques a été d'abord traité par Brito ([BRI 91, BRI 94a]) et développé ensuite par Polaillon ([POL 98a, POL 98b, POL 99]). Dans un papier récent, Brito et Polaillon ([BRI 04]) ont défini les outils permettant d'obtenir directement des treillis de Galois sur des données probabilistes, sans qu'une transformation préalable soit nécessaire, et proposé un algorithme qui construit le treillis de concepts.

Le treillis de concepts est en général assez complexe et contient un nombre important de classes, ce qui rend son interprétation difficile. Une alternative consiste à limiter le nombre de classes formées, en imposant une structure plus simple. Une méthode de classification hiérarchique ou pyramidale a été proposée ([BRI 91, BRI 94b]) où chaque classe formée est un concept, pour les opérateurs considérés : chaque classe est représentée par un objet symbolique dont l'extension doit coïncider avec la classe elle-même. Un critère numérique additionnel est défini, une mesure de "généralité", qui permet, à chaque étape, de choisir la "meilleure" agrégation parmi les agrégations possibles. Le cas des données décrites par des variables modales a été d'abord traité dans ([BRI 98]) ; Brito et De Carvalho ont étendu la méthode pour permettre de prendre en compte l'existence de règles hiérarchiques entre variables catégoriques multi-valuées ([BRI 99]) et entre variables modales ([BRI 02]), en définissant de façon adéquate les opérateurs de généralisation et les mesures de généralité. Les classes formées correspondent à des "concepts", décrits en extension, par la liste de ses membres, et en intention, par un objet symbolique qui exprime ses propriétés. La méthode rentre dans le cadre de la classification conceptuelle, puisque chaque classe formée est associée à une conjonction de propriétés portant sur les variables descriptives, qui constitue une condition nécessaire et suffisante d'appartenance à la classe.

Rappelons que les pyramides ([DID 84, DID 86, BER 85, BER 86]) étendent le modèle hiérarchique en permettant des classes recouvrantes qui ne sont pas emboîtées, mais le modèle pyramidal impose l'existence d'un ordre total sur l'ensemble des individus à classifier, tel que chaque classe formée soit un intervalle de cet ordre. La classification pyramidale produit une classification plus riche qu'une hiérarchie, en ce sens qu'elle permet la formation d'un plus grand nombre de classes, et elle fournit une sériation de l'ensemble donné.

Beaucoup d'autres méthodes de classification hiérarchique ont maintenant été proposées pour classifier des données symboliques, qui diffèrent selon le type des données qu'elles permettent de traiter et/ou le critère de formation des classes, citons [GOW 91, GOW 92, GOW 95b, GOW 95a, CHA 98, ELS 98].

Dans la Section 2, on commence par rappeler la définition de variable modale, et formaliser les notions d'événement modal et objet modal. On détaille ensuite, dans la Section 3, les résultats permettant de définir des treillis de Galois sur des données décrites par des variables modales. Dans la Section 4, on présente la méthode de classification hiérarchique / pyramidale pour ce type de données. Enfin, on illustre les méthodes présentées par un exemple.

2. Données Probabilistes

DÉFINITION 1 ([BOC 00])
Une *variable modale* Y définie sur un ensemble $E = \{\omega_1, \omega_2, \ldots\}$ de domaine $O = \{m_1, \ldots, m_k\}$ est une application $Y(\omega) = (U(\omega), \pi_\omega)$, pour $\omega \in E$, où π_ω est une distribution (fréquence ou probabilité) sur le domaine O des valeurs possibles (complétée par une σ-algèbre convenable), et $U(\omega) \subseteq O$ est le support de π_ω dans O.

En général, le support $U(\omega)$ peut être omis de la définition, et une variable modale considérée comme une application $Y : E \to M(O)$, de E dans la famille $M(O)$ des mesures non-négatives π sur O, à valeurs $Y(\omega) = \pi_\omega = \{m_1(p_1(\omega)), \ldots, m_k(p_k(\omega))\}$.

DÉFINITION 2

Un *événement modal* est une expression de la forme $e = [Y(\omega)R\{m_1(p_1), m_2(p_2), \ldots, m_k(p_k)\}]$ où $O = \{m_1, m_2, \ldots, m_k\}$ est le domaine de Y, et p_ℓ est la probabilité, fréquence ou poids de $m_\ell, \ell = 1, \ldots, k$. Il n'est pas imposé que $p_1 + p_2 + \ldots + p_k = 1$. R est une relation définie sur l'ensemble des distributions sur O. On considère les relations suivantes :

– $'\sim'$ tel que $[Y(\omega) \sim \{m_1(p_1), \ldots, m_k(p_k)\}]$ est vrai ssi $p_\ell(\omega) = p_\ell, \ell = 1, \ldots, k$
– $'\leq'$ tel que $[Y(\omega) \leq \{m_1(p_1), \ldots, m_k(p_k)\}]$ est vrai ssi $p_\ell(\omega) \leq p_\ell, \ell = 1, \ldots, k$
– $'\geq'$ tel que $[Y(\omega) \geq \{m_1(p_1), \ldots, m_k(p_k)\}]$ est vrai ssi $p_\ell(\omega) \geq p_\ell, \ell = 1, \ldots, k$

Un *objet modal* est formé d'une conjonction d'événements modaux.

Chaque individu $\omega \in E$ est associé à un objet modal :

$$s(\omega) = \bigwedge_{j=1}^{p} \left[Y_j(w) \sim \{m_1^j(p_1^j(\omega)), \ldots, m_{k_j}^j(p_{k_j}^j(\omega))\}\right]$$

Dans la description de chaque individu $w \in E$, on a toujours $p_1^j(\omega) + \ldots + p_{k_j}^j(\omega) = 1, j = 1, \ldots, p$. En fait, quand $p_1 + p_2 + \ldots + p_k = 1$, on a une distribution de probabilités ou de fréquences, alors que quand $p_1 + p_2 + \ldots + p_k \geq 1$ (resp. $p_1 + p_2 + \ldots + p_k \leq 1$) on a une enveloppe supérieure (resp. inférieure) d'une distribution de probabilités ou de fréquences.

DÉFINITION 3

On définit un ordre partiel sur l'ensemble des objets modaux définis sur le même ensemble de variables $\{Y_1, \ldots, Y_p\}$, comme suit :

Si $s_1 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(p_1^j), \ldots, m_{k_j}^j(p_{k_j}^j)\}\right]$ et $s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(q_1^j), \ldots, m_{k_j}^j(q_{k_j}^j)\}\right]$ alors $s_1 \leq s_2$ ssi $p_\ell^j \leq q_\ell^j, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

Si $s_1 \leq s_2$ on dira que s_1 est *plus spécifique* que s_2 et que s_2 est *plus général* que s_1.

3. Treillis de concepts sur des données probabilistes

Rappelons qu'une correspondance de Galois est une paire d'applications (f, g) entre deux ensembles ordonnés $(A, \leq_A)$ et $(B, \leq_B)$ qui sont antitones et dont les applications composées $h = g \circ f$ et $h' = f \circ g$ sont extensives.

Soit S l'ensemble des objets modaux, avec $0 \leq p_\ell^j \leq 1, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

THÉORÈME 1 :

Le couple d'applications

$$\begin{aligned} f_u : S &\rightarrow P(E) \\ s &\rightarrow ext_E s = \{\omega \in E : s(\omega) \leq s\} \end{aligned}$$

$$\begin{aligned} g_u : P(E) &\rightarrow S \\ C = \{w_1, \ldots, w_H\} &\rightarrow int(C) = s = \bigwedge_{j=1}^{p} \left[Y_j \leq \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right] \end{aligned}$$

avec $t_\ell^j = \text{Max}\ \{p_\ell^j(w_h), h = 1, \ldots, H\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ forment une correspondance de Galois entre $(P(E), \subseteq)$ et $(S, \geq)$.

Exemple :

Considérons le tableau de données présenté dans la section 5.

Alors, $g_u(\{\text{Ann, Bob}\}) = s_u =$
$[\text{ Tinnitus } \leq \{\text{ fréquent } (1),\text{ rare } (0.2)\}] \wedge [\text{ Maux de Tête } \leq \{\text{ fréquents } (0.9),\text{ rares } (1)\}] \wedge$
$[\text{ Pression Sanguine } \leq \{\text{ haute } (0.8),\text{ normale } (0.4),\text{ basse } (0.0)\}]$
et $f_u(s_u) = \{\text{Ann, Bob}\}$

THÉORÈME 2 :
Le couple d'applications

$$\begin{aligned} f_i : S &\rightarrow P(E) \\ s &\rightarrow ext_E s = \{\omega \in E : s(\omega) \geq s\} \end{aligned}$$

$$\begin{aligned} g_i : P(E) &\rightarrow S \\ C = \{w_1, \ldots, w_H\} &\rightarrow int(C) = s = \bigwedge_{j=1}^{p} \left[Y_j \geq \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right] \end{aligned}$$

avec $t_\ell^j =$ Min $\{p_\ell^j(w_h), h = 1, \ldots, H\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ forment une correspondance de Galois entre $(P(E), \subseteq)$ et $(S, \leq)$.

Exemple :
Considérons à nouveau le tableau de données présenté dans la section 5.
Alors, $g_i(\{\text{Ann, Bob}\}) = s_i =$
$[\text{ Tinnitus } \geq \{\text{ fréquent } (0.8),\text{ rare } (0.0)\}] \wedge [\text{ Maux de Tête } \geq \{\text{ fréquents } (0.0),\text{ rares } (0.1)\}] \wedge$
$[\text{ Pression Sanguine } \geq \{\text{ haute } (0.6),\text{ normale } (0.2),\text{ basse } (0.0)\}]$,
et $f_i(s_i) = \{\text{Ann, Bob}\}$

Pour les démonstrations des théorèmes 1 et 2, voir [BRI 04].

DÉFINITION 4
Un objet probabiliste s est *complet* si $h(s) = g(f(s) = s$.

DÉFINITION 5
Un *concept* est une paire (A, s), où $A \subseteq E$, $s \in S$, s est complet et $A = f(s)$.

Exemple :
D'après les exemples précédents, on peut conclure que s_u est un objet modal complet et que $(\{\text{Ann, Bob}\}, s_u)$ est un concept, pour la correspondance de Galois du théorème 1. De même, s_i est un objet modal complet et $(\{\text{Ann, Bob}\}, s_i)$ est un concept pour la correspondance de Galois du théorème 2.

Les théorèmes 1 et 2 établissent des correspondances de Galois entre deux treillis ; en conséquence, on obtient les théorèmes 3 et 4 ci-dessous ([BIR 40], [BAR 70], [BRI 04]).

THÉORÈME 3
Soit (f_u, g_u) la correspondance de Galois du théorème 1.
Si $s_1 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(r_1^j), \ldots, m_{k_j}^j(r_{k_j}^j)\}\right]$ et $s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(q_1^j), \ldots, m_{k_j}^j(q_{k_j}^j)\}\right]$ on définit
$s_1 \cup s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right]$, avec $t_\ell^j =$ Max $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ et
$s_1 \cap s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(z_1^j), \ldots, m_{k_j}^j(z_{k_j}^j)\}\right]$, avec $z_\ell^j =$ Min $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

Alors, l'ensemble des concepts, ordonnés par $(A_1, s_1) \leq (A_2, s_2) \Leftrightarrow A_1 \subseteq A_2$ forme un treillis, où le supremum et l'infimum de chaque paire d'éléments sont donnés par :

$\inf\left((A_1, s_1), (A_2, s_2)\right) = (A_1 \cap A_2, (g_u \circ f_u)(s_1 \cap s_2))$
$\sup\left((A_1, s_1), (A_2, s_2)\right) = ((f_u \circ g_u)(A_1 \cup A_2), s_1 \cup s_2)$

Ce treillis sera appelé “treillis de l’union”.

THÉORÈME 4 :
Soit (f_i, g_i) la correspondance de Galois du théorème 2.
Si $s_1 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(r_1^j), \ldots, m_{k_j}^j(r_{k_j}^j)\}\right]$ et $s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(q_1^j), \ldots, m_{k_j}^j(q_{k_j}^j)\}\right]$ on définit

$s_1 \cup s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(t_1^j), \ldots, m_{k_j}^j(t_{k_j}^j)\}\right]$, avec $t_\ell^j =$ Min $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$ et

$s_1 \cap s_2 = \bigwedge_{j=1}^{p} \left[Y_j \sim \{m_1^j(z_1^j), \ldots, m_{k_j}^j(z_{k_j}^j)\}\right]$, avec $z_\ell^j =$ Max $\{r_\ell^j, q_\ell^j\}, \ell = 1, \ldots, k_j, j = 1, \ldots, p$.

Alors, l’ensemble des concepts, ordonnés par $(A_1, s_1) \leq (A_2, s_2) \Leftrightarrow A_1 \subseteq A_2$ forme un treillis, où le supremum et l’infimum de chaque paire d’éléments sont donnés par :

$\inf\left((A_1, s_1), (A_2, s_2)\right) = (A_1 \cap A_2, (g_i \circ f_i)(s_1 \cap s_2))$
$\sup\left((A_1, s_1), (A_2, s_2)\right) = ((f_i \circ g_i)(A_1 \cup A_2), s_1 \cup s_2)$

Ce treillis sera appelé “treillis de l’intersection”.

Un algorithme permettant de cosntruire ces deux treillis est ptrsenté dans [BRI 04] ; il utilise l’algorithme de ([GAN 99]) pour déterminer la liste des concepts.

4. Classification hiérarchique / pyramidale

Les résultats présentés dans la Section 3 permettent d’obtenir tous les concepts associés à un tableau de données (étant donnés les fonctions f et g) et d’en construire le treillis de concepts. Mais ce treillis est en général trop complexe et contient trop de classes (noeuds) pour être aisément interprété. Une alternative consiste à limiter le nombre de classes formées, en leur imposant une structure plus simple. Dans cette section, on propose d’utiliser les modèles de classification hiérarchique ou pyramidale, en définissant un algorithme qui permette de construire une hiérarchie ou une pyramide dont les classes soient des éléments du treillis de concepts.

L’objectif général d’une méthode de classification est de grouper les éléments d’un ensemble E en classes homogènes. Dans le cas de la classification hiérarchique ou pyramidale, les classes formées sont organisées dans une structure arborescente. Suivant une approche ascendante, les éléments qui se ressemblent le plus sont d’abord réunis, après les classes similaires sont réunies, jusqu’à ce qu’une seule classe, réunissant tous les éléments de E soit formée. Dans le cas d’une hiérarchie, chaque niveau correspond à une partition ; dans le cas d’une pyramide on a, à chaque niveau, une famille de classes recouvrantes, mais toutes les classes sont des intervalles d’un ordre linéaire sur E ([DID 84, DID 86, BER 85, BER 86]).

La méthode de classification symbolique hiérarchique /pyramidale proposée dans ([BRI 91, BRI 94b]) construit une hiérarchie ou une pyramide en imposant que chaque classe soit un concept pour les opérateurs considérés. Le cas des données décrites par des variables modales a été traité dans ([BRI 98]) et puis dans ([BRI 02]).

Le critère qui guide la formation des classes est la dualité intention-extension : chaque classe de la hiérarchie ou de la pyramide doit correspondre à un concept, c’est-à-dire, chaque classe, qui est un sous-ensemble de E, est représentée par un objet modal complet dont l’extension est la classe elle-même. Ce qui veut dire que chaque classe est par construction associée à un objet qui généralise ses membres, et qu’aucun élément extérieur à la classe n’appartient à son extension.

Les classes, et les concepts correspondants , sont formées de façon récursive. L’ensemble initial des concepts est $\{(\omega_1, s_1), \ldots, (\omega_n, s_n)\}, s_i = s(\omega_i), i = 1, \ldots, n$; on suppose que tous les $(w_i, s_i),\ i = 1, \ldots, n$ sont des

concepts. À chaque étape, un nouveau concept (C, s) est formé, par l'union de concepts existants (C_α, s_α) et (C_β, s_β), avec $C = C_\alpha \cup C_\beta$, $s = s_\alpha \cup s_\beta = g(C) = int(C)$ et en imposant que $f(s) = ext_E s = C$. Selon que les fonctions f et g sont choisies comme dans le théorème 1 ou comme dans le théorème 2, on obtient une hiérarchie ou pyramide qui est un sous ensemble du treillis de l'union ou du treillis de l'intersection, respectivement.

4.1. *Degré de généralité*

Un critère additionnel doit être considéré, qui permette de choisir entre les agrégations possibles à une étape donnée. Le principe sera que les classes associées à des objets plus spécifiques doivent être d'abord formées. Comme la relation d'ordre (voir définition 3) est seulement un ordre partiel, un critère numérique a été défini, le "degré de généralité", qui permet d'évaluer la généralité d'un objet. Ainsi, à chaque étape, parmi les classes qui peuvent être formées, on choisira de former celle dont l'objet modal associé présente un moindre degré de généralité.

Pour des variables modales Y_j avec k_j modalités, $m_1^j, \ldots, m_{k_j}^j$, sur lesquelles on a une distribution de probabilités ou de fréquences $\{m_1^j(p_1^j), \ldots, m_{k_j}^j(p_{k_j}^j)\}$, et $s = \bigwedge_{j=1}^{p} e_j = \bigwedge_{j=1}^{p} [Y_j R_j \{m_1^j(p_1^j), \ldots, m_{k_j}^j(p_{k_j}^j)\}]$ avec $R_j \in \{\sim, \leq, \geq\}, j = 1, \ldots, p$, deux mesures ont été proposées, selon l'opérateur de généralisation utilisé :

- Si $R_j \in \{\sim, \leq\}, j = 1, \ldots, p$, et que la généralisation est effectuée comme indiqué dans le théorème 1 (c'est-à-dire, en prenant le maximum des probabilités ou fréquences associées aux différentes modalités), alors on considère

$$G_1(s) = \prod_{j=1}^{p} G_1(e_j) = \prod_{j=1}^{p} \frac{\sum_{\ell=1}^{k_j} \sqrt{p_\ell^j}}{\sqrt{k_j}} \qquad [1]$$

- Si $R_k \in \{\sim, \geq\}, j = 1, \ldots, p$, et que la généralisation est effectuée comme indiqué dans le théorème 2 (c'est-à-dire, en prenant le minimum des probabilités ou fréquences associées aux différentes modalités) alors on considère

$$G_2(s) = \prod_{j=1}^{p} G_2(e_j) = \prod_{j=1}^{p} \frac{\sum_{\ell=1}^{k_j} \sqrt{(1 - p_\ell^j)}}{\sqrt{k_j(k_j - 1)}} \qquad [2]$$

Ces mesures évaluent, dans chaque cas, la similarité entre la distribution donnée et la distribution uniforme. En fait, $G_1(e_j)$ est le coefficient d'affinité ([MAT 51]) entre $(p_1^j, \ldots, p_{k_j}^j)$ et la distribution uniforme $(\frac{1}{k_j}, \ldots, \frac{1}{k_j})$, $G_2(e_j)$ est le coefficient d'affinité entre $(1 - p_1^j, \ldots, 1 - p_{k_j}^j)$ et$(\frac{1}{k_j}\frac{1}{k_j - 1}, \ldots, \frac{1}{k_j}\frac{1}{k_j - 1})$, respectivement. Cela signifie que l'on considère un objet modal d'autant plus général que les distributions associées sont proches de la distribution uniforme. Pour les distributions où $p_1^j + \ldots + p_{k_j}^j = 1$, $G_1(e_j)$ et $G_2(e_j)$ sont maximaux, $G_1(e_j) = G_2(e_j) = 1$, quand la distribution associée à e_j est la distribution uniforme : $p_\ell^j = \frac{1}{k_j}, \ell = 1, \ldots, k_j$.

4.2. *Algorithme*

L'algorithme suivant construit une hiérarchie indicée (H, I) ou une pyramide indicée au sens large (P, I), où I est la fonction d'indexation $I : H \to \mathbb{R}_0^+$ (respec. $I : P \to \mathbb{R}_0^+$), telle que chaque classe formée correspond à un concept. Soit P_t l'ensemble des classes formées après l'étape t, Q_t le correspondant ensemble de concepts et $S_t \subseteq P_t \times P_t$ l'ensemble de paires d'éléments de P_t qui peuvent être agrégées à l'étape t+1, selon le modèle choisi. Pour simplifier, on supposera que $S_t \neq \emptyset$ à chaque étape.

– Initialisation :
$P_0 = E$, $Q_0 = \{(\omega_1, s_1), \ldots, (\omega_n, s_n)\}$, $S_0 = P_0 \times P_0$, $C_i = \{\omega_i\}$, $I(C_i) = 0, i = 1, \ldots, n$.
– Agrégation/Généralisation :
Après l'étape t : $P_t = \{C_h, h = 1, \ldots, m\}$, $Q_t = \{(C_h, s_h), h = 1, \ldots, m\}$, $S_t = \{(C_h, C_{h'}) \subseteq P_t \times P_t :$
C_h peut être agrégé avec $C_{h'}\}$
Tant que $E \notin P_t$:
1 Soit $(\alpha, \beta) : G(s_\alpha \cup s_\beta) = Min\{G(s_h \cup s_{h'})$ pour $(C_h, C_{h'}) \in S_t\}$
Si $ext_E(s_\alpha \cup s_\beta) = C_\alpha \cup C_\beta$
Alors
$C_{m+1} = C_\alpha \cup C_\beta$
$s_{m+1} = s_\alpha \cup s_\beta$
$I(C_{m+1}) = G(s_\alpha \cup s_\beta)$
$P_{t+1} = P_t \cup \{C_{m+1}\}$
$Q_{t+1} = Q_t \cup \{(C_{m+1}, s_{m+1})\}$
Sinon
$S_t = S_t \setminus (C_\alpha, C_\beta)$
Aller à **1**

5. Application

On a appliqué les méthodes décrites dans les sections précédentes au tableau de données présenté dans [HER 96], qui décrit 5 individus par 3 variables modales, exprimant la fréquence de certains paramètres pertinents pour l'étude de l'hypertension. Les variables sont les suivantes : *Tinnitus - fréquent ou rare, Maux de Tête - fréquents ou rares, Pression sanguine - haute, normale ou basse*.

Le tableau de données probabilistes est le suivant :

	Tinnitus		Maux de Tête		Pression Sanguine		
Nom	Fréquent	Rare	Fréquents	Rares	Haute	Normale	Basse
Anne	0.8	0.2	0.9	0.1	0.8	0.2	0.0
Bob	1.0	0.0	0.0	1.0	0.6	0.4	0.0
Chris	1.0	0.0	0.1	0.9	0.9	0.1	0.0
Doug	0.3	0.7	0.7	0.3	0.0	0.6	0.4
Eve	0.6	0.4	0.7	0.3	0.0	0.8	0.2

Sur ce tableau de données, on a construit le treillis de l'union (see Fig. 1), ainsi que la hiérarchie (see Fig. 2) et la pyramide (see Fig. 3) symboliques construites avec la fonction de généralisation f_u correspondante. Dans le treillis, les concepts sont indiqués par leur extension.

Comme prévu, la hiérarchie est contenue dans la pyramide, qui est elle-même une partie du treillis. Le nombre de concepts formés a été réduit de 31, dans le treillis, à 15 dans la pyramide et seulement 9 dans la hiérarchie. Le treillis organise tous les concepts, mais, sous cette forme, il ne permet pas de décider quels sont les concepts pertinents.

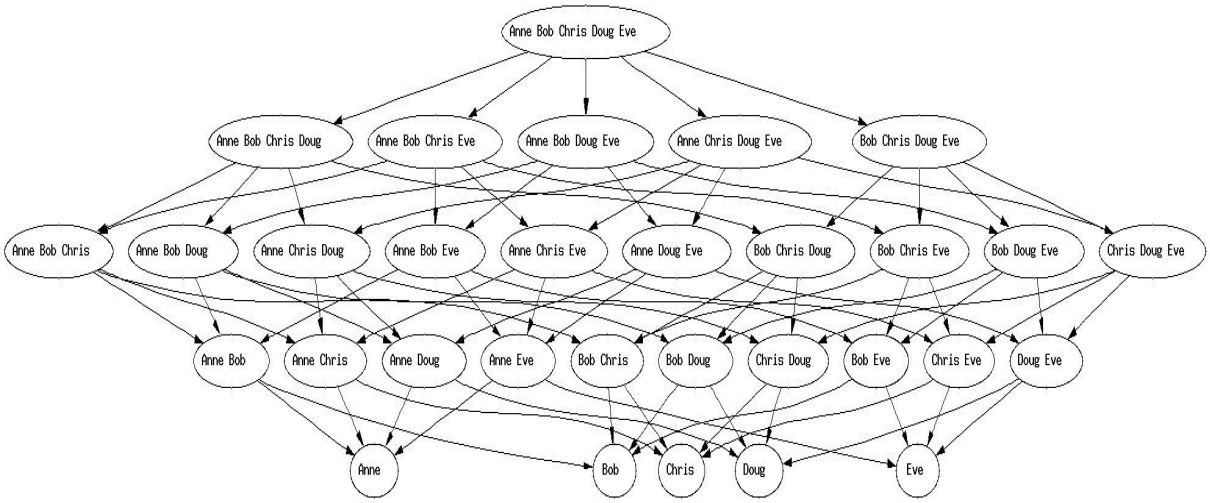


FIG. 1. *Données Herrman, Treillis de l'Union*

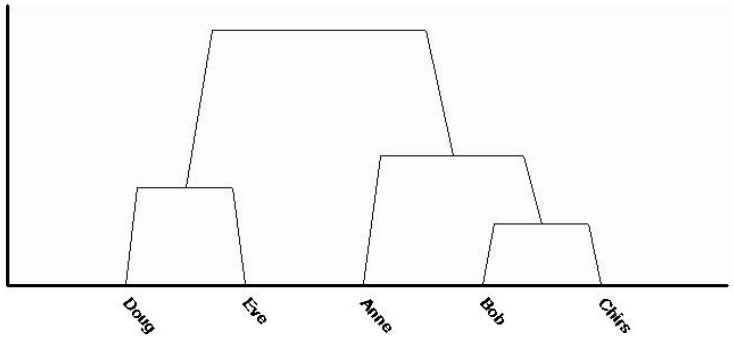


FIG. 2. *Données Herrman, Hiérarchie, Généralisation par le Maximum*

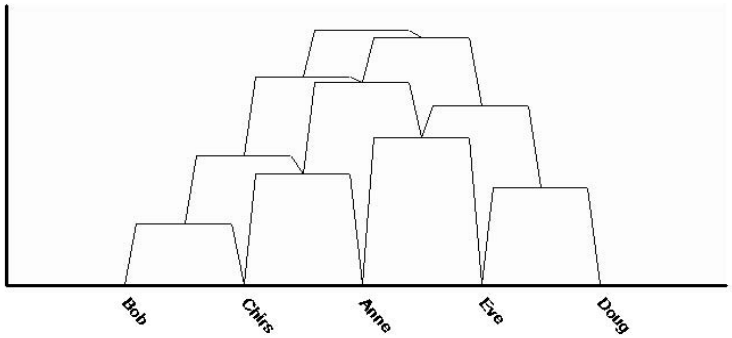


FIG. 3. *Données Herrman, Pyramide, Généralisation par le Maximum*

Dans la hiérarchie, on retient les concepts ((Anne, Bob, Chris), s1) et ((Doug, Eve), s2), avec
$s1 =$ [Tinnitus $\leq$ { fréquent (1), rare (0.2)}] $\wedge$ [Maux de Tête $\leq$ { fréquents (0.9), rares (1)}]$\wedge$
[Pression Sanguine $\leq$ { haute (0.9), normale (0.4), basse (0.0)}];
$s2 =$ [Tinnitus $\leq$ { fréquent (0.6), rare (0.7)}] $\wedge$ [Maux de Tête $\leq$ { fréquents (0.7), rares (0.3)}]$\wedge$
[Pression Sanguine $\leq$ { haute (0.0), normale (0.8), basse (0.4)}]

La pyramide permet la formation de concepts qui ne sont pas présents dans la hiérarchie, retenons par exemple le concept ((Anne, Eve), s3), avec
$s3 =$ [Tinnitus $\leq$ { fréquent (0.8), rare (0.4)}] $\wedge$ [Maux de Tête $\leq$ { fréquents (0.9), rares (0.3)}]$\wedge$
[Pression Sanguine $\leq$ { haute (0.8), normale (0.8), basse (0.2)}]
L'ordre induit par par pyramide, Bob-Chris-Anne-Eve-Doug, paraît traduire une importance décroissante du symptome *Tinnitus - fréquent*.

6. Conclusion

Dans ce papier, on a présenté des résultats permettant de construire deux treillis de Galois sur des données modales. L'avantage principal de l'approche proposée réside dans le fait qu'elle permet d'organiser les données modales directement, sans qu'aucune transformation préalable ne soit nécessaire. L'application pratique de la méthode reste cependant limitée par la taille des treillis obtenus, puisque le nombre de concepts tend à augmenter exponentiellement avec le nombre d'individus et de variables.

Une alternative consiste à limiter le nombre de concepts formés, en imposant un modèle de classification plus simple. Dans ce sens, une méthode de classification est proposée, qui utilise les modèles hiérarchique ou pyramidale. La méthode construit une hiérarchie ou une pyramide dont chaque classe est un concept du treillis correspondant, et permet d'ordonner les concepts par le *degr é de généralité* de leurs intentions, mesuré par l'affinité des distributions associées avec la distribution uniforme.

Le pas suivant consistera à prendre en compte l'ordre entre les modalités, quand il existe, dans la formation des concepts et dans l'évaluation de la généralité.

7. Bibliographie

[BAR 70] BARBUT M., MONJARDET B., *Ordre et Classification, Algèbre et Combinatoire*, vol. I et II, Hachette, Paris, 1970.

[BER 85] BERTRAND P., DIDAY E., A Visual Representation of the Compatibility between an Order and a Dissimilarity Index : The Pyramids, *Computational Statistics Quarterly*, vol. 2, n^o 1, 1985, p. 31-42.

[BER 86] BERTRAND P., Étude de la Représentation Pyramidale, PhD thesis, Université Paris IX Dauphine, Paris, France, 1986.

[BIR 40] BIRKHOFF G., *Lattice theory*, vol. XXV, 1st edition (3rd edition, 1967), American Mathematical Society Colloquium Publications, 1940.

[BOC 00] BOCK H. H., DIDAY E., *Analysis of Symbolic Data - Exploratory Methods for Extracting Statistical Information from Complex Data*, Springer Verlag, Berlin-Heidelberg, 2000.

[BRI 91] BRITO P., Analyse de données symboliques. Pyramides d'héritage, PhD thesis, Université Paris IX Dauphine, Paris, France, 1991.

[BRI 94a] BRITO P., Order Structure of Symbolic Assertion Objects, *IEEE Transactions on Knowledge and Data Engineering*, vol. 6, n^o 5, 1994, p. 830-835.

[BRI 94b] BRITO P., Use of Pyramids in Symbolic Data Analysis, DIDAY E., *et al*., Eds., *New Approaches in Classification and Data Analysis*, Springer-Verlag, Berlin-Heidelberg, 1994, p. 378-386.

[BRI 98] BRITO P., Symbolic Clustering of Probabilistic Data, RIZZI A., VICHI M., BOCK H.-H., Eds., *Advances in Data Science and Classification*, Springer-Verlag, Berlin-Heidelberg, 1998, p. 385-390.

[BRI 99] BRITO P., DE CARVALHO F., Symbolic Clustering in the Presence of Hierarchical Rules, *Proceedings of the Conference on Knowledge Extraction and Symbolic Data Analysis (KESDA'98)*, Office for Official Publications of the European Communities, Luxembourg, 1999, p. 119-128.

[BRI 02] BRITO P., DE CARVALHO F. A. T., Symbolic Clustering of Constrained Probabilistic Data, OPITZ O., SCHWAIGER M., Eds., *Exploratory Data Analysis in Empirical Research*, Springer Verlag, Heidelberg, 2002, p. 12-21.

[BRI 04] BRITO P., POLAILLON G., Structuring probabilistic data by Galois lattices, *Mathématiques, Informatique et Sciences Humaines*, vol. à paraître, 2004.

[CHA 98] CHAVENT M., A monothetic clustering method, *Pattern Recognition Letters*, vol. 19, 1998, p. 989-996.

[DID 84] DIDAY E., Une Représentation Visuelle des Classes Empiétantes : Les Pyramides, rapport n^{o}291, 1984, INRIA, Rocquencourt, Le Chesnay.

[DID 86] DIDAY E., Orders and Overlapping Clusters by Pyramids, LEEUW J. D., *et al.*, Eds., *Multidimensional Data Analysis*, DSWO Press, Leiden, 1986, p. 201-234.

[DUQ 86] DUQUENNE V., GUIGUES J., Familles minimales d'implication informatives résultant d'un tableau de données binaires, *Mathématiques, Informatique et Sciences Humaines*, vol. 95, 1986, p. 5-18.

[DUQ 87] DUQUENNE V., Contextual implications between attributes and some representation properties for finite lattices, GANTER B., WILLE R., WOLFF K. E., Eds., *Beitrage zur Begriffsanalyse*, Darmstadt, 1987, p. 149-172.

[ELS 98] EL-SONBATY, Y. ISMAIL M. A., On-line hierarchical clustering, *Pattern Recognition Letters*, vol. 19, 1998, p. 1285-1291.

[GAN 99] GANTER B., WILLE R., *Formal Concept Analysis - Mathematical Foundations*, Springer Verlag, New York, 1999.

[GIR 99] GIRARD R., RALAMBONDRAINY H., Recherche de concepts à partir de données arborescentes et imprécises, *Mathématiques, Informatique et Sciences Humaines*, vol. 147, 1999, p. 87-111.

[GOD 95] GODIN R., MINEAU G., MISSAOUI R., MILI H., Méthodes de classification conceptuelle basées sur les treilllis de Galois et applications, *Revue d'intelligence artificielle*, vol. 9(2), 1995, p. 105-137.

[GOW 91] GOWDA K. C., DIDAY E., Symbolic clustering using a new dissimilarity measure, *Pattern Recognition*, vol. 24, n^o 6, 1991, p. 567-578.

[GOW 92] GOWDA K. C., DIDAY E., Symbolic clustering using a new similarity measure, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, n^o 2, 1992, p. 368-378.

[GOW 95a] GOWDA K. C., RAVI T. R., Agglomerative clustering of symbolic objects using the concepts of both similarity and dissimilarity, *Pattern Recognition Letters*, vol. 16, 1995, p. 647-652.

[GOW 95b] GOWDA K. C., RAVI T. R., Divisive clustering of symbolic objects using the concepts of both similarity and dissimilarity, *Pattern Recognition*, vol. 28, n^o 8, 1995, p. 1277-1282.

[GUé 93] GUÉNOCHE A., Hiérarchies conceptuelles de données binaires, *Mathématiques, Informatique et Sciences Humaines*, vol. 121, 1993, p. 23-34.

[HER 96] HERRMANN C. S., HÖLLDOBLER S., STROHMAIER A., Fuzzy conceptual knowledge processing, *Proceedings of the ACM Symposium on Applied Computing, Philadelphia*, ACM Press, New York, 1996, p. 628-632.

[MAT 51] MATUSITA K., Decision rules based on distance for problems of fit, two samples and estimation, *Ann. Math. Stat.*, vol. 3, 1951, p. 1-30.

[MEP 93] MEPHU NGUIFO E., Une nouvelle approche basée sur le treillis de Galois, pour l'apprentissage de concepts, *Mathématiques, Informatique et Sciences Humaines*, vol. 124, 1993, p. 19-38.

[POL 98a] POLAILLON G., Interpretation and reduction of Galois lattices of complex data, RIZZI A., VICHI M., BOCK H.-H., Eds., *Advances in Data Science and Classification*, Springer-Verlag, Berlin-Heidelberg, 1998, p. 433-440.

[POL 98b] POLAILLON G., Organisation et interprétation par les treillis de Galois de données de type multivalué, intervalle ou histogramme, PhD thesis, Université Paris IX Dauphine, Paris, France, 1998.

[POL 99] POLAILLON G., DIDAY E., Reduction of symbolic Galois lattices via hierarchies, *Proceedings of the Conference on Knowledge Extraction and Symbolic Data Analysis (KESDA'98)*, Office for Official Publications of the European Communities, Luxembourg, 1999, p. 137-143.

[WIL 82] WILLE R., Restructuring lattice theory : an approach based on hierarchies of concepts, REIDEL D., Ed., *Ordered Sets*, Dordrecht-Boston, 1982, p. 445-470.

Les contrôles de jury de dégustations de vins

Bernard Burtschy

Ecole Nationale Supérieure des Télécommunications
46, rue Barrault 75013 Paris France
burtschy@enst.fr

RÉSUMÉ. Le contrôle du Grand Jury Européen, un organisme qui regroupe une trentaine de dégustateurs européens de vins les plus en vue, est entièrement effectué par analyse de données. L'analyse en composantes principales sert à vérifier la cohérence du jury et à établir le classement des vins, les techniques de classification à déterminer leur style.

MOTS-CLÉS : classification, vins , analyse en composantes principales

1. Introduction

La constitution d'un jury de dégustation et l'agrégation des notes sont le talon d'Achille de tous les concours internationaux. Le premier est en général résolu en prenant des dégustateurs provenant de l'univers du vin, sommeliers, œnologues, ce qui ne constitue nullement un brevet de dégustateur et il n'existe pas de formation en ce domaine. L'agrégation des notes s'effectue par moyenne, d'où l'effet pervers bien connu qui donne une pondération proportionnelle à la dispersion des notes de chaque dégustateur. L'utilisation des rangs ne résout que partiellement le problème. Divers auteurs ont proposé d'utiliser les techniques d'analyse des données (Cliff and King, 1999; Scaman and al. 2001; Giacco and del Signore 2004). Ces techniques sont utilisées dans le Grand Jury Européen depuis son origine en 1996.

2. Le Grand Jury Européen

Le Grand Jury Européen, association sans but lucratif de droit luxembourgeois créée par François Mauss en 1996, est un collège de grands dégustateurs ayant pour but d'offrir aux amateurs de grands vins, une cotation alternative aux notes des critiques individuels. Il regroupe une trentaine dégustateurs européens provenant de l'ensemble de la filière vins, critiques, sommeliers, restaurateurs, marchands, grands producteurs, journalistes spécialisés. La plupart d'entre eux sont connus dans leurs pays respectifs pour leurs guides de vins, leurs articles et leurs chroniques.

Le Grand Jury se devant de rester strictement indépendant du monde direct des producteurs et n'ayant aucun but lucratif, s'appuie sur des partenaires, sponsors, fournisseurs venus d'autres horizons. Le Grand Jury publie un communiqué de presse à la suite de chaque session. Les résultats font l'objet d'une présentation et d'une analyse approfondie dans la publication régulière du Grand Jury et sur son site Internet : grandjuryeuropeen.com. Le site winexcellence.com présente une analyse complète de toutes les sessions effectuées depuis les débuts du GJE en 1996.

2.1. *Des conditions strictes de dégustation*

Le Grand Jury s'est doté, dès sa première session, de règles très strictes de dégustation.
- des dégustations strictement à l'aveugle, seuls le thème et le millésime étant connus, parfois il n'y a aucune indication,
- une sélection cohérente de vins, sur un thème donné, le choix des vins se faisant après consultation des Membres Permanents,
- une organisation des dégustations en demi-journées où sont goûtés, par session, un maximum de 30 vins,
- un système de notation associant une note sur 100 et un commentaire, couplé à un traitement statistique élaboré corrigeant les défauts des simples additions de points ou de rangs,

- un contrôle juridique des opérations par une entité légale assermentée,
- l'achat des vins sur le marché,
- l'assemblage des deux bouteilles nécessaires au service du vin (quand la session réunit plus de 15 dégustateurs) afin d'éviter des distorsions dues à des vins qui auraient évolué différemment d'une bouteille à l'autre,
- ordre de service tiré au hasard par l'entité juridique contrôlant la dégustation,
- obligation faite aux dégustateurs de commencer la dégustation par le verre portant le même numéro que leur table, afin de ne pas pénaliser le premier vin goûté.

2.2. *La cohérence du jury*

«Une somme de subjectivités est un début d'objectivité» : la mise en pratique de ce concept se traduit par la constitution d'un Jury, opération délicate car il faut que ses membres aient une certaine homogénéité dans leur concept du « grand vin» et puissent cependant exprimer leur propre sensibilité en fonction de leur culture nationale et des spécificités de leur profession. A priori, on peut penser qu'un restaurateur allemand ne goûte pas de la même manière qu'un journaliste italien, qu'un sommelier français est différent d'un oenologue espagnol. En outre, la compétence et l'éthique professionnelle d'un tel collège de dégustateurs se doivent d'être à la fois reconnues par le monde des producteurs et par celui des amateurs.
La constitution d'un jury est une opération longue et fastidieuse, toujours renouvelée, qui est entièrement pilotée par le traitement statistique des données de dégustation. Les membres du jury sont systématiquement contrôlés selon plusieurs critères qui s'appliquent à l'expérimentation scientifique en général avec de véritables plans d'expérience :
- La répétitivité.
Un dégustateur se doit de donner systématiquement la même note au même vin. Chaque session comprend au moins deux fois le même vin, certains vins sont identiques d'une séance à l'autre. La disparité de notation de vins identiques conduit à l'élimination du jury. Il faut remarquer que seuls des dégustateurs de haut niveau réussissent avec régularité cette épreuve pourtant toute simple qui est à la base du contrôle d'un jury.
- La cohérence.
Des vins pirates sont introduits dans la dégustation et il faut les repérer que ce soit par leur non typicité, un bourgogne dans des bordeaux pour faire simple, soit par leur différence de niveau, un bordeaux « de base » dans des crus classés.
- Capacité de hiérarchisation et pertinence.
Le principe de la dégustation est la notation des vins et donc leur hiérarchisation. Certains hiérarchisent peu, d'autres beaucoup. Evidemment, tout statisticien sait que la hiérarchisation peut être augmentée facilement, comme tout indicateur de dispersion, d'une manière artificielle par dilatation. Ce critère est donc indissociable de la pertinence de la dégustation, critère difficile à mesurer, sauf dans un jury collectif et par analyse multidimensionnelle. Nous sommes au cœur du contrôle du jury.

2.3. *La notation des vins*

La notation des vins, en particulier, pose problème. La France en particulier et l'Europe en général ont l'habitude d'utiliser les échelles de 0 à 20 ou de 0 à 10. La tradition de l'université américaine, adoptée en particulier par Robert Parker, est de noter sur une échelle de 100. Cette échelle comporte ses propres règles qui ne sont pas toujours comprises. Les notes s'échelonnent de 50 à 100 en théorie, mais en pratique, il n'existe pas de note inférieure à 72 (zéro pointé). Le maximum est de 100, rarissime lui aussi. L'écart est donc de 25 à 30 points. L'échelle adoptée par l'OIV (l'Office International du Vin) en 1993 et qui est de règle dans les concours internationaux, est aussi une échelle sur 100, mais décomposée selon les principes basiques de la dégustation (visuel, arômes, gustatif, synthèse) que d'ailleurs peu d'experts utilisent.
Il est demandé au GJE de noter les vins sur une échelle de 100 pour que les notes soient comparables à la notation américaine. Une analyse statistique des notes des dégustateurs montre que trois systèmes de notation coexistent :
- un premier groupe note selon les règles traditionnelles de la notation américaine. Les notes s'échelonnent entre 72 et 100, soit 28 points d'écart.
- un second groupe utilise une échelle qui lui était probablement plus familière, de 0 à 10 ou de 0 à 20, et a multiplié, selon le cas, sa note par un facteur de 10 ou de 5. Les notes s'étagent entre 20 et 100, soit 80 points d'écart.
- un troisième groupe enfin a effectué une sorte de mélange entre les deux approches précédentes et note entre 50 et 100, soit 50 points d'écart, quelques-uns uns utilisant l'échelle de l'Office International du Vin (OIV) qui conduit au même résultat.

Il est impossible d'imposer, à un dégustateur, un autre système de notation que celui dont il a l'habitude. Toutes les tentatives qui ont été faites, ont été vouées à l'échec. Il vaut mieux qu'il utilise son système de notation

habituel, plutôt que de le distordre, pour entrer dans un autre, car cette distorsion est non linéaire et peu fiable. Au traitement statistique de se débrouiller pour normaliser les données.

3. Le contrôle du jury

Chaque dégustateur notant l'ensemble des vins, le tableau de données qui en résulte, est un tableau classique pour l'analyse des données multidimensionnelles (Tab. 1). Chaque ligne décrit la dégustation d'un vin. Les colonnes décrivent successivement le nom du vin dégusté, la séance où il a été dégusté, l'ordre de dégustation et les notes de divers dégustateurs. On notera que le vin Haut-Condissas a été dégusté dans la séance B et dans la séance C. Ce tableau, ainsi que les analyses le concernant, sont extraits d'une dégustation des Bordeaux 1999, sur quatre séances, qui a eu lieu à Villa d'Este (Italie) en mars 2004.

Nom du vin	Séance	Ordre	D1	D2	D3	D4	D5	D6	Etc.
Ch. Canon-La-Gaffelière	**C**	16	88	96	91	86	92	88	88
Ch. Pavie-Decesse	**A**	25	90	91	92	92	89	90	95
Ch. De Valandraud	**D**	26	85	94	97	87	90	86	92
Ch. Ausone	**C**	4	88	91	97	85	91	90	90
La Mondotte	**A**	5	87	93	92	89	88	88	89
Ch. Sociando-Mallet	**C**	32	93	88	96	83	85	87	88
Haut-Condissas	**B**	29	91	90	93	85	87	86	92
Haut-Condissas	**C**	34	91	91	93	85	88	86	93
Ch. Giscours	**D**	18	83	92	85	90	88	92	90
Ch. Léoville-las-Cases	**C**	22	88	93	90	92	87	86	89

TAB 1. Tableau de données (Bordeaux 1999 dégustés en mars 2004)

3.1. *Le pilotage par analyse en composantes principales*

Ce tableau est soumis à l'analyse en composantes principales (Saporta, 1990, Lebart, Morineau et Piron 2000, Jolliffe 2002) ce qui permet à la fois une représentation des membres du Jury et des vins. Les unités statistiques sont les vins, les données sont normalisées par dégustateur en raison de la forte hétérogénéité des échelles de notation.

Par ses notes, chaque dégustateur fournit un classement des vins. Ce classement est symbolisé sur le graphique par une flèche (Fig. 1). Deux dégustateurs sont d'autant plus cohérents entre eux que leurs classements des vins sont plus proches, comme par exemple les dégustateurs D1 et D12. Un dégustateur hypothétique qui serait en désaccord total avec D1 et D12, aurait sa flèche en sens inverse. Dans un jury cohérent, toutes les flèches vont du même côté, ce qui est le cas ici, et l'axe horizontal reflète ce consensus. La longueur de la flèche représente la capacité de hiérarchisation d'un dégustateur, certains prenant franchement position, d'autres non, les notes ayant été préalablement normalisées.

La même analyse en composantes principales fournit la représentation des vins (Fig. 2), le premier axe donnant le classement qui sera publié, le deuxième axe donnant une idée de la disparité des notations. Dégustation après dégustation, les deux premiers axes donnent l'essentiel de l'information.

Les deux représentations, celle des dégustateurs et celle des vins, peuvent être superposées avec les précautions d'usage. En utilisant cette méthode statistique bien balisée, cette double représentation avec à la fois les vins et les dégustateurs permet de comprendre très intimement l'appellation analysée en faisant apparaître, en un clin d'œil, ce qu'il faut souvent de longues années de dégustation pour appréhender.

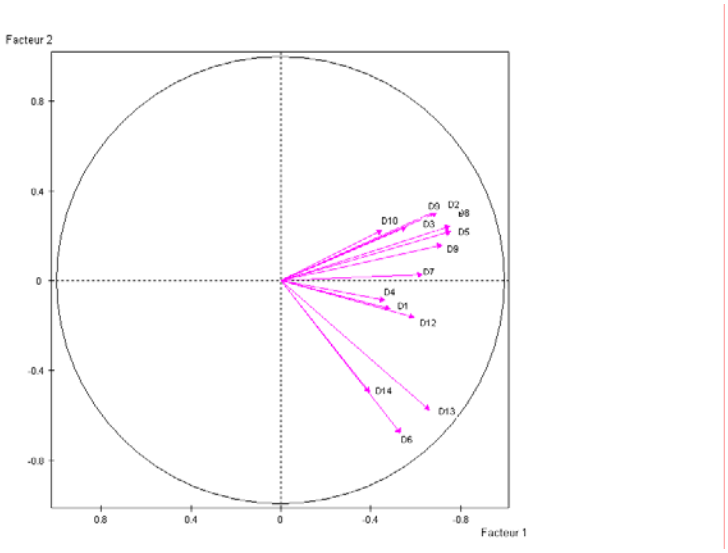


FIG. 1 : La représentation des dégustateurs

Mais l'analyse des résultats est bien plus riche. Elle permet de « gérer » au mieux un jury de dégustation en vérifiant la pertinence de chaque membre du jury : gare à celui qui se trouverait sur la partie gauche du graphique ! Elle permet aussi de mettre en évidence l'indispensable pluralité des opinions, sans que cette pluralité ne devienne cacophonie.
Les résultats fournissent aussi une véritable « carte du tendre des vins » en regroupant les vins similaires, en faisant apparaître leurs styles, les styles des dégustateurs, les analogies et les oppositions. A cet égard, cette carte est, dans le domaine, d'une richesse incomparable. Il est bien entendu possible d'utiliser des méthodes de représentation plus sophistiquées, mais l'analyse en composantes principales est à la fois une méthode statistique bien balisée pour le statisticien et déjà considérée comme révolutionnaire dans le monde du vin.

3.2. *Vers des résultats plus généraux*

Dans les nombreuses analyses effectuées suite à de nombreuses dégustations, elle a permis d'engranger nombre de résultats.

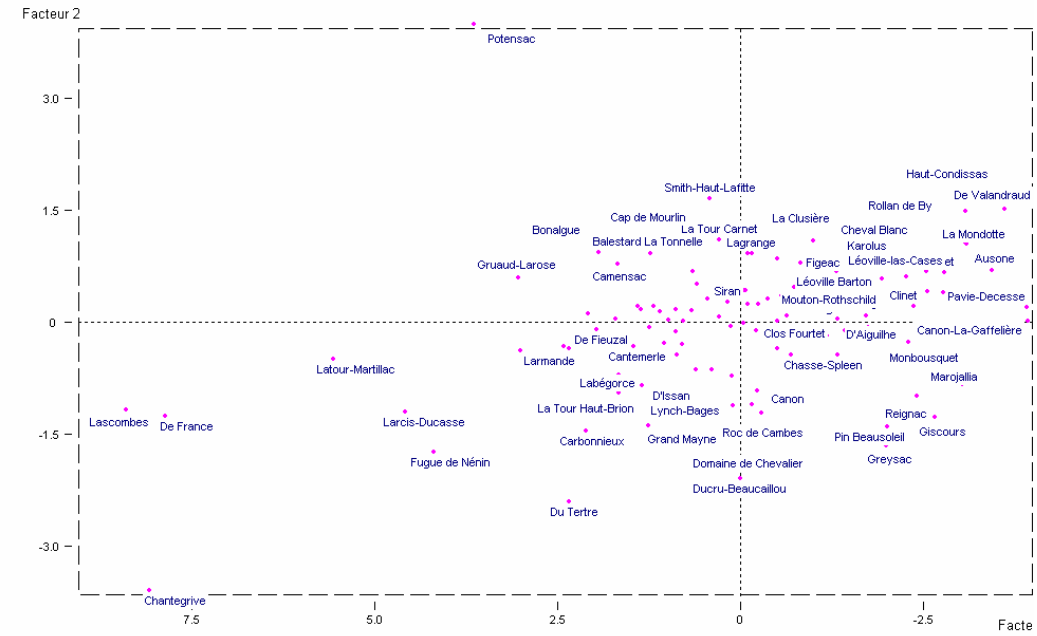


FIG. 2 : La représentation des vins

- il n'existe pas de différence culturelle dans la dégustation des vins rouges entre les différents pays d'Europe. Il existe des différences de sensibilité, mais elles ne s'expliquent ni par la nationalité, ni par le métier d'origine des dégustateurs.
- des dégustations conjointes avec un jury américain, pendant une semaine en novembre 2001 à Las Vegas, ont montré que les jurys européens et américains donnaient des résultats très similaires, que se soient dans les vieux vins, les vins récents européens ou les cabernets américains. En revanche, sur les cabernets américains, positionnée en éléments supplémentaires grâce aux notes publiées, la presse spécialisée américaine se situe dans une position caricaturale en privilégiant les vins outrageusement concentrés.
- La dégustation des vins en dehors de la sphère de référence du GJE, tels que les Zinfandels américains en raison de leur fort niveau d'alcool, déstabilise certains membres du jury, alors que le jury américain est plus cohérent et que la presse américaine ne prend guère position.
- Si la dégustation des vins rouges, quels que soient leur origine, conduit toujours à une grande cohérence du GJE, les blancs montrent une forte opposition culturelle entre le nord de l'Europe qui préfère des vins plus souples et le sud qui recherche des vins à l'acidité plus vive.

4. Les styles des vins

L'analyse des dégustations collectives permet de dégager, très rapidement, les styles des vins. Penons comme exemple les grands bourgognes rouges, qui sont d'une complexité effroyable, le même cépage pinot noir prenant d'infinies nuances selon le terroir et le vinificateur. Il faut beaucoup de temps pour en pénétrer les arcanes et une vie de dégustateur assidu suffit à peine pour en décoder les variations. Des classifications effectuées (Hartigan 1975, Gordon 1999) sur le tableau de données permettent, en deux temps, trois mouvements, de définir les styles avec des techniques statistiques usuelles.
L'arbre hiérarchique qui s'en dégage permet d'un coup d'œil, de repérer les deux grands styles de Bourgogne, avec d'un côté les anciens, les traditionnels aux vins peu colorés basés sur la finesse et les modernes aux vins fortement colorés et puissants, ces deux styles opposés divisant la région et ses aficionados. Les branches permettent ensuite de distinguer les vins de classe, les vins denses, les veloutés et bien d'autres qui font les délices des dégustations. Sans analyse de données, ces distinctions sont beaucoup plus incertaines et seuls quelques dégustateurs blanchis sous le harnais de nombreux flacons, s'y hasardent.
D'une manière identique, les méthodes de partitionnement permettent d'isoler des groupes de vins similaires, de préciser quels sont les dégustateurs qui les préfèrent et ceux qui les arborent. Cet arsenal, très classique pour les praticiens de l'analyse de données, n'est pas plus détaillé par manque de place, ce qui n'empêche pas ses résultats d'être impressionnants.

5. Perspectives

Toutes ces analyses sont basées sur la note donnée à chaque vin, ce qui est une simple note numérique. Il est d'ailleurs remarquable de voir la richesse des résultats en exploitant ces notes et leurs cohérences. Pourtant, on est loin d'avoir exploité toute l'information. Chaque vin dégusté est accompagné, par chaque dégustateur d'une note en clair qui en définit ses caractéristiques. Il est tentant de les exploiter par les méthodes de la statistique textuelle (Lebart et Salem, 1994).Le caractère multilingue, l'Europe est une mosaïque de langages, en interdit l'exploitation directe. La solution passe probablement, dans un premier temps, par l'utilisation d'un vocabulaire réduit traduit dans chaque langue.

6. Bibliographie

Cliff, M.A. and M.C. King (1999) Use of principal component analysis for evaluation of judge performance at wine competitions. *J Wine Research*. 10(1):25-32.

Giaccio M., Del Signore A (2004). Multivariate classification of Montepulciano d'Abbruzzo wine samples according to vintage year. *Journal of the Science of Food and Agriculture*. Vol 84, 2, pp. 164-172.

Gordon, A.D. (1999). Classification. Second edition. London: Chapman and Hall.

Hartigan, J.A. (1975). Clustering Algorithms. New York:Wiley.

Jolliffe I.T. (2002). Principal Component Analysis. Springer Verlag, Berlin.

Lebart L., Salem A. (1994). Statistique textuelle. Dunod, Paris.

Lebart L., Morineau A., Piron M (2000). Statistique exploratoire multidimensionnelle. Dunod, Paris, 3ème édition.

Saporta G. (1990). Probabilités, analyse des données et statistique. Technip, Paris.

Scaman C.H., J. Dou, M.A. Cliff, D. Yuksel and M.CC. King (2001). Evaluation of Wine Competition Judge Performnace Using Principal Component Similarity Analysis. *Journal of Sonsory Studies.* V16, N 3, June 2001, pp 287-300.

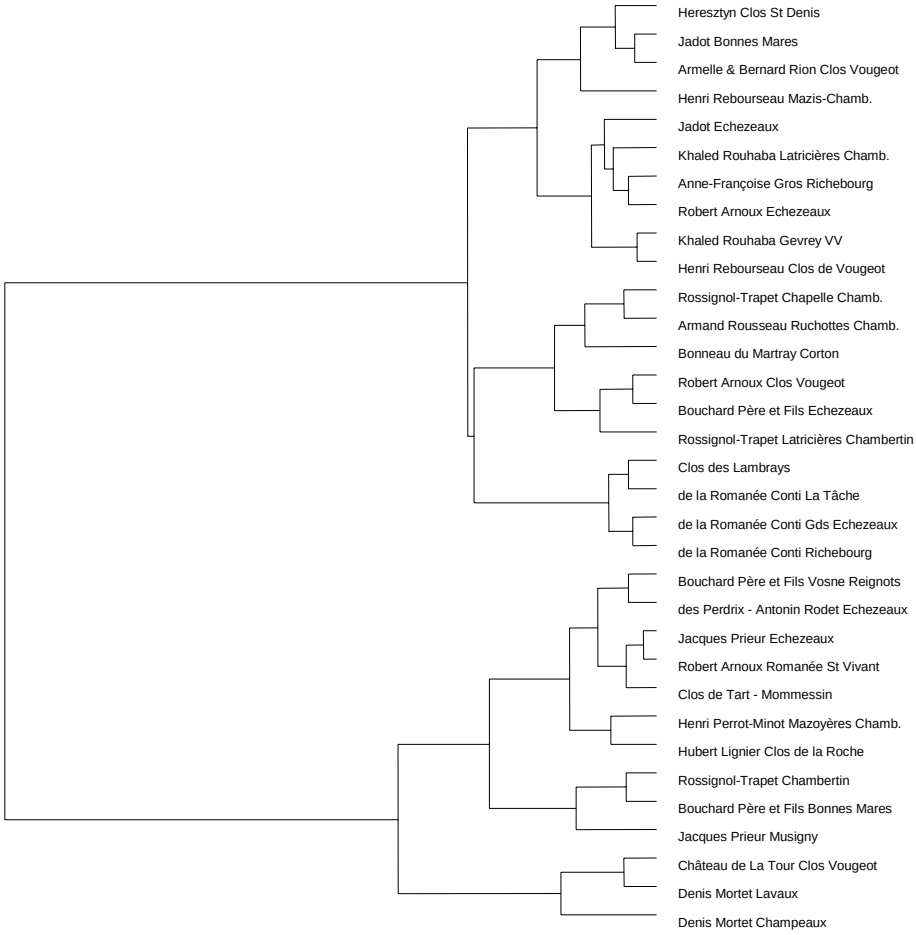


FIG. 3 : le style des grands vins de Bourgogne

Reconnaissance des Formes et Analyse d'Images à Tours

Hubert CARDOT

LI, Université de Tours
Polytech'Tours - DI
64 avenue Jean Portalis
37200 Tours
hubert.cardot@univ-tours.fr

RÉSUMÉ. Plusieurs études sont présentées successivement : les SVM appliqués à de grandes bases de données, les réseaux de neurones récurrents pour la prévision, la fusion d'information en utilisant la théorie de l'évidence, une adaptation des contours actifs en segmentation d'images, une évolution des liens hypertextes afin de mieux représenter la connaissance, une analyse de vidéos pour aider à l'évaluation de l'autisme chez les enfants, une analyse de signaux vidéos et sonores dans le domaine médical, et enfin, l'authentification de personnes à partir de signatures manuscrites ou de l'utilisation de périphériques simples.

MOTS-CLÉS : Reconnaissance des formes, analyse d'images, classification, authentification.

1. Introduction

Cet article est une description sommaire de plusieurs études menées principalement au sein de l'équipe RFAI (Reconnaissance des Formes et Analyse d'Images) du LI (Laboratoire d'Informatique) de l'Université de Tours. Cette équipe est constituée d'une vingtaine de chercheurs en comptant les doctorants. L'objectif est de montrer à la communauté une partie significative de nos activités actuelles sachant que des descriptions plus approfondies pourront être trouvées dans la bibliographie.

La première partie va s'intéresser aux outils et méthodes développés pour la reconnaissance des formes. En particulier, nous verrons des études sur la classification de données, la prévision dans des séries temporelles, la fusion d'information, la segmentation d'images, et la structuration des connaissances.

Dans la deuxième partie, c'est plutôt l'aspect applicatif qui est décrit : analyse de vidéos pour aider à l'évaluation de l'autisme chez les enfants, analyse de signaux vidéos et sonores dans le domaine médical, et authentification de personnes à partir de signatures manuscrites ou de l'utilisation de périphériques simples.

2. Reconnaissance des Formes

2.1. *SVM*

Les Machines à Vecteurs Supports (SVM) [VAP 98] sont une approche très efficace pour résoudre les problèmes de classification. Nos travaux sur les SVM se font en collaboration avec des chercheurs du LUSAC de l'Université de Caen.

L'utilisation des SVM devient problématique quand on l'applique sur des grandes bases de données, même si on se sert d'algorithmes dédiés à ce type de problème (comme SMO [PLA 99]). En effet, la complexité, en temps de calcul, augmente plus que proportionnellement avec la taille de la base d'apprentissage. De plus, le

nombre de vecteurs supports augmente aussi avec la taille de la base d'apprentissage surtout si les données sont bruitées.

Notre idée principale pour résoudre ce problème repose sur l'utilisation de la Quantification Vectorielle (QV) [GER 91] pour construire une base d'apprentissage de taille réduite en produisant un ensemble de prototypes exemples. Les résultats expérimentaux obtenus (tableau 1) montrent que la méthode proposée réduit fortement le temps d'apprentissage avec une très faible détérioration des taux de classification [LEB 04]. De plus cette méthode produit une fonction de décision de complexité réduite lorsque les données sont bruitées.

	notre méthode		méthode classique	
bases	temps	erreur	temps	erreur
satimage	8.5	10.7	29.9	10.1
letter	19	6.34	827	6.34
shuttle	3.8	0.09	412	0.09

Tableau 1 : Taux d'erreur et temps d'apprentissage (en heures)

2.2. *Réseaux de neurones récurrents*

Les réseaux de neurones récurrents ou bouclés sont par leur structure adaptés au traitement de données séquentielles [BON 03]. En particulier, la prévision de séries temporelles est une application indispensable dans de nombreux domaines tels que la météo, la finance ou le marketing.

Le plus souvent le modèle mathématique qui a généré la série est inconnu. Les relations entre les valeurs passées de la série et les valeurs futures doivent alors être déduites par apprentissage à partir des valeurs passées. Cette relation peut être décrite par une fonction f telle que : $x_t = f(x_{t-1}, x_{t-2}, \ldots)$ avec x_t la valeur future à estimer et $(x_{t-1}, x_{t-2}, \ldots)$ les valeurs récentes de la série temporelle.

Le nombre de valeurs récentes à prendre en compte dans la fonction f dépend du contexte. L'algorithme BPTT (Back Propagation Through Time) [WER 90] [WIL 90] permet d'entraîner des réseaux de neurones récurrents en s'adaptant à ce contexte.

Nous développons plusieurs approches pour en améliorer les performances [ASS 03]. Une de ces approches est connue sous le nom de *Boosting* [SCH 90] [FRE 90]. Le principe est de réaliser l'apprentissage plusieurs fois sur tout ou partie de la base d'apprentissage en favorisant à chaque étape les exemples qui ont été précédemment estimés difficiles. Ensuite, il faut combiner les modèles issus de ces apprentissages.

Les résultats actuels, obtenus sur deux séries temporelles (taches solaires, Mackey-Glass 17), montrent une amélioration des performances grâce à notre algorithme de *Boosting* comparées à l'utilisation d'un seul réseau de neurones récurrent déjà très performant.

Nous continuons la recherche d'améliorations à appliquer au problème de l'apprentissage des dépendances temporelles pour la prévision de valeurs futures. Une application en cours de développement est la reconnaissance de phonèmes à partir de séquences vidéo.

2.3. *Fusion d'information*

Nous travaillons sur la fusion de l'information et sur son application au problème de l'authentification de signatures graphiques hors-ligne.

La base de données consiste en 525 images réelles de signatures hors-ligne manuscrites obtenues auprès de 35 personnes qui ont signé chacune une quinzaine fois. Deux classifieurs de distance à base de distance euclidienne sont employés pour la classification de ces images de signatures après l'extraction de leurs caractéristiques. Le premier classifieur représente les caractéristiques globales et générales de la forme de signature, extraites avec des primitives basées sur l'histogramme. Le deuxième classifieur représente les caractéristiques globales et locales, et aussi la lisibilité et la complexité d'une signature, extraites avec un ensemble de primitives comme la dimension fractale, la dimension de masse, la pente de la signature et la fraction du contour d'une image de la signature, etc. La performance de ces deux classifieurs a été testée avec la méthode du « Leave-one-out ». Les taux de reconnaissance de ces deux classifieurs sont de 69,52 % et 69,33 % respectivement. Ici, la décision repose sur un classifieur des 5 plus-proches-voisins à la majorité simple.

Le taux de reconnaissance des classifieurs a été amélioré en utilisant la théorie de l'évidence de Dempster-Shafer [SHA 76] avec notre approche floue non paramétrique pour la modélisation des fonctions de croyance. Le résultat obtenu [ARI 04] est montré dans le tableau 2. La fusion a été réalisée en prenant les 5 premières classes

d'après leur rang (k = 5) pour chaque classifieur. Les éléments focaux possibles ont été choisis automatiquement par notre algorithme et leurs masses de croyance ont été calculées à l'aide de la fonction du degré d'appartenance. La décision de la fusion a été prise avec un argument de maximum de la valeur de masse de croyance. En cas de non fusion ou d'ambiguïté, notre algorithme continue son travail en prenant k = 10, ou 15, jusqu'à 20. Si l'ambiguïté persiste avec k = 20, le résultat initial avec k = 5 est retenu.

La validation et l'exécution de notre méthodologie ont été vérifiées en comparant les résultats avec ceux obtenus avec une méthode de fusion se fondant sur la matrice de confusion des classifieurs et l'intégration de croyance basée sur la formule bayésienne qui est expliquée dans [XU 92]. Le résultat obtenu avec cette méthode est aussi montré dans le tableau 2.

Méthode	Taux de Reconnaissance
Classifieur 1	69,52%
Classifieur 2	69,33%
Théorie de l'évidence (notre approche)	91,85 %
Matrice de confusion	91,15 %

Tableau 2 : Performance des méthodes

Ce travail se poursuit dans le sens d'une généralisation de notre fonction de degré d'appartenance floue en vue de l'application à la combinaison de différents types de classifieurs.

2.4. *Analyse d'images*

Nos travaux de reconnaissance des formes portent principalement sur des formes extraites à partir d'images. Dans ce cas, une étape importante est la segmentation qui permet de partitionner l'image en régions dont l'analyse pourra aboutir à des formes connues.

Plusieurs approches de la segmentation co-existent avec chacune leur domaine de prédilection. Celle que nous privilégions est basée sur les contours actifs (*snakes*) [KAS 87] car elle est adaptée au suivi d'objets déformables dans des séquences d'images.

Le principe est d'associer au contour une fonctionnelle d'énergie qu'ensuite l'algorithme va tendre à minimiser [ROU 03]. Il existe plusieurs possibilités pour cet algorithme ; nous privilégions l'algorithme glouton (*greedy*) qui est très rapide pour une qualité satisfaisante.

La fonctionnelle d'énergie du contour est composée de trois termes d'énergie : l'énergie interne, externe, et de contexte. Ces trois termes d'énergie, qui peuvent eux-mêmes être subdivisés, vont être pondérés par un coefficient proportionnel à l'importance de ce terme d'énergie dans le résultat final. Ces coefficients dépendent des images et des objets à retrouver : contours anguleux ou arrondis, région contrastée, … Le réglage de ces coefficients peut se faire par essais-erreurs par un expert du traitement d'images ou même par un expert du domaine d'application. Cette étape est fastidieuse, c'est pourquoi nous recherchons des méthodes pour les déterminer automatiquement. Nos essais actuels portent notamment sur une méthode basée sur l'algorithme Tabou. Cette méthode est appliquée aussi bien pour l'optimisation de la fonctionnelle d'énergie que pour les coefficients (poids) qu'elle contient.

Pour répondre à des besoins applicatifs imminents pour le suivi d'objets 3D dans des séquences d'images 3D, nous allons généraliser nos contours actifs à des surfaces actives.

2.5. *Liens intelligents*

La reconnaissance des formes s'applique naturellement à des objets dans des images mais elle peut aussi s'étendre à des formes moins tangibles comme le contexte d'un lien hypertexte dans une page web.

Cette étude part de la constatation que les auteurs qui publient des connaissances sous la forme de documents électroniques lisibles sur un écran utilisent de plus en plus la technologie des liens hypertextes pour améliorer la

présentation et la lisibilité de leur travail [VER 00]. Il est alors logique de se demander si ces liens ne représenteraient pas une structure formelle intéressante pour intégrer les outils conceptuels et technologiques favorisant la structuration des connaissances et leur partage par une communauté.

Pour répondre à cette question, nous réalisons une typologie des besoins auxquels répondent les liens que les auteurs placent dans les liens hypertextes électroniques sur le web. En d'autres termes, nous explorons toute la richesse sémantique explicite ou implicite contenue dans les liens hypertextes présents dans un domaine précis, puis nous vérifions la présence de cette richesse sémantique dans les outils technologiques existants avec l'objectif de confirmer que l'exploitation de cette richesse sémantique peut faciliter le partage des connaissances sur le web.

3. Applications

3.1. *Autisme*

Cette étude a pour but de quantifier de façon précise les différences de préhension existant entre les enfants sains et les enfants atteints d'autisme [MAR 03]. Elle se déroule en collaboration avec l'Equipe 1 « Autisme et troubles du développement : psychopathologie, physiopathologie et thérapeutique » de l'Unité Inserm 619 « Dynamique et pathologie du développement cérébral » et dans le cadre de l'IFR 135 « Imagerie fonctionnelle ».

Pour réaliser ces analyses, l'enfant est assis dans une chaise adaptée à sa taille et fait reposer son avant-bras sur une table spécifiquement ajustée en fonction de sa taille et de la hauteur de la chaise. La consigne donnée à l'enfant est de saisir l'objet lors du signal de départ et de le reposer à un endroit cible de la table. Plusieurs formes d'objets sont testées (cube, cylindre, sphère) de différentes tailles. Les objets peuvent être composés d'une forme simple ou de plusieurs formes (cube, cylindre, fourchette en plastique…).

L'enfant est filmé pendant ses mouvements par 3 caméras permettant d'obtenir une vision en 3 dimensions de la main lors de la saisie (vue de dessus, vue de face, vue de profil). A partir de ces séquences, des opérateurs de traitement d'images (contours actifs) permettent de suivre la main et de calculer des caractéristiques (angle global, angle des phalanges, écartement des doigts autour de l'objet manipulé, vitesse de déplacement…).

Quand cette étape d'acquisition et d'extraction de caractéristiques sera au point, des analyses des données obtenues [LEZ 01] avec des enfants sains et des enfants atteints d'autisme permettront de rechercher des caractéristiques discriminantes pour l'évaluation fine de cette pathologie.

3.2. *Analyse de signaux vidéos et sonores : application à l'étude de signaux médicaux*

La problématique considérée concerne l'étude de séquences multimédia constituées d'images et de sons dont il s'agit d'étudier les corrélations de manière à aider à la compréhension de l'origine des bruits.

L'analyse des séquences d'images consiste à suivre les objets en mouvement de manière à permettre leur étude. Une méthode générique, reposant sur une combinaison de suivi de régions et de contours, et une méthode adaptée aux objets homogènes, reposant sur la théorie des ensembles de niveaux, sont proposées [DEL 03].

L'analyse des données sonores consiste en l'élaboration d'un système d'identification reposant sur l'étude de la structure des signaux grâce à des codages adaptés et à leur modélisation par les lois de Zipf.

Ces méthodes ont été évaluées sur des séquences acoustico-radiologiques dans le cadre de l'étude de la pathologie du reflux gastro-oesophagien, en collaboration avec l'équipe Acoustique et Motricité Digestive de l'Université de Tours.

3.3. *Authentification basée sur l'écriture manuscrite*

L'authentification basée sur l'écriture manuscrite et en particulier sur la signature manuscrite est la manière la mieux acceptée par les utilisateurs parmi les différentes méthodes d'authentification biométrique. En effet, c'est un moyen simple et encore très utilisé d'authentifier les documents ou les chèques. Nous parlons dans ces cas de signatures hors-ligne, c'est-à-dire que seule l'image est disponible, par opposition aux signatures en ligne où l'information dynamique a pu être conservée.

La difficulté dans de tels systèmes est la détermination de caractéristiques stables et discriminantes. C'est pourquoi nous proposons de nouvelles caractéristiques basées sur la dimension fractale [HUA 00], [WIR 04]. Nos résultats actuels montrent que nos caractéristiques associées à des caractéristiques classiques permettent d'obtenir de meilleures performances qu'avec les caractéristiques classiques seules. L'application d'une méthode de sélection de caractéristiques basée sur les algorithmes génétiques a permis de confirmer la pertinence de nos nouvelles caractéristiques.

3.4. *Authentification basée sur le clavier et la souris*

Aujourd'hui, les exigences de sécurité pour l'accès à des ressources informatiques, ne permettent plus de se limiter au traditionnel couple login et mot de passe pour authentifier un utilisateur. En effet les utilisateurs ne prêtent souvent pas assez attention à sa sécurité, ils choisissent ainsi des mots de passe trop courts ou trop simples, de plus ils n'accordent pas assez d'importance à la confidentialité de ceux-ci entraînant souvent leurs divulgations de façon intensionnel ou non. Pour remédier à ce problème une solution prometteuse est la biométrie. Mais celle-ci nécessite l'ajout de capteurs coûteux et est souvent mal acceptée par les utilisateurs. Il semble donc intéressant d'étudier la possibilité de se limiter aux données pouvant être extraites à l'aide de périphériques se trouvant sur tout ordinateur, c'est-à-dire dans un premier temps uniquement à l'aide du clavier et de la souris.

Dans le cas du clavier, l'authentification va utiliser l'étude de la dynamique de frappe des utilisateurs [DOW 01] [GUV 03]. Cette dynamique, en fait le style de l'utilisateur au clavier, va être étudiée en s'intéressant à la frappe de touches successives qui vont être regroupées par deux. Pour chacun de ces couples, nous allons extraire des donnés temporelles (temps entre touches, temps de pression d'une touche) afin de caractériser le comportement de l'utilisateur. A ces données quantitatives, il va être possible de rajouter des données qualitatives comme par exemple l'ordre de relâchement des touches lors de la réalisation d'une majuscule.

Pour la souris, l'étude va se concentrer dans un premier temps sur des séquences d'interactions prédéfinies. Ces séquences pourront être une signature [SYU 98], un mot de passe graphique ou encore le comportement au cours d'un mini-jeu. Au cours ce ces séquences, on extrait des caractéristiques permettant de définir un utilisateur, ces caractéristiques pourront être spatiales (longueur parcourue, courbure…) ou dynamiques (vitesse, accélération, vitesse angulaire…). Les problèmes principaux de cette étude sont, d'une part d'identifier les caractéristiques qui vont nous permettre de séparer un utilisateur authentique d'un imposteur et, d'autre part la mise en place des outils de comparaison.

Les premiers résultats obtenus à l'aide du clavier nous donne 7 % pour les deux taux d'erreur (faux acceptés et vrais rejetés) sur une petite phrase connue. A partir de 10 lignes de texte tapées, les taux d'erreur tombent à 0 %. Ces résultats encourageants restent bien sûr à valider et à améliorer. Pour la souris, les résultats ne sont pas encore significatifs.

4. Conclusion

Nous avons eu, dans cet article, l'occasion de découvrir un nombre important d'études menées par des membres de l'équipe RFAI du LI de Tours. Je n'ai d'ailleurs pas mentionné leur nom pour ne pas alourdir le texte mais il est évident que ces travaux sont le résultat de nombreux chercheurs et d'étudiants encadrés par eux.

Toutefois, cette description est loin d'être exhaustive, ce qui n'était pas l'objectif ; en particulier, nous travaillons aussi sur les chaînes de Markov cachées [LEF 03] et sur l'hybridation de modèles.

Ainsi, nous couvrons un large spectre de méthodes du domaine de la reconnaissance des formes que nous avons l'intention de faire évoluer et d'appliquer dans les années à venir à des objets obtenus à partir d'une ou plusieurs sources vidéos et à l'authentification de personnes.

5. Bibliographie

[VAP 98] V. N. Vapnik. *Statistical Learning Theory*. New York, Wiley edition, 1998.

[PLA 99] J. Platt. *Fast Training of Support Vector Machines using Sequential Minimal Optimization, Advances in Kernel Methods-Support Vector Learning*. MIT Press, pp. 185-208, 1999.

[GER 91] A. Gersho et R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic, 1991.

[LEB 04] Gilles LEBRUN Christophe CHARRIER Olivier LEZORAY, *Réduction du temps d'apprentissage des SVM par Quantification Vectorielle* , CORESA, mai 2004.

[BON 03] Boné R. (2003), *Les dépendances temporelles dans les réseaux de neurones. Application à la prévision à un pas et multipas* , conférence plénière, 10ème rencontre internationale Approches Connexionnistes en Sciences Economiques et de Gestion (ACSEG 2003), Nantes, France, Novembre 2003, pp 117-129.

[WER 90] P.J. Werbos. *Backpropagation through time: what it does and how to do it*. Proceedings of IEEE, Special issue on neural networks, vol. 78, No. 10, pp.1550-1560, October 1990.

[WIL 90] R.J. Williams, J. Peng. *An efficient gradient-based algorithm for on line training of recurrent network trajectories*. Neural Computation 2: 490-501, 1990.

[ASS 03] Assaad M., Boné R. (2003), *Apprentissage itératif de réseaux de neurones récurrents pour la prévision de séries temporelles*, 10ème rencontre internationale Approches Connexionnistes en Sciences Economiques et de Gestion (ACSEG 2003), Nantes, France, Novembre 2003, pp. 63-74.

[SCH 90] Schapire R. E., *The strenght of weak learnability*, Machine Learning, 5, 197-227, 1990.

[FRE 90] Freund Y., *Boosting a weak learning algorithm by majority*, 3rd annual workshop on Computational Learning Theory, 202-216, 1990.

[SHA 76] G. Shafer. *A mathematical Theory of Evidence*. Princeton Univ Press., Princeton New Jersey, 1976.

[ARI 04] M. Arif, T. Brouard, N. Vincent, *Amélioration de la reconnaissance des formes par la fusion de l'information*, SETIT 2004, Sousse, Tunisie, 15-20 mars 2004.

[XU 92] L Xu, A Krzyżak, C. Y Suen, *Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition*, *IEEE Trans. Sys., Man & Cyber.*, vol. 22(3), pp. 418-435, 1992.

[KAS 87] M. Kass, A. Witkin, D. Terzopoulos, *Snakes : Active Contour Models*, Proceedings of the first International Conference on Computer Vision, juin 1987, pp. 259-268.

[ROU 03] J-J. ROUSSELLE. *Les contours actifs, une méthode de segmentation - Application à l'imagerie médicale*. Thèse de doctorat en Informatique. Laboratoire Informatique - Université de Tours. Juillet 2003. p. 162.

[VER 00] G. Verley, J-J. Rousselle, *An Evolved Link-specification Language for Creating and Sharing Documents on the Web*, CRIS 2000, 25-27 mai 2000.

[MAR 03] MARTINEAU J, COCHIN S. *Visual perception in children : human, animal and virtual movement activates different cortical areas*. International Journal of Psychophysiology, 2003, 51(1):37-44.

[LEZ 01] LEZORAY O, CARDOT H. *A Neural Network Architecture for Data Classification, International Journal of Neural Systems*, Vol. 11, Numéro 1, pp. 33-42, février 2001.

[DEL 03] E. DELLANDREA. *Analyse de signaux vidéos et sonores : application à l'étude de signaux médicaux*. Thèse de Doctorat en Informatique. Université de Tours. Octobre 2003.

[HUA 00] K. Huang and H. Yan, *Signature Verification using Fractal Transformation*, International Conference on Pattern Recognition (ICPR'00), pp. 2851-2854, Barcelona, Spain, September 2000.

[WIR 04] M. Wirotius, J.Y. Ramel, N. Vincent, *New features for authentication by on-line handwritten signatures*, International Conference on Biometric Authentication (ICBA), Japan, July 2004.

[DOW 01] Dowland, P.S., H. Singh, and S.M. Furnell. *A Preliminary Investigation of User Authentication Using Continuous Keystroke Analysis,* IFIP 8th Annual Working Conference on Information Security Management & Small Systems Security. 2001. Las Vegas.

[GUV 03] Guven, A. and I. Sogukpinar, *Understanding users'keystroke patterns for computer access security*. Computers & Security, 2003. 22(8).

[SYU 98] Syukri, A.F., E. Okamoto, and M. Mambo. *A User Identification System Using Signature Written with Mouse*. Third Australasian Conference on Information Security and Privacy. 1998.

[LEF 03] S. LEFEVRE, E. BOUTON, T. BROUARD, N. VINCENT, *A new way to use hidden Markov Models for Object Tracking in Video Sequences*, International Conference on Image Processing (ICIP2003), 14-17 septembre 2003, Barcelone (Espagne).

Classifying Classification Problems

J. C. Gower

The Open University
Department of Statistics
Walton Hall
Milton Keynes, MK7 6AA, U.K.

RÉSUMÉ. Classification problems, both for assignment and class construction, are specified either in probabilistic form or not. Underlying issues are (i) the types of sampling unit under consideration: in particular, are they differentiated into previously determined classes (possibly with identical members) or are they undifferentiated and (ii) considerations of the types of variable used; are they quantitative or categorical? Rather than a simple data-matrix, the fundamental form of data is taken to be the between and within-group structure. These considerations lead to a simple cross-classification of familiar, and some novel, classification problems.

MOTS-CLÉS: Probabilistic Classification, Non-probabilistic Classification, Classes, Groups, Assignment, Class Construction, Approximation.

1. Introduction

In the following, I shall attempt a simple classification of a range of classification problems. A major, and well-recognised, division is into problems of forming classes and problems of assigning to previously recognised classes. The other elements of the classification proposed here are the role of probabilistic formulations and the structure of the data. These are discussed first in a general context and in subsequent sections in more detail.

When forming classes, we must define just what kind of thing it is that we are classifying. In the early days of numerical taxonomy, Sneath and Sokal (1973) coined the term Operational Taxonomic Unit (OTU) to describe the units under study. This term is little used nowadays, perhaps because taxonomy is now subsumed into a wider classification discipline. Nevertheless, it is useful to have some such term, so I shall use Operational Unit (OU).

Perhaps, it is not fully appreciated that very often we form classes from OUs that are already recognized as classes. For example, Linnaeus classified animal and plant life but life-forms such as “cats”, and “lilies” and so on, had long been universally recognised. These are the relevant OUs. Just by giving names to groups recognizes them as classes with associated widely accepted properties; nouns are very often the names of classes. Probabilistic notions play little or no part in defining the class of cats, nor of classifying them further into genera, families, etc. Thus, at the outset the OUs may be differentiated into recognised classes. Even though these initial classes are recognised, there remains the goal of further classification to organise what may be a very large body of information.

On other occasions, the OUs are undifferentiated but the possibility is recognised that there may be some heterogeneity that may be useful as a basis for differentiating the OUs into two,

or more classes. This is the position with mixture problems which naturally, though not essentially, are given probabilistic formulations. In its purest form, instead of forming classes of previously named OUs, mixture problems are concerned with the possibility of deriving totally new classes from undifferentiated units – which may then be named, if we wish. Historically, Karl Pearson (1898) had a sample of ancient bones that were almost certainly a mixture of male and female bones which he wanted to separate; in this case he knew the names and number of classes involved. Nowadays, his problem might have been treated by discriminant analysis, assigning each bone to a population either of known female or known male bones. Nevertheless, there remains the possibility that modern known populations might not reflect the variability of ancient times, so justifying the mixture formulation.

Fundamentally, we have a division into where the OUs to be classified are (i) differentiated into named classes and (ii) where they are undifferentiated.

Probabilistic and non-probabilistic approaches have been mentioned. An influence on the more appropriate formulation is associated with the types of variable used to describe the OUs. With quantitative variables, there is certainly going to be variation within classes. When variation is large within OUs compared to the variation between OUs, overlap between the OUs is likely and probabilistic ideas become relevant. With categorical, including binary, variables there may still be overlapping variation but often it is possible to choose characters that do not vary within OUs (e.g. all cats have retractable claws) and then probability is less irrelevant. Clearly, when possible, it is better to base classifications on non-variable characteristics of the OUs.

2. Data structure

The starting point of most algorithms is a data matrix $\boldsymbol{X}$ with N rows (often referring to the OUs) and P columns of variables. The above remarks on structured and unstructured OUs suggest that a data matrix may be of several types that at first sight may look very similar. Nevertheless the differences can be fundamental. I believe that the failure to distinguish between superficially similar types of data matrix is at the root of many misuses of classification software. The situation may be clarified by considering the usual between and within groups sample-structure that is familiar in simple analysis of variance and in canonical variate analysis. Thus, the N rows of a data matrix $\boldsymbol{X}$ refer to objects drawn from K groups the kth of which has n_k members and $N = \sum_{k=1}^{K} n_k$. The K groups are the OUs whereas the n_k objects are undifferentiated samples pertaining to the kth OU.

When $K = 1$, so that $N = n_1$, we have the classical data-matrix of statistical multivariate analysis. Typically, its rows represent a random sample of size $N = n_1$. The samples have no *a priori* structure, so are said to be *unstructured*. Note that although the *data* are unstructured, this does not preclude the possibility that they might be fitted to a highly structured model. Indeed, that is the approach of the multivariate mixture problem which seeks a representation of the underlying probability distribution as a mixture of M more simple distributions. After analysis, the previously unstructured samples can be assigned to the M newly-found distribution-groups which may become named structured OUs of interest.

By contrast, when $n_k = 1$ for $k = 1,2,...,K$ we have $N = K$ OUs at the outset, where each OU is represented by *one* object, or row, of $\boldsymbol{X}$. Normally, the OUs will bear names indicating an initial classification. Botanists recognise species like daisies and dandelions, linguists recognise languages, geographers recognise rivers and mountains, librarians recognise book-titles and so on. The setting $n_k = 1$ need not imply that the kth OU class has only one member; merely that all its members are indistinguishable on the basis of the P chosen variables. This initial classification does not inhibit a desire for further classification; rather it encourages classification to organise what can be very large bodies of information.

This structure with $N = K$, $n_k = 1$ is quite different from the familiar statistical data-matrix with $K = 1$, $N = n_1$ and would be an inadequate representation of reality when variability within

groups is substantial. However, there are many classification problems where within-group variability may be ignored. For example, the *k*th OU may have a unique member (there is only one Bordeaux) or we may be able to choose variables which do not vary within groups (all cats have claws) but do vary between groups. As we have seen, this is a very common situation and those interested in classifying groups will seek to describe the groups by variables which are constant within groups, or essentially so (see below for "typical" OUs in taxonomy); indeed, they would be foolish not to do this when it is a possibility. When constant within-group variables are unavailable, the only remaining possibility is to base classifications on variables that do vary within groups, in which case probabilistic methods become relevant and the choice $n_k = 1$ becomes untenable; then replication of samples within groups is essential to capture the within-group variability. Within groups variation is a familiar component of the assignment problem of classical discriminant analysis but we show below, in section 3, how it can also be important when constructing classes.

To recapitulate, the objects to be classified are said to be structured when they fall into named OUs, which implies that a preliminary classification is recognised. Otherwise they are said to be unstructured, which usually means that the rows of $\boldsymbol{X}$ are undifferentiated as with a random sample assumed to be drawn from some notional population or mixture of populations. All non-probabilistic classification problems are concerned with structured data. In probabilistic classification problems we meet with both structured (e.g. canonical variate analysis) and unstructured (e.g. multivariate mixtures) sets of OUs.

When there is replication within groups, it is important to know whether or not the n_k replicates are chosen by some random process. It is not obvious that simple random samples are necessarily a desirable basis for forming classifications. For example, the number of speakers of a language is unlikely to be relevant for classifying languages thus indicating that some better representation of speakers than is given by a random sample. Indeed, it is the *distribution* of variables within the groups which is important, not the relative frequencies of speakers. Each language might be represented by a single set of characteristics as in taxonomy where within-group variation is often handled by representing each group by a single invented object with *typical* values; this is acceptable when within group variation is small relative to between group variation. Choosing $n_k = 1$ gives an extreme form of non-random sampling but nevertheless may provide a better representation of the language OU.

We have considered the most simple between-within structure where the K groups represent the totality of objects to be classified. This does not preclude the possibility of additional groups being added at a later stage. Also the groups might be regarded as a random sample of some larger set, and then probabilistic methods might have a role to play; this seems an artificial set-up. Rather than a basic K groups structure, the groups might have an elaborate *a priori* imposed structure of the crossed and nested kinds, indicating an equally elaborate *a priori* classification. It seems unlikely that one would wish to use this classification as the starting point for further classification. Rather, one might begin again by subsuming the complex structure into a K-group structure and comparing any new classification with the old one. These possibilities point the way to some areas of future research in classification theory.

The above has said little about the types of variable used in classification. Variables may be numerical or categorical. If numerical, they may be continuous on ratio or interval scales, a distinction which rarely affects classification. Categorical variables may be nominal, ordinal or dichotomous, a special binary categorical variable with one category merely defined to be not the other; all these types contain different kinds of information which may be exploited when classifying things. Numerical variables are less important for non-probabilistic classification than are categorical variables. This is because numerical variables are likely to vary within groups, so when non-varying categorical variables can be found, they are to be preferred. Just as objects may be structured, so may variables. Structure in variables has been long-recognised in survey design and Gower (1971) took it into account when designing a general similarity coefficient in which primary variables could be associated with sets of secondary variables, in turn associated with tertiary variables, and so on. More generally, we may have multiphase sampling, where the variables themselves have a nested structure, and multistage sampling,

with the sample units at each stage being described by their own variables. This level of sophistication is not known in classification work but perhaps it should be given some attention.

Many methods of classification are based on a measure of distance or dissimilarity between OUs. Overwhelmingly, the same definition of dissimilarity is assumed for all pairs of OUs but Friedman and Meulman (2004) recognise the possibility of using a different metric within, and by implication, between groups.

3. Probabilistic and non-probabilistic classification

Probabilistic methods such as discriminant analysis for assigning to classes and mixture problems for constructing classes are familiar in the statistical literature. Non-probabilistic methods are less familiar – at least to statisticians. They usually pertain to structured data with $n_k = 1$.

Identification keys constructed by *ad hoc* methods have been known to botanists for several centuries and foreshadow recent developments, such as Regression Trees and certain aspects of Expert Systems. To use a key a single object has to be identified, that is it has to be named. The simplest thing to do is to compare the object with each row of $\boldsymbol{X}$ until a match is found, when identification is achieved. This is inefficient because an enormous number of comparisons may have to be made, and it may be impracticable because variables included in $\boldsymbol{X}$ may be unavailable (e.g. a flowering plant may give no information on the characteristics of its seeds). Diagnostic keys overcome these difficulties. A key is essentially a tree with one binary variable associated with each node. The value of the binary variable for the object to be identified determines which of the two possible branches of the tree one traverses to reach the next node. In this way one answers a series of questions until one reaches an end-point of the tree, where the identification is given. Many interesting problems in constructing keys have been reviewed by Payne and Preece (1980). We may require the tree with fewest nodes or using fewest different among the binary variables or with minimum average numbers of steps to achieve identification. Costs may be associated with ascertaining the values of the binary variables, in which case we may require the key that is cheapest to use. When tests take time (as with some biochemical tests) it may be efficient to do a group of tests simultaneously and then it has to be decided how best to group the tests. Probability concepts are irrelevant for these interesting problems but may enter if we recognise that the OUs have different frequencies of occurrence, so affecting average numbers of steps to identification and average costs. However, any probabilistic distribution associated with the binary variables themselves is neither required nor material.

To construct classifications with binary variables, we can classify the K objects into $M<K$ groups in such a way that on being told that an object belongs to one of these groups, more correct statements about the likely value of its binary variables can be made than for any other classification. This is maximal predictive classification (Gower, 1975) which models the dictum of a distinguished botanical taxonomist, Gilmour (1937), that *a system of classification is the more natural the more propositions there are that can be made regarding its constituent classes*. No probability distribution is associated with the binary variables, although again the relative frequencies of the objects may be accommodated. The maximal predictive classes have optimal assignment properties. In general, optimal future assignment is a valid criterion for constructing classes.

We may apply the K-means algorithm to numerical data with $n_k = 1$, thus forming homogeneous groups among the $N = K$ OUs without any appeal to probability. In this approach, we would ignore the unknown within-group variation, which may or may not be a reasonable thing to do. When $\boldsymbol{X}$ is regarded as a multivariate data-matrix (with $K = 1$ and $n_1 = K$) then the K-means algorithm finds the maximum likelihood solution to the mixture problem modelled as a combination of multinormal populations. Thus, the same algorithm may be used to compute solutions to two different classification problems, one probabilistic and the other not. Of course, different mixture models have different maximum likelihood solutions whereas the K-means algorithm is only fully valid for multinormal mixtures. A similar distinction

occurs with principal components analysis (PCA). On the one hand, multivariate PCA is based on multinormal assumptions and is concerned with deriving significance tests for zero eigenvalues, thus identifying subspaces, perhaps characterised by reified latent variables, while on the other hand, a PCA of differentiated OUs is concerned with approximating the distances by configurations in few dimensions. Any kind of multidimensional scaling (MDS) of undifferentiated samples seems of little interest.

Many, mostly heuristic, algorithms applied to differentiated OUs give a hierarchical classification of the K OUs. Heuristic algorithms are acceptable when there is no practicable way of optimising an objective criterion (e.g. NP complete problems). Ultrametrics and additive trees give objective criteria for fitting trees to dissimilarity data by least squares, thus providing well-defined models. The use of least-squares does not necessarily imply an appeal to probability. One may note the eighteenth century work of mathematicians like Legendre and Laguerre who used L_1, L_2, or L_∞ norms to approximate complicated functions (e.g. Bessel functions) by polynomials. Polynomials give an acceptable approximation to the function - probability is irrelevant. Analogously, we may regard non-probabilistic classifications as giving similar approximations to $\boldsymbol{X}$ where the goodness of fit may be used to assess the adequacy of the tree approximation or of the class-predictors fitted to maximise prediction. Similarly, MDS approximates a dissimilarity matrix. Such measures of approximation are at least as useful as significance tests associated with probabilistic models; *significant* is not synonymous with *important* and *not significant* is not synonymous with *unimportant*.

Other non-probabilistic objectives for classification concern mixtures of hierarchical and non hierarchical organisation. We may wish to classify the K groups into $M<K$ classes arranged hierarchically with each class containing undifferentiated members. Indeed, it seems to us that this is more frequently required than full hierarchical classification but it rarely gets mentioned, except when pointing out that branches of a full tree may be amalgamated, perhaps governed by specifying some threshold level in a dendrogram. Again with a multivariate data-matrix we could formulate a variant of the mixture problem in which the M classes are to be arranged hierarchically. It is strange that the specification of M in a K means classification into M disjoint classes is fully accepted, the corresponding problem for nested classes is not. Gower (1975) gives a method for constraining maximal predictive classes to have hierarchical organisation which immediately extends to any classification criteria C_m into m classes. One only has to optimise $\sum_{m=1}^{M} W_m C_m$ where the classes are constrained to be hierarchically arranged and W_m is an optional weighting function, perhaps a function of m. Although the computational problems of optimising this criterion, as with other objective classification criteria, are formidable, it easily allows different hierarchical classifications to be compared.

4. The Classification

	OUs	*Assignment*	*Construction*
-probabilistic	*Structured*	Matching Diagnostic keys	Maximal predictive classes *Non* Cluster analysis: *M*-groups, hierarchic, other.
Probabilistic	*Unstructured*	(Null)	Mixture problems
	Structured	Discrimination	Undeveloped

Table 1: Types of classification problem depending on whether (i) the problem is probabilistic or non-probabilistic, (ii) is for assignment to classes or construction of

classes, or (iii) is concerned with structured or unstructured OUs. (Simplification of a similar table in Gower, 1998)

We are now in a position to show the classification of classification methods. We have a three-way cross classification as shown in Table 1. The classifying factors are (i) whether the method is probabilistic or not (ii) whether the problem is one of forming classes or one of assigning to classes and (iii) whether the data is structured or not. We have seen that non-probabilistic methods depend only on structured data so the *unstructured, non-probabilistic* classification does not exist. Similarly, the *probabilistic, unstructured, assignment* problem is null because there are no named classes to assign to. The statistical literature is mostly concerned with the two cells (i) *probabilistic, structured, assignment* and (ii) *probabilistic, unstructured, construction*. There is a vast classification literature on *non-probabilistic, structured, construction* problems.

There remains the cell labelled *probabilistic, structured, construction*. This problem seems not to be addressed in the literature, yet it is very interesting. Suppose we wish to classify K normal populations into $M<K$ groups. Following discrimination ideas, we could seek boundaries which minimise the overlap between the M groups, thus minimising future errors of misclassification. With limiting point-densities *any* set of boundaries give no overlap and no possibility of misclassification. Nevertheless, it remains reasonable to require a grouping of the populations into M classes, possibly nested, by grouping together pairs of point-populations that are closer than others, as judged by distances based on the non-probabilistic information contained in the values of the variables. When the point densities expand to conventional distributions, it seems that a combination of probabilistic and non-probabilistic information should be used for constructing classifications; assignment to these classes might be entirely probabilistic.

By focussing attention on different types of data structure, some of the issues that underlie probabilistic and non-probabilistic problems have been brought into focus. I hope to have shown that many non-probabilistic classification problems are relevant and interesting in their own right and should not be regarded, as they sometimes are, as heuristics for a more desirable fully stochastic formulation. Further, the non-probablilistic formulations support a renewed interest into eighteenth century ideas of approximation for assessing fit, in addition to statistical ideas based on stochastic variability.

The classification of Table 1 is oversimplified. For example, we have already mentioned that one purpose for forming classes is so that future assignment to these classes is optimal. Also, in discriminant analysis we may derive discriminant functions from a training sample and validate the process by assigning the remaining samples. Thus, the cells of Table 1 are not independent. Nevertheless, I hope that the classification will be of some use in focussing attention on and clarify major issues arising in classification n theory.

5. Bibliographie

[FRE 04] FRIEDMAN J., MEULMAN, J., Clustering objects on subsets of attributes (with discussion), *J. R. Statist. Soc. B.,* vol. 66, 2004, p. 1-25.

[GOW 71] GOWER J., A general coefficient of similarity and some of its properties, *Biometrics,* vol. 27, 1971, p. 857-871.

[GOW 75] GOWER J., Maximal predictive classification, *Biometrics,* vol. 30, 1975, p. 634-654.

[GOW 98a] GOWER J., Classification, overview, in: *Encyclopaedia of Biostatistics*, Armitage, P., Colton, T. (Eds.), Wiley, Chichester p. 656-667.

[GOW 98b] GOWER J., ROSS, G., Non-probabilistic classification, in: *Advances in Data science and Classification*, Rizzi, A. Vichi, M., Bock, H.-H. (Eds.), Springer, Berlin, p. 21-28.

[PAY 80] PAYNE J., PREECE, D., Identification keys and diagnostic tables: a review (with discussion), *J. R. Statist. Soc. A.,* vol. 143, 1980, p. 253-292.

[Pea 98] PEARSON K., Mathematical contributions to the theory of evolution, V. On the reconstruction of the stature of prehistoric races. *Philosophical Transactions of the Royal Society of London, Series A,* 192, 1898, 169-244.

[SNE 73] SNEATH P., SOKAL R., *Numerical Taxonomy*, Freeman, San Francisco, 1973.

Les méthodes de classification et de détermination du nombre de classes : du classique au symbolique

André HARDY

Unité de Statistique
Département de Mathématique
Université de Namur
8 Rempart de la Vierge
B - 5000 Namur Belgique
andre.hardy@fundp.ac.be

RÉSUMÉ. *Le but de cet exposé est de montrer comment certaines méthodes de classification et de détermination du nombre de classes classiques ont pu être étendues en des méthodes "symboliques". On insistera plus particulièrement sur les travaux effectués dans l'équipe de statistique de l'Université de Namur : les méthodes classiques et symboliques de classification basées sur les processus de Poisson homogène et non homogène, le module de détermination du nombre de classes symbolique NBCLUST. Des applications à des ensembles de données symboliques, artificielles et réelles illustrent le travail.*

MOTS-CLÉS : *Classification, Détermination du nombre de classes, Objet symbolique, Processus de Poisson, Enveloppe convexe, Critère des Hypervolumes*

1. Le problème de classification

Le problème de classification auquel nous nous intéressons est le suivant.

$E = \{x_1,\ x_2,\ ...,\ x_n\}$ est un ensemble de n objets sur lesquels on mesure la valeur de p variables $Y_1,\ Y_2,\ ...,\ Y_p$. Nous recherchons une partition $P = \{C_1,\ C_2,\ ...,\ C_k\}$ de l'ensemble E des objets en k classes.

2. Les méthodes de classification classiques

2.1. *Les méthodes basées sur une matrice de dissimilarité*

Nous considérerons en premier lieu quatre méthodes de classification hiérarchiques agglomératives classiques bien connues : les méthodes du lien simple, du lien complet, de la moyenne et de Ward, et une méthode de partitionnement : la méthode des Nuées dynamiques [CEL 89].

2.2. *Les méthodes basées sur les processus de Poisson*

2.2.1. *Introduction*

Pour éviter le choix (bien souvent arbitraire) d'une distance ou d'une dissimilarité en classification, nous utilisons des méthodes statistiques basées sur les processus de Poisson homogène et non homogène [RAS 96],

[KAR 91]. Le point de départ de ces approches est le problème suivant : "Etant donné la réalisation d'un processus de Poisson homogène dans un domaine convexe compact D, estimer D en utilisant des méthodes d'inférence statistique". La solution de ce problème fut trouvée par Rasson et Ripley [RIP 77]. L'estimateur du maximum de vraisemblance du domaine D, qui est également une statistique exhaustive pour D, est l'enveloppe convexe des points. L'estimateur non biaisé correspondant est une dilatation de l'enveloppe convexe à partir de son centre de gravité.

2.2.2. *La méthode de classification et le critère des Hypervolumes*

Sur base du résultat précédent, une première méthode de classification automatique fut élaborée [HAR 82], [HAR 83]. Elle suppose que les points observés sont générés par un processus de Poisson homogène dans un domaine D de R^p, où D est l'union de k domaines convexes compacts disjoints $D_1,\ D_2,\ ...,\ D_k$; $C_i \subset \{x_1,\ x_2,\ ...,\ x_n\}$ est le sous-ensemble des observations appartenant à $D_i\ (1 \leq i \leq k)$. Le problème revient à estimer les domaines inconnus D_i dans lesquels les points ont été générés. Les estimateurs du maximum de vraisemblance des k domaines inconnus $D_1,\ D_2,\ ...,\ D_k$ sont les k enveloppes convexes $H(C_i)$ des k sous-groupes C_i de points tels que la somme des mesures de Lebesgue des enveloppes convexes disjointes $H(C_i)$ est minimale. Le critère des Hypervolumes est alors défini par

$$W_k = \sum_{i=1}^{k} m(H(C_i))$$

où $m(H(C_i))$ est la mesure de Lebesgue multidimensionnelle de l'enveloppe convexe des points appartenant à C_i.

2.2.3. *Une méthode polythétique divisive basée sur le processus de Poisson homogène*

Un des inconvénients souvent cité pour des algorithmes divisifs de classification est qu'une partition obtenue à un niveau de la procédure n'est jamais remise en cause. L'algorithme divisif de classification polythétique basé sur le critère des Hypervolumes élaboré par A. Hardy [HAR 96a] , propose une solution à ce problème. La première partie de l'algorithme est une procédure hiérarchique divisive qui produit une partition de l'ensemble des données en k classes (k fixé). La solution obtenue à la fin de cette première partie ne correspond pas toujours à la structure "naturelle" des données. La seconde partie de l'algorithme est basé sur la propriété d'admissibilité par rapport à l'omission de classes de Fisher et Van Ness [FIS 71]. Elle consiste en une procédure de "recollement-division" qui améliore, lorsque c'est possible, la partition obtenue à la fin de la première étape de l'algorithme, et la valeur correspondante du critère à optimiser. Une approche comparable a été utilisée plus tard par Pirçon [PIR 04].

2.2.4. *Cinq nouvelles méthodes de classification monothétiques*

J.-Y. Pirçon [PIR 04] propose cinq nouvelles méthodes de classification monothétiques divisives. Elles sont toutes basées sur le processus de Poisson. Une première méthode (HOPP) est développée en faisant l'hypothèse que les points sont la réalisation d'un processus de Poisson homogène. Les quatre autres méthodes font l'hypothèse que les points sont la réalisation d'un processus de Poisson non homogène (UNHOPPHI et UNHOPPKI), ou d'une superposition de processus de Poisson non homogènes (SONHOPPHI et SONHOPPKI). Dans ces quatre derniers cas l'intensité du processus de Poisson doit être estimée. Deux estimations d'intensité non paramétriques sont utilisées : les histogrammes et les noyaux.

Un arbre est construit. Le critère de coupure est obtenu par la méthode du maximum de vraisemblance. Il s'agit de trouver la variable pour laquelle le "vide" dans les données selon cette variable est maximum. Pour trouver le meilleur sous-arbre de l'arbre construit, un processus d'élagage est nécessaire. Deux critères sont proposés. Un premier est basé sur l'inertie, et l'autre sur le Gap test [KUB 96], qui sera développé dans le paragraphe suivant.

Les cinq méthodes proposées sont monothétiques et les coupures sont faites perpendiculairement aux axes. Dans certains cas, une classe compacte est divisée en plusieurs parties. Une des originalités des méthodes proposées par Pirçon est d'inclure à la fin de la procédure une étape de "recollement" qui a pour objet de tester si deux

feuilles de l'arbre doivent être fusionnées. Une condition nécessaire de recollement est que les deux groupes soient connexes. Les critères utilisés pour le recollement sont à nouveau le critère de l'inertie et le Gap test.

La méthode UNHOPPKI a été étendue en une méthode de classification symbolique, appelée SCLASS. Elle sera présentée dans la deuxième partie de ce papier.

3. Les méthodes de détermination du nombre de classes

3.1. *Méthodes basées sur une matrice de dissimilarité*

De nombreuses méthodes de détermination du nombre de classes utilisent principalement une matrice de dissimilarité entre les objets. Nous considérerons tout d'abord cinq méthodes classiques de détermination du nombre de classes, les "meilleures" du classement de Milligan et Cooper [MIL 85] : la méthode de Caliński et Harabasz [CAL 74], l'index J [DUD 73], l'index C [HUB 76], l'index Γ [BAK 75] et le test de Beale [BEA 69].

3.2. *Méthodes basées sur les processus de Poisson*

Soit $x = (x_1,\ x_2,\ ...,\ x_n)$ un échantillon aléatoire généré par un processus de Poisson homogène dans k domaines convexes compacts disjoints $D_1,\ D_2,\ ...,\ D_k$ d'un espace Euclidien à p dimensions.

3.2.1. *Un test du quotient de vraisemblance*

Pour un entier $k \geq 2$, on teste l'hypothèse nulle d'une structure naturelle en k classes contre l'alternative d'une structure en $k-1$ classes. La statistique du test est déduite du modèle statistique par la méthode du quotient de vraisemblance. Elle est donnée par [HAR 96b]

$$S(x) = \frac{W_k}{W_{k-1}}.$$

Le test est réalisé de manière séquentielle. Si k_0 est la première valeur de $k \geq 2$ pour laquelle on rejette H_0, alors on considérera $k_0 - 1$ comme le nombre approprié de classes naturelles.

3.2.2. *Le Gap test*

Notons par C l'ensemble des points et par $P = \{C_1, C_2\}$ une partition de C en deux classes. On teste les hypothèses suivantes :

- H_0 : les $n = n_1 + n_2$ points sont une réalisation du processus de Poisson homogène dans le domaine D
- H_1 : n_1 points sont la réalisation d'un processus de Poisson homogène dans le domaine D_1 et n_2 points dans D_2 où $D_1 \cap D_2 = \emptyset$.

Nous développons le test dans le cas unidimensionnel. Une description dans le cas multidimensionnel est faite dans [KUB 96].

Un test du quotient de vraisemblance donne [RAS 94] :

$$Q(x) = \frac{max\ L_{H_0}(x)}{max\ L_{H_1}(x)} = \left(1 - \frac{m(\triangle)}{m(D)}\right)^n$$

où $\triangle$ est le plus grand "vide" dans les données et $m(D)$ la mesure de Lebesgue du domaine D.

Par conséquent, la région critique du Gap test, au niveau α, est donnée par

$$W_\alpha = \{x : \frac{m(\triangle)}{m(D)} \geq t_\alpha\}.$$

Le seuil t_α est obtenu par Kubushishi en utilisant des lois limites ; il est donné par $t_\alpha = -\log(-\log(1-\alpha))$.

Les méthodes de détermination du nombre de classes de Milligan et Cooper, le test des Hypervolumes et le Gap test ont été appliqués et évalués sur des ensembles de données artificielles et réelles [HAR 96b], [BEA 02].

4. Les données symboliques

Considérons un ensemble d'objets $E = \{x_1, ..., x_n\}$ sur lesquels on mesure un ensemble de p variables symboliques $Y_1, ..., Y_p$. Ces variables peuvent être des variables intervalles, multivaluées ou modales [BOC 00]. La plupart des méthodes de classification utilisent une matrice de dissimilarité, qui reflète la structure de l'ensemble des n objets symboliques. Un module implémentant des méthodes de détermination du nombre de classes pour des données symboliques, appelé NBCLUST, a été intégré dans la méthode de classification symbolique SCLUST. C'est pourquoi nous nous concentrerons tout d'abord sur les mesures de dissimilarité utilisées dans le logiciel SCLUST.

4.1. *Variables intervalles*

Soit $E = \{x_1, \ldots, x_n\}$ un ensemble de n objets décrits par p variables intervalles $Y_1, \ldots, Y_p$ de domaines $\mathcal{Y}_1, \ldots, \mathcal{Y}_p$ respectivement. La variable Y_j sera donnée par

$$Y_j : E \to \mathcal{B}_j : x_k \mapsto Y_j(x_k) = x_{kj} = [\alpha, \beta] \subset \mathcal{R}.$$

$\mathcal{B}_j$ est donc l'ensemble des intervalles bornés fermés de $\mathcal{R}$.

On associe p indices de dissimilarité δ_1, ..., δ_p définis sur les ensembles $\mathcal{B}_j$ de manière à obtenir une mesure de dissimilarité globale sur E. Si $x_{kj} = [\alpha_{kj}, \beta_{kj}]$ et $x_{\ell j} = [\alpha_{\ell j}, \beta_{\ell j}]$, on définit les trois distances suivantes pour des variables intervalles.

$$\begin{aligned}
\text{La distance de Hausdorff :} \quad \delta_j(x_{kj}, x_{\ell j}) &= \max\{\,|\,\alpha_{kj} - \alpha_{\ell j}\,|, |\,\beta_{kj} - \beta_{\ell j}\,|\,\} \\
\text{La distance } L_1 : \quad \delta_j(x_{kj}, x_{\ell j}) &= |\,\alpha_{kj} - \alpha_{\ell j}\,| + |\,\beta_{kj} - \beta_{\ell j}\,| \\
\text{La distance } L_2 : \quad \delta_j(x_{kj}, x_{\ell j}) &= (\alpha_{kj} - \alpha_{\ell j})^2 + (\beta_{kj} - \beta_{\ell j})^2.
\end{aligned}$$

On obtient ainsi une mesure de dissimilarité globale sur E

$$d : E \times E \longrightarrow R^+ : (x_k, x_\ell) \longmapsto d(x_k, x_\ell) = \left(\sum_{j=1}^{p} \delta_j^2(x_{kj}, x_{\ell j}) \right)^{1/2}.$$

4.2. *Variables multivaluées*

$E = \{x_1, \ldots, x_n\}$ est un ensemble de n objets décrits par p variables multivaluées $Y_1, \ldots, Y_p$ dont les domaines sont respectivement $\mathcal{Y}_1, \ldots, \mathcal{Y}_p$. $Y_j(x_k)$ est un ensemble de catégories et $\mathcal{B}_j = \mathcal{P}(\mathcal{Y}_j)$. Notons m_j

le nombre de catégories prises par Y_j. Nous transformons la matrice des données initiales en une matrice de fréquences. La fréquence $q_{j,x_k}(c_s)$ associée à la catégorie c_s $(s = 1, \ldots, m_j)$ de $Y_j(x_k)$ est donnée par

$$q_{j,x_k}(c_s) = \begin{cases} \frac{1}{|Y_j(x_k)|} & \text{si} \quad \mathrm{c_s} \in \mathrm{Y_j(x_k)} \\ 0 & \text{sinon.} \end{cases}$$

Les distances L_1 et L_2 sur $\mathcal{B}_j$ sont respectivement définies par :

$$\delta_j(x_{kj}, x_{\ell j}) = \sum_{i=1}^{|\mathcal{Y}_j|} |q_{j,x_k}(c_i) - q_{j,x_\ell}(c_i)| \quad \text{et} \quad \delta_{\mathrm{j}}(\mathrm{x_{kj}}, \mathrm{x_{\ell j}}) = \sum_{\mathrm{i}=1}^{|\mathcal{Y}_{\mathrm{j}}|} (\mathrm{q_{j,x_k}(c_i) - q_{j,x_\ell}(c_i)})^2$$

et la distance de De Carvalho par

$$\delta_j(x_{kj}, x_{\ell j}) = \sum_{i=1}^{|\mathcal{Y}_j|} (\gamma q_{j,x_k}(c_i) + \gamma' q_{j,x_\ell}(c_i))$$

où

* $\gamma = \begin{cases} 1 & \text{si} \quad \mathrm{c_i} \in \mathrm{Y_j(x_k)} \text{ et } \mathrm{c_i} \notin \mathrm{Y_j(x_\ell)} \\ 0 & \text{sinon} \end{cases}$

* $\gamma' = \begin{cases} 1 & \text{si} \quad \mathrm{c_i} \notin \mathrm{Y_j(x_k)} \text{ et } \mathrm{c_i} \in \mathrm{Y_j(x_\ell)} \\ 0 & \text{sinon.} \end{cases}$

On obtient une mesure de dissimilarité globale sur E.

$$d : E \times E \longrightarrow R^+ : (x_k, x_\ell) \longmapsto d(x_k, x_\ell) = \left(\sum_{j=1}^{p} \delta_j^2(x_{kj}, x_{\ell j}) \right)^{1/2}.$$

4.3. *Variables modales*

Le cas des variables modales est semblable à celui des variables multivaluées. Les fréquences $q_{j,x_k}(c_s)$ sont simplement remplacées par les valeurs de la distribution $\pi_{j,k}$ associées à chacune des catégories de $Y_j(x_k)$.

5. Les méthodes de classification symboliques

5.1. *Les méthodes basées sur une matrice de dissimilarité*

L'existence d'une matrice de dissimilarité pour des objets symboliques décrits par des variables intervalles, multivaluées et modales permet l'application directe des algorithmes de classification classiques basés sur des dissimilarités. Ce sera entre autres le cas des méthodes du lien simple, du lien complet, du centroïde et de Ward considérées dans ce travail.

Il en va de même pour la méthode de classification SCLUST [VER 00] disponible dans le logiciel SODAS2. SCLUST est une extension symbolique de la méthode des Nuées Dynamiques [CEL 89]. Elle détermine d'une manière itérative une série de partitions qui améliore, à chaque étape, la valeur d'un critère mathématique.

L'algorithme est basé sur la définition de prototypes pour représenter les classes et d'une fonction de proximité qui assigne les objets aux classes.

Dans le cas de variables intervalles, le prototype de la classe C_ℓ, noté $g^{(\ell)}$, est l'hyperrectangle de gravité de C_ℓ, défini par

$$g^{(\ell)} = \left(\left[\frac{1}{n_\ell} \sum_{x_i \in C_\ell} \alpha_{i1}, \frac{1}{n_\ell} \sum_{x_i \in C_\ell} \beta_{i1} \right], \ldots, \left[\frac{1}{n_\ell} \sum_{x_i \in C_\ell} \alpha_{ip}, \frac{1}{n_\ell} \sum_{x_i \in C_\ell} \beta_{ip} \right] \right).$$

Pour des variables multivaluées, le prototype $g^{(\ell)}$ est défini par

$$g^{(\ell)} = (\frac{1}{n_\ell} \sum_{x_i \in C_\ell} q_{1,x_i}(c_1), \ldots, \frac{1}{n_\ell} \sum_{x_i \in C_\ell} q_{1,x_i}(c_{m_1})), ..., (\frac{1}{n_\ell} \sum_{x_i \in C_l} q_{p,x_i}(c_1), \ldots, \frac{1}{n_l} \sum_{x_i \in C_\ell} q_{p,x_i}(c_{m_p}))).$$

Le cas des variables modales est similaire au cas des variables multivaluées. Les fréquences $q_{j,x_k}(c_s)$ sont simplement remplacées par les valeurs de la distribution $\pi_{j,k}$ associées à chacune des catégories de $Y_j(x_k)$. Les fonctions de proximité qui assignent les objets aux classes sont définies à partir des mesures de dissimilarité définies dans le paragraphe précédent.

5.2. *Une méthode basée sur le processus de Poisson non homogène : SCLASS*

La méthode de classification SCLASS suppose que les points observés sont générés par un processus de Poisson non homogène d'intensité $q(.)$ dans un ensemble D où D est l'union de k domaines convexes disjoints D_i $(i = 1, \cdots, k)$. La fonction de vraisemblance, pour les observations $x = (x_1, ..., x_n)$ avec $x_i \in R^p$ $(i = 1, \ldots, n)$, vaut

$$L_D(x) = \frac{1}{(\rho(D))^n} \prod_{i=1}^{n} I_D(x_i) q(x_i)$$

où

$$\rho(D) = \int_D q(x)dx$$

est l'intensité intégrée du processus et I_D la fonction indicatrice de l'ensemble D.

Si l'intensité du processus est connue, les estimateurs du maximum de vraisemblance des k domaines inconnus D_i seront les k enveloppes convexes $H(C_i)$ des k sous-groupes C_i de points tels que la somme des intensités intégrées est minimale.

L'intensité du processus de Poisson non homogène est estimée par la méthode des noyaux [SIL 81] en utilisant un noyau normal. Le paramètre de lissage choisi est celui pour lequel l'estimation change de l'unimodalité à la multimodalité [SIL 86]. Le critère de coupure est le suivant : pour chaque variable on partitionne C en deux classes C_1 et C_2 de telle manière que la somme des intensités intégrées soit minimale. On retient alors la meilleure variable et la partition correspondante. Un critère d'arrêt classique est utilisé.

Dans le cas de variables intervalles, chaque intervalle est représenté par ses coordonnées (Milieu, Longueur) dans l'espace $(M, L) \subset RXR^+$. La valeur de coupure est obtenue en minimisant

$$\int_{M_i}^{M_{i+1}} \rho_1(m)dm + \int_{min(L_i, L_{i+1})}^{max(L_i, L_{i+1})} \rho_2(l)dl$$

où ρ_1 est l'intensité sur l'axe M et ρ_2 l'intensité sur l'axe L.

SCLASS est donc une méthode de classification hiérarchique monothétique divisive. Le résultat est un arbre de décision où chaque classe correspond à un objet symbolique.

6. Les méthodes de détermination du nombre de classes symboliques

Différentes approches ont été analysées.

- Nous avons appliqué les cinq règles d'arrêt issues de l'analyse de Milligan et Cooper aux hiérarchies de partition fournies par les quatre méthodes de classification hiérarchiques, en utilisant la matrice de dissimilarité calculée dans le cadre de ces méthodes hiérarchiques [TRO 04].
- SCLUST n'est pas une méthode de classification hiérarchique. Parmi les cinq règles de Milligan et Cooper, seuls la méthode de Calinski et Harabasz, l'indice C et l'index Γ sont applicables à des ensembles de partitions non emboîtées en ℓ classes $(1 \leq \ell \leq K)$ où K est une constante fixée par l'utilisateur [HAR 04], [HAR 02b], [HAR 02a].
- Le test des Hypervolumes est basé sur le calcul d'enveloppes convexes de points ; il ne requiert pas la connaissance d'une matrice de dissimilarité, mais seulement la position des points. Ces positions sont connues dans chacune des p représentations Milieu-Longueur. Nous sélectionnons parmi les p variables intervalles celle qui contribue le plus à l'inertie de l'ensemble des objets symboliques. Nous retenons alors le nombre de classes donné par le test des Hypervolumes associé à cette variable.

7. Exemples

Les différentes méthodes de classification et de détermination du nombre de classes ont été appliquées à des ensembles de données symboliques simulées tests, mais également à des ensembles de données réelles : les huiles d'Ichino, les boucles mérovingiennes, les magasins "e-fashion stores", ...

8. Conclusion

Nous avons voulu montrer, dans cet exposé, comment certaines méthodes classiques de classification et de détermination du nombre de classes pouvaient être étendues en des méthodes symboliques, en insistant sur les recherches effectuées dans l'unité de statistique de l'Université de Namur. Parmi les autres contributions importantes dans le domaine, soulignons la méthode symbolique de classification hiérarchique monothétique divisive DIV [CHA 97], la méthode de classification hiérarchique et pyramidale HIPYR [BRI 00] ainsi que des extensions importantes de la méthode de classification dynamique [VER 00], [VER 04].

Le logiciel SODAS2 a été développé dans le cadre du projet européen ASSO (Analysis System of Symbolic Official Data). Il intègre un nombre important de méthodes d'analyse des données symboliques parmi lesquelles on trouve les méthodes de classification symboliques SCLUST (dans laquelle est intégrée le module de détermination du nombre de classes NBCLUST), DIV, HIPYR et SCLASS.

9. Bibliographie

[BAK 75] BAKER F., HUBERT L., Measuring the power of hierarchical cluster analysis, *Journal of the American Statistical Association*, vol. 70, 1975, p. 31-38.

[BEA 69] BEALE E., Euclidean cluster analysis, *Bulletin of the International Statistical Institute*, vol. 43, 2, 1969, p. 92-94.

[BEA 02] BEAUTHIER C., Comparaison entre le Gap test et le test des Hypervolumes en classification, 2002, Mémoire, Université de Namur.

[BOC 00] BOCK H.-H., DIDAY E., *Analysis of Symbolic Data. Exploratory Methods for Extracting Statistical Information from Complex Data.*, Springer Verlag, 2000.

[BRI 00] BRITO P., Hierarchical and Pyramidal Clustering with Complete Symbolic Objects, *Analysis of Symbolic Data Analysis, H.-H. Bock and E. Diday (Eds)*, Studies in Classification, Data Analysis, and Knowledge Organization, 2000, p. 312–323.

[CAL 74] CALINSKI T., HARABASZ J., A dendrite method for cluster analysis, *Communication in Statistics*, vol. 3, 1974, p. 1-27.

[CEL 89] CELEUX G., DIDAY E., GOVAERT G., LECHEVALLIER Y., RALAMBONDRAINY H., *Classification Automatique des Données*, Dunod, 1989.

[CHA 97] CHAVENT M., Analyse des Données Symboliques : Une méthode divisive de classification, PhD thesis, Université Paris IX-Dauphine, 1997.

[DUD 73] DUDA R., HART P., *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.

[FIS 71] FISHER L., VAN NESS J., Admissible Clustering Procedures, *Biometrika*, vol. 58, 1971, p. 91–104.

[HAR 82] HARDY A., RASSON J.-P., Une nouvelle approche des problèmes de classification automatique, *Statistique et Analyse des données*, , 1982, p. 41–56.

[HAR 83] HARDY A., Statistique et Classification Automatique : un modèle - un nouveau critère - des algorithmes - des applications, PhD thesis, Université de Namur, 1983.

[HAR 96a] HARDY A., A heuristic approach for the Hypervolumes method in cluster analysis, *Jorbel*, vol. 36, 1996, p. 43–55.

[HAR 96b] HARDY A., On the Number of Clusters, *Computational Statistics and Data Analysis*, vol. 23, 1996, p. 83–96.

[HAR 02a] HARDY A., LALLEMAND P., Determination of the number of clusters for symbolic objects described by interval variables, *Studies in Classification, Data Analysis and Knowledge Organisation*, , 2002, p. 311-318.

[HAR 02b] HARDY A., LALLEMAND P., LECHEVALLIER Y., La determination du nombre de classes pour la méthode de classification symbolique SCLUST, *Actes des huitièmes rencontres de la Société Francophone de Classification*, 2002, p. 27–31.

[HAR 04] HARDY A., LALLEMAND P., Clustering of Symbolic Objects described by multi-valued and modal variables, *Proceedings IFCS 2004*, , 2004.

[HUB 76] HUBERT L., LEVIN J., A general statistical framework for assessing categorical clustering in free recall, *Psychological Bulletin*, vol. 83, 1976, p. 1072-1080.

[KAR 91] KARR A. F., *Point Processes and their Statistical Inference*, Marcel Dekker, Inc., 1991.

[KUB 96] KUBUSHISHI T., On some Applications of the Point Process Theory in Cluster Analysis and Pattern Recognition, PhD thesis, Université de Namur, 1996.

[MIL 85] MILLIGAN G., COOPER M., An examination of procedures for determining the number of clusters in a data set, *Psychometrika*, vol. 50, 1985, p. 159-179.

[PIR 04] PIRÇON J.-Y., Le clustering et les processus de Poisson pour de nouvelles méthodes monothétiques, PhD thesis, Université de Namur, 2004.

[RAS 94] RASSON J.-P., KUBUSHISHI T., The Gap Test : an Optimal Method for Determining the Number of Natural Classes in Cluster Analysis, DIDAY E., LECHEVALLIER Y., SCHADER M., BERTRAND P., BURTSCHY B., Eds., *New Approaches in Classification and Data Analysis*, 1994, p. 186–193.

[RAS 96] RASSON J.-P., GRANVILLE V., Geometrical tools in classification, *Computational Statistic and Data Analysis*, vol. 23, 1996, p. 105–123.

[RIP 77] RIPLEY B., RASSON J.-P., Finding the edge of a Poisson Forest, *Journal of Applied Probability*, , 1977, p. 483–491.

[SIL 81] SILVERMAN B. W., Using Kernel Density Estimates to Investigate Multimodality, *Journal of Royal Statistical Society, B*, vol. 43, 1981, p. 97–99.

[SIL 86] SILVERMAN B., *Principal Component Analysis*, Springer-Verlag, 1986.

[TRO 04] TROCLET J., Méthodes de détermination du nombre de classes pour des objets symboliques, 2004, Mémoire, Université de Namur.

[VER 00] VERDE R., CARVALHO F. D., LECHEVALLIER Y., A Dynamical Clustering Algorithm for Multi-Nominal Data, *Data Analysis, Classification, and Related Methods, H.A.L. Kiers, J.-P. Rasson, P.J.F. Groenen and M. Schader (Eds.)*, Springer Verlag, Heidelberg, 2000, p. 387–394.

[VER 04] VERDE R., Clustering Methods in Symbolic Data Analysis, *Proceedings IFCS 2004* , 2004.

Transformation de longues séries temporelles en descriptions symboliques

Georges Hébrail

ENST Paris, LTCI-UMR 5141 CNRS
Département Informatique et Réseaux
46, Rue Barrault
75013 Paris, France

RÉSUMÉ. De nombreuses longues séries temporelles sont disponibles sous forme informatique (courbes de consommation d'électricité, d'eau, de gaz, courbes de ventes des grandes surfaces, courbes de trafic automobile, …). Leur forme brute (succession de données numériques) est peu adaptée aux traitements de fouille de données où l'on cherche à extraire une information de haut niveau. Cette communication décrit des méthodes de construction de descriptions symboliques de longues séries temporelles. Deux approches sont exposées : la segmentation en épisodes à niveaux constants, et la classification automatique d'épisodes de durée fixe. Dans une dernière partie, il est montré que les descriptions symboliques obtenues peuvent trouver de nombreuses applications.

MOTS-CLÉS : Séries temporelles, courbes, représentation symbolique, classification automatique.

1. Introduction

De nos jours, la plupart des activités humaines sont assistées par des systèmes informatiques qui permettent, au-delà de la fonction assurée, de disposer d'une trace chronologique de ces activités. Par exemple, les entreprises de distribution d'électricité, d'eau, de gaz disposent de courbes de consommation de leurs clients, courbes initialement générées pour les opérations de facturation. De même, les banques disposent des historiques des soldes des comptes de leurs clients, les opérateurs de télécommunications disposent de courbes de consommations téléphoniques, les grandes surfaces de distribution disposent de courbes de ventes magasin par magasin, produit par produit, au fil du temps. D'autres sources d'informations de nature chronologique peuvent être générées par des mesures réalisées périodiquement, comme par exemple les mesures de trafic automobile dans les grandes agglomérations ou sur les grands axes. La Figure 1 donne un exemple de courbe de consommation d'électricité d'une entreprise sur une durée de trois mois.

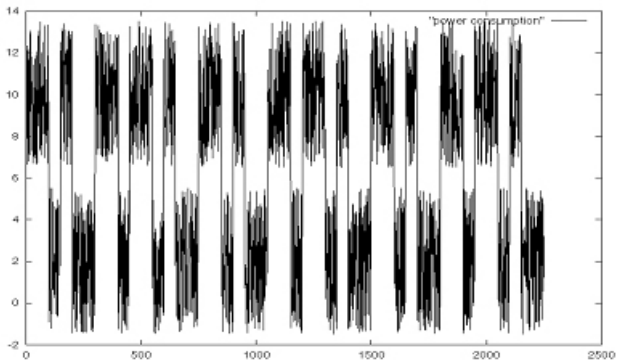


Figure 1 : Longue série temporelle

Dans cette communication, on s'intéresse à l'analyse exploratoire d'une ou plusieurs longues séries temporelles de type numérique, typiquement des séries d'une année, comportant un point par heure. L'objectif

est de synthétiser cet ensemble de séries temporelles, afin de mieux comprendre les comportements sous-jacents : le plus souvent il s'agit du comportement de clients d'entreprises, dans une optique marketing.

Nous proposons de transformer chaque série temporelle en une séquence de symboles d'un alphabet de faible taille (typiquement une dizaine de symboles), constituant ainsi une information résumée de la série, directement interprétable par l'humain ou pouvant être traitée par des techniques de fouilles de données. Deux approches complémentaires ont été étudiées dans la thèse de Bernard Hugueney [HUG 03] :

- La segmentation de la série en épisodes contigus de durée variable, chaque épisode étant alors représenté par une courbe constante à un niveau prenant ses valeurs dans un ensemble discret de valeurs. Les symboles correspondent dans ce cas aux différents niveaux discrets.
- Le découpage de la série en épisodes contigus de même durée (par exemple les jours), et l'association à chaque épisode d'une forme type obtenue par classification automatique des formes observées sur une ou plusieurs séries. Les symboles correspondent dans ce cas aux formes type.

La Section 2 présente l'approche par segmentation et la Section 3 présente l'approche par classification automatique. La Section 4 décrit les applications possibles des représentations symboliques des longues séries temporelles.

2. Représentation symbolique par segmentation

L'approche de base consiste à découper la série en *P* épisodes contigus (encore appelés segments) et à approcher la série par une fonction constante sur chacun des segments, où *P* est un paramètre de l'algorithme fixé à l'avance. L'écart entre la série initiale et son approximation par des segments à niveau constant est évalué par la somme des erreurs quadratiques en chacun des points de la série. Le problème posé est donc la détermination de *P-1* points de coupure de la série, minimisant la somme des erreurs quadratiques. A l'optimum, chaque segment est approché par une fonction constante dont le niveau est égal à la moyenne des points du segment. Ce problème trouve sa solution exacte par programmation dynamique ([BEL 61]) avec une complexité en $O(PN^2)$, où *N* est le nombre de points de la série à segmenter. Des algorithmes approchés ont été proposés ([KEO 01]), permettant de réduire la complexité à $O(ln(P)N)$ par exemple.

Des critères ont été proposés et évalués pour choisir le nombre d'épisodes *P* (voir [HUG 03]). Le choix de *P* est fortement lié à l'utilisation qui est faite du résultat de la segmentation. La Figure 2 montre le résultat d'une segmentation avec *P=6*.

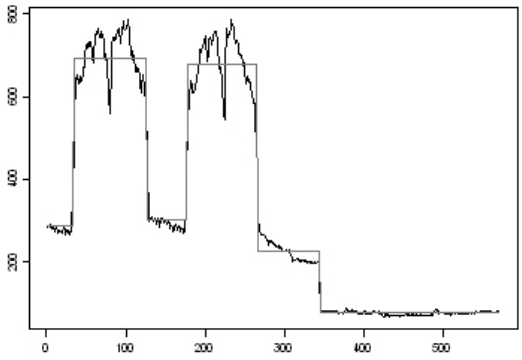


Figure 2 : Segmentation d'une série

La segmentation décrite ci-dessus définit, pour chaque épisode, un niveau constant égal à la moyenne des valeurs de l'épisode. Les différents épisodes ont donc des niveaux tous (ou presque tous) différents. Dans [HUG 02] et [HUG 03], il est proposé de rechercher les points de coupure de la série de telle sorte que le niveau des segments ne prenne sa valeur que dans une liste prédéfinie de niveaux discrets fixés à l'avance (environ une dizaine). On parle alors de segmentation *prototypique*. On peut donc considérer dans ce cas que la série est transformée en une séquence de symboles, chaque symbole correspondant à l'un des niveaux discrets définis à l'avance. La complexité de l'algorithme optimal est alors en $O(KPN^2)$, où *K* est le nombre de niveaux discrets de l'alphabet. Les niveaux discrets peuvent être définis par une classification automatique de l'ensemble des valeurs

prises par la ou les séries temporelles à traiter. La Figure 3 donne un exemple de segmentation prototypique de la série de la Figure 2, avec *K=3 et P=6*.

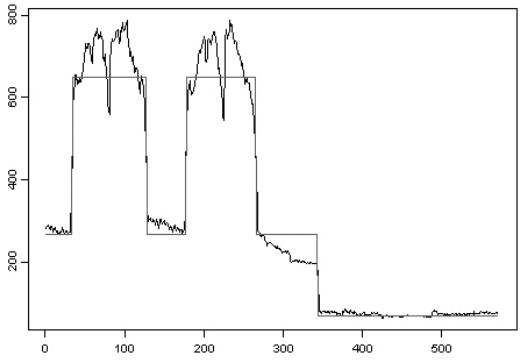


Figure 3 : Segmentation prototypique d'une série

3. Représentation symbolique par classification automatique

Dans l'approche par segmentation, la durée des épisodes est variable, mais la forme de la courbe au sein de chaque épisode est imposée. L'autre approche explorée dans [HUG 03] consiste à donner plus de richesse à la forme de la courbe au sein de chaque épisode, mais en imposant que la durée de tous les épisodes soit la même. Cette durée est supposée fixée par l'utilisateur (par exemple une journée ou une semaine dans la pratique), ainsi que le point de la série où commence le premier épisode (*offset*).

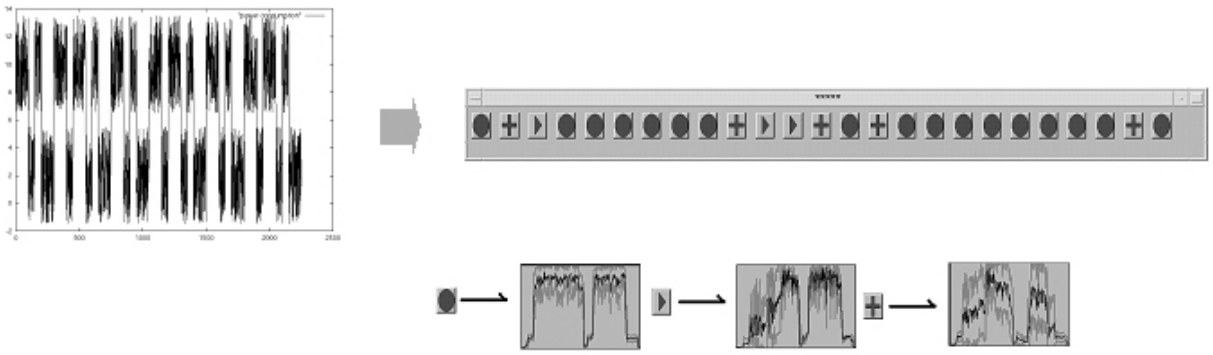


Figure 4: Représentation symbolique par classification automatique

Le principe de construction de la description symbolique par classification automatique est le suivant :

- La ou les séries à transformer sont découpées en épisodes de durée fixe, à partir de l'offset. On obtient donc un ensemble de « petites » courbes de même durée.
- Une classification automatique est réalisée sur l'ensemble des courbes des épisodes extraits dans l'étape précédente. Les classes obtenues définissent les symboles de l'alphabet, en leur associant la courbe moyenne de la classe.
- Les séries initiales sont transformées en une séquence de symboles (chaque épisode est représenté par le symbole de la classe à laquelle sa courbe appartient), comme l'illustre la Figure 4. Chaque épisode peut également être décrit par des attributs temporels (comme le jour, le mois, la saison, …).

Dans ses expérimentations, B.Hugueney a utilisé les cartes auto-organisatrices de Kohonen ([KOH 95]) comme méthode de classification automatique, mais d'autres méthodes peuvent être employées, comme les méthodes de la famille des nuées dynamiques.

Différents paramètres sont à régler pour construire les descriptions symboliques par classification automatique :

- La durée des épisodes : celle-ci peut être guidée par une analyse fréquentielle de la série par transformation de Fourier.
- L'offset depuis le début de la série : celui-ci peut être choisi arbitrairement en début de journée (0h) ou de semaine, ou peut être optimisé, par exemple en recherchant l'offset dont tous les points anniversaire sont de variance minimum.
- Le nombre de classes de la classification automatique (qui correspond au nombre de symboles) : il peut être guidé par les indicateurs standard en classification automatique, mais il est préférable de faire intervenir l'utilisateur pour valider les classes et supprimer éventuellement les classes correspondant à des individus atypiques. Dans tous les cas, le nombre de classes doit rester petit (en général une dizaine) afin que la description symbolique de la série reste intelligible.

4. Applications des descriptions symboliques

La représentation symbolique d'une série temporelle n'est intéressante que si le nombre de symboles différents reste faible et si chaque symbole porte une information de haut niveau conceptuel. Les niveaux prototypiques d'une segmentation et les formes type d'une représentation par classification sont des bons candidats pour définir des symboles de haut niveau conceptuel, mais il faut veiller à ce que chaque symbole retenu trouve une interprétation intuitive auprès de l'utilisateur. Par exemple, pour des séries de trafic automobile, trois niveaux prototypiques peuvent être définis, correspondant à des trafics 'fluide', 'ralenti', et 'congestionné'. Sous la condition que les symboles élaborés lors de la construction des descriptions symboliques aient un sens, de nombreuses applications sont possibles :

- Visualisation des descriptions symboliques : ainsi que le montre la Figure 4, les descriptions symboliques par classification automatique peuvent être facilement visualisées avec une lecture plus facile que la visualisation de la série initiale.

- Statistique descriptive : les techniques standard de statistique descriptive et de visualisation peuvent être appliquées aux descriptions symboliques. Par exemple, sur des séries de trafic automobile, il est immédiat (et informatif) de compter le nombre de jours dans l'année où un axe routier est congestionné.

- Détection d'aléas et correction de valeurs aberrantes : en comparant la série initiale avec sa représentation symbolique, il est possible de détecter des épisodes s'éloignant de leur forme type associée. Ces épisodes peuvent correspondre à des aléas pour lesquels une alarme est déclenchée, ou bien à des défauts de mesures, pour lesquels la forme type permet d'estimer de bonnes valeurs pour les valeurs aberrantes.

- Fouille de données exploratoire : la forme « symbolique » de la représentation symbolique permet d'utiliser facilement la plupart des méthodes de fouille de données exploratoire. Les résultats produits par ces méthodes sont facilement interprétables du fait de l'existence d'une interprétation des symboles. Les méthodes de recherche de séquences fréquentes ([AGR 95]) sont particulièrement adaptées au traitement de séquences de symboles que sont les descriptions symboliques de séries temporelles.

- Fouille de données décisionnelle : les descriptions symboliques de séries temporelles peuvent facilement être utilisées par les méthodes de fouilles de données décisionnelles (arbres de décision, réseaux de neurones, …). Leur caractère symbolique leur permet de pouvoir intervenir dans les modèles de prédiction soit comme prédicteur, soit comme variable à prédire (prévision de séries temporelles).

- Bases de données de séries temporelles : les descriptions symboliques de séries temporelles sont adaptées au stockage et à l'indexation de séries temporelles dans les bases de données. Par exemple, dans une base de données stockant une description symbolique de séries temporelles de trafic automobile, il est facile de répondre à la requête suivante : « trouver les grands axes où le trafic est toujours fluide le week-end ».

5. Conclusion

Cette communication a présenté rapidement les deux approches développées dans [HUG 03] de transformation de longues séries temporelles en descriptions symboliques. Le lecteur intéressé se reportera à [HUG 03] pour une définition précise du modèle de représentation symbolique de séries temporelles, ainsi que pour le détail des algorithmes de construction de ces représentations.

Dans [HUG 02] il est également proposé une définition de représentations symboliques *floues* de séries temporelles, avec des algorithmes permettant de les construire. L'introduction du flou permet d'améliorer la précision de la représentation par rapport à la série initiale, tout en conservant un petit nombre de symboles pour une interprétation aisée de la représentation.

Sur les nombreuses applications proposées dans la Section 4, seules quelques-unes ont été effectivement explorées : beaucoup de travail reste à faire dans la continuité de ce qui a été décrit.

6. Bibliographie

[AGR 95] AGRAWAL, R. SRIKANT, R., " Mining Sequential Patterns ", Proceedings of 11th Int'l Conf. on Data Engineering (DE'95), Taipei, Taiwan, 1995.

[BEL 61] BELLMAN R., " On the approximation of curves by line segments using dynamic programming ", *Communications of the ACM*, VOL.4, N°6, Juin 1961.

[HEB 01] HÉBRAIL G., HUGUENEY B., " Symbolic representation of long time series ", Conference on Applied Statistical Models and Data Analysis (ASMDA'2001), Compiègne, Juin 2001.

[HUG 02] HUGUENEY B., BOUCHON-MEUNIER B., HEBRAIL G., LE P., " Segmentation de séries temporelles en segments de niveaux prototypiques et de durées floues ", Rencontres francophones sur la logique floue et ses applications, Montpellier, France, Octobre 2002.

[HUG 03] HUGUENEY B., " Représentations symboliques de longues séries temporelles ", Thèse de doctorat de l'Université Paris VI, Janvier 2003.

[KEO 01] " An online algorithm for segmenting time series ", IEEE International Conference on Data Mining, 2001.

[KOH 95] KOHONEN T., *Self organizing maps*, Springer, Berlin, Heidelberg, 1995.

Mapping gene family data onto evolutionary trees

Boris Mirkin

School of Computer Science and Information Systems,
Birkbeck University of London
Malet Street, London, WC1E 7HX, UK

RÉSUMÉ. Des modèles destinés à placer des données de familles de gènes individuels sur un arbre évolutionnaire d'espèces sont présentés. Plus précisément, une famille de gènes est vue comme un ensemble de protéines homologues appartenant à différentes espèces. Les données peuvent être présentées soit sous la forme de (a) une matrice de distances entre protéines, soit (b) un arbre de « gènes » dérivé de cette matrice, ou simplement (c) par le profil phylogénétique qui est un vecteur booléen indiquant la présence ou l'absence de la famille dans les feuilles de l'arbre. Des algorithmes seront présentés. Ils sont destinés à appliquer tous ces types de données sur un arbre évolutionnaire d'espèces préexistant pour pouvoir reconstruire l'histoire de l'évolution du gène. Quelques résultats expérimentaux seront aussi présentés.

MOTS-CLÉS : Arbre évolutionnaire, arbre de gènes, reconstruction de l'histoire des gènes, optimisation.

1. Introduction

The rooted tree with leaves labelled by taxa is a natural form of the biological taxonomy related, from the Darwin's times, to evolution. With the development of molecular biology and genomics, evolutionary trees, or phylogenies, became a major instrument for aggregated presentation and visualisation of interrelation among species. Methods for building evolutionary trees based on the premise: the more similar proteins the more recently they diverged, became an indispensable tool of a molecular biologist or bioinformatician (see, for instance [NEI 00], [FEL 01]).

To further advance into the understanding of natural phenomena and fuller exploit the aggregating capabilities of evolutionary trees, methods for mapping other data onto the trees should be advanced as well. This, however, is a twilight area at which only occasional efforts have been made so far.

In this paper some models for biologically meaningful mapping of data of individual gene families onto an evolutionary species tree are considered. A gene family, typically, is observed as a set of homologous proteins belonging to different species. The data of it can be presented either with a between-protein distance matrix, or a "gene" tree derived from the matrix, or just the phyletic profile, that is, a Boolean vector indicating presence/absence of the family in species under consideration. Theoretical and computational models will be presented for mapping of these data types to a pre-specified evolutionary tree so that the evolutionary history of individual genes can be reconstructed. Results found jointly with E. Koonin (NCBI NIH USA) and collaborators will be presented [MIR03], [MIR04].

In our view, the approach may be more generally applicable to situations at which a general classification tree is present along with some partial data on the same entities, so that the partial data could be meaningfully mapped to the tree. Perhaps such is the situation in which a number of texts is to be interpreted by mapping them onto a semantic structure tree (ontology).

2. Mapping individual gene trees

2.1. *Duplication and reconciled tree*

Duplication of genes is considered a major mechanisms of evolution: after a gene is duplicated on a chromosome, its copies may acquire different mutations so that eventually the copies may come to bear different functions [NEI 00]. Duplications of genetic material with follow-up losses may be used to explain the empirical

fact that an evolutionary tree built on similarities between proteins from a gene family may differ from a species tree built using many sources of information.

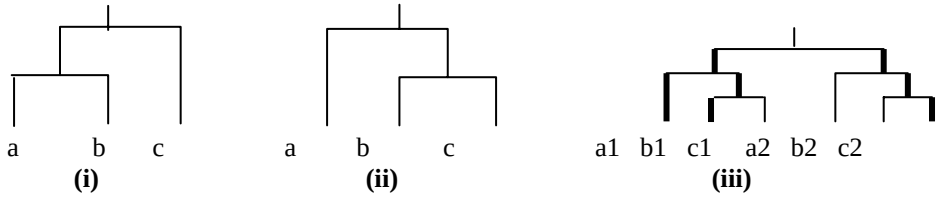


Figure 1. A gene tree (i) versus species tree (ii): the reconciled tree (iii) explains the difference by the duplication of the species tree with losses of lineages leading to c1, a2 and b2.

For example, gene tree (i) on Figure 1 may reflect similarities between haemoglobin proteins in horse (a), chimpanzee (b) and human (c) if those in horse and chimp belong to the haemoglobin alpha-lineage and that in human to the beta-lineage. Figure 1(iii) shows the "reconciled" correct evolutionary tree that would emerge if all alpha- and beta-haemoglobin proteins in the species have been collected. According to this tree, a duplication of the gene of haemoglobin occurred before the species' common ancestor appeared so that the "wrong" shape of gene tree (i) is an artefact caused by the data available at drawing it, which is expressed by solid lines in Figure 1(iii).

Here, as well as further on, all trees considered are binary rooted trees with leaves labelled by corresponding species.

The strategy of copying those subtrees of the species tree at which the gene tree differs from the species tree has been promoted in cladistics for building the so-called reconciled species tree [PAG 94, 97]. It allows to draw intuitively appealing pictures and, also, easily accommodates the presence of the so-called paralogs, protein products of duplicate copies of a gene. However, the reconciled tree is difficult to handle both theoretically and computationally in the situations at which the number of inconsistencies grows large as inevitably happens when treating a number of different gene trees.

2.2. *Annotating model of duplication*

Another formalisation of the concept of duplication was proposed in [MIR 95] according to the rules delineated in Figure 2.

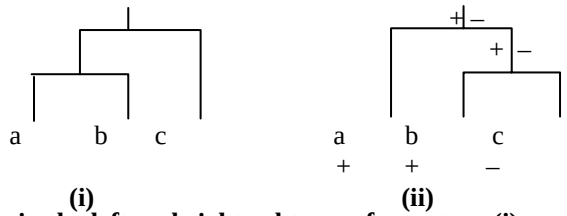


Figure 2. Species in the left and right subtrees of gene tree (i) annotated by symbols + and – in the species tree (ii), with the follow-up raising the symbols along the tree.

A duplication is manifested in the difference between the contents of a divergence in the gene tree and species tree. If this is the case, the species tree can be annotated by differently labelling, on the species tree, the leaves of each of the two diverged subtrees of the gene tree. Then these labels, + and – on Figure 2, are extended bottom-up by annotating each internal node with the labels of its leaf contents to represent the history of the corresponding duplication at the species tree. A loss event then is declared if a node has only one of the labels whereas its parent has both. This concept is less visually intuitive than that of the reconciled tree, but it allows to map different gene trees onto the same species tree without changing it and, moreover, can be analysed mathematically. In particular, let us refer to a pair of nodes (g,s), g being of the gene tree and s of the species tree, as a crossing if the leaf contents of s overlaps that of each of the children of g. Such is cluster s comprising leaves b and c at tree (ii) with regard to g being the root of tree (i) on Figure 2. If g-contents is part of s-contents, then crossing (g,s) corresponds to a duplication event, otherwise, pair (g ,s) will be referred to as incompatible. It has been proven that the total number of losses at mapping a gene tree to the species tree is equal to the number of incompatible pairs plus twice the number of duplications. This implies that the total number of losses is at least three times greater than the number of duplications [MIR 95], [EUL 98].

2.3. *Lca mapping as a computationally effective approach*

Every node g of a gene tree G can be uniquely mapped to that node s(g) of a species tree S whose leaf contents is the minimal among those containing the leaf contents of g. Thus, s(g) is the last common ancestor (lca) in S to all

species descending from g in G. The lca mapping is not computationally intensive: it can be done in a linear time over the size of the leaf set.

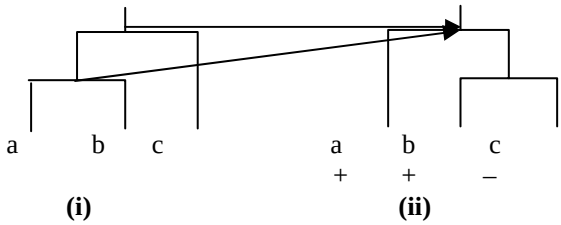


Figure 3. Lca mapping of internal nodes of gene tree (i) onto species tree (ii).

The lca mapping M: G $\rightarrow$S maps a pair node-parent (g, pg) of gene tree G to a pair (M(g),M(pg)) of nodes in S in such a way that either: (a) M(pg) is the parent of M(g) in S (consistency), or (b) M(pg)=M(g) (contraction); a contraction is one-sided if M(g')$\neq$M(pg) where g' is sibling of g in G, or (c) there are intermediate nodes between M(pg) and M(g) in S (retraction). The inconsistencies of the lca mapping, contraction (b) and retraction (c), appear to be closely related to the duplications and losses above: the contractions correspond to duplications and intermediate nodes in the retractions correspond to those duplications for which the collateral children are losses [EUL 98]. The total number of losses in annotating S by G-duplication based events is the number of all the intermediate nodes plus the number of one-sided contractions under the lca mapping M [EUL 98].The latter, lca mapping based measure, was used in a slightly different form by [GUI 96]; actually, the development of the annotating duplication model in [MIR 95] was motivated by the need to give a substantiation to the heuristic measure used in [GUI 96]. The fact that the lca mapping based measure is equal to the number of losses under the model of annotating duplication is proven in [ZHA 97], [EUL 98].

2.4. *Consensus trees*

Given a number of individual gene trees, the total number of losses and duplications can be used as the scoring function to derive a consensus tree with local search heuristics. This strategy was pursued, starting from the pioneering work [GUD 79], in [GUI 96], [PAG 97], [ARV 03] and others. Arguably, none of the attempts has been quite successful. This probably can be attributed to the combinatorial complexity of the problem, potentially biased representation of gene families, and missing of other important mechanisms of the evolution such as the horizontal transfer from the model. An attempt of incorporating the horizontal transfer to the setting is described in [ADD 03].

3. Parsimoniously mapping an individual gene set to the tree

3.1. *The problem*

Given an evolutionary tree over species, the phyletic profile of a gene family is specified by the subset of tree leaves labelled by the extant species at which the family is present. An evolutionary scenario leading to the observed phyletic profile may involve the evolutionary events of *emergence, inheritance, horizontal transfer* and *loss*. No duplication concept has been used so far in this context, since the phyletic profile does not take into account the number of homologous proteins within a species. We refer to both emergence and horizontal transfer of a gene as a *gain*. The total number of loss and gain events in a scenario shows the extent of incompatibility between the evolutionary histories of the given gene according to this particular scenario and that implied by the topology of the species tree. Among all possible scenarios, we select those that are most parsimonious, i.e. require the minimum number of events to explain the observed phyletic profile. The two types of events, loss and gain, are likely to require different weighting in order to construct realistic evolutionary scenarios, which can be achieved by introducing the gain penalty *gp* that can differ from 1. Then a parsimonious scenario should minimize the total score; thus reaching the *minimum inconsistency* of the gene family.

3.2. *Algorithm PARS and its properties*

This approach has been pursued in [SNE 02], [MIR 03], and [KUN 03] with different computational schemes, of which only [MIR 03] gives a genuine optimisation algorithm. This algorithm builds a parsimonious scenario for each parent node using parsimonious scenarios for its children. At each node of the tree, sets of loss and gain events are maintained under both the assumption that the gene has been inherited at the node and the assumption that it has not been inherited. A loss can occur only under the former assumption and a gain under the latter.

Consider the parent-children triple as shown in Figure 4, with each node assigned sets of loss and gain events under the above two inheritance assumptions. Let us denote the total number of events under the inheritance and

non-inheritance assumptions by e_i and e_n, respectively, where gains are weighted by the gain penalty *gp*. An evolutionary scenario at a given node is defined by a pair of sets (G, L), representing the gains and losses in the subtree rooted at the node. We use (G_i, L_i) and (G_n, L_n) to denote scenarios under the inheritance and non-inheritance assumptions, respectively. In a parsimonious scenario, the parental inconsistency score can be derived from those of its children as $e_i = \min (e_{n1} + e_{n2} + 1, e_{i1} + e_{i2})$ or $e_n = \min (e_{i1} + e_{i2} + g, \quad e_{n1} + e_{n2})$, under the inheritance or non-inheritance assumption, respectively. These lead to a recursive algorithm PARS for building parsimonious scenarios described in detail in [MIR 03]. At a leaf node the four sets G_i, L_i, G_n and L_n are empty, except that $G_n = \{a\}$ if gene a is present in the given leaf or $L_i = \{a\}$ if a is not present. The algorithm then computes parsimonious scenarios for parental nodes according to the topology of the tree using the rules given above, proceeding from the leaves to the root.

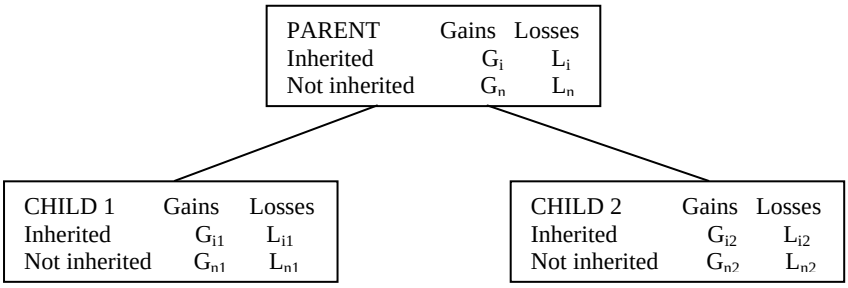


Figure 4. Patterns of events in a parent-children triple according to a parsimonious scenario.

The recursive structure of the algorithm PARS enables one to derive all parsimonious scenarios compatible with the phyletic profile of a gene. In some cases their number can be rather high. To have a unique outcome, a secondary criterion of minimising the number of gains was utilised in [MIR 03]. Some other properties of the criterion [MIR 03]:

(1) The total number of gain and loss events in a parsimonious scenario is a monotone function of the difference between gain penalty *gp* and 1, getting its minimum at *gp*=1.

(2) No gene family can emerge in a node being ascendant to the last common ancestor to the set of organisms to which the family belong.

(3) No gene can be lost at the children of a node at which it emerged.

The method can incorporate a maximum likelihood approach, and take into account co-functioning of genes.

3.3. *Reconstructing LUCA*

This method has been used at an evolutionary tree involving 26 extant species of bacteria, archaea and yeast for reconstructing the evolutionary histories of about 2700 gene families represented by the so-called clusters of orthologous groups (COGs) in [MIR 03] at different gain penalty values ranging from 0.1 to 10. The reconstructed contents of the root represents the last ultimate common ancestor (LUCA); it was used as an external criterion for selecting the gain penalty value. At *gp*=1 the set of 572 genes comprising LUCA was recognised as best approximating an organism capable of survival and reproduction; the relatively small inconsistencies were caused by mutual inconsistencies between the method and data, e.g., the mitochondrial origin of some gene families. This led the authors to conclude that the events of horizontal transfer in the process of evolution have been as frequent as loss events, which contradicted to previously expressed opinions, as well as to conclusions of [SNE 02] and [KUN 03]; note however that these authors did not use any external criterion for selecting the gain penalty in [SNE 02] or equivalent constants in [KUN 03].

3.4. *Relation to reconstruction of ancestral characters: principles of maximum likelihood and parsimony*

Let us treat a gene family as a character that may be present or not in any node of the tree, so that the character's state change from 1 to 0 corresponds to a loss and from 0 to 1 to a gain. Then the problem of building of an evolutionary scenario becomes equivalent to the problem of reconstructing the character's ancestral states. Two popular approaches to this latter problem are the principle of Maximum Parsimony (MP) and the principle of Maximum Likelihood (ML) [NEI 00].

The principle of Maximum Parsimony is exactly that used above. A method of doubly running through the tree, differing from that in [MIR 03], has been proposed in [FIT 71] and [HAR 73]. The principle is implemented as part of the maximum parsimony principle for building evolutionary trees [FEL 01], [MAD 92]. It is simple and

intuitive. Its shortcoming is that it frequently leads to situations with several equally parsimonious scenarios drawing very different histories of the gene evolution.

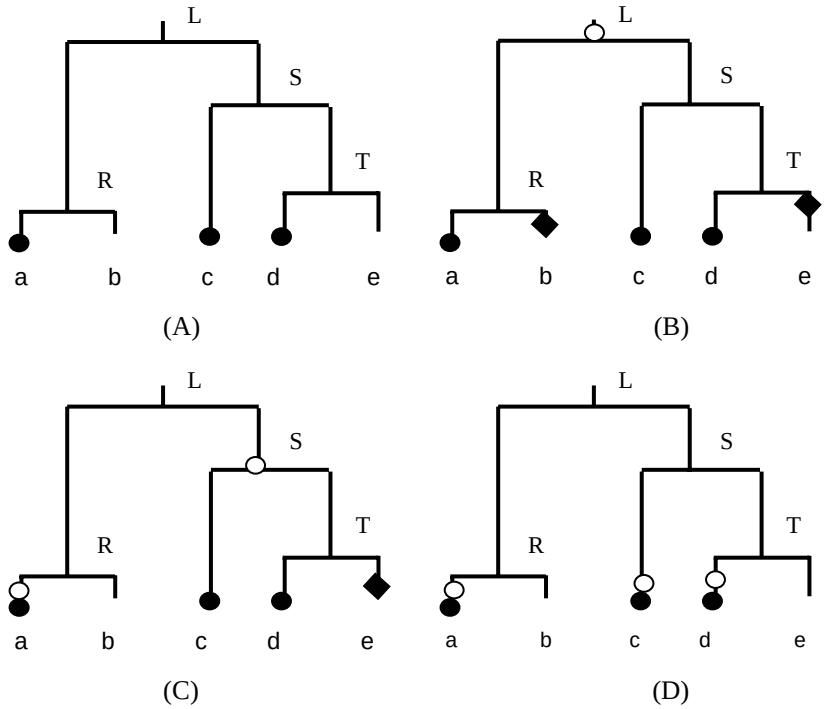


Figure 5. An evolutionary tree with a gene family's presence at a, c, and d denoted by a black circle (A). Equally parsimonious scenarios for the family's history are at (B), (C) and (D) with a white circle denoting a gain event and a black diamond denoting a loss.

Consider, for example, the case of a gene family that is present at three of the five extant species of Figure 5 (A). There are three equally parsimonious scenarios of its history so that an external criterion to choose one scenario from the set should be specified. A number of general selecting criteria were considered in [MAD 92], [MIR 03].

Another set of criteria emerges in the setting of the Maximum Likelihood (ML) approach. This approach involves two types of quantitative information, the evolutionary tree branch lengths and a probabilistic model of transformation of a distribution of presence/absence of the gene family along tree branches. Typically, a continuous-time Markov model is utilized involving constant rates q_{ij} of change of character states, that is, gene's presence or absence, from *i* to *j* where both *i* and *j* can be 1, for presence, or 0, for absence [YAN 95, 96].

In a typical situation such a model would lead to a unique most likely scenario for the gene history, which, at the first glance, supports the idea of superiority of the ML over MP. However, there are other issues that hinder the use of the Maximum Likelihood. First, the estimates of the change rates are solutions to a complex optimisation problem that can be solved only locally so that the solution much depends on the initial setting, thus biased towards a specific system of penalty weighting. Second, there is a clash between the maximum likelihood used for deriving the probabilistic model from data and the maximum likelihood used for deriving the gene histories from the model. The probabilistic model implies a probabilistic distribution over a number of possible evolutionary scenarios, of which only one, the most likely, is selected each time. This procedure, when applied to a multitude of gene families, obviously would lead to a biased empirical distribution of gene gains and losses.

4. Within-family distances and principle of maximum correlation

4.1. *Principle of maximum correlation*

Deriving evolutionary histories should involve not the phyletic profile only but similarities between proteins constituting a gene family, the approach being currently developed jointly with E. Koonin, Y. Wolf and T. Fenner [MIR 04]. We consider that any gene family C embracing organisms of subset S of the set I of organisms under consideration is accompanied with matrix B_C of dissimilarity coefficients b_{jk} for all $j,k \in S$. Any evolutionary scenario h of gene C history should lead to a scenario-based distance matrix $D_C(h)$ so that the quality of the scenario could be scored according to the correlation between $D_C(h)$ and B_C. The higher the

correlation, the better the history. This can be formulated as the principle of Maximum Correlation (MC). Obviously, the MC principle should supplement the MP and ML rather than substitute them.

4.2. *Directed scenario and scenario-generated tree distance*

To implement the MC principle, an evolutionary tree will be considered timed, that is, each node in it assigned with an estimate of time at which the corresponding ancestor species diverged in the process of evolution to give rise its "children" species. Given a set of gains, that is, nodes in the evolutionary tree, ordered over the times assigned to them, this can be further extended in what we call a directed scenario, as follows. The elder gain node is postulated to be the node at which the gene under consideration has emerged.

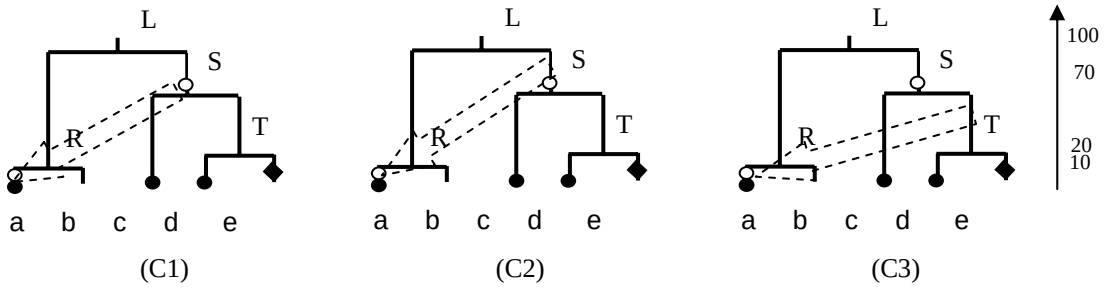


Figure 6. Three versions of the horizontal transfer event: (C1) directly from node S to node a, (C2) from the middle of S's life span to the middle of a's life span, and (C3) from the middle of T's life-span to the middle of a's life span.

Each of the other gains, g, is assigned with its source, s, which must be a node belonging to a subtree generated by an older gain such that s's life span either being earlier than that of g or overlapping with it. Figure 6 illustrates three possible directed scenarios for the case of a gene family comprising three extant species corresponding to the "static" scenario C from Figure 5. Each of these scenarios can be differently reflected in the matrix of distances between organisms in the family.

Indeed, if the horizontal transfer itself takes no time, the distance between a and c, which is 200 in Figure 5, becomes equal to just 70, the distance between S and c, according to C1, or, with mid-life-spans added, 90=70+15+5 according to C2, or 100=70+25+5 according to C3, with 25 being the distance from S to the middle point of its edge to T. Overall the within gene distance matrices will be

```
      c   d          c  d          c  d          c   d
D=|200  200|  D1=|70 70|  D2=|90  90|  D3=|100  50| a
  |      140|     |  140|     |   140|     |      140| c
```

according to the original timed tree and scenarios C1, C2, C3, respectively.

This distance matrix can be proven to be uniquely defined by a directed scenario.

4.3. *Synchronization of a horizontal transfer*

The directed scenarios at Figure 6 are not entirely appropriate because the transfers between sources and targets in them are placed at different moments of time. To synchronize the target of a horizontal transfer and its source, one should take into account the pattern of interrelation between their life spans. Three generic patterns are shown on Figure 7: (i) and (ii), target's span overlaps that of the source, (iii) the source had died out before the target emerged. In case (iii) the transfer is to have been through a "relation lineage" which is absent from the tree as highlighted on the right-hand side of pattern (iii).

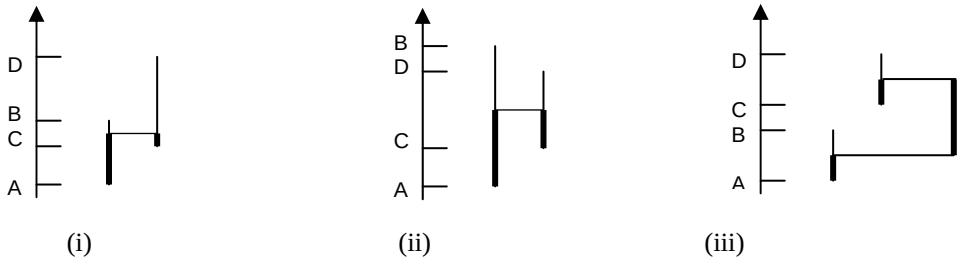


Figure 7. Generic patterns of interrelation between life spans of a horizontal transfer target, AB, (vertical line on the left) and its source, CD (vertical line on the right), to generate the synchronising distance between them, AB at (i) or AD at (ii) and (iii).

For the sake of simplicity, the midranges of the life span intervals are taken as the transfer time points.

With the synchronization, the within gene family distance matrices corresponding to scenarios C2 and C3 of Figure 6 will become D2' and D3' as follows:

```
          c    d              c    d
D2'=    |170 170|   D3'=|100  90| a
        |    140|       |    140| c
```

These distances are relevant only at distances between extant organisms descending from the target and the source. Otherwise, the distances should take into account the transfer points only.

4.4. *Virtual change of tree topology effected by a directed scenario*

In many aspects, a directed scenario is equivalent to changing the evolutionary tree topology by joining each subtree rooted at a target to the edge above its source (as presented on Figure 8).

4.5. *Experimental results*

A newly developed algorithm MaCor implements the line of thought described above and involving procedures for (1) deriving directed-scenario-based tree distances between organisms within a family, (2) measuring correlation between protein-based within-family distance matrices and those scenario-based ones and for (3) greedy-wise building of the directed scenario maximally correlated with the protein based distance matrix. Our algorithms have been applied to the updated evolutionary tree built by E. Koonin and Y. Wolf over 66 organisms and 4873 gene families (COGs) produced in NCBI NIH USA.

Figure 8. Changed tree topologies corresponding to directed scenarios C2 and C3 on Figure 6.

In particular, algorithm PARS produced a LUCA with 489 gene families at *gp*=1. The number decreased from the 572 found at the tree of 26 organisms (see section 3.3) because of more complex topology of the evolutionary tree at 66 organisms. This result was compared to that found with the algorithm MaCor applied to all parsimonious scenarios derived at *gp* varying through 1, 2 and 3. This way we have been able to select the gain penalty weight according to MC principle. Of 4873 gene families, the best correlation of the directed scenarios and protein-based distance matrices was achieved for 3975 of them at *gp*=1, 602 at *gp* =2, 295 at *gp*=3 and 35 at greater gain penalties. Altogether, only 444 families appear to be in LUCA according to MaCor, of which only 325 are common to those found with algorithm PARS (note that the latter always selects the minimum number of gains thus forcing a family to belong to LUCA anytime when it is possible.) These results show that further improvements of MaCor should be made in all of its three major procedures above.

5. Conclusion

The frameworks presented can be seen as different approaches to mapping within gene family similarity data onto an evolutionary tree. One approach first produces a gene tree, then maps it to the species tree, whereas the other approach first takes the gene phyletic profile, maps it to the species tree and then fine tunes it into a directed scenario based on the similarity data. The first approach is natural for modelling duplications, the second for horizontal transfers, though there are ways for extending either to both types of events.

The difference between the approaches increases when looking at their scoring mechanisms: at the mapping of gene trees, differences at deeper parts of the species tree may formally translate into an excessive number of duplication and loss events. Consider, for example, a group of species descending, in the species tree, from a node p whose another child is just a leaf s. If the group and s have the same last common ancestor in the gene tree, but s now is not an outsider but sibling of another leaf in the group, then this difference will be translated into as many duplications as there are nodes between s and p in the gene tree, because each pair child-parent on the path will be contracted under the lca mapping. Intuitively, one may think that the difference between the species and gene trees should be attributed to just one event occurred at the level p, not many, which points to an inadequacy of the analysed concept of duplication. In terms of phyletic patterns such a difference would be

attributed to a horizontal transfer event. It is quite clear, at least in principle, how to separate duplications from horizontal transfers in the phyletic profile approach. This is not so in terms of tree mapping.

On the data analysis level, the phyletic pattern concepts seems more flexible because it leads to easier mathematical problems and is better suitable to the biological intuition.

6. References

[ADD 03] ADDARIO-BERRY, L., HALLETT, M.T., LAGERGREN, J., "Towards identifying lateral gene transfer events", *Pacific Symposium on Biocomputing*, 2003, p. 279-290.

[ARV 03] ARVESTAD, L., BERGLUND, A.-C., LAGERGREN, J.,SENNOBLAD, B., "Bayesian gene/species tree reconciliation and orthology analysis using MCMC", *Bioinformatics*, vol. 19, 2003 , p. i7-i15.

[EUL 98] EULENSTEIN, O., MIRKIN, B., VINGRON, M., "Duplication-based measures of difference between gene and species trees ", *Journal of Comp.Biology*, vol. 5, num. 1, 1998, p. 135-148.

[FEL 01] FELSENSTEIN, J. *PHYLIP 3.6*: *Phylogeny inference package*, http://evolution.genetics. washington.edu/phylip/.

[FIT 71] FITCH, W.M., "Towards defining the course of evolution: Minimum change for a specific tree topology",. *Systematic. Zoology*, v. 20, 1971, p. 406-416.

[GOO 79] GOODMAN, M., CZELUSNIAK, J., MOORE, G.W., ROMERO-HERRERA, A.E., MATSUDA, G., "Fitting the gene lineage into its species lineage. A parsimony strategy illustrated by cladograms constructed from globin sequences", *Systematic Zoology*, vol. 28, 1979, p. 132-163.

[GUI 96] GUIGÓ, R., MUCHNIK, I., SMITH, T.F., "Reconstruction of ancient molecular phylogeny", *Molecular Phylogenetics and Evolution*, vol. 6. 1996, p. 189-213.

[HAR 73] HARTIGAN, J.A:, "Minimum evolution fits to a given tree ", *Biometrics*, vol. 19, 1973, p. 53-65.

[KUN 03] KUNIN V., OUZONIS, C.A., " GeneTRACE – reconstruction of gene content of ancestral species", *Bioinformatics*, vol. 19, 2003, p. 1412-1416.

[MAD 92] MADDISON, W.P., MADDISON, D.R:, *MacClade 3.0*. Sunderland, MA: Sinauer Associates, 1992.

[MIR 03] MIRKIN, B., FENNER, T., GALPERIN, M., KOONIN, E., "Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes", *BMC Evolutionary Biology* 2003, 3:2 (www.biomedcentral.com/1471-2148/3/2/).

[MIR 95] MIRKIN, B., MUCHNIK, I., SMITH, T. "A biologically consistent model for comparing molecular phylogenies", *Journal of Comp.Biology*, vol. 2, num. 4, 1995, p. 493-507.

[MIR 04] MIRKIN, B., WOLF, Y., FENNER, T., KOONIN, E., "Modelling horizontal transfer events with directed evolutionary scenarios and the principle of maximum correlation", 2004, *in progress*.

[NEI 00] NEI, M., KUMAR, S, *Molecular Evolution and Phylogenetics*, Oxford Univ. Press, 2000.

[PAG 94] PAGE, R.D.M., "Maps between trees and cladistic analysis of historical associations among genes, organisms, and areas", *Systematic Biology*, vol. 43, 1994, p. 58-77.

[PAG 97] PAGE, R.D.M., CHARLESTON, M.A., "From gene to organismal phylogeny: Reconciled trees and the gene tree/species tree problem", *Molecular Phylogenetics and Evol..*, vol. 7, 1997, p. 231-240.

[SNE 02] SNEL, B., BORK, P., HUINEN, M.A., " Genomes in flux: The evolution of archaeal and proteobacterial gene content ", *Genome Research*, vol. 2, 2002, p.17-25.

[YAN 96] YANG, Z., "Phylogenetic analysis using parsimony and likelihood methods", *Journal of Molecular Evolution*, vol. 42, 1996, p. 294-307.

[YAN 95] YANG, Z., KUMAR, Z., NEI, M., "A new method of inference of ancestral nucleotide and amino acid sequences", *Genetics*, vol. 141, 1995, p. 1641-1650.

[ZHA 97] ZHANG, L., "On a Mirkin-Muchnik-Smith conjecture for comparing molecular phylogenies", *Journal of Comp. Biology*, vol. 4, num. 1, 1997, p. 177-187.

Un noyau d'alignement local pour la classification de séquences biologiques

Jean-Philippe Vert* — **Hiroto Saigo**** — **Tatsuya Akutsu****

** Ecole des Mines de Paris*
Centre de Géostatistique
35 rue Saint-Honoré
77300 Fontainebleau
Jean-Philippe.Vert@ensmp.fr
*** Kyoto University*
Institute for Chemical Research
Bioinformatics Center
Uji 611-0011, Japan
{hiroto,takutsu}@kuicr.kyoto-u.ac.jp

RÉSUMÉ. *La recherche d'alignements entre séquences biologiques est un outil couramment utilisé pour la recherche d'homologie et donc de similarité entre séquences. Nous montrons comment ce concept peut être adapté à des problèmes de classification supervisée de séquences biologiques à l'aide de machines à vecteurs de support, ou de toute autre méthode utilisant des noyaux positifs. Pour ce faire nous montrons comment un noyau défini positif peut être construit à partir du score d'alignement local utilisé pour la recherche d'homologie, à l'aide d'opérations de convolution entre noyaux. Des expériences de classification de séquences en super-familles structurelles valident cette approche.*

MOTS-CLÉS : *Séquences biologiques, alignement local, noyau défini positif, machine à vecteurs de support*

1. Introduction

Alors que les quantités de séquences biologiques générées par les différents programmes de séquençages continuent de croître à grande vitesse, les besoins en algorithmes performants pour analyser et classer ces séquences se font de plus en plus pressants. En particulier, la classification automatique de gènes en classes structurelles ou fonctionnelles est un pré-requis pour la compréhension des fonctions et interactions au sein de systèmes vivants. De nombreuses méthodes ont été proposées pour la recherche d'homologie entre séquences biologiques, permettant la classification de séquences dans des classes fonctionnelles ou structurelles. La recherche d'homologie est depuis plus de 20 ans basée sur la recherche d'alignements entre séquences et sur le calcul d'un score d'alignement, via par exemple l'algorithme de Smith-Waterman [SMI 81] ou ses variantes plus rapides BLAST [ALT 90] et FASTA [PEA 90]. A partir des années 1990, de meilleures performances ont été obtenues par des méthodes construisant des modèles pour différentes classes de séquences, et comparant une séquence à classer à ces différents modèles. Ces modèles incluent par exemple la méthode des profiles [GRI 90], les chaînes de Markov cachées [KRO 94, BAL 94], PSI-BLAST [ALT 97] ou SAM-T98 [KAR 98]. Ces méthodes sont dites *génératives*, au sens ou elles créent des modèles pour différentes classes de séquences, et vérifient à quel degré ces modèles expliquent une séquence à classer.

De nouveaux gains en performance ont été réalisés depuis 5 ans, avec l'utilisation de méthodes *discriminantes* pour la classification supervisée. Par opposition aux méthodes génératives, ces méthodes apprennent des règles de classification qui prennent en compte les différences entre classes. Une attention particulière a été portée sur l'uti-

lisation de machines à vecteurs de support (SVM) pour la classification de séquences en familles d'homologues. Les SVM sont des algorithmes d'apprentissage statistique [VAP 98, CRI 00, SCH 02] pour la classification supervisée en différentes classes. Un élément important des SVM est l'utilisation d'une fonction, appelée noyau, pour mesurer la similarité entre n'importe quelle paire d'éléments à classer, des séquences dans notre cas. En utilisant différents noyaux, on peut obtenir une grande variété de SVM avec des performances différentes sur un problème de classification donné. Dans le cas de séquences biologiques, plusieurs noyaux ont été développés au cours des dernières années. La première utilisation des SVM pour la classification de séquences biologiques a été proposée par [JAA 00] à l'aide du noyau de Fisher déduit d'un modèle statistique de séquences. D'autres tentatives incluent la méthode "SVM-pairwise" [LIA 02], ou les noyaux de spectre [LES 02] et de mismatch [LES 03].

Une fonction noyau peut souvent être considérée comme une mesure de similarité entre objets à classer. En particulier, une SVM apprend une fonction telle que des objets "similaires" (au sens de la fonction noyau) tendent à appartenir à des classes similaires. Cette observation suggère que des noyaux intéressants peuvent être construits à partir de mesures de similarité pertinentes. Dans le cas des séquences biologiques, les mesures de similarité par alignements de séquences sont couramment utilisées pour la mesure directe de similarité entre séquences, car elles quantifient de manière naturelle des phénomènes biologiques responsables de l'évolution de séquences (notamment les mutations, insertions, et délétions). Le but de cette contribution est donc d'analyser sous quelles conditions ces mesures de similarité peuvent être utilisées comme fonction noyau par des SVM dans un contexte de classification.

Nous montrons dans un premier temps que les scores d'alignement, même s'il constituent des mesures de similarité intéressantes, ne peuvent pas être utilisés directement par des SVM, car ils ne sont pas définis positifs. Dans un deuxième temps, nous montrons comment des opérations de convolution permettent de construire un noyau défini positif utilisant les mêmes informations que les scores d'alignement. Les preuves des résultats énoncés dans cette contribution et des extensions de ce travail se trouvent dans les références [SAI 03] et [VER 04].

2. Machines à vecteurs de support et noyaux positifs

Les SVM pour la classification supervisée sont des algorithmes introduits par Vapnik et ses collègues dans les années 1990 [BOS 92]. Dans le cas de la classification supervisée binaire, une SVM apprend un fonction de classification à partir d'un ensemble d'exemples positifs $\mathcal{X}_+$ et négatifs $\mathcal{X}_-$ de la forme :

$$f(x) = \sum_{i:x_i \in \mathcal{X}_+} \lambda_i K(x, x_i) - \sum_{i:x_i \in \mathcal{X}_-} \lambda_i K(x, x_i), \tag{1}$$

où les poids positifs λ_i associés aux exemples d'entraînement sont calculés par maximisation d'une fonctionnelle quadratique. La fonction $K(.,.)$ est appelée un noyau. Une nouvelle séquence x est classée dans la classe positive (resp. négative) si la fonction $f(x)$ est positive (resp. négative). Une présentation plus détaillée de l'algorithme SVM peut être trouvée dans différents ouvrages [VAP 98, CRI 00, SCH 02].

Toute fonction $K(.,.)$ peut être utilisée comme noyau dans (1) à condition d'être symétrique et définie positive, ce qui signifie que pour tout nombre n et tout choix de n séquences $\{x_1, \ldots, x_n\}$, la matrice de taille $n \times n$ définie par $K_{i,j} = K(x_i, x_j)$ doit être symétrique et semi-définie positive. De tels fonctions seront appelées des noyaux de séquences dans la suite. Dans la section suivante, nous montrons comment définir un tel noyau à l'aide du concept d'alignement local, couramment utilisé pour la comparaison de séquences biologiques.

3. Alignement local de séquences

Commençons par quelques notations. L'alphabet dans lequel les séquences sont écrites est un ensemble fini $\mathcal{A}$ (de 20 lettres dans le cas de séquences protéiques, ou de 4 lettres dans le cas de séquences nucléiques). Une séquence est une suite finie de lettres, et nous notons $\mathcal{X} = \{\epsilon\} \cup \bigcup_{i=1}^{\infty} \mathcal{A}^i$ l'ensemble des séquences finies sur $\mathcal{A}$, ϵ représentant la séquence vide. La longueur d'une séquence $x \in \mathcal{X}$ est notée $|x|$, et la concaténation de séquences x et y est notée xy.

La notion d'alignement entre séquences, couramment utilisée pour comparer des séquences biologiques, est définie formellement de la manière suivante.

Définition 1 *Un alignement avec gaps π de $p \geq 0$ positions entre deux séquences x et y de $\mathcal{X}$ est une paire de p-uples $\pi = ((\pi_1(1), \ldots, \pi_1(p)), (\pi_2(1), \ldots, \pi_2(p))) \in \mathbb{N}^{2p}$ qui satisfait :*

$$1 \leq \pi_1(1) < \pi_1(2) < \ldots < \pi_1(p) \leq |x|,$$
$$1 \leq \pi_2(1) < \pi_2(2) < \ldots < \pi_2(p) \leq |y|.$$

Une manière courante de représenter un alignement entre deux séquences est de les écrire l'une au-dessus de l'autre, en alignant les lettres définie par l'alignement et en rajoutant des signes '-' pour représenter les gaps. Par exemple, si $x =$ `GAATCCG` et $y =$ `GATTGC`, alors l'alignement de 4 lettres $\pi = ((1, 2, 4, 6), (1, 3, 4, 5))$ est représenté par :

```
G-AATCCG-
GAT-T-G-C
```

Soit $\Pi(x, y)$ l'ensemble des alignements entre deux séquences x et y, et soit $|\pi|$ le nombre de lettres alignées dans l'alignement $\pi \in \Pi(x, y)$. Afin de trouver un "bon" alignement entre deux séquences, différentes fonctions de score $s : \Pi(x, y) \to \mathbb{R}$ ont été développées, parmi lesquelles le score d'alignement local défini formellement comme suit :

Définition 2 *Etant donné une matrice de substitution $S \in \mathbb{R}^{\mathcal{A} \times \mathcal{A}}$ et une fonction de pénalité de gaps $g : \mathbb{N} \to \mathbb{R}$ telle que $g(0) = 0$, on défi nit le score d'alignement local d'un alignement $\pi \in \Pi(x, y)$ par :*

$$s_{S,g}(\pi) := \sum_{i=1}^{|\pi|} S\left(x_{\pi_1(i)}, y_{\pi_2(i)}\right) - \sum_{i=1}^{|\pi|-1} \left[g(\pi_1(i+1) - \pi_1(i) - 1) + g(\pi_2(i+1) - \pi_2(i) - 1)\right]. \quad (2)$$

En d'autres termes, le score d'alignement local de π est la somme des scores de substitutions entre lettres alignées, moins la somme des pénalités de gaps quand des gaps sont présents. De ce score, on déduit le score d'alignement entre deux séquences :

Définition 3 *Le score d'alignement local, ou score de Smith-Waterman (noté score SW) entre deux séquences $(x, y) \in \mathcal{X}^2$ est le score d'alignement local de leur meilleur alignement, i.e.,*

$$SW_{S,g}(x, y) := \max_{\pi \in \Pi(x,y)} s_{S,g}(\pi). \quad (3)$$

Le score de SW est couramment utilisé pour mesurer la similarité entre séquences, et peut être calculé avec une complexité $O(|x||y|)$ par programmation dynamique [SMI 81].

Ce score étant une mesure de similarité "naturelle" entre séquences biologiques, il est naturel de se demander si il peut être utilisé comme noyau par des SVM. Etant clairement symétrique, il suffit de vérifier s'il est défini positif ou non. Des résultats expérimentaux montrent que la réponses est négative en général, en particulier pour des matrices de similarité et des pénalités de gaps utilisées en pratique : il est possible de trouver des ensembles de séquences telles que la matrice de similarité résultante ait des valeurs propres négatives. Comme le montre la proposition suivante, ce résultat négatif dépend cependant des paramètres choisi :

Proposition 1 *Soit $g = 0$ (pas de pénalité de gap) et S la matrice de substitution nulle sauf pour une lettre $a \in \mathcal{A}$ sur la diagonale, i.e., $S(a, a) = 1$ et $S(u, v) = 0$ sauf si $u = v = a$. Alors le score $SW_{S,g}$ est un noyau pour séquence défi ni positif.*

4. Noyau d'alignement local

Afin d'utiliser la notion d'alignement local avec des SVM, nous allons maintenant définir des noyaux définis positifs à partir de scores d'alignement. Notre travail repose sur une opération définie par [HAU 99] laissant invariant l'espace des noyaux définis positifs sur un ensemble : la convolution. Dans le cas de noyaux pour séquences, la convolution est l'opération qui à deux noyaux K_1 et K_2 associe le noyau pour séquence $K_1 \star K_2$ défini par :

$$\forall (x,y) \in \mathcal{X}^2, \quad K_1 \star K_2(x,y) = \sum_{x_1 x_2 = x, y_1 y_2 = y} K_1(x_1,y_1) K_2(x_2,y_2).$$

Si K_1 et K_2 sont des noyaux de séquences définis positifs, alors leur convolution $K_1 \star K_2$ est également un noyau défini positif [HAU 99]. Pour tout noyau de séquences K, on note $K^{(n)}$ le noyau obtenu par n convolutions de K avec lui-même.

Les noyaux de convolution ainsi définis sont utiles pour comparer des séquences de différentes longueurs, mais qui ont des parties communes. Par exemple, [WAT 00] et [HAU 99] montrent que la probabilité d'émettre deux séquences par une "pair-HMM" est un noyau de convolution, et peut donc être utilisé comme noyau par les SVM. Nous allons à présent étendre cette idée pour définir, par convolution, un noyau qui imitent des mesures de similarité par recherche d'alignement local.

Pour cela, nous allons définir formellement trois noyaux de séquence de base. Le premier est un noyau trivial, toujours égal à 1 :

$$\forall (x,y) \in \mathcal{X}^2, \quad K_0(x,y) = 1.$$

Deuxièmement, afin de quantifier l'alignement entre deux lettres, nous définissons le noyau :

$$K_a^{(\beta)}(x,y) = \begin{cases} 0 & \text{if } |x| \neq 1 \text{ or } |y| \neq 1, \\ \exp\left(\beta S(x,y)\right) & \text{otherwise,} \end{cases} \tag{4}$$

où $\beta \geq 0$ est un paramètre and $S : \mathcal{A}^2 \to \mathbb{R}$ est une matrice de similarité symétrique telle que la matrice $\left(\exp\left(\beta S(a,b)\right)\right)_{a,b \in \mathcal{A}}$ soit semi-définie positive (ce qui est par exemple le cas pour tout β si S est conditionnellement définie positive [BER 84]).

Troisièmement, nous définissons le noyau suivant pour quantifier la pénalité des gaps :

$$K_g^{(\beta)}(x,y) = \exp\left[\beta\left(g(|x|) + g(|y|)\right)\right],$$

où $\beta \geq 0$ est un paramètre et $g(n)$ est le coût d'un gap de longueur n donné par :

$$\begin{cases} g(0) = 0 & \text{if } n = 0, \\ g(n) = d + e(n-1) & \text{if } n \geq 1. \end{cases} \tag{5}$$

d et e sont des paramètres appelés coût d'ouverture et d'extension.

Il est facile de vérifier que ces trois noyaux sont bien des noyaux de séquences définis positifs. Il en résulte que le noyau suivant, défini pour tout $n \in \mathbb{N}$ est également défini positif :

$$K_{(n)}^{(\beta)}(x,y) = K_0 \star \left(K_a^{(\beta)} \star K_g^{(\beta)}\right)^{(n-1)} \star K_a^{(\beta)} \star K_0.$$

Ce noyau quantifie la similarité entre deux séquences x et y à travers des alignements de exactement n lettres. En effet, l'opération de convolution consiste à sommer sur toutes les décompositions de x et y en une parties initiales (dont la similarité est mesurée par K_0), puis une succession de n lettres (dont la similarité est mesurée par $K_a^{(\beta)}$) éventuellement séparées par $n-1$ gaps (dont la similarité est mesurée par $K_g^{(\beta)}$), puis des parties terminales (dont la similarité est mesurée par K_0).

Afin de prendre en compte des alignement d'un nombre quelconque de lettres, nous définissons finalement le noyau suivant, appelé *noyau d'alignement local* :

$$K_{LA}^{(\beta)} = \sum_{i=0}^{\infty} K_{(i)}^{(\beta)}. \tag{6}$$

Ce noyau est défini comme une limite ponctuelle de noyaux définis positifs, et est donc lui-même bien défini positif [BER 84]. L'intérêt de ce noyau réside dans le théorème suivant, qui le relie au score d'alignement local :

Théorème 1 *Le noyau d'alignement local s'écrit en fonction du score d'alignement local de la manière suivante :*

$$K_{LA}^{(\beta)}(x,y) = \sum_{\pi \in \Pi(x,y)} \exp\left(\beta s_{S,g}(x,y,\pi)\right). \tag{7}$$

En particulier, le score de SW peut être vu comme une limite quand β tend vers l'infi ni :

$$\lim_{\beta \to +\infty} \frac{1}{\beta} \ln K_{LA}^{(\beta)}(x,y) = SW_{S,g}(x,y). \tag{8}$$

Ces équations clarifient le lien entre le noyau d'alignement local et le score de SW, et mettent en évidence pourquoi ce score n'est pas défini positif. Premièrement, le score de SW ne conserve que la contribution du meilleur alignement, alors que le noyau fait une somme sur tous les alignements. Deuxièmement, le score de SW est le logarithme (à la limite) d'un noyau défini positif, et le passage au logarithme est une opération qui ne conserve pas la propriété d'être défini positif en général [BER 84].

5. Implémentation

Une implémentation naïve du noyau d'alignement local à partir de la formule (7) nécessiterait une somme sur $|\Pi(x,y)|$ alignements, et résulterait en une complexité exponentielle en $|x|$ et $|y|$. Cependant, tout comme le score de SW, le calcul peut être factorisé par programmation dynamique pour aboutir à une implémentation en $O(|x||y|)$ (voir détails dans [VER 04]).

Dans la pratique, cependant, ce noyau souffre comme d'autres noyaux pour séquences du problème de la dominance de la diagonale, c'est-à-dire du fait que $K_{LA}^{(\beta)}(x,x)$ peut couramment être des ordres de magnitude plus grand que $K_{LA}^{(\beta)}(x,y)$ pour deux séquences x et y. Cela est particulièrement vrai pour les grandes valeurs du paramètre β, car :

$$\frac{K_{LA}^{(\beta)}(x,x)}{K_{LA}^{(\beta)}(x,y)} \sim \exp\beta\left(SW_{S,g}(x,x) - SW_{S,g}(x,y)\right)$$

quand $\beta \to \infty$. Dans la pratique, il est connu que les SVM ne fournissent pas de bon résultats dans ce cas, car l'apprentissage consiste essentiellement à mémoriser les données observées et la généralisation revient essentiellement à rechercher le plus proche voisin.

Afin d'utiliser le noyau d'alignement local en pratique, nous proposons de prendre son logarithme via la formule suivante :

$$\tilde{K}_{LA}^{(\beta)}(x,y) = \frac{1}{\beta} \ln K_{LA}^{(\beta)}(x,y). \tag{9}$$

Cette opération pose problème, car $\tilde{K}_{LA}^{(\beta)}$ risque de ne pas être défini positif. Dans la pratique, la matrice de similarité entre exemple d'apprentissage utilisée par les SVM risque de posséder des valeurs propres négatives. Pour remédier à ce problème, nous proposons de retrancher à la diagonale de cette matrice la plus petite valeur propre (si elle est négative), afin que la matrice devienne semi-définie positive. Cette astuce n'est bien sûr utile que dans la phase d'apprentissage.

6. Expériences et conclusion

Nous avons testé le noyau d'alignement local dans un problème de classification de séquences de domaines protéiques en super-familles de la base de données SCOP [MUR 95] version 1.53. Nous avons suivi l'expérience décrite dans [LIA 02]. Les données[1] consistent en 4352 séquences groupées en familles et super-familles. Pour chaque famille, les séquences de cette familles sont des exemples de test positifs, et les séquences de la même super-famille mais de familles différentes sont les exemples positifs d'entraînement. Les exemples négatifs sont pris en dehors de la super-famille, et sont séparés aléatoirement en exemples d'entraînement et de test. En ne considérant que les familles avec au moins 10 exemples positifs en entraînement et 5 en test, on aboutit à 54 familles. Pour chaque famille, la surface sous la courbe des vrai positifs contre les faux positifs (courbe ROC), normalisée entre 0 et 1, est calculée (indice ROC). De même, la surface sous cette courbe jusqu'à 50 faux positifs est calculée (ROC50), ainsi que le nombre de faux positifs ayant un score supérieur au score médian des vrais positifs (RFP).

Le noyau d'alignement local est comparé avec 3 autres noyaux représentant l'état de l'art en classification supervisée de séquences protéiques : le noyau de Fisher [JAA 00], le noyau "pairwise" [LIA 02], et le noyau mismatch [LES 03].

La table 1 résume les résultats obtenus pour différentes valeurs de β, ainsi que les scores obtenus par les autres méthodes testées. Ces résultats montrent que les meilleurs résultats sont obtenus quand β est de l'ordre de

Kernel	Mean ROC	Mean ROC50	Mean mRFP
LA ($\beta = +\infty$)	0.908	0.591	0.0654
LA ($\beta = 1$)	0.912	0.612	0.0626
LA ($\beta = 0.8$)	0.908	0.597	0.0679
LA ($\beta = 0.5$)	**0.925**	**0.649**	0.0541
LA ($\beta = 0.2$)	0.923	**0.661**	0.0637
LA ($\beta = 0.1$)	0.868	0.429	0.111
Pairwise	0.896	0.464	0.0837
Mismatch	0.872	0.400	0.0837
Fisher	0.773	0.250	0.204

TAB. 1. *ROC, ROC50 et RFP moyens obtenus sur 54 familles pour différentes noyaux. LA-eig représente le noyau d'alignement local. $\beta = +\infty$ correspond au score de SW.*

$0.2 - 0.5$, et qu'ils sont meilleurs que l'état de l'art représenté par les autres noyaux. Les distributions des scores ROC, ROC50 et RFP sur les 54 familles pour différentes noyaux sont montrés sur les figures 1, 2 et 3. Ces résultats illustrent d'une part l'intérêt d'utiliser une mesure de similarité naturelle pour obetnir de bonnes performance en classification, et d'autre part le gain obtenu en prenant en compte l'ensemble des alignements entre deux séquences plutôt que le meilleur alignement uniquement.

7. Bibliographie

[ALT 90] ALTSCHUL S., GISH W., MILLER W., MYERS E., LIPMAN D., A basic local alignment search tool, *Journal of Molecular Biology*, vol. 215, 1990, p. 403–410.

[ALT 97] ALTSCHUL S., MADDEN T., SCHAFFER A., ZHANG J., ZHANG Z., MILLER W., LIPMAN D., Gapped BLAST and PSI-BLAST : A new generation of protein database search programs, *Nucleic Acids Research*, vol. 25, 1997, p. 3389–3402.

[BAL 94] BALDI P., CHAUVIN Y., HUNKAPILLER T., MCCLURE M., Hidden Markov models of biological primary sequence information, *Proc. Natl. Acad. Sci. USA*, vol. 91(3), 1994, p. 1053–1063.

1. Accessibles à `www.cs.columbia.edu/compbio/svm-pairwise`

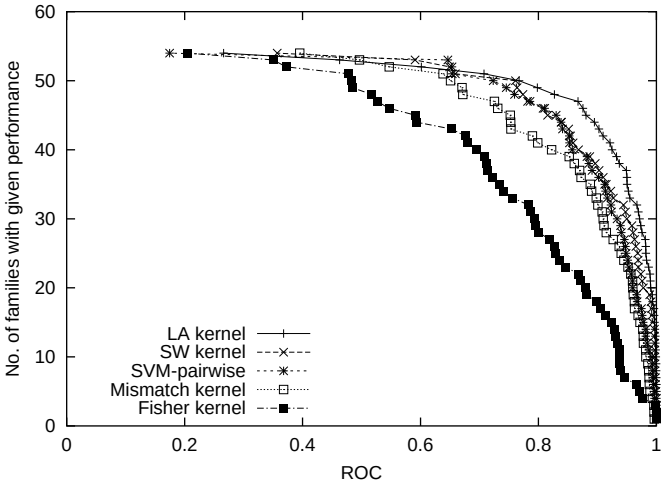


FIG. 1. *Distribution du score ROC pour différents noyaux. La courbe noté "LA kernel" correspond au noyau d'alignement local avec $\beta = 0.5$.La courbe "SW kernel" correspond au score de SW.*

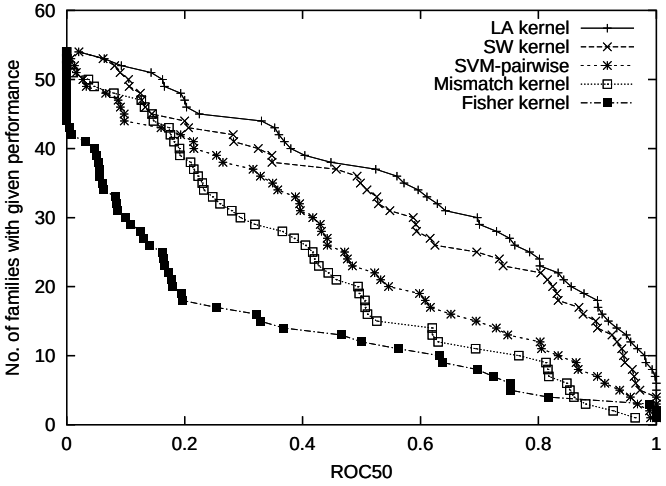


FIG. 2. *Distribution du score ROC50 pour différents noyaux.*

[BER 84] BERG C., CHRISTENSEN J., RESSEL P., *Harmonic analysis on semigroups*, Springer-Verlag, New-York, 1984.

[BOS 92] BOSER B. E., GUYON I. M., VAPNIK V. N., A training algorithm for optimal margin classifiers, *Proceedings of the 5th annual ACM workshop on Computational Learning Theory*, ACM Press, 1992, p. 144–152.

[CRI 00] CRISTIANINI N., SHAWE-TAYLOR J., *An introduction to Support Vector Machines and other kernel-based learning methods*, Cambridge University Press, 2000.

[GRI 90] GRIBSKOV M., LÜTHY R., , EISENBERG D., Profile Analysis, *Methods in Enzymology*, vol. 183, 1990, p. 146–159.

[HAU 99] HAUSSLER D., Convolution Kernels on Discrete Structures, rapport, 1999, UC Santa Cruz.

[JAA 00] JAAKKOLA T., DIEKHANS M., HAUSSLER D., A Discriminative Framework for Detecting Remote Protein Homologies, *Journal of Computational Biology*, vol. 7, n^o 1,2, 2000, p. 95–114.

[KAR 98] KARPLUS K., BARRETT C., HUGHEY R., Hidden Markov Models for Detecting Remote Protein Homologies, *Bioinformatics*, vol. 14, n^o 10, 1998, p. 846–856.

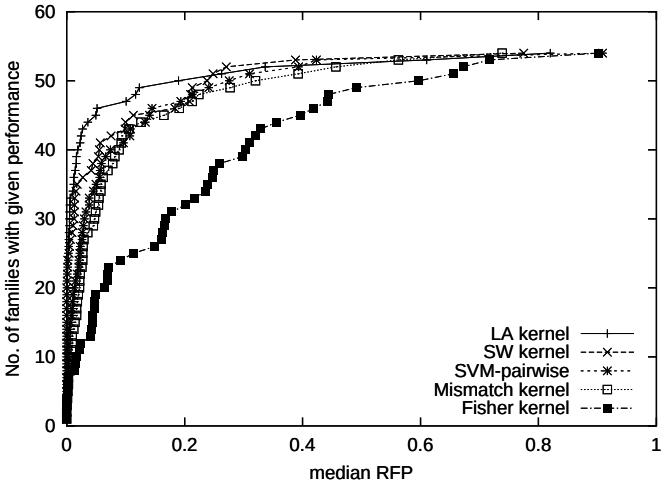


FIG. 3. *Distribution du score RFP pour différents noyaux.*

[KRO 94] KROGH A., BROWN M., MIAN I., SJOLANDER K., HAUSSLER D., Hidden Markov models in computational biology : Applications to protein modeling, *Journal of Molecular Biology*, vol. 235, 1994, p. 1501–1531.

[LES 02] LESLIE C., ESKIN E., NOBLE W. S., The spectrum kernel : a string kernel for SVM protein classification, ALTMAN R. B., DUNKER A. K., HUNTER L., LAUERDALE K., KLEIN T. E., Eds., *Proceedings of the Pacific Symposium on Biocomputing 2002*, World Scientific, 2002, p. 564–575.

[LES 03] LESLIE C., ESKIN E., WESTON J., NOBLE W. S., Mismatch String Kernels for SVM Protein Classification, BECKER S., THRUN S., OBERMAYER K., Eds., *Advances in Neural Information Processing Systems 15*, MIT Press, 2003.

[LIA 02] LIAO L., NOBLE W. S., Combining pairwise sequence similarity and support vector machines for remote protein homology detection, *Proceedings of the Sixth International Conference on Computational Molecular Biology*, ACM Press, 2002, p. 225–232.

[MUR 95] MURZIN A., BRENNER S., HUBBARD T., CHOTHIA C., SCOP : A structural classification of proteins database for the investigation of sequences and structures, *Journal of Molecular Biology*, vol. 247, 1995, p. 536–540.

[PEA 90] PEARSON W., Rapid and sensitive sequence comparisons with FASTP and FASTA, *Methods in Enzymology*, vol. 183, 1990, p. 63–98.

[SAI 03] SAIGO H., VERT J.-P., UEDA N., AKUTSU T., Protein homology detection using string alignment kernels, *Bioinformatics*, , 2003, To appear.

[SCH 02] SCHÖLKOPF B., SMOLA A. J., *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, 2002.

[SMI 81] SMITH T., WATERMAN M., Identification of common molecular subsequences., *Journal of Molecular Biology*, vol. 147, 1981, p. 195-197.

[VAP 98] VAPNIK V. N., *Statistical Learning Theory*, Wiley, New-York, 1998.

[VER 04] VERT J.-P., SAIGO H., AKUTSU T., Convolution and local alignment kernels, SCHÖLKOPF B., TSUDA K., VERT J.-P., Eds., *Kernel Methods in Computational Biology*, The MIT Press, 2004, (to appear).

[WAT 00] WATKINS C., Dynamic alignment kernels, SMOLA A., BARTLETT P., SCHÖLKOPF B., SCHUURMANS D., Eds., *Advances in Large Margin Classifiers*, p. 39–50, MIT Press, Cambridge, MA, 2000.

Normalisation et analyse des classes pour la classification sous hypothèse de connexité

Catherine Aaron

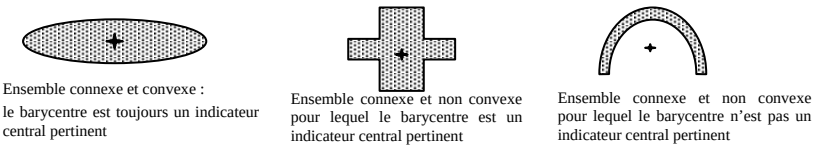
SAMOS-MATISSE, Université Paris1
90 rue de Tolbiac,
75013 Paris

RÉSUMÉ. Le but de cet article est de construire des méthodes d'analyse des classes sous hypothèses de connexité. On proposera plus spécifiquement une méthode de normalisation liée à la connexité par le biais d'un travail sur le plus petit arbre connexe qui nous mènera à une détermination « automatique » du nombre de voisins à considérer dans des méthodes type k plus proches voisins et à la construction d'un indicateur central comme point minimisant une inertie construite sur la distance curviligne.

MOTS-CLÉS : dimension, normalisation, connexité, classification

1. Introduction

Nous partons de l'hypothèse suivante : $X = \left\{ \vec{x}_1,...,\vec{x}_N \right\} \subset R^p$ est un ensemble connexe par arc[1]. Nous nous intéressons alors à la construction de méthodes permettant de décrire et d'analyser X. Dans un premier temps soulignons que, sous la seule hypothèse de connexité, les outils de statistique « classique » ne sont pas adaptés et requièrent souvent une hypothèse, plus restrictive, de convexité. En effet, l'espérance (estimée comme un barycentre) est un indicateur central qui peut être fort éloignée de l'ensemble des points du nuage comme le montrent les exemples suivants :



Ensemble connexe et convexe : le barycentre est toujours un indicateur central pertinent

Ensemble connexe et non convexe pour lequel le barycentre est un indicateur central pertinent

Ensemble connexe et non convexe pour lequel le barycentre n'est pas un indicateur central pertinent

Figure 1 : barycentre et hypothèse de connexité

De même, la variance, qui caractérise la dispersion autour du barycentre, pourra se révéler inappropriée à la caractérisation de la dispersion du nuage (notamment dans le cas où le barycentre n'est pas un indicateur central pertinent).

Dans le cadre de la connexité par arc comme caractéristique d'un nuage, il vient assez naturellement l'idée de remplacer la notion de distance euclidienne par celle de distance curviligne.

Des méthodes d'analyse des données reposant sur cette dernière distance existent, telle que la méthode ISOMAP [Joshua 2000] mais, à notre connaissance, il n'y a pas de méthode permettant de normaliser les données dans un tel cadre. De plus les algorithmes existants de calcul de la distance curviligne utilisent des graphes des $k-$ plus proches voisins sans qu'il existe une manière automatique de choisir k.

[1] Pour la description d'une méthode d'obtention de classes connexes se référer à [AARON 04]

Ainsi nous nous proposons, ici, de construire une méthode de normalisation des données qui mènera à un choix de k , puis un indicateur central reposant uniquement sur la connexité.

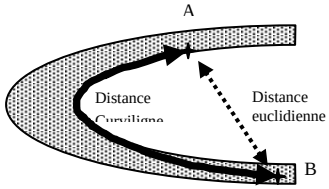


Figure 2 : distance curviligne

2. Algorithme de normalisation

2.1. *Problématique*

Les algorithmes de construction des distances curvilignes reposent principalement sur la construction de graphes sur le nuage de points, le plus souvent bâtis par la méthode des $k-$ plus proches voisins. De tels algorithmes sont très sensibles à la normalisation préliminaire effectuée sur les données (voir figure 3 où une dilatation de l'axe vertical change complètement la structure des voisinages).

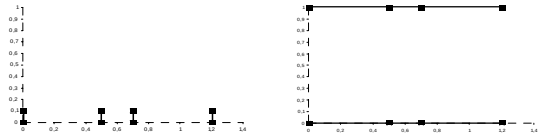


Figure 3 : plus proches voisins pour deux ensembles identiques a une homothétie près

Nous proposons alors, ci-dessous, une méthode de normalisation visant à donner le même poids à chacune des variables.

2.2. *Principe*

Sous hypothèse de connexité de l'ensemble de points, il semble pertinent de considérer l'arbre connexe issu de la classification hiérarchique par la distance minimum. Cet arbre est considéré comme le plus « solide » dans le sens où la plus grande distance entre deux points connectés est minimale. On normalisera alors les données en cherchant à rendre le déplacement sur cet arbre de même valeur moyenne, parallèlement à chaque direction.

Pour cela on définit le poids d'un axe comme la moyenne projection des liaisons sur cet axe :

$$poids(\vec{u}) = \frac{1}{N-1}\sum_{j>i} 1_{\delta(i,j)=1} \cdot \left| \overrightarrow{x_i x_j} . \vec{u} \right| \text{ avec } \begin{cases} \delta(i,j) = 1 \text{ si } i \text{ est lié à } j \\ \delta(i,j) = 0 \text{ sinon} \end{cases}$$

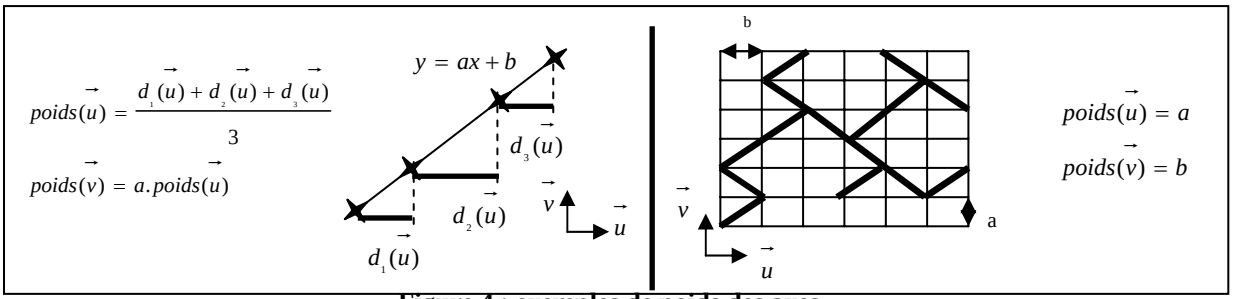


Figure 4 : exemples de poids des axes

On cherche alors à donner le même poids à chacun des axes canoniques. Pour cela, on est obligé d'adopter un algorithme itératif car, après redimensionnement d'un axe, les voisinages peuvent changer (cf figure 3) et, par conséquent, l'arbre connexe aussi.

L'algorithme est alors le suivant :

tant que le vecteur des poids est différent de 1

- calcul de l'arbre connexe
  - calcul des poids

- pondération de chaque axe par 1/poids(axe)

On propose alors de choisir le nombre de plus proches voisins pour la construction d'un graphe des $k-$ plus proches voisins, après la normalisation, comme le $\max_i(num(i,j))$ avec $num(i,j)=0$ si i et j ne sont pas connectés et j est le $num(i,j)^{eme}$ voisin de i.

2.3. *Résultats*

Le graphique suivant présente les résultas obtenus sur des données simulées. Pour chaque exemple la partie supérieure présente les résultats pour le plus petit arbre connexe et le graphe des plus $k-$ plus proches voisins associés après la normalisation proposée. La partie inférieure présente les mêmes graphes pour une normalisation « classique » division par l'écart type. Dans un souci de lisibilité des figures les graphes (connections entre les points qui dépendent de la normalisation) sont présentés sur le nuage de points initial.

Les figures a,b,c,d et e ont été obtenues en tirant X suivant une loi uniforme sur $[-1,1]$ et $Y=\sin(\omega X)$ pour plusieurs valeurs de ω (10,20,30 et 40). La figure f correspond à un tirage uniforme sur $[-1,1]$ sur les deux axes.

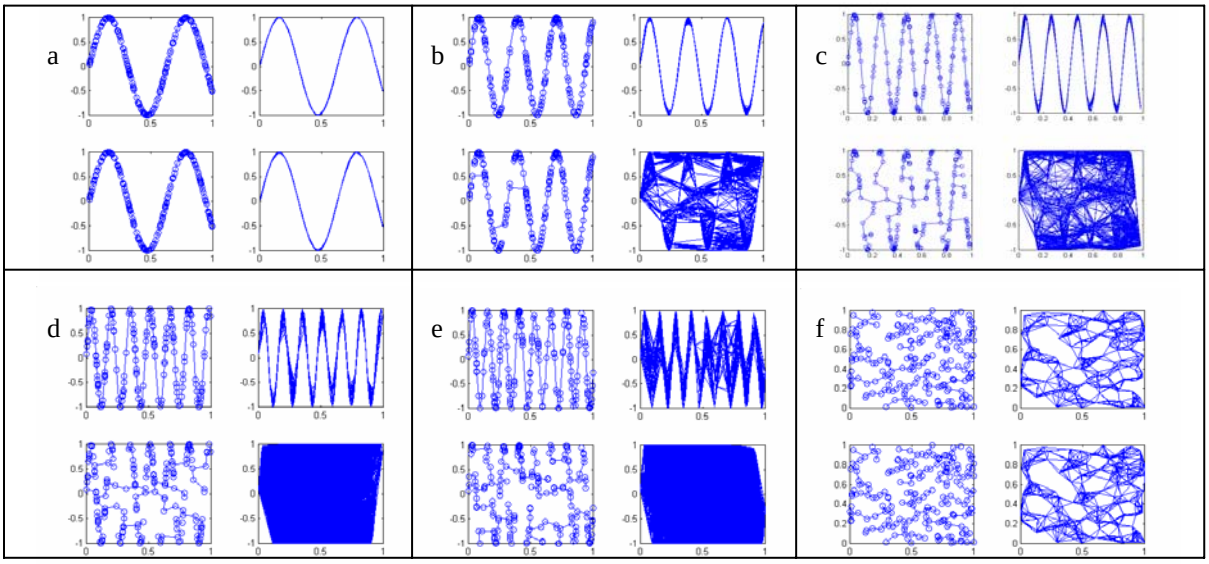


Figure 4 : plus petit graphe connexe (gauche) pour normalisation proposée (haut) ou classique (bas) x suit une uniforme [0,1] et y=sin(wx), a)w=10 ,b)w=20 ,c)w=30 ,d)w=40, e)w=50 pour des tirages de 200 points

On voit ainsi que la méthode de normalisation proposée permet de retrouver à la fois un nombre de voisins à choisir et des voisinages pour chaque point permettant de retrouver des structures de données qu'une normalisation classique, reposant sur la distance euclidienne ne retrouve pas.

3. Construction d'un indicateur central

A partir du graphe des $k-$ plus proches voisins, on peut construire une matrice des distances curvilignes entre les individus en utilisant, par exemple, l'algorithme de Dijkstra. A partir de cette matrice, on peut alors trouver le « point » central du nuage en élargissant la caractérisation inertielle du barycentre classique à la distance curviligne :

$$G_{classique}=\arg\min\int\|G-x\|_2\,dx \text{ donne, par analogie } G_{connexe}=\arg\min_{x_i}\sum_j\|x_j-x_i\|_{curviligne}$$

Cette méthode doit être rendue robuste afin de pouvoir « sortir » de l'ensemble des points du nuage (pour l'instant le barycentre connexe est obligatoirement un point du nuage) et pour ne pas trouver d'indicateur lorsque celui-ci n'est pas valide, ceci pouvant arriver dans les cas les plus « exotiques » tels que l'étude d'un anneau. Les graphiques suivants présentent les résultats obtenus dans le cas d'une parabole, celui d'un anneau et de tirages « classiques » uniforme et gaussien. La première figure représente le nuage (croix) et le barycentre connexe (rond), la deuxième représente l'ensemble des inerties curvilignes triées par ordre croissant, et la troisième, en trois dimensions : les deux axes du nuages et l'inertie curviligne associée à chaque point (en z).

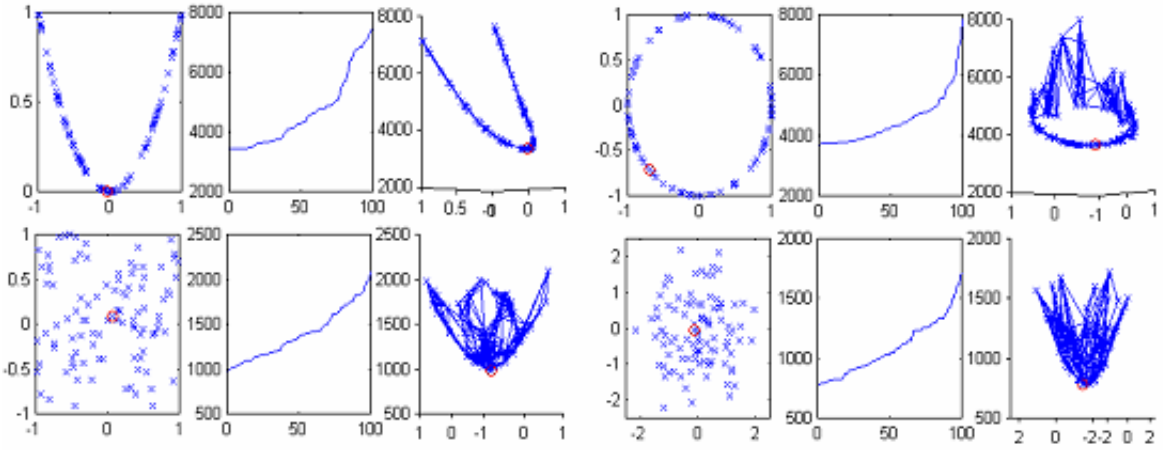


Figure 6 : indicateur central pour différentes formes de tirages

On voit ainsi que l'on retrouve des indicateurs centraux pertinents (proches du barycentres) dans le cas de nuages « classiques ». Dans le cas de la parabole notre indicateur central est meilleur que le barycentre, en revanche, dans le cas de l'anneau ni le barycentre ni notre indicateur ne sont vraiment intéressants.

4. Conclusion

Les résultats de la normalisation par le plus petit graphe connexe sont encourageants tant en capacité à retrouver les structures non linéaires sous-jacentes qu'en obtention automatique du nombre de voisins à prendre en compte. Il faut cependant noter que les calculs sont relativement longs : le calcul du plus petit arbre connexe est en $\theta(N^4)$ et il faut itérer le calcul jusqu'à obtenir la normalisation finale. Par ailleurs le nombre d'itérations ne semble pas dépendre directement de N et a été, dans les exemples étudiés, d'environ une dizaine d'itérations. A ce jour l'existence et l'unicité de la normalisation n'ont pas encore été établies. Le principal problème vient des changements incessants de l'arbre connexe lorsqu'on effectue une homothétie. Cette étude théorique de la normalisation est notre prochain objectif.

5. Bibliographie

[Aaron, 2004] C. Aaron. Clustering under connectivity hypothesis. *Student*, 2004. (a paraître)

[Coeurjolly, 2002] D. Coeurjolly. *Algorithme et géométrie discrète pour la visualisation des courbes et des surfaces*. Thèse de doctorat.

[Gordon 1996] A. Gordon hierarchical classification. *tree and other network model for representing proximity data*, World Scientific Publ, River Edge, NJ, 1996.

[Joshua 2000] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford *A Global Geometric Framework for Nonlinear Dimensionality Reduction*, Science N°299 Dec 22 2000: 2319-2323.

[Li 1992] K. Li. On principal Hessian direction for data visualisation and dimension reduction : another application of stein's lemma. *Journal of the American Statistical Association*, Vol.87, N°420, 1025-1039, 1992.

[Ripley 1981] Brian D. Ripley. Spatial Statistics, Wiley series in probability and mathematical statistics, 1981.

[Rohlf 1974] J. Rohlf. Graphs implied by the Jardine-Sibson Overlapping Clustering Methods B_k. *Journal of the American Statistical Association*, Vol 69, N°347, 705-710, 1974.

Extension de l'algorithme Apriori et des règles d'association au cas des données symboliques diagrammes

Filipe Afonso[*,**], **Edwin Diday**[**]

[*]*Lamsade et* [**]*Ceremade*
Université Paris 9 Dauphine
Place du Maréchal de Lattre de Tassigny, 75775 Paris Cedex 16, France
afonso@ceremade.dauphine.fr, diday@ceremade.dauphine.fr

RÉSUMÉ. *Ce papier concerne l'extension de l'agorithme Apriori et des règles d'association au cas des données symboliques diagrammes. La méthode proposée va nous permettre de découvrir des règles au niveau des concepts. Notamment, plutôt que d'extraire des règles entre articles venant des mêmes transactions enregistrées dans un magasin comme dans le cas classique, nous découvrons des règles à partir d'une matrice décrivant les achats des concepts clients afin d'étudier leurs comportements.*

MOTS-CLÉS : *Régles d'association, Algorithme Apriori, Données Symboliques*

1. Introduction

Dans la pratique, nous sommes souvent intéressés non pas par les individus statistiques eux-mêmes mais par des individus de plus hauts niveaux appelés concepts. Dans ce cas, une agrégation des données va nous amener à manipuler des variables qui ne sont pas à valeurs uniques mais à valeurs intervalles, histogrammes et diagrammes ([BOC 00]). C'est dans cette optique que nous étendons l'Algorithme Apriori ([AGR 94]) et les règles d'association aux variables symboliques diagrammes. L'intérêt réside dans la découverte de règles au niveau des concepts. Si nous prenons l'exemple du panier de la ménagère, au lieu d'extraire des règles au niveau des transactions, nous allons extraire des règles au niveau des clients en agrégeant tous les articles achetés par un même client grâce à un diagramme construit avec la proportion de chaque article par rapport aux achats totaux du client.

2. Algorithme Apriori étendu aux données diagrammes

Nous étendons l'algorithme Apriori ([AGR 94]) aux données diagrammes. Au lieu d'avoir une valeur unique par case dans notre matrice de données ou bien un ensemble d'items par transaction comme dans le cas classique, nous avons un diagramme dans chaque case. Nous donnons un exemple table 1 avec quatre concepts et une seule variable diagramme X mais l'algorithme que nous allons décrire se généralise en présence de plusieurs variables.

2.1. *Principe de la méthode*

Pour étendre l'algorithme Apriori, au lieu de considérer les fréquences $P_{Xi,c}$ pour chaque catégorie c de chaque variable X_i qui sont des valeurs continues, nous allons regrouper les fréquences en un nombre fini d'intervalles. Ainsi, nous regardons les supports des intervalles de fréquences $0 < P_{Xi,c} \leq 1/h, 1/h < P_{Xi,c} \leq 2/h, 2/h < P_{Xi,c} \leq 3/h, ..., (h-1)/h < P_{Xi,c} \leq 1$ où h détermine la précision de notre regroupement en intervalles. Dans un deuxième temps, nous allons faire l'union 2 à 2 des intevalles de poids connexes ayant des supports strictement positifs $0 < P_{Xi,c} \leq 2/h, 1/h < P_{Xi,c} \leq 3/h, ..., (h-2) < P_{Xi,c} \leq 1$. Nous répétons l'opération jusqu'à

Concepts=Clients	X=items	Concepts=Clients	X=items
1	1/2v, 1/4p, 1/4c	3	2/3v, 1/3p
2	1/2v, 1/3p, 1/6c	4	2/3p,1/3c

TAB. 1. *Matrice de données symboliques composée d'une variable diagramme*

obtenir un unique intervalle $0 < P_{Xi,c} \leq 1$. Ainsi, nous travaillerons avec des objets symboliques (OS) booléens et les intervalles de fréquences seront les propriétés de l'OS avec comme intensions a(w)=$[\frac{a}{h} < P_{Xi,c}(w) \leq \frac{b}{h}]$ (a=0..h-1, b=1..h, a<b). Finalement, un k-OS sera une assertion booléenne définit à partir de k propriétés. Par exemple, si a et a' sont deux catégories de deux variables diagrammes X et X' avec P_{X_a} et $P_{X'_{a'}}$ leurs fréquences respectives alors $[\frac{1}{3} < P_{X_a} \leq \frac{2}{3}] \wedge [0 < P_{X'_{a'}} \leq \frac{1}{3}]$ sera un 2-OS. Il faudra alors veiller à ne pas croiser des intervalles de même catégorie et à utiliser les plus petits intervalles de fréquences possible pour un même support.

2.2. *Support, Confiance, et "Confiance Diagramme" (CD) dans le cas de données diagrammes.*

Soient Ω un ensemble d'individus, X et Y deux OS ayant pour intensions $a_x(w) = \bigwedge_{i,u}[\frac{a_{i,u}}{h} < P_{Xi,u}(w) \leq \frac{b_{i,u}}{h}]$ et $a_y(w) = \bigwedge_{j,v}[\frac{c_{j,v}}{h} < P_{Yj,v}(w) \leq \frac{d_{j,v}}{h}]$ avec $\forall i, u, j, v\ X_{i,u} \neq Y_{j,v}$ où $P_{x_{i,u}}$ ($P_{y_{j,v}}$) est la fréquence de la catégorie u (v) de la variable diagramme X_i (Y_j), $\frac{a_{i,u}}{h}$ et $\frac{b_{i,u}}{h}$ ($\frac{c_{j,v}}{h}$ et $\frac{d_{j,v}}{h}$) les bornes des intervalles de fréquences.

Définitions : A) Support. $Sup(X \rightarrow Y) = \frac{card(ext(X \wedge Y) = \{w \in \Omega / a_x(w) = vrai, a_y(w) = vrai\}}{card(\Omega)}$

B) Confiance. $Conf(X \rightarrow Y) = \frac{card(ext(X \wedge Y) = \{w \in \Omega / a_x(w) = vrai, a_y(w) = vrai\}}{card(ext(X) = \{w \in \Omega / a_x(w) = vrai\}} = \frac{sup(X \rightarrow Y)}{sup(X)}$

C) CD. Dans le cas des diagrammes, nous définissons un nouvel indicateur de qualité (confiance diagramme ou CD) pénalisant les règles ayant les plus grands intervalles de poids et donc la plus grande imprécision en conclusion : $CD(X \rightarrow Y) = conf(X \rightarrow Y)/(1 + \frac{\sum_{j,v}(d_{j,v} - c_{j,v})}{n_v \times h})$ où n_v est le nombre de propriétés en conclusion.

2.3. *Règles d'association symboliques*

Dans le cas classique, pour tous les itemsets (ensembles d'items) fréquents X et $Y \subset X$, nous générons la règle $Y \rightarrow X - Y$ ([AGR 93]). L'algorithme classique génère uniquement les règles ayant un support et une confiance supérieurs à deux seuils minimum *minsup* et *minconf* respectivement. Dans le cas diagramme, nous allons générer les règles ayant un support et un CD supérieurs à deux seuils minimum *minsup* et *minCD* respectivement de la forme : $\bigwedge_{i,u}[\frac{a_{i,u}}{h} < P_{Xi,u} \leq \frac{b_{i,u}}{h}] \rightarrow [\frac{c_{j,v}}{h} < P_{Yj,v} \leq \frac{d_{j,v}}{h}]$ où $\forall i, u\ X_{i,u} \neq Y_{j,v}$.

2.4. *Algorithme Apriori Diagramme*

Nous allons détailler les étapes de l'algorithme "Apriori diagramme" à l'aide de l'exemple table 1. Pour cet exemple, nous donnons une précision h=3, un support miminum $minsup$ = 35% (i.e. 2 unités) et *minCD*=55% :

1. Regrouper les fréquences de chaque catégorie en intervalles (voir section 2.1). Nous codons les intervalles de fréquences 1, 2, 3,... car nous utiliserons l'ordre lexicographique par la suite. Pour la matrice, table 1, nous considérons les poids P_v, P_p et P_c des catégories v, p et c. Ces intervalles de poids sont montrés table 2 colonnes C_1 (OS 1 à 9).

2. Calculer les supports des intervalles de poids avec un passage dans la matrice des données. Nous faisons alors l'union 2 à 2 des intervalles connexes de supports positifs (voir section 2.1). Nous répétons les unions 2 à 2 de nos nouveaux intervalles jusqu'à obtenir un unique intervalle $0 < P_{Xi,c} \leq 1$. Les supports sont alors calculés sans aucun passage dans la matrice des données puisque si A et B connexes $Sup(A \cup B) = Sup(A) + Sup(B)$. les 1-OS fréquents sont ajoutés à $L_{k=1}$. Dans notre exemple, les supports des intervalles du point (1.) sont calculés

OS	C_1	Sup	OS	C_1	Sup	OS	C_2	Sup	OS	C_3	Sup
1	$0<P_v\leq\frac{1}{3}$	0	6	$\frac{2}{3}<P_p\leq 1$	0	11	2,4	3	16	2,4,7	2
2	$\frac{1}{3}<P_v\leq\frac{2}{3}$	3	7	$0<P_c\leq\frac{1}{3}$	3	12	2,7	2			
3	$\frac{2}{3}<P_v\leq 1$	0	8	$\frac{1}{3}<P_c\leq\frac{2}{3}$	0	13	2,10	3			
4	$0<P_p\leq\frac{1}{3}$	3	9	$\frac{2}{3}<P_c\leq 1$	0	14	4,7	2			
5	$\frac{1}{3}<P_p\leq\frac{2}{3}$	1	10	$0<P_p\leq\frac{2}{3}$	4	15	7,10	3			

TAB. 2. *k-Objets Symboliques fréquents*

N°	Règles	Sup %	Conf %	CD %
1	$1/3 < P_v \leq 2/3 \rightarrow 0 < P_p \leq 1/3$	75	100	75
2	$0 < P_p \leq 1/3 \rightarrow 1/3 < P_v \leq 2/3$	75	100	75
3	$0 < P_c \leq 1/3 \rightarrow 0 < P_p \leq 2/3$	75	100	60
4	$0 < P_p \leq 2/3 \rightarrow 1/3 < P_v \leq 2/3$	75	75	56
5	$0 < P_p \leq 2/3 \rightarrow 0 < P_c \leq 1/3$	75	75	56

TAB. 3. *Règles d'association symboliques*

(table 2 colonne sup). Nous remarquons que $0 < P_p \leq 1/3$ et $1/3 < P_p \leq 2/3$ ont un support supérieur à 0. $0 < P_p \leq 2/3$ devient donc candidat (OS=10) et il est fréquent car son support est la somme des supports des intervalles précédents, soit 3+1=4. Finalement, nous ajoutons à l'ensemble L_1 les OS fréquents 2, 4, 7 et 10.

3. Faire tant que l'ensemble des k-OS fréquents $L_k \neq \oslash$:

(a) Générer les k+1-OS candidats en calculant le produit cartésien entre les OS de L_k tout en respectant l'ordre (Nous générons un k+1-OS avec 2 k-OS si les k-1 premiers codes sont égaux et le k^{ieme} code du premier est inférieur au k^{ieme} code du second). Aussi, nous générons les k+1-OS entre intervalles de catégories différentes (et "non marqués" voir point (c)). Ainsi, l'ensemble des candidats C_{k+1} est généré. De plus, nous supprimons de C_{k+1} tout k+1-OS I tel qu'il existe un k-OS $J \subset I$ n'appartenant pas à L_k. Pour notre exemple, l'algorithme génère les candidats de C_2 : (2,4), (2,7), (2,10), (4,7) et (7,10) (voir table 2, OS=11 à 15). (4,10) n'est pas généré car 4 et 10 sont des intervalles de la même catégorie.

(b) Pour tout $c \in C_{k+1}$, calculer le support avec un passage dans la matrice de données. Tout k+1-OS $I \in C_{k+1}$ fréquent est ajouté à L_{k+1} . (2,4), (2,7), (2,10), (4,7) et (7,10) sont fréquents.

(c) Marquer tout k+1-OS $I \in L_{k+1}/\exists J \in L_{k+1}$ avec $J \subset I$ et $sup(I) = sup(J)$. Il s'agit de k+1-OS définis avec les mêmes catégories mais avec des intervalles de poids différents et nous conservons les plus petits intervalles pour un même support. Nous les marquons au lieu de les supprimer car ces k+1-OS ne seront pas utilisés pour la génération de k+2-OS mais ils seront utilisés pour la génération de règles. Dans notre exemple, (2,10) ne sera pas utilisé pour la génération de 3-OS car (2,10) $\supset$ (2,4) et sup(2,10) = sup(2,4).

(d) Générer les k+1-règles (règles avec k+1 propriétés) avec un CD supérieur à *minCD*, voir table 3.

Finalement, A l'itération suivante le 3-OS fréquent (2,4,7) (table 2, OS=16) sera généré et l'algorithme s'arrêtera.

3. Applications : Régles d'association classiques versus symboliques

"Apriori Diagramme" permet d'étudier des concepts. Par exemple, dans l'Apriori classique, les unités statistiques pour étudier "le panier de la ménagère" sont des transactions. Par opposition, avec notre méthode nous sommes capables d'étudier les clients plutôt que les transactions. Nous considérons la matrice classique, table 4, avec 4 clients, 3 articles (i.e. 3 catégories) (v = viandes, p = poissons, c = pâtes et céréales) et 11 transactions. Pour appliquer l'analyse symbolique sur les concepts clients nous créons ces concepts (table 1). Pour chaque client, cette matrice agrège tous les articles achetés grâce à un diagramme construit avec la proportion de chaque article

Transaction	Client	X=items	Transaction	Client	X=items	Transaction	Client	X=items
t_1	1	v	t_5	2	v,p	t_9	3	v
t_2	1	v,p,c	t_6	2	v,p,c	t_{10}	4	p,c
t_3	1	v,p,c	t_7	2	v	t_{11}	4	p
t_4	1	v	t_8	3	v,p			

TAB. 4. *Matrice de transactions pour l'algorithme Apriori classique*

N°	Régle	Support %	Confiance %	N°	Règle	Support %	Confiance %
1	c → p	36	100	3	p → c	36	57
2	p → v	45	71	4	v → p	45	55

TAB. 5. *Règles d'association classiques*

par rapport aux achats totaux du client. Nous rentrons en paramètre de l'algorithme diagramme, la précision h=3 (= maximum d'articles pour un client) et $minsup$ = 35% (i.e. 2 clients). Les OS fréquents sont donnés table 2.

Nous donnons les règles obtenues table 5 pour le cas classique avec $minsup = 35\%$ et $minconf = 55\%$ et pour le cas diagramme table 3 avec $minDC = 55\%$. Dans les deux cas, nous remarquons que l'achat de pâtes et céréales implique, avec une confiance de 100%, l'achat de poissons. Toutefois, la méthode diagramme nous fournit plus d'informations. En effet, nous savons en plus que les clients de pâtes et céréales achètent environ deux fois plus de poissons que de pâtes : $0 < P_c \leq 1/3 \rightarrow 0 < P_p \leq 2/3$ avec un support de 75% et une DC de 60%. Deuxièmement, avec l'étude classique, nous obtenons les règles $v \rightarrow p$ avec $conf(v \rightarrow p) = 55\%$ et $p \rightarrow v$ avec $conf(p \rightarrow v) = 71\%$. Par conséquent, la meilleure règle serait $p \rightarrow v$ alors qu'en symbolique nous avons "l'inverse". En effet, la règle 1/3< P_v ≤2/3→0< P_p ≤2/3 est meilleure que la règle 0< P_p ≤2/3→1/3< P_v ≤2/3 selon la confiance (100%, 75% resp.). Ainsi, nous voyons que si le "degré d'inclusion" de l'achat de poissons dans l'achat de viandes dans les transactions est grand, l'analyse symbolique montre que ce sont plutôt les clients de viandes qui sont aussi clients de poissons et non l'inverse. Et comme le montre la règle 1, les clients de viandes sont aussi clients de poissons même s'ils achètent environ deux fois plus de viandes que de poissons. Si nous prenons un autre exemple, dans un tabac, la vente de cigarettes est importante et par conséquent le "degré d'inclusion" des jeux à gratter dans les cigarettes est grand mais en fait ce sont les clients de cigarettes qui vont acheter des jeux et non l'inverse comme l'aurait suggéré le cas classique.

4. Conclusions et perspectives

Nous avons étendu l'algorithme Apriori au cas des variables symboliques diagrammes dans le but d'extraire des règles d'association à partir de concepts. Nous avons pris comme exemple des clients de n'importe quel type de magasins où nous trouvons des règles entre les articles achetés au niveau des clients et non plus au niveau des transactions. Nous avons constaté que nous découvrions des informations supplémentaires par rapport aux règles classiques. Il serait alors intéressant d'étendre cette méthode à d'autres variables symboliques afin d'extraire des règles d'association plus riches.

5. Bibliographie

[AGR 93] Agrawal, R. and Imielinski, T. and Swami., A, Mining Association Rules between Sets of Items in Large Databases, *ACM SIGMOD Records*, p. , 207-216.

[AGR 94] Agrawal, R. and Srikant, R., Fast Algorithms for Mining Association Rules, *Proceedings of the 20th International Conference on Very Large Databases*, p. , 478-499.

[BOC 00] Bock, H.H. and Diday, E., *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data*, Springer Verlag, 2000.

Segmentation d'images couleurs par la méthode de classification floue T-LAMDA

J.C. Atine, A. Doncescu, J. Aguilar-Martin

LAAS-CNRS
Avenue du Colonel Roche 31077 Toulouse France

RÉSUMÉ. Nous présentons une nouvelle méthode de segmentation d'images couleurs basée sur les arbres de décision floue. Cette méthode fait partie de la classe des méthodes de segmentation d'union-find avec un critère d'homogénéité des régions fondée sur les composantes colorimétriques. Elle est basée sur la réconciliation des avantages de la loi bayésienne avec la structure des méthodes neuronales tout en utilisant des mesures floues, qui se fait sur des pixels adjacents.

MOTS-CLÉS : Classification floue, arbre de décision, image couleur.

1. Introduction

La segmentation d'images est primordiale pour la classification des objets présents dans une image. Si les méthodes à base des filtres fréquentiels ou statistiques permettent d'extraire les contours des objets, les méthodes de clustering permettent d'extraire les régions qui ont des couleurs différentes.
Il existe différentes méthodes de segmentation d'une image couleur basée entre autre sur des méthodes floues [PHILI], basées sur la construction des fonctions d'appartenance introduites dans des algorithmes de type « *Split_and_merge* ».
L'algorithme « *Split_and_merge* » a été proposé par Horowitz et Pavlidis en 1974 [JOLI]. Cet algorithme a comme opposée la méthode ascendante « *Region growing* » que nous utilisons avec la technique d'*union-find*.
La méthode *union-find* consiste dans la construction des arbres de décisions à partir de chaque pixel de l'image, qui seront fusionnés jusqu'à l'obtention des régions optimales. Le critère de fusion est basé sur la méthode de classification non supervisée LAMDA.
La nouvelle méthode de segmentation baptisée TREE-LAMDA a été développée au Laboratoire d'Analyse et d'Architecture des Systèmes de Toulouse. Nous présentons dans ce papier une amélioration des résultats par un tri des pixels avant l'utilisation de LAMDA. Ce tri nous permet d'éviter l'introduction de la position des pixels dans le vecteur d'attributs *[R, V, B]*.

2. La méthode LAMDA

La méthodologie LAMDA permet de représenter un système de classes ou de concepts au moyen de la connexion logique de toutes les informations marginales disponibles pour l'élément. Ensuite on calcule l'adéquation globale qui est fonction de l'adéquation marginale de chaque attribut. L'objet appartient à la classe qui présente le plus haut degré d'adéquation. LAMDA est une méthodologie de classification basée sur un concept car les objets non classifiés sont confrontés à un prototype, formé de chacune des classes existantes. Le caractère flou des prototypes modélise l'imprécision dans la formalisation des concepts. Une propriété importante qui différencie LAMDA des autres méthodes de classification, consiste dans sa capacité a modéliser de façon naturelle l'indistingabilité totale, ou l'homogénéité dans l'univers de description où l'information est obtenue. Ceci est dû à une classe spéciale, la classe non-informative (NIC). La classe NIC accepte tous les objets qui peuvent être contenus dans l'univers de description avec le même degré d'adéquation. D'après le principe

d'adéquation maximale, cette classe représente un seuil minimal d'assignation d'un objet à une classe considérée comme significative.
Nous ne ferons qu'une présentation succincte de LAMDA (Learning Algorithm for Multivariate Data Analysis) car celle ci a déjà été présentée plus en détails dans [PIE 89] [WAI 00] [AGU 90]. L'originalité de LAMDA est de concilier les avantages de la loi bayésienne avec la structure des méthodes neuronales tout en utilisant des mesures floues. LAMDA repose sur l'agrégation d'informations marginales ; chaque information marginale étant calculée grâce à la généralisation de la loi binomiale suivante :

$$\rho_{ij}^{1-\alpha(x_i,c_{i,j})}(1-\rho_{ij})^{\alpha(x_i,c_{i,j})} \qquad [1]$$

où $c_{i,j}$ représente la composante i du centre c_j de la classe J, x_i est la composante i de l'élément x à classer, ρ_{ji} est la probabilité qu'un élément appartienne à la classe c_j et $\alpha(x_i, c_{i,j})$ représente la distance entre x_i et $c_{i,j}$. La grande particularité de LAMDA est son interactivité avec l'utilisateur : celui-ci peut en effet regrouper certaines classes (grâce a une visualisation simple et conviviale de la classification), et réitérer une nouvelle classification pour le reste des éléments. Mais ici l'intervention de l'utilisateur se fait en aval de la classification et non pendant celle-ci.
Nous n'utiliserons pas l'interactivité avec l'expert afin de rester dans le cadre d'une classification non supervisée. Le principal problème de la classification provient des incertitudes au niveau des transitions des classes : celles-ci proviennent à la fois du bruit présent et des données elles-mêmes. Une méthode utilisant la transformée en ondelettes a été proposée pour résoudre ce problème [ALE 97]. L'image est décomposée avec une succession de filtres afin de former une approximation de l'image de plus haute résolution. Cette amélioration consiste à introduire dans la classification les points d'inflexions les plus significatifs des signaux biochimiques. Ces points sont détectés grâce au maximum du module de la transformée en ondelettes et sont introduits sous forme de fonction par palier dans la classification. Cet ajout permet non seulement de résoudre le problème des incertitudes au niveau des transitions entre classes mais aussi de réduire le nombre de classes qui peut être très important dans la classification LAMDA classique. En effet la méthode est capable de réaliser un apprentissage supervisé et non supervisé. Il est aussi possible d'utiliser des attributs qualitatifs et/ou quantitatifs. La méthode représente une stratégie générale de classification.
La complexité des algorithmes d'apprentissage et de reconnaissance dépend directement des fonctions spécifiques adoptées. Avec certaines fonctions, la méthode LAMDA est très performante vis-à-vis du volume de calcul aussi bien pour l'apprentissage que pour la reconnaissance. Dans le cas de nos images, le pourcentage d'erreur observé dépend de la fonction de présence choisie relative au type de donnée dans l'image. Pour une image de simple primitive $Lamda_0$ sera suffisante.

$$Lamda_0 = |\rho - X| \qquad [2]$$

Dans cet article, nous utiliserons pour nos résultats *Lamda1* pour laquelle nous obtenons de meilleurs résultats pour les descripteurs de couleurs choisis.

$$Lamda_1 = \rho^x (1-\rho)^{1-x} \qquad [3]$$

3. Les arbres de décision floue

Les notions de chemin et de voisin introduisent implicitement des notions de connexité entre les pixels. Les plus couramment utilisées sont la 4-connexite et la 8-connexite.
Nous avons utilisé une structure basée sur les arbres de décisions, pour représenter les données classées. La technique des arbres de décision [9] est fondée sur l'idée de réaliser la classification d'un objet par une suite de tests sur les attributs qui le décrivent. Ces tests sont organisés de telle façon que la réponse à l'un deux indique à quel prochain test on doit soumettre cet objet. Le principe est d'organiser l'ensemble des tests possibles comme un arbre. Une feuille de cet arbre désigne une des C classes (mais à chaque classe peut correspondre plusieurs feuilles) et à chaque noeud est associé un test (un sélecteur) portant sur un attribut, dans notre cas les valeurs R, G, B des pixels.
Nous utilisons la structure de données habituelle pour la résolution du problème de fusion, par la méthode LAMDA, dans l'algorithme classique Union-Find [FIO 94].
En utilisant la méthode de « région growing » on fait des fusions de régions. Ce n'est pas le cas pour l'union-find où l'on rajoute un élément à une classe.
La « région growing » ou grossissement de régions [MIN] est souvent utilisée dans le cadre de la segmentation d'images aériennes [BIC]. On part de petites régions, soit des pixels ou des points et on les regroupe jusqu'à ce que l'on considère que l'on est dans le cas optimal. On améliore encore l'efficacité de la structure en utilisant le procédé dit de compression des chemins (ou path compression) : lors de la recherche de la classe d'un élément x, on remonte dans la structure jusqu'au père. On en profitera pour relier directement chaque nœud visité au père.

Pour valider l'utilité du critère Lamda pour la segmentation, nous avons fait des tests avec une simple distance pour la classification, et nous remarquons cependant que la classification est toujours meilleure avec T-Lamda.

3.1. *Les opérations*

Nous utilisons une structure de données qui peut manipuler les opérations suivantes :
a) chaque pixel représente un ensemble
b) *Union* : remplacer les deux ensembles x et y par leur union si le critère flou donné par LAMDA est satisfait.
c) *Find* : envoie les pixels appartenant au même père dans une nouvelle structure de type arbre.
d) Accord : dire si deux classes peuvent être regroupées ou pas. Pour cela nous calculons le degré d'agrégation floue de la classe de plus faible poids par rapport à l'autre classe.
Les autres classes utilisées pour la comparaison sont connexes à la région (X, Y)

4. Résultats

Tout d'abord nous nous plaçons dans le cas d'apprentissage non supervisé où nos données sont représentées par une structure.
L'opération de recherche dit si un élément appartient à une classe, en calculant son degré d'agrégation par rapport à chaque classe existante pour inférer sa classe d'appartenance probable.
Les résultats que nous avons obtenus lors de la classification demeurent acceptables du point de vue reconnaissance mais un problème se pose au niveau de la gestion des composantes connexes de l'image voir *Figure1*.

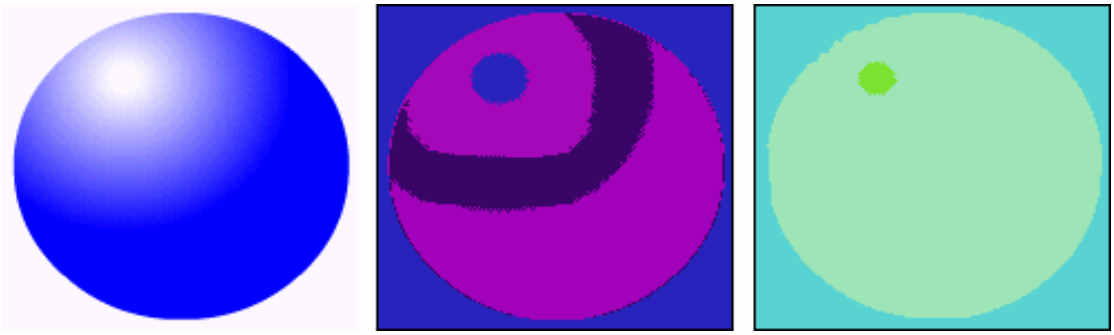


Figure 1. Résultat de la segmentation. A gauche l'image originale, au centre la segmentation obtenue avec Lamda et à droite la segmentation obtenue avec T-Lamda. On note que pour l'image du centre que deux régions de même couleur séparée sont considérées comme faisant partie du même ensemble sans prise en compte de la localisation spatiale, ce qui n'est pas le cas avec T-Lamda.

Nous avons tenté plusieurs méthodes afin de prendre en compte la position spatiale des pixels pour éviter les erreurs de regroupement. Tout d'abord nous avons tenté d'introduire les coordonnées x et y des pixels dans notre algorithme de classification en utilisant le même critère $Lamda_1$ pour tous les paramètres, puis $Lamda_1$ pour les descripteurs *R, V, B* et $Lamda_3$ pour les descripteurs de position. Les résultats ne se sont pas révélés satisfaisants (voir. *figure 2*) ; de plus le temps de calcul s'avère très long lors du calcul des distances par rapport à une classe ou lors de la recherche de la classe voisine dite classe d'appartenance d'un voisin du pixel considéré. En effet nous considérons que si un pixel a une probabilité d'appartenir à une classe K alors son voisin « proche » a la même probabilité d'y appartenir. Cette opération s'avère coûteuse et peu intéressante car le calcul du degré d'agrégation d'un élément par rapport à une classe s'effectue par rapport à toutes les classes existantes ; dans l'utilisation de Lamda même le pixel n'a aucune chance d'y appartenir du fait de sa connexité.
Nous avons aussi défini un seuil variable, pour la mesure de la distance entre deux pixels dans un premier cas et d'un pixel à une classe dans un second cas. Cette mesure s'ajoute au critère de classification Lamda lors de la phase de regroupement.

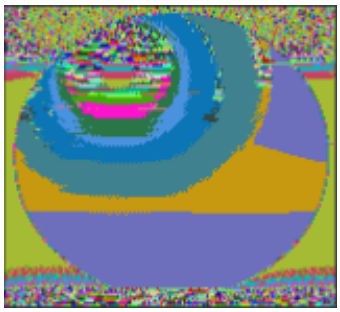


Figure 2. Résultat de la classification en appliquant $Lamda_1$ aux différentes observables R, V, B, x, y.

Figure 3. Résultat de la segmentation en utilisant une structure d'arbre de décision associée à une simple distance.

En appliquant $Lamda_1$ aux descripteurs RVB et $Lamda_3$ à x et y et du fait que les données arrivent ligne par ligne, cela nous génère un sous regroupement.

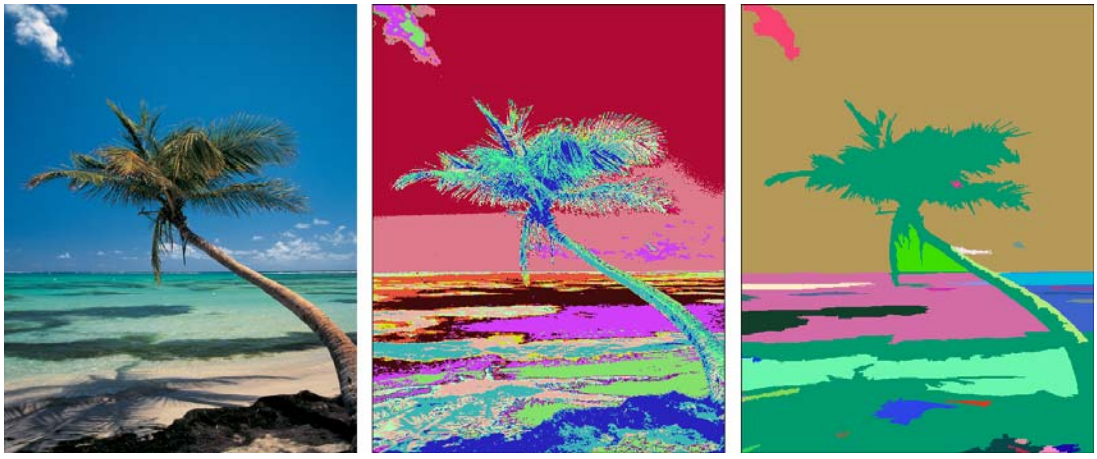


Figure 4. Résultat de la segmentation en utilisant l'algorithme de segmentation Lamda. A gauche l'image originale, au centre la segmentation obtenue avec Lamda, et à droite le résultat obtenu avec T-Lamda.

Nous avons introduit dans notre méthode de classification la méthode d'arbre de décision en changeant le mode d'apprentissage et l'initialisation des données.

Lamda-Tree
La méthode consiste à adapter notre algorithme de classification LAMDA à une structure d'arbre de décision [EYR]. Nous utilisons la structure de données habituelle pour la résolution du problème dit Union-Find [8] adaptée à l'algorithme de classification Lamda

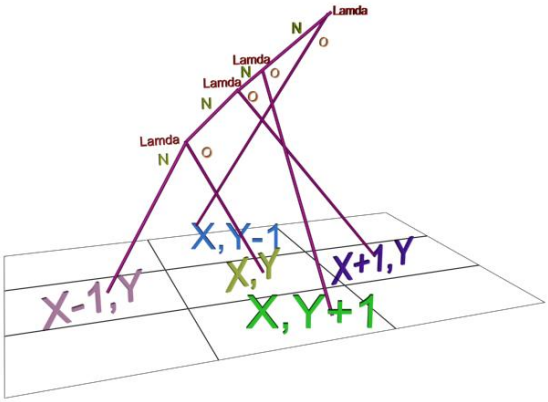


Figure 5. A chaque noeud Lamda, on calcule le degré d'agrégation des régions connexes à X Y selon la structure de décision.

5. Conclusion

Nous présentons dans ce papier une méthode originale basée sur le tri des pixels avant l'utilisation de LAMDA. Ce tri nous permet d'éviter l'introduction de la position des pixels dans le vecteur d'attributs *[R, V, B]*.
La méthode Lamda fonctionne relativement bien pour la classification, elle est sensible au bruit et fournit un nombre de régions trop important sur ces types d'images. Cependant la méthode T-Lamda est une alternative robuste au bruit, elle permet de prendre aussi en compte les caractéristiques spatiales de l'élément. Ceci nous permet de diminuer les erreurs de classification, un pixel n'étant comparé qu'aux classes voisines auxquelles il est susceptible d'appartenir. Cette méthode fournit des résultats satisfaisants, cependant elle n'a pas réellement été comparée à d'autres méthodes non supervisées, plus classiques.

6. Bibliographie

[AGU 90] J. Aguilar Martin, F. Jarachi, M. Chan, *Partitioned identification techniques from Poisson observations: application to cerebral blood flow estimation*, Rapport LAAS No89244 11th IFAC World Congress, Tallinn (URSS), 13-17 Août 1990, pp.24-27.

[ALE 97] Aleksandra Mojsilovic, Miodrag V. Popovic, Aleksandar N. Neskovic, Aleksandar D. Popovic, *Wavelet Image Extension for Analysis and Classification of Infarcted Myocardial Tissue*, IEEE Transaction on biomedical engineering, vol. 44, n°9, septembre 1997.

[BIC] Manuele Bicego, Silvio Dalfini, Vittorio Murino, *Aerial images analysis for the extraction of geographical entities*.

[EYR] *Apprentissage artificiel, concepts et algorithmes*, ed. EYROLLES, p 334-362.

[FIO 94] Christophe Fiorio, Jens Gustedt, *Two linear time Union-Find strategies for image processing*, LIRMM octobre 1994.

[MIN] Mingkun Li, Ishwar K. Sethi, Dongge Li and Nevenka Dimitrova, Region *Growing Using Online Learning*.

[PIE 89] N. Piera, P.desproches, J. Aguilar Martin, *Lamda : An incremental conceptual clustering method*, rapport n°89420 LAAS décembre 1989.

[WAI 00] Julio Waissman-Vilanova, *Construction d'un modèle comportemental pour la supervision de procédés : application à une station de traitement des eaux*, Rapport LAAS n°00601, 2000.

[LAMB] P. Lambert, H. Grecu, *A quick and coarse color image segmentation*, ICIP 2003, 14-17 Septembre, 2003.

[DT ID3] *Tutorial: Decision Trees: ID3, Faculty of Information Technology*, CSE5230 Data Mining, semester 2, 2003, Monash University

[JOLI] Jean-Michel Jolion, *Approches de l'image*, RFV-INSA Lyon 1997, DEA DISIC 2001-2002

[PHILI] Sylvie PHILIPP-FOLIGUET, *Segmentation d'images en régions floues*, Logique Floue et Applications, LFA 2000, La Rochelle, 2000.

[SMITH] John R. Smith, Shi-Fu Chang, *Joint Adaptive space and frequency basis selection*, IEEE ICIP'97, Santa Barbara

[SUZUK] Hirotaka Suzuki, Pascal Matsakis, Jacky Desachy, *Exploitation de connaissances structurelles en classification d'images : utilisation de méthodes heuristiques d'optimisation combinatoire.*

Classification automatique de documents : application au web

H. Azzag[*], **C. Guinot**[**], **G. Venturini**[*]

[*] *École Polytechnique de l'Université de tours, Laboratoire d'Informatique, 64, Avenue Jean Portalis, 37200 Tours, France*
hanene.azzag@etu.univ-tours.fr, venturini@univ-tours.fr

[**] *CE.R.I.E.S., 20, rue Victor Noir, 92521 Neuilly-sur-Seine Cédex, France*
christiane.guinot@ceries-lab.com

RÉSUMÉ. *Nous présentons dans cet article un nouvel algorithme appelé AntTree pour la classification hiérarchique de documents et son application à la génération automatique de sites portails. Il utilise le principe d'auto-assemblage observé chez les fourmis réelles pour la construction d'un arbre (une hiérarchie) dont les noeuds sont connus a priori et les arcs restent à déterminer. Chaque fourmi va représenter un noeud du graphe à assembler, donc un document à classer. Partant d'un point initial (un support) les fourmis vont se déplacer et se fixer successivement les unes aux autres, construisant ainsi plusieurs niveaux de la hiérarchie. Tous ces déplacements dépendent de la fonction de similarité calculée entre les documents. Nous avons testé notre modèle sur une série de pages web extraites d'Internet. Nous avons comparé avec succès les résultats obtenus avec ceux de la CAH (classification ascendante hiérarchique). Enfin nous montrons que notre modèle apporte des améliorations significatives au problème posé.*

MOTS-CLÉS : *Classification hiérarchique, page web, site portail, fourmis artificielles*

1. Introduction

La taille des serveurs web et le nombre de documents qu'ils proposent augmentant sans cesse, la recherche d'information devient de plus en plus difficile. Les outils de recherche disponibles actuellement offrent des possibilités de recherche basées essentiellement sur des mots clés. Cette formulation de requête limite les moteurs de recherche et les réponses qu'ils apportent sont généralement peu précises, même pour des requêtes bien détaillées. Les sites portails peuvent êtres considérés comme une bonne alternative. Ce sont des outils efficaces lorsque l'utilisateur désire une information d'un certain type ou d'un certain sujet. Mais malheureusement leur construction requiert un effort considérable. C'est pour cette raison qu'il est nécessaire de penser à concevoir des méthodes automatiques de génération de sites portails.

La conception de chaque outil de recherche ou site portail doit commencer par la collecte de documents à indexer. Il faut ensuite extraire des pièces d'information à partir des documents trouvés (titres, mots clés, etc.) et enfin, présenter le site portail avec une classification hiérarchique de ces documents basée sur la similarité entre les textes. D'une manière plus générale un site portail peut être vu comme une classification hiérarchique d'un ensemble de documents en catégories et sous catégories, de sorte que chaque sous catégorie soit la plus similaire possible à sa catégorie mère et la plus dissimilaire possible aux autres. En suivant cette optique, le problème majeur à résoudre est de définir de manière automatique cette hiérarchie qui est actuellement réalisée manuellement par des expert humains [FIL 97][KUM 01][SAN 99][MCC 00]. Si on travaille avec un grand nombre de documents, ou si on souhaite que la machine puisse de manière autonome construire un tel site, les approches existantes seront inopérantes.

Le but de notre travail est de construire de manière automatique une hiérarchie tout en classant de façon arborescente des pages web. Pour ce faire nous définissons une nouvelle méthode basée sur les fourmis artificielles que nous présentons dans cet article avec des résultats expérimentaux obtenus sur plusieurs bases de tests.

2. Modèle biologique et algorithme de fourmis

Dans la nature, les fourmis offrent un modèle stimulant pour le problème de la classification. En effet, leur stratégie a été sélectionnée sur plusieurs millions d'années d'évolution et s'est par conséquent révélée très efficace. Le problème du regroupement d'objets ou d'individus est en effet très présent dans la nature et de nombreuses espèces ont dû développer des comportements souvent sociaux pour le résoudre. Citons l'exemple du tri du couvain chez les fourmis [FRA 92] ou encore les déplacements collectifs chez de nombreuses espèces [CAM 01]. Ces algorithmes peuvent bénéficier de propriétés intéressantes comme l'optimisation locale et globale de la classification, l'absence d'information sur une classification initiale des données, le parallélisme, etc.

Le modèle que nous définissons est lié au phénomène d'auto-assemblage observé chez certaines fourmis [LIO 00]. Ces dernières ont la capacité de construire des structures vivantes ayant différentes fonctionnalités. Les fourmis peuvent ainsi construire des "chaînes de fourmis" leur permettant de passer d'un point à un autre, ou de rapprocher des bords d'une feuille pour y placer leur nid, ou encore des "gouttes de fourmis" ce qui semble être une fonctionnalité encore inexpliquée. À partir de ces éléments, nous définissons brièvement le modèle informatique utilisé, et précédemment validé sur des données numériques dans [AZZ 03].

Les fourmis $f_1, ..., f_n$ sont placées initialement sur le support f_0 (f_0 représente le support sur lequel va être construit le graphe). Chaque fourmi f_i représente un document d_i de la base de documents à classer. Nous simulons successivement une action pour chaque fourmi. Cette dernière peut avoir deux états : elle est soit libre de se déplacer ou de se connecter, soit assemblée à la structure sans la possibilité de se déplacer mais seulement de se décrocher. Les fourmis ne perçoivent la structure que localement. Pour une fourmi f_i en déplacement et positionnée sur une fourmi f_{pos} assemblée à la structure, le voisinage V_{pos} perceptible par f_i est limité à f_{pos}, à la fourmi mère de f_{pos} (du niveau précédent dans l'arbre), aux fourmis filles de f_{pos}. La fourmi f_i peut donc percevoir les valeurs de similarité entre le document qu'elle représente et les documents représentés par les fourmis de V_{pos}. En fonction de ces valeurs de similarité, elle peut soit se connecter à f_{pos}, soit se déplacer sur une des fourmis de V_{pos}. Ainsi, une fois toutes les fourmis accrochées les unes aux autres (ou sur le support), l'algorithme s'arrête. L'arbre résultant représente une classification des documents. Les propriétés visées pour une classification de documents représentant un site portail sont les suivantes : chaque sous-arbre A représente une catégorie composée de toutes les fourmis de A. Soit f la fourmi qui est à la racine d'un sous-arbre A. Nous souhaitons que 1) f soit représentative de cette catégorie (les fourmis placées dans A sont les plus similaires possible à f), 2) les fourmis filles de f qui représentent des sous-catégories soient les plus dissimilaires possible entre elles. Autrement dit, un bon site portail est constitué de catégories homogènes, et pour une catégorie donnée, les sous-catégories sont judicieusement choisies (les plus dissimilaires entre elles possible).

3. Résultats

Nous avons évalué AntTree sur un ensemble de 4 bases de 258 à 1025 textes (voir figure 1(a)). La base *Reuters* contient 1025 textes extraits de la base *reuteurs21578*. La base *CE.R.I.E.S.* contient 258 textes sur la peau humaine saine [GUI 03]. La base *Database 1* contient des textes sur des sujets scientifiques. Enfin la base *Database 2* contient des textes sur des sujets différents (médecine, informatique, téléphonie, ...). Les bases *Database 1* et *Database 2* sont extraites d'Internet, les documents sont ensuite classés en catégories afin de pouvoir évaluer la qualité de la classification obtenue. Cette classe n'est bien entendu pas fournie à AntTree.

La classification obtenue est évaluée à la fois en terme de nombre de classes trouvé C_t, de pureté des classes P_r et d'erreur couple Ec. Pour une classe donnée la pureté représente le pourcentage de pages bien classées ; Ec représente une mesure d'erreur de classification fondée sur les couples de documents de la base. Nous utilisons la

Bases	Taille (# de documents)	Taille (Mb)	# de classes
Reuters	1025	4.05	9
CE.R.I.E.S.	258	3.65	17
Database 1	319	13.2	4
Database 2	524	20	7

(a)

Bases	AntTree	CAH
Reuters	1.51	120
CERIES	0.04	4
Database 1	0.12	6
Database 2	0.34	25

(b)

FIG. 1. *Descriptif des bases utilisées (a) avec les temps d'exécution en secondes (b)*

		CAH			AntTree		
Bases	C_r	Ec	P_r	C_t	$Ec[\sigma_{Ec}]$	$P_r[\sigma_{P_r}]$	$C_t[\sigma_{C_t}]$
Reuters	9	0.21	0.50	5	0.35 [0.004]	0.40 [0.007]	12 [0.00]
CERIES	17	0.21	0.30	7	0.15 [0.001]	0.37 [0.012]	17 [0.00]
Database 1	4	0.09	0.82	7	0.29 [0.011]	0.68 [0.012]	7 [0.00]
Database 2	7	0.23	0.52	3	0.10 [0.006]	0.80 [0.009]	8 [0.00]

Ec L'erreur de classification moyenne obtenue sur 15 essais
C_r Nombre de classe réelle
C_t Nombre de classe moyen trouvé obtenu sur 15 essais
σ_x : Ecarts types
P_r La pureté moyenne obtenu sur 15 essais

FIG. 2. *Resultats obtenus par AntTree et CAH sur les différentes bases*

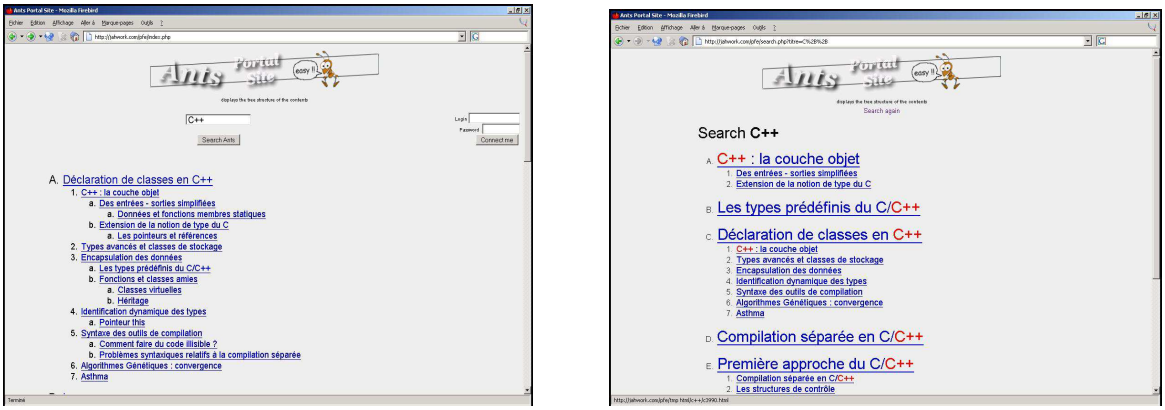
mesure de similarité *cosinus*, où chaque document est représenté par un vecteur de poids calculé suivant le schéma *tf-idf* [SAL 88].

Pour toutes les bases, nous comparons les résultats obtenus par notre algorithme avec ceux obtenus par la classification ascendante hiérarchique [JAI 99]. Notre analyse des résultats (voir figure 2) est la suivante : en moyenne, les deux algorithmes obtiennent les mêmes performances. AntTree est meilleur que CAH pour le même nombre de cas. Généralement AntTree apporte de meilleurs résultats que CAH quand les données sont suffisamment dissimilaires entre elles et trouve un nombre de classes plus proche du nombre de classes réel que CAH. Les résultats obtenus sont donc très encourageants compte tenu du fait que notre algorithme est de 50 à 100 fois plus rapide que CAH (voir tableau 1 (b)).

De plus on peut facilement générer de manière automatique le site portail une fois les pages classées en arbre. La hiérarchie de documents ainsi construite est stockée dans une base de données. Les pages HTML du site sont générées de manière dynamique en PHP. La figure 3(a) représente l'interface du portail obtenu sur la base *Database 2* (524 documents). La figure 3(b) montre un exemple d'intégration d'un outil de recherche utilisant un index inversé généré automatiquement dans la base de données.

4. Conclusion

Dans cet article nous avons décrit une nouvelle approche de génération automatique de site portail fondée sur les colonies de fourmis. Nous avons comparé notre algorithme avec la méthode CAH. Les résultats obtenus sont très satisfaisants, notamment en ce qui concerne le nombre de classes trouvé et le temps d'exécution.



(a) (b)

FIG. 3. *Interface du site portail généré (a) et Recherche avec le mot C++ (b)*

Comme perspective nous comptons appliquer notre algorithme à une large collection de textes et également implémenter le décrochage des fourmis. Chaque fourmi aura donc la possibilité de se déconnecter de sa position et de se déplacer vers d'autres fourmis peut-être plus similaires à elle. On s'intéresse en outre à généraliser notre algorithme à la construction de graphes (pas seulement des arbres) avec lesquels on peut générer automatiquement des hypertextes avec le même principe d'auto-assemblage.

5. Bibliographie

[AZZ 03] AZZAG H. MONMARCHÉ N., SLIMANE M., VENTURINI G., GUINOT C., Algorithme AntTree : classification non supervisée par des fourmis artificielles, *Revue des nouvelles technologie de l'information*, cépadués éditions, 2003, p. 75–86.

[CAM 01] CAMAZINE S., DENEUBOURG J.-L., FRANKS N. R., SNEYD J., THERAULAZ G., BONABEAU E., *Self-Organization in Biological Systems*, Princeton University Press, 2001.

[FIL 97] FILO D., YANG J., Yahoo !. http ://www.yahoo.com, 1997.

[FRA 92] FRANKS N., SENDOVA-FRANKS A., Brood sorting by ants : distributing the workload over the work surface, *Behav. Ecol. Sociobiol*, vol. 30, 1992, p. 109-123.

[GUI 03] GUINOT C., MALVY D. J.-M., MORIZOT F., TENENHAUS M., LATREILLE J., LOPEZ S., TSCHACHLER E., DUBERTRET L., Classification of healthy human facial skin, Textbook of Cosmetic Dermatology Third edition (to appear), 2003.

[JAI 99] JAIN A. K., MURTY M. N., FLYNN P. J., Data clustering : a review, *ACM Computing Surveys*, vol. 31, n^o 3, 1999, p. 264–323.

[KUM 01] KUMAR R., RAGHAVAN P., RAJAGOPALAN S., TOMKINS A., On semi-automated Web taxonomy construction, *WebDB*, Santa Barbara, May 2001.

[LIO 00] LIONI A., Auto-assemblage et transport collectif chez oecophylla, Thèse de doctorat, Université libre de bruxelles, Université Paul Sabatier, 2000.

[MCC 00] MCCALLUM A. K., NIGAM K., RENNIE J., SEYMORE K., Automating the Construction of Internet Portals with Machine Learning, *Information Retrieval*, vol. 3, n^o 2, 2000, p. 127–163, Kluwer Academic Publishers.

[SAL 88] SALTON G., BUCKLEY C., Term weighting approaches in automatic text retrieval, *information processing and management*, vol. 25, 1988, p. 513–523.

[SAN 99] SANDERSON M., CROFT W. B., Deriving Concept Hierarchies from Text, *Research and Development in Information Retrieval*, 1999, p. 206-213.

Extraction de métadonnées sur les prototypes issus de la classification d'objets

Abdourahmane Baldé[1], Yves Lechevallier[1], Marie-Aude Aufaure[2]

(1)*INRIA Rocquencourt*
Domaine de Voluceau
Rocquencourt - B.P. 105
78153 Le Chesnay Cedex
(2)*Supélec Plateau du Moulon*
3, rue Joliot-Curie
91192 Gif-sur-Yvette cedex

RÉSUMÉ. Nous présentons ici une méthodologie d'extraction des métadonnées sur des prototypes issus de la classification d'objets (pouvant être élargie aux objets symboliques). Les métadonnées, utilisées généralement pour une meilleure gestion de l'information, seront créées dans le but d'archiver des informations jugées pertinentes lors du processus de classification. Cette étude a été validée en s'appuyant sur des données issues d'une enquête. L'objectif étant de pouvoir décrire un ensemble de données en fournissant toute l'information les concernant et pas seulement l'information d'ordre bibliographique.

MOTS-CLÉS : classification, métadonnées, objets symboliques, ontologies, rdf, dublin core

1. Introduction

La maîtrise sur des grands ensembles d'information devient de plus en plus complexe et de plus en plus fastidieux. Les métadonnées constituent une voie pour aider l'utilisateur ou le gestionnaire d'information à comprendre, retrouver, comparer des informations sans forcément avoir recours directement au contenu de celles-ci.

En effet, les métadonnées peuvent être vues comme étant des données structurées qui décrivent les données et qui peuvent s'appliquer à tous types de données.

L'objectif dans ce travail aura été de construire des métadonnées issues depuis la génération des données jusqu'à leur classification. Celles-ci devant rendre compte du contenu des classes créées, des méthodes de classification utilisées et des informations d'ordre général (l'auteur, l'éditeur, la date de création etc.).

Ce travail pourrait ainsi être élargi au domaine de l'analyse des données symboliques. En effet, les données qui seront manipulées peuvent être des objets symboliques (objets qui constituent les individus de l'analyse des données symboliques, permettant de représenter des individus complexes ou des classes d'individus par des conjonctions de propriétés ou des descripteurs) [Diday, 1998].

Dans le paragraphe 1, nous présentons une idée sur les notions de métadonnées, de classification et d'objet symbolique[1]. Dans le paragraphe 2, nous précisons nos idées en décrivant notre approche. Une simulation est fournie dans le paragraphe 3. Et nous concluons.

2. Notions générales

Dans cette étude, nous utiliserons les définitions données par [Bui thi et al. , 2001], [Diday, 2003] et [Diday, 1998] en matière de métadonnées, de classification et d'objet symbolique respectivement.

Ainsi, les *métadonnées* sont définies comme des informations émises à un niveau d'abstraction supérieur et relatives à un niveau d'abstraction inférieur. Ce qui fait intervenir les notions de réflexivité et d'abstraction.

[1] Nous utiliserons le terme «objet symbolique» pour parler de données agrégées et ce dans un souci de généralisation de ce travail à tous types de données agrégées y compris les objets dits symboliques

La *classification automatique* quant à elle, est définie comme un ensemble de méthodes et algorithmes consistant à découper une population d'objets en plusieurs classes, en tenant compte des variables qui les caractérisent et de la mesure de ressemblance choisie.
L'*objet symbolique* est quant à lui défini par une description notée 'd' ; une relation binaire 'R' sur D permettant de comparer d à une autre description de D ; une fonction 'a' permettant d'évaluer le résultat de la comparaison (à l'aide de R) de la description d'un individu du monde réel par rapport à la description donnée 'd'.
Notre approche tiendra compte aussi des normes déjà existantes. Ainsi, les éléments du *Dublin Core* ont largement été utilisés pour constituer les métadonnées (de type bibliographique) de nos fichiers de données/métadonnées.
En effet, le *Dublin Core* est un ensemble de 15 éléments simples qui définissent les catégories d'information à enregistrer à propos d'une ressource (page Web, document ou image) pour que celle-ci puisse être trouvée.
Le schéma *RDF* (composé de trois éléments : *ressource, propriété, déclaration*) sera utilisé pour fournir une description des éléments de métadonnées extraits. *RDF* définit la signification, les caractéristiques et les relations d'un ensemble de propriétés.
Résultant du travail du W3C (*Consortium du world Wide Web:* créateur des standards pour le Web), RDF définit une structure de métadonnées pour décrire le contenu du Web à l'aide du langage *XML* [Gardarin, 2002] ainsi que les relations entre ressources.
Voici un exemple faisant usage de la norme RDF, dans cet exemple on veut expliquer que l'auteur de la ressource *«Web et ontologies»* est *«Marie-Aude» :*

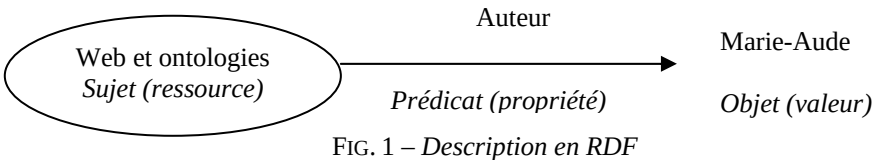


FIG. 1 – *Description en RDF*

3. Idées de base

Pour réaliser ce processus d'extraction, nous partons des données recueillies lors d'une enquête. L'idée étant de définir un ensemble d'éléments de métadonnées pouvant rendre compte d'informations portant sur les données recueillies. Ensuite, définir de nouveaux éléments de métadonnées lors de la phase de classification automatique.
Pour ce faire, nous nous sommes inspirés des éléments de [Csernel, 2002] et du Dublin Core. L'objectif poursuivi étant, bien évidemment, la définition d'éléments pouvant rendre compte des informations sur, d'une part, les données originales et agrégées, et sur les classes obtenues d'autres parts.

4. Méthodologie d'extraction des métadonnées

Les métadonnées que nous avons extraites ont la spécificité d'être intégrées avec les données qu'elles décrivent. Ce qui a l'avantage d'être simple, clair et facile à comprendre pour les utilisateurs.
Ce travail consiste donc à extraire de manière automatique des informations jugées pertinentes au cours du processus de classification et de collecte des données. Cette étape d'extraction de métadonnées s'est déroulée en plusieurs phases que nous développons ci-après.

4.1. *Méthodologie*

La méthodologie d'extraction utilisée est celle qui consiste à réaliser l'extraction en 3 phases :

1. Renseigner les éléments de l'entête. Ces éléments sont constitués essentiellement par les éléments du Dublin Core.
2. Renseigner les éléments de métadonnées spécifiques à la classification automatique. Au cours de cette phase particulièrement périlleuse et ce, compte tenu du nombre important d'éléments à renseigner, notre travail a été tout d'abord de donner une sémantique claire à chacun de ces éléments qui étaient déjà définis par [Csernel, 2002].

Extraire, lors de l'application des différentes méthodes de classification, les informations afin de créer un historique des prototypes d'objet. En effet, nous devons être en mesure, lorsque nous disposons d'une classe d'objets, de connaître des informations sur l'origine de ces données, sur le critère de classification choisi, etc.

4.2. *Création et maintenance des métadonnées*

Ici, nous tentons de faire une analyse sur le processus de création et de maintenance des métadonnées dans le cadre général.
Ce processus de création de métadonnées peut se diviser en quatre (4) étapes :

- La définition des besoins : définir les besoins de l'organisation qui souhaite l'intégration des métadonnées. Cette phase doit tenir compte des normes déjà en vigueur afin de faciliter l'interopérabilité avec d'autres organismes.
- L'extraction et l'intégration des métadonnées : intervient après la définition de l'ensemble des éléments de métadonnées à renseigner.
- La promotion des métadonnées : rien ne sert de créer des métadonnées si l'on ne peut les faire découvrir à d'autres gens. Ainsi, le meilleur moyen de faire connaître «ses» métadonnées, c'est de les publier par le Web.
- La maintenance des métadonnées : mettre à jour les métadonnées dès que les données qu'elles décrivent changent. La réussite de cette étape dépendra de deux facteurs importants : le taux d'obsolescence des données décrites et les moyens mis à disposition par les organisations concernées.

5. Simulation

Dans cette section, nous allons illustrer nos propos (des sections précédentes) en fournissant les résultats issus d'extraction automatique de métadonnées dans le cadre de la classification d'objets.
Pour ce faire, nous allons partir d'une base de données qui décrit 150 iris. Dans cette base, les iris sont décrits par quatre variables (longueur et largeur du sépale, longueur et largeur du pétale). A ces individus, on applique une méthode de classification non supervisée. Cette dernière nous fournit trois classes d'iris avec pour chacune la description des individus qui la composent.
A la suite de cette classification, nous avons obtenu trois classes homogènes. Les métadonnées extraites et relatives aux individus ont cette structure :

```
<Entête>
   <dc : title>Iris</dc : title>
   <dc : author>Yves Lechevallier</dc :author>
   <dc : date>11/03/04</dc : date>
   <dc : language>Français</dc : language>
</Entête>
<OrigInfo>
   <NbOrigVar>4</NbOrigVar>
   <NbOrigMat>1</NbOrigMat>
   <PopSampSize>150</PopSampSize>
</OrigInfo>
```

```
<OrigVar>
        <Num>1</Num>
        <Name>Iris</Name>
        <Label>longueur du sépale</Label>
        <Computed>select * from IRIS</Computed>
</OrigVar>
<MetaInd>
        <Num>1</Num>
        <Name>Setosa</Name>
        <Operator>Native Data</Operator>
</MetaInd>
```

Nos fichiers de métadonnées se composent de trois parties : la première est relative aux informations d'ordre général (titre, auteur, etc.), la seconde est relative à la description des variables (qui décrivent nos individus), la dernière est relative aux objets qui sont agrégées (nous avons ainsi des informations sur l'opérateur d'agrégation, le nombre d'individus qui ont été agrégés pour former notre objet, etc.). Ainsi, pour chacune des classes, nous avons, en plus de l'entête, la description suivante :

```
<MetaHistory>
   <SqlQuery>select * from IRIS_Classe</SqlQuery>
   <Source>c:\user\yves\asso\iris</Source>
   <OdbcSource>11/03/04</OdbcSource>
</MetaHistory>
```

```
<MetaInd>
    <Num>2</Num>
    <Name>classe3/3</Name>
    <NbInObj>42</NbInObj>
    <GNbInObj>42</GNbInObj>
   < Operator >agregrated</Operator>
</MetaInd>
```

Nous voyons, à partir de ces deux bouts de résultats, que la construction d'un historique est relativement facile si nous prenons les métadonnées comme base à cette opération.
A la suite de cette étape, nous avons obtenu des métadonnées relatives aux individus à classifier et celles qui sont relatives aux classes d'individus. Pour exemple, nous avons la description de la 3ème classe (classe3/3) qui nous donne le nombre d'individus de la classe (42) ainsi que le nombre d'individus agrégés pour créer celle-ci (42).

6. Conclusion

Les métadonnées sont un instrument qui transforme les données brutes en connaissances. Elles représentent une valeur ajoutée à l'information en permettant leur compilation et leur repérage. Malgré la différence de structure, tous les types de métadonnées poursuivent un objectif commun : offrir des éléments de description pour faciliter l'accès à des ressources données en fournissant toute l'information les concernant. Le W3C travaille énormément dans le but de donner une dimension supplémentaire à l'utilisation des métadonnées. Elles

constituent un véritable moyen de capitalisation des connaissances et du savoir-faire. C'est d'ailleurs la perspective qui paraît la plus prometteuse.
En effet, on pourrait faire intervenir les ontologies dans ce processus de capitalisation et de représentation des connaissances (cf. [Kassel, 2002] et de [Kassel et al, 2000]). Les ontologies peuvent apporter une dimension sémantique aux métadonnées et permettre surtout de faire face à la complexité d'organisations taxonomiques.
Une autre perspective serait d'utiliser RDF pour représenter les liens entre les différentes ressources de métadonnées obtenues au cours d'un processus de classification (cf. au document traduit par Karl Dubost *(http://www.la-grange.net/w3c/REC-rdf-syntax/),* relatif aux spécificités de RDF).
Pour faciliter la définition des métadonnées, RDF aura un système de classe comme dans tout environnement de programmation orienté objet et de modélisation. Ces classes, organisées en hiérarchie, offrent une extensibilité grâce à la subtilité des sous-classes. De cette façon, pour créer un schéma légèrement différent d'un autre déjà existant, il n'est pas nécessaire de "réinventer la roue" mais il faut juste fournir des modifications incrémentales au schéma de base.

7. Bibliographie

[Bui thi et al. , 2001] M.P. Bui thi, P. Joly, P. Faudemay, La description du contenu de la vidéo selon les points de vue de la production audiovisuelle, CIDE 01 (Conférence Internationale du Document Electronique), Toulouse, octobre 2001.

[Csernel, 2002] M. Csernel, Meta Data Specification for ASSO (Analysis System of Symbolic Official data) Library, 20-01-2002

[Diday, 2003] E. Diday, Cours de DEA-Dauphine, année universitaire 2002-2003

[Diday, 1998] E. Diday. L'analyse des Données symboliques : un cadre théorique et des outils. Rapport du CEREMADE n°9821, 1998.

[Gardarin, 2002] G. Gardarin, XML : des bases de données aux services Web, Dunod, 2002

[Kassel, 2002] G. Kassel, OntoSpec : une méthode de spécification semi-informelle d'ontologies, in Actes des journées francophones d'Ingénierie des Connaissances (IC'2002), pages 75-87, 2002.

[Kassel et al. , 2000] C. Barry, C. Irastorza, G. Kassel, M. Abel, P. Boulitreau et S. Perpette, Construction et exploitation d'une ontologie pour la gestion des connaissances d'une équipe de recherche, in Actes des journées francophones d'Ingénierie des Connaissances (IC'2000), 2000.

Evaluation de la stabilité d'une partition à l'aide de la mesure de Loevinger d'une règle logique

Ghazi Bel Mufti * — **Patrice Bertrand** **

* *ESSEC de Tunis, Dept. Economie*
4 rue Abou Zakaria El Hafsi,
Montfleury, 1089 Tunis, Tunisie
belmufti@yahoo.fr

** *GET-ENST Bretagne, Dept. Lussi*
Technopôle Brest-Iroise CS 83818,
29238 Brest Cedex 3, France
patrice.bertrand@enst-bretagne.fr

RÉSUMÉ. *Une méthode est développée pour mesurer la stabilité d'une classification lorsqu'on retire quelques objets de l'ensemble des objets à partitionner. Des mesures de stabilité d'une classe sont définies comme des mesures de Loevinger de la qualité d'une règle. La stabilité d'une classe peut être ineterprétée comme une moyenne pondérée des stabilités inhérentes, respectivement, à l'isolation et la cohésion de la classe examinée. La conception de la méthode permet en outre de mesurer la stabilité d'une partition, qui peut être perçue comme la moyenne pondérée des mesures de stabilité de toutes les classes de la partition. Comme concéquence, une approche est déduite pour la détermination du nombre de classes optimal d'une partition. Par ailleurs, en utilisant le test de Monte Carlo, un niveau de signification probabiliste est calculé afin de donner une valeur intrinsèque de la mesure de stabilité, sous un modèle nul spécifiant l'absence de stabilité d'une classe. Pour illustrer les potentialités de la méthode, nous présentons des mesures de stabilité qui ont été obtenues en utilisant l'algorithme des K-means sur des données simulées ainsi que sur les iris de Fisher.*

MOTS-CLÉS : *Stabilité d'une classe, Test de Monte Carlo, Isolation et cohésion d'une classe, Mesure de Loevinger, Nombre de classes d'une partition.*

1. Introduction

Les méthodes de classification sont fréquemment utilisées pour analyser les données qui sont recueillies dans diverses disciplines scientifiques. Parallèlement, peu de procédures standard sont disponibles pour valider les résultats générés par ces méthodes (pour une revue des méthodes de validation, voir [JAI 88, MIL 96, GOR 99]). Une approche pertinente en validation d'une classification consiste à définir un indice qui mesure l'adéquation de la structure en classes avec l'ensemble des données étudiées, puis à comparer cette mesure avec les valeurs qui seraient obtenues sur des jeux de données de même taille mais ne possédant pas de structure en classes (*cf.* par exemple [BAI 82, GOR 94]). Plus précisément, cette comparaison consiste à estimer le niveau de signification statistique (p-valeur) de la valeur observée de l'indice de validité d'une classification pour le test de l'hypothèse d'absence de structure dans l'ensemble des données.
Une autre approche en validation est basée sur la stabilité des résultats d'une classification. Une classification est généralement considérée comme étant stable si les classes initiales restent inchangées après de petits changements sur les données. Un aperçu de la littérature sur la stabilité en classification est donné dans [CHE 96]. Plus récemment, différentes méthodes (*cf.* [ROB 97, LEV 01, BEN 02, TIB 01]) ont été proposées pour estimer le ou les nombres optimaux de classes d'une partition, à l'aide de mesures de stabilité en classification. Dans une

approche qui possède plusieurs points communs avec [BEN 02] et [TIB 01], nous avons propososé un indice de stabilité pour une seule classe en mesurant son isolation et sa cohésion (*cf.* [BEL 01]). Dans ce qui suit, nous proposons d'estimer non seulement la stabilité d'une classe mais aussi celle d'une partition, en utilisant la mesure de Loevinger qui a pour but d'évaluer la qualité d'une règle logique (*cf.* [BER 04] pour un exposé plus détaillé). Bien que notre approche s'applique à tout type de bruitage des données, nous présentons ici, par souci de simplification, seulement le cas du bruitage consistant à retirer un faible pourcentage d'objets de l'ensemble des données.

2. Définition des indices de stabilité

Nous considérons un ensemble de n objets à classer, noté $\mathcal{X}$, et supposons qu'un algorithme de partitionnement en k classes, choisi arbitrairement et noté P_k, a été appliqué à $\mathcal{X}$. Nous notons $\mathcal{P}$ la partition de $\mathcal{X}$ en k classes ainsi obtenue : $\mathcal{P} = \mathrm{P}_k(\mathcal{X})$. Notre objectif est d'évaluer la stabilité de $\mathcal{P}$ ainsi que la stabilité de chacune de ses classes. Nous utilisons pour cela deux critères : l'isolation et la cohésion d'une classe. Le principe de base est que la partition $\mathcal{P}$ et ses classes, sont valides si la plupart des ensembles de données bruitées possèdent une structure en classes très proche de la partition $\mathcal{P}$. Ici, nous nous intéressons uniquement au cas où l'ensemble de données bruitées est un échantillon aléatoire tiré dans la population $\mathcal{X}$. Afin de garantir que chaque classe de $\mathcal{P}$ est correctement représentée dans l'échantillon aléatoire, nous utilisons une procédure dite d'échantillonnage stratifié proportionnel. Plus pécisément, si l'on note f le taux d'échantillonnage (f est supposé assez grand pour que l'échantillon puisse être considéré comme étant une perturbation de $\mathcal{X}$, e.g. $f > 0.7$), cette procédure consiste à sélectionner aléatoirement et sans remise n'_A éléments dans chaque classe A de $\mathcal{P}$, en ayant noté n'_A la partie entière de fn_A et n_A la taille de la classe A. Tous les échantillons considérés par la suite, sont obtenus selon cette procédure. Ils sont donc tous de même taille $n' = \sum_{A\in\mathcal{P}} n'_A$ qui est proche de fn par valeurs inférieures.

Définissons tout d'abord l'indice de stabilité qui évalue l'isolation d'une classe. Soit $\mathcal{Q} = \mathrm{P}_k(\mathcal{X}')$ la partition obtenue en appliquant P_k sur un échantillon $\mathcal{X}'$ de $\mathcal{X}$. Nous dirons, de façon naturelle, qu'une classe A de $\mathcal{P}$ est isolée si la règle suivante est vérifiée pour tout échantillon $\mathcal{X}'$ de $\mathcal{X}$:

(R) *Règle d'isolation de A* : Si deux objets de l'échantillon $\mathcal{X}'$ ne sont pas classés ensemble par la partition $\{A, \mathcal{X} \setminus A\}$, alors ils ne le sont pas non plus par la partition $\mathcal{Q}$.

Nous mesurons la qualité de cette règle à l'aide de la mesure de Loevinger ([LOE 47]). Rappelons que la mesure de Loevinger, qui est égale à $1 - P(E \cap \neg F)/P(E)P(\neg F)$ si $E \Rightarrow F$ est la règle examinée, possède des propriétés intéressantes. En effet, elle vaut 0 dans le cas d'indépendance entre les évènements E et F et atteint la valeur maximale 1 dans le cas d'une implication parfaite (cf. aussi [LEN 03]). En notant $t^{\,is}(A, \mathcal{Q})$ la mesure de Loevinger de la qualité de la règle (R), on a :

$$t^{\,is}(A, \mathcal{Q}) = 1 - \frac{n'(n'-1)m_{(\mathcal{Q};A,\overline{A})}}{2n'_A(n'-n'_A)\; m_{(\mathcal{Q})}} \,,$$

où :

- $m_{(\mathcal{Q})}$ est le nombre de paires d'objets qui sont classés ensemble par la partition $\mathcal{Q}$,
- $m_{(\mathcal{Q};A,\overline{A})}$ est le nombre de paires d'objets échantillonnés qui sont dans la même classe de la partition $\mathcal{Q}$ et tels que seulement l'un des deux objets appartient à A.

Considérons un grand nombre N d'échantillons $\mathcal{X}'_i$ pour $i = 1, \ldots, N$, et notons $\overline{t}^{\,is}_N(A)$ la moyenne des N valeurs $t^{\,is}(A, \mathcal{Q}_i)$ obtenues pour $\mathcal{Q}_i = \mathrm{P}_k(\mathcal{X}'_i)$. Par définition de la règle (R), $\overline{t}^{\,is}_N(A)$ est une mesure de stabilité qui évalue l'isolation de la classe A. Par ailleurs, $\overline{t}^{\,is}_N(A)$ est une estimation sans biais de l'espérance de la variable aléatoire $t^{\,is}(A, \mathcal{Q})$, où $\mathcal{Q}$ désigne la partition $\mathrm{P}_k(\mathcal{X}')$ générée par l'algorithme P_k sur un échantillon aléatoire $\mathcal{X}'$. Une démarche analogue permet de définir des mesures de stabilité qui évaluent d'autres caractéristiques d'une classe, *i.e.* son isolation par rapport à une autre classe, sa cohésion et sa validité. Nous définissons de la même manière des mesures de stabilité évaluant ces caractéristiques (isolation, cohésion et validité), mais pour la partition $\mathcal{P}$. Nous avons montré que quelle que soit la caractéristique considérée, la mesure de stabilité évaluant $\mathcal{P}$, est une moyenne pondérée des mesures de stabilité évaluant les classes de $\mathcal{P}$ pour la même caractéristique.

3. Méthodologie statistique

De façon générique, nous notons $\bar{t}_{\mathcal{X},N}$ l'un quelconque des indices de stabilité défini en section 2. Nous avons $\bar{t}_{\mathcal{X},N} = (t_{\mathcal{X},1} + \ldots + t_{\mathcal{X},N})/N$, où chaque $t_{\mathcal{X},i}$ est la valeur prise par la mesure de Loevinger calculée sur la base de l'information donnée par la partition $\mathrm{P}_k(\mathcal{X}'_i)$ de l'échantillon $\mathcal{X}'_i$ de $\mathcal{X}$. Par exemple, quand $\bar{t}_{\mathcal{X},N}$ désigne la mesure de stabilité de l'isolation de la classe A, on a $\bar{t}_{\mathcal{X},N} = \bar{t}^{\,is}_N(A)$ et $t_{\mathcal{X},i} = t^{\,is}(A, \mathrm{P}_k(\mathcal{X}'_i))$. Chaque $t_{\mathcal{X},i}$ ($i = 1, \ldots, N$) est donc la valeur observée de la mesure de Loevinger, notée par la suite $T_{\mathcal{X}}$, qui mesure la qualité de la règle logique évaluant une caractéristique, comme par exemple l'isolation d'une classe. Donc $T_{\mathcal{X}}$ est une variable aléatoire, puisque ses valeurs dépendent du choix de l'échantillon aléatoire $\mathcal{X}'$, et $\bar{t}_{\mathcal{X},N}$ est un estimateur sans biais de $E(T_{\mathcal{X}})$.

Nous examinerons trois questions méthodologiques. La première est de déterminer un nombre N d'échantillons de $\mathcal{X}$ de telle sorte que $\bar{t}_{\mathcal{X},N}$ estime $E(T_{\mathcal{X}})$ de façon précise et fiable, tout en évitant une trop grande valeur de N qui augmenterait inutilement le temps calcul. Pour cela, nous utilisons l'intervalle de confiance (standard) au niveau de confiance 95% de $E(T_{\mathcal{X}})$ qui est basé sur l'estimation $\bar{t}_{\mathcal{X},N}$. On choisit alors N en respectant la double contrainte suivante : d'une part N doit être assez grand ($N \geq 75$) pour que le théorème Central Limite s'applique, et d'autre part la taille de l'intervalle standard doit être plus petite que le double de la marge d"erreur souhaitée par l'utilisateur. Nous avons par ailleurs remarqué que l'intervalle de confiance "bootstrap studentisé" diffère peu de l'intervalle standard, sauf pour $50 \leq N \leq 100$, auquel cas l'intervalle "bootstrap studentisé" est préférable.

La seconde question concerne l'interprétation des valeurs prises par $\bar{t}_{\mathcal{X},N}$. Jain et Dubes ([JAI 88] p. 144) ont observé qu'il est facile de proposer des indices de validation de classes, mais très difficile de fixer un seuil sur l'indice qui définisse quand cet indice est exceptionellement grand ou petit. Pour résoudre cette difficulté, qui concerne aussi les indices de stabilité, nous suivons la procédure que ces auteurs ont proposée (cf. [JAI 88, GOR 94]), en admettant comme cela semble raisonnable, que l'absence de structure entraîne l'absence de stabilité :

> **Etape 1.** Définir une hypothèse nulle H_0 qui traduit l'absence de structure dans l'ensemble des données étudié ;
> **Etape 2.** Déterminer la distribution de l'indice sous l'hypothèse nulle H_0 ;
> **Etape 3.** Tester l'hypothèse nulle H_0. Le niveau de signification de la valeur observée $\bar{t}_{\mathcal{X},N}$ indique si cette valeur est exceptionellement grande.

Par exemple, si le niveau de signification de la mesure $\bar{t}_{\mathcal{X},N} = 0.899$ par rapport à H_0, est inférieur à 5 ou 10 %, alors on conclut que la valeur 0.899 indique une stabilité élevée. Les niveaux de signification sont estimés par la méthode de Monte-Carlo, tout d'abord en générant des ensembles d'objets distribués uniformément dans l'enveloppe convexe du jeu de données étudié, puis après partitionnement de ces jeux simulés, en estimant la distribution de $E(T_{\mathcal{X}})$ sous H_0.

La dernière question porte sur le choix du nombre de classes. Il est naturel (*cf.* par exemple [LEV 01]) de considérer que le nombre de classes est optimal lorsqu'il maximise (voire localement) la validité de la partition. Ici, ce choix peut être affiné grâce aux informations sur chaque classe (mesures de stabilité de l'isolation et de la cohésion) et de plus, peut être remis en question par les niveaux de signification des mesures de stabilité.

4. Illustration sur des jeux de données

Dans cette section, nous illustrons brièvement notre approche en indiquant les mesures de stabilité de la partition des iris de Fisher en 3 classes, déterminée par la méthode des K-means appliquée avec seulement les deux variables longueur et largeur des pétales. Selon notre expérience sur d'autres jeux de données, et selon les recommandations données dans [BEN 02], nous avons utilisé un taux d'échantillonnage de 0.8. Le tableau 1 ci-dessous indique les mesures de stabilité de la partition en 3 classes des iris de Fisher, obtenu avec ces paramètres. Pour que chacune de ces mesures soient précises, c'est-à-dire proches de leur espérance à ± 0.01, il a été nécessaire pour calculer certaines d'entre elles, de partitionner $N = 671$ échantillons des iris de Fisher. On observe que la classe 3 (qui coincide avec l'ensemble des iris virginica) est très stable aussi bien pour le critère de l'isolation que celui

de la cohésion. Les valeurs de stabilité pour la validité des classes 1 et 2 sont élevées, *i.e.* $.932$ et $.940$, mais ne sont pas significatives, puisque leurs niveaux de signification sont compris entre $24 - 32$ % et $32 - 41$ % (avec une probabilité approximative de 95%). De plus, l'isolation (partielle) des deux classes est plutot faible (.875). Il en résulte que le découpage des données selon les classes 1 et 2, relève plus d'une dissection que d'une classification comprenant deux classes bien séparées. Avec un niveau de signification entre 6.7 et $11.9\,\%$ au seuil de confiance (approximatif) $95\,\%$, la validité globale de la partition en 3 classes peut être considérée comme réelle.

TAB. 1. *Mesures de stabilité de la partition en 3 classes des iris de Fisher (précision ± 0.01). Les niveaux de signification (%) sont estimés par des intervalles de confiance de niveau approximatif 95 %.*

		Isolation		*Cohésion*		*Validité*	
Classe	**1**	.942	*17 – 25 %*	.922	*34 – 44 %*	.932	*24 – 32 %*
	2	.935	*33 – 42 %*	.964	*28 – 37 %*	.940	*32 – 41 %*
	3	1.	*0 – 1 %*	1.	*0 – 1 %*	1.	*0 – 1 %*
Partition		.959	*6.8 – 11.2 %*	.959	*7 – 12 %*	.959	*6.7 – 11.9 %*

Nous avons également appliqué notre approche de validation par stabilité à d'autres partitions : par exemple, les partitions des iris de Fisher à 2, 4 et 5 classes, et des partitions de nombreux jeux de données répartis uniformément dans un carré. Dans tous les cas, les niveaux de signification permettent une meilleure interprétation des mesures de stabilité observées (voir [BER 04] pour plus de détails).

5. Bibliographie

[BAI 82] BAILEY T. A., DUBES R., Cluster validity profiles, *Pattern Recognition*, vol. 15, 1982, p. 61–83.

[BEL 01] BEL MUFTI G., BERTRAND P., Stability of individual clusters, rapport n^{o}0137, 2001, laboratoire Ceremade, Université Paris-Dauphine, Paris, France.

[BEN 02] BEN-HUR A., ELISSEEFF A., I. G., A stability based method for discovering structure in clustered data, *Pacific Symposium on Biocomputing*, 2002, p. 6 –17.

[BER 04] BERTRAND P., BEL MUFTI G., Loevinger's measures of rule quality for assessing cluster stability, rapport, 2004, Dept Tamcic - Lussi, GET - ENST Bretagne, Brest, France.

[CHE 96] CHENG R., MILLIGAN G. W., Measuring the influence of individual data points in a cluster analysis, *Journal of Classification*, vol. 13, 1996, p. 315–335.

[GOR 94] GORDON A. D., Identifying genuine clusters in a classification, *Computational Statistics and Data Analysis*, vol. 18, 1994, p. 561-581.

[GOR 99] GORDON A. D., *Classification*, Chapman & Hall, 1999.

[JAI 88] JAIN A. K., DUBES R., *Algorithms for clustering data*, Prentice-Hall, Englewood Cliffs, NJ., 1988.

[LEN 03] LENCA P., MEYER P., PICOUET P., VAILLANT B., LALLICH S., Critères d'évaluation des mesures de qualité en ECD, *Revue des Nouvelles Technologies de l'Information (Entreposage et Fouille de données)*, vol. 1, 2003, p. 123–134.

[LEV 01] LEVINE E., DOMANY E., Resampling method for unsupervised estimation of cluster validity, *Neural Computation*, vol. 13, 2001, p. 2573 – 2593.

[LOE 47] LOEVINGER J., A systemic approach to the construction and evaluation of tests of ability, *Psychological monographs*, vol. 61, 1947.

[MIL 96] MILLIGAN G. W., Clustering validation : results and implications for applied analyses, ARABIE P., HUBERT L. J., DE SOETE G., Eds., *Clustering and Classification.*, World Scientific Publ., River Edge, NJ, 1996.

[ROB 97] ROBERTS S. J., Parametric and non-parametric unsupervised cluster analysis, *Pattern Recognition*, vol. 30, 1997, p. 261–272.

[TIB 01] TIBSHIRANI R., WALTHER G., BOTSTEIN D., BROWN P., Cluster validation by prediction strength, rapport, 2001, Université de Stanford, Standford, Etats Unis (http ://www-stat.stanford.edu/ tibs/ftp/predstr.pdf).

Liens entre critère métrique et critère probabiliste en classification croisée dans le cas discret

Khemal Bencheikh yamina, Abir Ammar

UFA de Sétif
Faculté des sciences,
Département de mathématiques,
Sétif 19000, Algérie.

RÉSUMÉ. Les méthodes de classification se ramènent souvent à l'optimisation d'un critère numérique défini à partir d'une distance. Dans certain cas, il est possible de montrer que cela revient à estimer les paramètres d'un modèle probabiliste par une approche classification. Dans ce travail nous étendons cette étude au cas de la classification croisée et dans le cas de données discrètes. Pour ceci nous définissons la notion de critère métrique et de critère probabiliste, nous montrons ensuite qu'un critère probabiliste peut toujours être considéré comme un critère métrique et établissons les conditions pour que la réciproque soit vraie. Cette approche nous permet de montrer d'une part, que les critère utilisant la distance L_1 correspondent à un mélange de lois de Bernoulli et d'autre part de proposer de nouveaux critères utilisant des distances adaptatives pouvant améliorer la qualité des résultats.

MOTS-CLÉS : Classification croisée, mélange de lois de probabilité, distance L_1.

1. Introduction

L'une des principales difficultés pour les méthodes de classification automatique est le choix du critère et de la métrique utilisée. Lorsqu'il est possible de trouver un modèle de mélange de lois de probabilités tel que l'estimation des paramètres du modèle par l'approche classification ([SCO 71], [SCH 74], [CEL 88] , [GOV 89], [BEN 92]) conduisent à l'optimisation d'un critère numérique de classification, on obtient un éclairage nouveau de ce critère et de la métrique sous jacente permettant de les justifier ou éventuellement de les rejeter. [GOV 89] s'est intéresser aux liens qui existent entre la classification automatique et les modèles probabilistes lorsque les données mettent en jeu un seul ensemble, [BEN 03] à repris cette étude lorsque les données mettent en jeu deux ensembles et dans le cas continu ; nous proposons de le faire ici lorsque les données sont discrètes. Dans les deux premiers paragraphes, nous définissons deux types de critères et nous étudions dans quelles conditions ces critères peuvent être équivalents. Dans le troisième paragraphe, on fait une application des résultats obtenus aux paragraphes précédents à une famille de critère métrique défini sur des données binaires. Nous retrouvons les liens déjà obtenus entre l'utilisation de la distance L_1 sur les données binaires et les modèles de distributions de Bernoulli ([GOV 90] et [BEN 02]), mais ce travail permet de compléter ces résultas en précisant, par exemple, comment justifier certains choix de pondérations.

2. Définition des deux types de critères

On suppose dans tout ce qui suit que les données initiales sont fournies sous la forme d'un tableau X de n lignes et p colonnes contenant les valeurs prises par n individus pour p variables, ces valeurs seront notées x_i^j , i=1, …, n et j=1, …, p et appartiennent à un ensemble E fini.

Ceci se vérifie lorsque les données sont binaire ou qualitatives. Par exemple, dans le cas de données binaires, l'espace E est l'ensemble {0, 1}. On peut sans difficulté étendre les définitions et propriétés établies lorsque les données étaient continues [BEN 03].

2.1. *Critère métrique*

Il s'agit de trouver une partition (P_1 ,…, P_k) de l'ensemble **I** des individus en K classes, une partition (Q^1, …, Q^M) de l'ensemble **J** des variables en M classes et un K.M-uple (λ_k^m) ; k = 1, …, K et m = 1, …, M (un par classe) minimisant le critère suivant :

$$\boldsymbol{W}(P \times Q, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} D(x_i^j , \lambda_k^m)$$

Ce critère qui dépend de la mesure de dissimilarité D sera appelé critère métrique et noté **CM**(**E**,**L**,**D**).

L =$\{ \lambda_k^m$, k = 1,…, K et m = 1, …, M$\}$.

2.2. *Critères métriques équivalents*

On dira que deux critères métriques sont équivalents si et seulement s'ils sont définis sur les mêmes ensembles E et L et s'il existent une bijection ϕ sur R strictement croissante vérifiant :

$$\mathbf{CM}(\mathbf{E},\mathbf{L}, \mathbf{D_1}) = \phi \circ \mathbf{CM}(\mathbf{E},\mathbf{L},\mathbf{D_2}).$$

où $\mathbf{D_1}$ et $\mathbf{D_2}$ sont les mesures de dissimilarité associées aux deux critères.

Proposition 1 :

$\forall\ \alpha \in R^{+*}$ et $\beta \in$ R les critères **CM**(**E**,**L**, **D**) et **CM**(**E**,**L**, α **D** + β) sont équivalents

2.3. *Critère probabiliste*

On reprend ici la représentation de [BEN 99].

Il s'agit de rechercher une partition P × Q = { $P_k \times Q^m$, k=1,…, K et m=1,…, M }, K et M étant supposés connus, telle que chaque classe $P_k \times Q^m$ soit assimilable à un sous-échantillon qui suit une distribution $p(x,\lambda_k^m)$. Il s'agit alors de maximiser le critère de vraisemblance classifiante suivant :

$$\mathbf{VC}(P \times Q, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \text{Log R} (P_k \times Q^m , \lambda_k^m)$$

où L est le K.M-uple (λ_k^m , k = 1, …, K et m = 1, …, M) et R($P_k \times Q^m$, λ_k^m) = $\prod_{x \in P_k \times Q^m} p(x,\lambda_k^m)$ est la vraisemblance que le sous-échantillon $P_k \times Q^m$ provient de la distribution $p(x,\lambda_k^m)$.

Ce critère qui dépend de la famille F de distributions de probabilités définies sur E sera appelé critère probabiliste et noté **CP**(**E**, **F**).

3. Liens entre les deux types de critères

3.1. *Critère métrique associé à un critère probabiliste*

Proposition 2 : **CP**(**E**, **F**) = - **CM**(**E**,**L**, **D**)

où L est l'ensemble de définition des paramètres de la famille F et D est définie par :

$$\forall\ x \in E,\quad \forall\ \lambda \in L \quad D(x, \lambda) = -\operatorname{Log}\ p(x,\lambda)$$

Le critère métrique ainsi défini à partir d'un critère probabiliste est appelé critère métrique associé.

3.2. *Conditions pour qu'un critère métrique soit associé à un critère probabiliste*

Proposition 3 : Un critère métrique **CM**(**E**,**L**, **D**) est associé à un critère probabiliste si et seulement si :

$$\forall\ \lambda \in L \qquad \sum_{x \in E} e^{-D(x,\lambda)} = 1.$$

3.3. *Critère probabiliste équivalent à un critère métrique*

Proposition 4 :

Etant donné le critère métrique **CM**(**E**,**L**, **D**), s'il existe un réel r $\succ$ 1 tel que la quantité :

s = $\sum_{x \in E} r^{-D(x,\lambda)}$ soit indépendante de λ, alors le critère probabiliste **CP**(**E**, **F**) où F est défini par les distributions de probabilité $p(x,\lambda) = \frac{1}{s}\ r^{-D(x,\lambda)}$ est un critère équivalent.

4. Application : Métrique L_1 et distribution de Bernoulli

Nous allons étudier un certain nombre de critères issus de la distance L_1 et définis sur un ensemble de données binaires.

4.1. *Distance L_1 fixe et identique pour toutes les classes*

Considérons E = {0, 1} et l'ensemble des noyaux coincide avec E. Soit la distance suivante :

$$\forall\ x \text{ et } \lambda_k^m \in E \qquad D(x, \lambda_k^m) = \alpha\ |\ x - \lambda_k^m\ | + \beta$$

où α est une constante réelle positive et β un réel quelconque. Puisque tous les critères métriques sont équivalents (proposition 1), on se limite alors à l'étude du critère métrique défini à l'aide de la distance suivante :

$\forall\ x$ et $\lambda_k^m \in E \quad \forall\ \alpha \in R^{+*} \quad D(x, \lambda_k^m) = \alpha\ |\ x - \lambda_k^m\ |$.

En appliquant la proposition 3 et en posant r = e, on obtient s = $\sum_{x \in E} e^{-\alpha|x-\lambda_k^m|} = 1 + e^{-\alpha}$ quantité indépendante de λ_k^m ; d'où $p(x,\lambda_k^m) = \frac{1}{1+e^{-\alpha}}\ e^{-\alpha|x-\lambda_k^m|} = \varepsilon^{1-|x-\lambda_k^m|}(1-\varepsilon)^{|x-\lambda_k^m|}$

avec $\varepsilon = \frac{1}{1+e^{-\alpha}} \in]0, \frac{1}{2}[$, $x \in \{0, 1\}$ et $\lambda_k^m \in \{0, 1\}$.

Cette expression correspond à une des deux lois de Bernoulli suivante :

1 avec la probabilité ε et 0 avec la probabilité 1-ε ou 1 avec la probabilité 1-ε et 0 avec la probabilité ε. Ce mélange de loi de Bernoulli dépend du paramètre ε celui-ci mesure l'écart d'une classe à son centre et ne dépend ni des variables ni des classes, ce qui dans certaines situations peut s'avérer irréaliste, pour cela nous proposons une métrique plus générale permettant de varier le paramètre ε suivant les partitions en lignes et en colonnes.

4.2. *Distance L_1 variable et dépendante de chaque classe*

Les noyaux sont de la forme $\lambda_k^m = (\alpha_k^m, \beta_k^m, \gamma_k^m)$ et la métrique L_1 dépend de chaque classe $P_k \times Q^m$ qui est défini par :

$\forall\ x$ et $\beta_k^m \in E\ \forall\ \alpha_k^m \in R^{+*} \quad D(x, (\alpha_k^m, \beta_k^m, \gamma_k^m)) = \alpha_k^m\ |\ x - \beta_k^m\ | + \gamma_k^m$ et γ_k^m quelconque.

Si s #1, il n'existe pas de critère probabiliste équivalent car s dépend de λ_k^m.

Si s = 1 alors $\gamma_k^m = \text{Log}(1+e^{-\alpha_k^m})$, le critère métrique définie par la métrique ci-dessus est associé à un critère probabiliste dont la distribution de probabilité est : $p(x,\lambda_k^m) = \frac{1}{1+e^{-\alpha_k^m}} e^{-\alpha_k^m \left| x-\beta_k^m \right|}$

Cette distribution peut encore s'écrire sous la forme suivante :

$$p(x,\lambda_k^m) = (\varepsilon_k^m)^{1-\left|x-\beta_k^m\right|} (1-\varepsilon_k^m)^{\left|x-\beta_k^m\right|} \quad \text{avec} \quad \varepsilon_k^m = \frac{1}{1+e^{-\alpha_k^m}} \in]0, \frac{1}{2}[.$$

Cette fois-ci le paramètre ε_k^m dépend de chaque classe $P_k \times Q^m$; il s'agit alors d'un mélange de lois de Bernoulli de paramètre ε_k^m ou de paramètre 1- ε_k^m .

On retrouve alors la méthode la plus générale que l'on avait développée dans l'approche des modèle de mélange de Bernoulli ([BEN 02]) où l'on a pu comparer les algorithmes classiques utilisant une métrique fixe et identique pour toutes les classes et les algorithmes utilisant des distances adaptatives qui s'adaptent à chaque itération aux différentes classes, ces essais ont montré l'intérêt d'utiliser des métriques qui dépendent des variables et des classes dans le cas de la classification simple et des métriques qui dépendent des classes en lignes et en colonnes dans le cas de la classification croisée, comme c'est le cas dans cette étude.

5. Bibliographie

[BEN 92] BENCHEIKH Y., *Classification automatique et modèles*, Thèse de Doctorat, Université de Metz, 1992.

[BEN 99] BENCHEIKH Y., *Classification croisée et modèles*, Rairo operations reseach, vol. 33, 1999, p. 525-541.

[BEN 02] BENCHEIKH Y., *Classification croisée et distance L_1 adaptative*, Revue de statistique appliquée, vol. 03, 2002, p. 53-72.

[BEN 03] KHEMAL BENCHEIKH Y., *Liens entre critère métrique et critère probabiliste en classification croisée dans le cas continu* , Actes de la 10ème rencontre de la société francophone de classification SFC03, p. 149-152, Neuchâtel Suisse, 2003.

[CEL 88] CELEUX G., *Classification et modèles,* , Revue de statistique appliquée, vol. 04, 1988, p. 43-58.

[GOV 89] GOVAERT G., " *Modèle de classification et distance dans le cas continu* ", Rapport n° 988, 1989, rapport de recherche, INRIA de Paris.

[GOV 90] GOVAERT G., " *Classification binaire et modèles* ", Revue de statistique appliquée, vol. 38, 1990, p. 67-81.

[SCH 74] SCHROEDER A., " *Reconnaissance des composants d'un mélange* ", Thèse de Doctorat 3ème cycle, Université Paris 6, 1974.

[SCO 71] SCOTT ., SYMONS " *Clustering methods based on likelihood ratio criteria* ", Biometrics, vol. 27, 1971, p. 387-397..

L'Enumération Optimisée sur 2 Séquences Proportionnelles

Farid BENINEL, Serge SABOURIN

IUT- STID Pôle Universitaire Niortais
8, rue Archimède, 79000 Niort, FRANCE

RÉSUMÉ. *Etant donnée une séquence réelle finie $\{a_i : i = 1, ..., n\}$ et un réel $\lambda \in [0, 1]$, on s'intéresse au calcul des quantités $K_{\lambda,a}(k, m, Z/0)$ donnant le nombre de combinaisons de somme associée inférieure ou égale à $Z \in \mathbb{R}$; ces combinaisons sont formées de k composantes de a et m composantes de λa, distinctes. Cette quantité intervient notamment dans le calcul de $p - value$ exacte associée à des indices d'association dont la statistique associée est une combinaison linéaire de variables à 3 modalités. L'algorithme proposé se base sur une relation de récurrence donnant $K_{\lambda,a}(k, m, Z/0)$. Il présente les avantages de son autonomie par rapport à d'autres logiciels numériques, de convenir aux différents liens de dépendance entre variables aléatoires intervenant dans la statistique étudiée et de tirer partie des propriétés intrinsèques de la séquence de scores.*

MOTS-CLÉS : *Concordances des réponses, Enumération optimisée, Indice d'association, Statistique permutationnelle, p-value exacte.*

1. Motivation

L'étude de certains indices d'association [BEN] conduit à l'étude des statistiques

$$T_{\lambda,a}(Z_1, ..., Z_n) = \sum_{i=1}^{n} a_i Z_i,$$

avec $\lambda \in IR$, , $a \in IR^n$ les paramètres déterministes et $Z_1, ..., Z_n$ les variables aléatoires associées à cette statistique. Ces variables sont identiquement distribuées et à valeurs dans$\{1, \lambda, 0\}$.

Etant donnée une hypothèse nulle H_0 on s'intéresse, dans ce travail, au calcul de $p - value(T_{\lambda,a}, t/H_0)$ la p-value exacte associée à une valeur t de la statistique$T_{\lambda,a}$ sous cette hypothèse.

Ici $p - value(T_{\lambda,a}, t/H_0) = \frac{1}{2} Min(P_{H_0}(T_{\lambda,a} \leq t),\ P_{H_0}(T_{\lambda,a} \geq t))$ ou $P_{H_0}(T_{\lambda,a} \leq t)$ ou $P_{H_0}(T_{\lambda,a} \geq t)$ selon que l'on rejette H_0 pour les petites et grandes valeurs ou les petites valeurs ou les grandes valeurs de $T_{\lambda,a}$.

D'autres façons de définir la p-value sont données dans [GIB 75].

Posons $N_1 = card\{i \in \{1, ..., n\}\ : Z_i = 1\}, N_\lambda = card\{i \in \{1, ..., n\} : Z_i = \lambda\}$.

L'hypothèse H_0 considérée dans ce travail est que le *n*-uple $(Z_1, ..., Z_n)$ est uniformément distribué conditionnellement aux valeurs de (N_1, N_λ).

Dans le cas où $(N_1, N_\lambda) = (k, m)$, les réalisations de $T_{\lambda,a}(Z_1, ..., Z_n)$ sont de la forme $\sum_{j=1}^{k} a(i_j) + \lambda \sum_{j=k+1}^{k+m} a(i_j)$ avec $(i_1, ..., i_{k+m}) \subset \{1, ..., n\}$.

Posons $E_{\lambda,a}(k, m, Z/j) = \{(i_1, ..., i_{k+m}) \subset \{j+1, ..., n\} : \sum_{j+1}^{j+k} a(i) + \lambda \sum_{j+k+1}^{j+k+m} a(i) \leq Z\}$ et $K_{\lambda,a}(k, m, Z\ /\ j) = card(E_{\lambda,a}(k, m, Z\ /\ j))$.

La décomposition de $P_{H_0}(T_{\lambda,a} \leq t)$ en une somme de probabilités conditionnellement aux valeurs du couple (N_1, N_λ) donne

$$P_{H_0}(T_{\lambda,\omega} \leq t) = \sum_{k,l} \mu_{H_0}(k,l) K_{\lambda,a}(k,m,t/0),$$

avec

$\cdot \mu_{H_0}(k,l) = P_{H_0}((N_1, N_\lambda) = (k,l))/(\binom{n}{k}\binom{n-k}{l})$,

$\cdot\ K_{\lambda,a}(k,m,t\ /\ 0)$ le nombre de cas favorables à l'évènement $\{T_{\lambda,\omega} \leq t\}$

sachant $\{(N_1, N_\lambda) = (k,m)\}$.

La distribution du couple (N_1, N_λ) est supposée connue et par suite, les valeurs $\mu_{H_0}(k,l)$ le sont aussi.

Ainsi, l'étude est restreinte à la calculabilité des quantités $K_{\lambda,a}(k,m,t\ /\ 0)$ et plus généralement à la calculabilité des quantités $K_{\lambda,a}(k,m,t\ /\ j)$ avec $j = 0, ..., (n-k-m+1)$.

La suite est dévolue aux résultats combinatoires permettant le calcul des quantités $K_{\lambda,a}(k,m,t\ /\ j)$, l'optimisation du temps de calcul et aux simulations nécessaires à la validation de l'algorithme de calcul.

2. Combinatoire

Dans [BEN 99] on donne un algorithme permettant le calcul des quantités $K_{\lambda,a}(k,0,Z\ /\ j)$, $K_{\lambda,a}(0,m,Z\ /\ j)$.

Plus précisément, cet algorithme permet, étant donnée une séquence $S = (s_1, ..., s_n)$, des entiers l , j et un réel Z , de calculer la quantité $N_S(l, Z\ /\ j) = card\{(i_1, ..., i_l) \subset \{j+1, ..., n\} : \sum_1^l s_{i_k} \leq Z\}$.

2.1. *Lien entre* $N_{\lambda a}$, N_a *et* $K_{\lambda,a}$

Le nombre de combinaisons issues de 2 séquences et le nombre de combinaisons de l'une ou de l'autre sont liés par les équations ci aprés :

$$K_{\lambda,a}(k,0,Z\ /\ j) = N_a(k,Z\ /\ j)\ , \tag{1}$$

$$K_{\lambda,a}(0,k,Z\ /\ j) = N_{\lambda a}(k,Z\ /\ j). \tag{2}$$

Et de manière à utiliser, dans l'algorithme, une seule des 2 fonctions $N_a, N_{\lambda a}$, on a, pour tout $\lambda > 0$

$$N_{\lambda a}(k,Z\ /\ j) = N_a(k,\frac{Z}{\lambda}\ /\ j)\ . \tag{3}$$

2.2. *Inégalités*

Soient $j,k,l \in IN$ tels que $k+l+j \leqslant n$ et $Z \in IR$,

$$N_a(k+l,Z\ /\ j) \leq K_{\lambda,a}(k,l,Z\ /\ j)/\binom{k+l}{k} \leq N_{\lambda a}(k+l,Z\ /\ j). \tag{4}$$

La double inégalité, ci-dessus, permet de majorer ou de minorer la $p-value$ qui nous intéresse ; ainsi pour détecter les valeurs extrêmes, le seul usage de l'algorithme concernant une unique séquence suffit.

2.3. *Cas général*

Posons $\sum_{\lambda,a}(k,m\ /\ j)$ l'ensemble des sommes associées aux combinaisons de $E_{\lambda,a}(k,m,\infty\ /\ j)$; désignons par $S^{+}_{\lambda,a}(k,m\ /\ j),\ S^{-}_{\lambda,a}(k,m\ /\ j)$ respectivement la somme maximale et la somme minimale associée.

$$K_{\lambda,a}(k,m,Z\ /\ j)=\begin{cases} C^{k}_{n-j}C^{m}_{n-j-k} & si\ \ Z\ \geq\ S^{+}_{\lambda,a}(k,m\ /\ j); \\ 0 & si\ \ Z\ <\ S^{-}_{\lambda,a}(k,m\ /\ j). \end{cases} \quad (5)$$

La relation de récurrence, sur laquelle se base l'algorithme *A.E.O*, est donnée par

$$\text{K}_{\lambda,a}\textit{(k,l,Z/0)}=\sum_{j=1}^{n-k-l+1} K_{\lambda,a}\textit{(k-1,l,Z-a(j)/j)}+K_{\lambda,a}(k,l-1,Z-\lambda a(j)/j) \quad (6)$$

et l'on convient que $K_{\lambda,\omega}(0,0,Z/j)\ =\ 1$; cela vient du fait que l'unique combinaison constituée de zéro éléments est l'ensemble vide. Cette relation de récurrence peut être réduite. En effet posons :

$M_1(k,l,Z)=Max\{m\in\{1,...,n\}:S^{+}_{\lambda,a}(k-1,l/m)\leq Z-a(m)\}$,

$m_1(k,l,Z)=Min\{m\in\{1,...,n\}:S^{-}_{\lambda,a}(k-1,l/m)>Z-a(m)\}$,

$M_\lambda(k,l,Z)=Max\{m\in\{1,...,n\}:S^{+}_{\lambda,a}(k,l-1/m)\leq Z-\lambda a(m)\}$,

$m_\lambda(k,l,Z)=Min\{m\in\{1,...,n\}:S^{-}_{\lambda,a}(k,l-1/m)>Z-\lambda a(m)\}$.

Et pour simplifier les écritures, posons pour $M,m\in\{1,...,n\}$ avec $M\leqslant m$,

$C(M,k,l,z)=\sum_{m=1}^{M}\binom{n-m}{k-1}\binom{n-m-k+1}{l}$,

$C(k,l,z)=C(M_1,k,l,z)+C(M_\lambda,k,l,z)$,

$q_1(m,M,k,l,Z)=\sum_{j=M+1}^{m-1}K_{\lambda,a}(k-1,l,Z-a(j)/j)$,

$q_2(m,M,k,l,Z)=\sum_{j=M+1}^{m-1}K_{\lambda,a}(k,l-1,Z-\lambda a(j)/j)$.

En conséquence de l'équation (5),on a

$$K_{\lambda,a}(k,m,Z\ /\ 0)=C(k,l,n)+q_1(m_1,M_1,k,l,Z)+q_2(m_\lambda,M_\lambda,k,l,Z). \quad (7)$$

3. Simulations

Les simulations consistent pour un grand nombre de séquences a, générées au hasard, et de valeurs λ, à observer, pour différentes valeurs de k,m, le temps de calcul de $K_{\lambda,a}(k,m,Z/0)$.[CAS 96]

Les valeurs Z envisagées sont réparties uniformément dans l'intervalle $\left[S^{-}_{\lambda,a}-\delta,S^{+}_{\lambda,a}+\delta\right]$ avec $\delta>0$ assez petit.

4. Bibliographie

[BEN] BENINEL F., GRUN-RÉHOMME M., O. V., A survey quality measure, *Soumis*.

[BEN 99] BENINEL F., HUSSON F., An optimized algorithm to determine values of the exact cumulative distribution function of some discrete statistics., *Comp. Stat, Springer Verlag, Berlin*, vol. 14,2, 1999, p. 251-261.

[CAS 96] CASTAGLIOLA P., An optimized algorithm for computing Wilcoxon's statistic when N is small., *Comp. Stat, Springer Verlag, Berlin*, vol. 11,2, 1996, p. 1-10.

[GIB 75] GIBBONS J., PRATT J., P-values : interpretation and methodology, *American Statistician*, vol. 29, 1975, p. 20-25.

Intervalles conservés et Neighbor-Joining

Anne Bergeron[1]**, Mathieu Blanchette**[2] **et Cedric Chauve**[1]

(1)LaCIM et Département d'Informatique, Université du Québec à Montréal, Canada.
[anne,chauve]@lacim.uqam.ca
(2)McGill Center for Bioinformatics et School of Computer Science, McGill University, Montréal, Canada.
blanchem@mcb.mcgill.ca

RÉSUMÉ. *Nous présentons dans cette note une application de la notion d'intervalles conservés (introduite par Bergeron et Stoye, [BER 03]) dans le cadre de la phylogénie, et plus particulièrement de l'algorithme Neighbor-Joining utilisé avec des données basées sur l'ordre des gènes. Nous proposons notamment une modification de cet algorithme permettant d'obtenir des informations sur de possibles génomes ancestraux. Nous étudions la pertinence de ces résultats dans le cadre d'un jeu de données constitué de génomes de mitochondries de Métazoaires, déjà considéré par Blanchette et al. [BLA 99] et par Bourque et Pevzner [BOU 02].*

MOTS-CLÉS : *Phylogénies, Ordre des gènes, Intervalles conservés, Neighbor-Joining, Mitochondries, Métazoaires.*

1. Introduction

Dans cet article, nous nous intéressons à la reconstruction de phylogénies basée sur l'étude de l'ordre des gènes. Cette approche, qui repose entre autres sur la modélisation de génomes par des permutations signées, a connu de nombreux développements récents [MOR 02, SAN 98, BOU 02] (voir aussi [SAN 00] pour une revue plus large des problèmes reliés à l'étude de l'ordre des gènes). Les algorithmes proposés dans ce cadre reposent sur le calcul du médian de trois génomes pour une distance donnée, basée sur la notion de réarrangements génomiques, (distance de breakpoints ou d'inversions, avec les méthodes BPAnalysis [SAN 98], GRAPPA [MOR 02], MGR-MEDIAN [BOU 02]). Ce problème du médian est NP-complet mais peut être résolu de manière non exacte à l'aide d'heuristiques efficaces, ce qui permet en particulier de proposer des génomes ancestraux des organismes étudiés, c'est-à-dire des génomes potentiels pour les nœuds internes de l'arbre phylogénétique reconstruit. Cette capacité à reconstruire un génome ancestral est un des points forts des méthodes basées sur l'ordre des gènes.

Récemment, Bergeron et Stoye [BER 03] ont introduit le concept d'*ensemble d'intervalles conservés* (Section 2) d'un ensemble de génomes représentés par des permutations signées, qui permet de décrire de manière compacte certains groupes de gènes qui sont consécutifs dans tous ces génomes, groupes de gènes qu'il est tentant de considérer comme des invariants, en termes d'ordre des gènes, provenant d'ancêtres communs. L'intérêt de cette notion repose entre autres sur le fait que, le nombre de scenarios d'évolution les plus parcimonieux, en termes de réarrangements génomiques, entre deux génomes pouvant être exponentiel en la taille de ces génomes [BER 02], le nombre de génomes ancestraux possibles est très grand, et le choix de l'un d'entre eux en particulier devient alors un choix arbitraire. De son côté, la notion d'intervalles conservés permet de représenter un ensemble de génomes ancestraux sans avoir à en exhiber un en particulier.

Le but de notre travail est d'effectuer une première exploration de l'utilisation de la notion d'intervalles conservés dans le cadre du calcul de phylogénies basées sur l'ordre des gènes. Plus précisément, nous proposons une adaptation de l'algorithme Neighbor-Joining, et en fait de tout algorithme de clustering hiérarchique, basée sur les intervalles conservés, permettant donc d'étudier des génomes décrits par l'ordre de leurs gènes et de déterminer des informations intéressantes sur leurs génomes ancestraux (Section 3). Nous utilisons notre méthode

sur un jeu de données de génomes de mitochondries de Métazoaires étudié par Blanchette et al. [BLA 99], ainsi que par Bourque et Pevzner [BOU 02] (Section 4). Nous obtenons des résultats qui sont cohérents avec ceux de [BLA 99, BOU 02], notamment en termes des génomes ancestraux proposés, tout en permettant d'apporter certaines précisions et corrections sur la structure de ces génomes et des scenarios d'évolution proposés.

2. Génomes, permutations signées et intervalles conservés

Soient $\mathcal{A}$ un ensemble de n gènes, et $\mathcal{G}$ un ensemble de génomes sur $\mathcal{A}$ sans répétition de gène : dans chaque génome, chaque gène apparaît une et une seule fois, et est signé $+$ ou $-$ dépendant du brin d'ADN sur lequel il est exprimé. Nous utilisons la représentation classique d'un tel ensemble de génomes par un ensemble de permutations signées sur $\{1, \ldots, n\}$, chaque gène $\mathcal{A}$ correspondant à un élément de $\{1, \ldots, n\}$. Voici par exemple, un ensemble de 3 génomes sur 10 gènes.

$$\begin{array}{lcccccccccc} G_1 = & +1 & -2 & +3 & +4 & +5 & +6 & -7 & -8 & +9 & +10 \\ G_2 = & +1 & -5 & +2 & -4 & -3 & +6 & +7 & -8 & +9 & +10 \\ G_3 = & +1 & -5 & +2 & -4 & -3 & +6 & +8 & -7 & +9 & +10 \end{array}$$

La notion d'intervalles conservés d'un ensemble de permutations signées a été introduite récemment par Bergeron et Stoye [BER 03] comme une mesure de similarité entre un ensemble de génomes. Soit $\mathcal{G}$ un ensemble de permutations signées sur $\{1, \ldots, n\}$. Un intervalle $[a, b]$ (où a et b sont deux entiers signés sur $\{1, \ldots, n\}$) est un intervalle conservé de $\mathcal{G}$ si les conditions suivantes sont vérifiées :

1. dans toute permutation de $\mathcal{G}$, soit a précède b, soit $-b$ précède $-a$,
2. les éléments situés entre a et b (ou $-b$ et $-a$) sont les mêmes, au signe près, dans toute permutation de $\mathcal{G}$.

On note $IC(\mathcal{G})$ l'ensemble des intervalles conservés de $\mathcal{G}$: $IC(\{G_1, G_2, G_3\}) = \{[1, 6], [3, 4], [6, 10], [6, 9], [9, 10], [1, 10]\}$.

Intuitivement, on peut appréhender l'ensemble des intervalles conservés de $\mathcal{G}$ comme l'ensemble des groupes de gènes qui ne devraient pas être séparés par un scénario d'évolution plausible pour ce groupe de génomes, et qui devraient donc se retrouver ensembles dans les génomes des ancêtres communs aux organismes considérés. $IC(\mathcal{G})$ peut aussi être vu comme une représentation de l'ensemble des permutations signées respectant ces intervalles, c'est-à-dire l'ensemble des ancêtres potentiels possibles vis-à-vis d'un scenario d'évolution, pour les génomes de $\mathcal{G}$, ne cassant aucun de ces intervalles.

Cette mesure de similarité entre génomes induit une distance entre deux ensembles de permutations signées : la distance entre $\mathcal{G}_1$ et $\mathcal{G}_2$, notée $d(\mathcal{G}_1, \mathcal{G}_2)$, est donnée par $|IC(\mathcal{G}_1)| + |IC(\mathcal{G}_2)| - 2|IC(\mathcal{G}_1 \cup \mathcal{G}_2)|$. Bergeron et Stoye [BER 03] ont proposé un algorithme linéaire en temps permettant, étant donnés deux ensembles d'intervalles conservés $\mathcal{I}_1$ et $\mathcal{I}_2$, correspondant à deux ensembles de permutations $\mathcal{G}_1$ et $\mathcal{G}_2$, de calculer l'union de ces deux ensembles d'intervalles conservés (c'est-à-dire l'ensemble d'intervalles conservés de $\mathcal{G}_1 \cup \mathcal{G}_2$).

3. Neighbor-Joining et intervalles conservés

Le principe général des algorithmes de clustering hiérarchiques basés sur l'analyse de matrices de distances est le suivant (voir [FEL 03] pour une description précise de plusieurs méthodes, incluant Neighbor-Joining) :

1. la matrice de distance est indexée par un ensemble de groupes de taxa, chaque groupe étant structuré en arbre phylogénétique, et au départ chaque groupe est réduit à un seul taxon,

2. deux groupes sont choisis, selon un critère numérique calculable à partir de la matrice de distances et variant selon les algorithmes, pour être fusionnés en un seul groupe,

3. puis les deux groupes choisis sont supprimés de la matrice et remplacés par le nouveau groupe créé, dont les distances aux autres groupes sont calculées, selon une méthode variant selon les algorithmes, et l'on retourne à l'étape 2 si la matrice comporte plus de deux groupes.

La caractéristique de base de notre algorithme consiste à utiliser la distance d'intervalles entre groupes de taxa (génomes).

La première innovation concerne l'étape 2 du schéma algorithmique décrit ci-dessus. Elle consiste à associer à chaque groupe de taxa $\mathcal{G}$ utilisé pour indexer la matrice de distance, c'est-à-dire à chaque nœud sommet de l'arbre phylogénétique calculé, l'ensemble des intervalles conservés des génomes de ce groupe, à savoir $IC(\mathcal{G})$. Ainsi, à la différence des versions classiques des algorithmes de Neighbor-Joining, chaque nœud interne N de l'arbre phylogénétique contient un codage compact de l'ensemble des génomes ancestraux possibles, pour les taxa $\mathcal{G}_N$ contenus dans le sous-arbre de racine N, compatibles avec les intervalles conservés des génomes de $\mathcal{G}_N$.

La seconde innovation consiste, lors de l'étape 3, à calculer la distance du nouveau groupe créé, notons-le $\mathcal{G}_N$, à tout groupe restant dans la matrice, par exemple un groupe $\mathcal{G}_M$, de manière non purement numérique, c'est-à-dire en calculant une moyenne de la distance à $\mathcal{G}_M$ des taxa présents dans $\mathcal{G}_N$, mais en calculant la distance d'intervalles entre les ensembles d'intervalles conservés $IC(\mathcal{G}_N)$ et $IC(\mathcal{G}_M)$.

Ces deux innovations induisent une augmentation, par rapport aux versions classiques de Neighbor-Joining, de la complexité en temps et en espace du calcul d'un arbre phylogénétique pour m taxa décrits par des génomes sur n gènes : l'étape 3 demande d'effectuer le calcul de l'union d'intervalles conservés, ce qui induit une augmentation du temps de calcul d'un facteur $O(n)$, et le stockage sur chaque nœud interne d'un ensemble d'intervalles conservés demande un accroissement de l'espace nécessaire au stockage d'un sommet en $O(n)$. Cependant, l'algorithme proposé reste polynomial en temps de calcul ($O(n^3m)$) et en espace ($O(m(mn))$).

4. Application aux génomes de mitochondries de Métazoaires

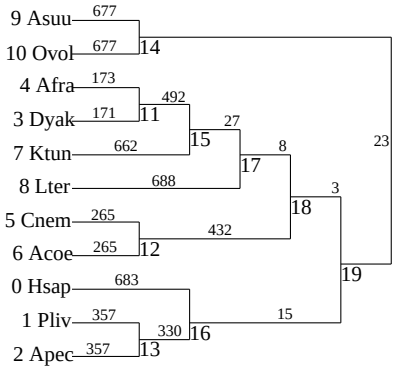
Nous avons appliqué notre méthode avec l'algorithme Neighbor-Joining (tel que décrit dans [FEL 03, p. 167]) à l'ensemble de génomes de mitochondries de Métazoaires étudié par Blanchette et al. [BLA 99] et par Bourque et Pevzner [BOU 02].

Ces données se composent des organismes suivants : *Homo sapiens*, (Hsap, groupe des Chordés), *Asterina pectinifera* et *Paracentrotus lividus* (Apec et Pliv, groupe des Echinodermes), *Drosophila yakuba* et *Artemia franciscana* (Dyak et Afra, groupe des Arthropodes), *Albinaria coerulea*, *Cepaea nemoralis* et *Katharina tunicata* (Acoe, Cnem et Ktun, groupe des Mollusques), *Lumbricus terrestris* (Lter, groupe des Annelida), *Ascaris suum* et *Onchocera volvulus* (Asuu et Ovol, groupe des Nematodes). Pour chacun de ces organismes, suivant [BOU 02], nous avons considéré leurs génomes mitochondriaux, dont l'ordre des gènes est disponible dans la base de données *Mitochondrial Gene Arrangement Guide 6.0* maintenue par Jeffrey L. Boore. Ces génomes partagent essentiellement le même ensemble de 36 gènes (les gènes `atp8` et `UNK` ont été supprimés des génomes les contenant car ils ne sont pas présents pas dans tous les génomes).

L'arbre obtenu est décrit en Figure 1. On peut notamment remarquer qu'à l'exception du taxon *Katharina tunicata*, dont la "mobilité" posait aussi problème dans [BLA 99, BOU 02], tous les clades importants ont été reconstruits, et notamment les deusterome (Chordés et Echinodermes), réunion que les méthodes Neighbor-Joining et Fitch classiques basées sur les distances d'inversions ou de breakpoints ne retrouvaient pas [BLA 99]. Ce point important semble indiquer que la distance d'intervalles est une bonne mesure de distance d'évolution en termes de réarrangements génomiques. De plus, les branchements différents entre l'arbre que nous exhibons et les arbres produits par [BLA 99, BOU 02], notamment la position du groupe des Arthropodes, ne concernent que des branchements correspondant à de courtes branches, à la fois dans notre arbre et dans ceux de [BLA 99, BOU 02], ce qui incite à relativiser ces différences.

Concernant les ensembles d'intervalles conservés étiquetant les nœuds internes de cet arbres (les "génomes ancestraux"), par manque de place nous ne pouvons les inclure dans cette note, mais ils sont disponibles à l'url `www.lacim.uqam.ca/~chauve/SFC04/`. On peut faire les constatations suivantes. Premièrement, les ensembles d'intervalles conservés assignés à chacun des ancêtres des groupes Echinodermes, deusteuromes (Echinodermes et Chordés), Arthropodes, Arthropodes+Ktun, Arthropodes+Mollusques+Annelidés, et Nematodes n'indiquent aucune contradiction avec les génomes ancestraux proposés dans [BLA 99, BOU 02], bien que nous considérions

FIG. 1. *Arbre obtenu par Neighbor-Joining et intervalles conservés, avec longueur de branches.*



dans notre étude l'ensemble des 36 gènes communs aux 11 génomes à la différence de [BLA 99] qui ne prenaient pas en compte les ARN de transfert. De plus, les structures d'intervalles conservés proposés permettent d'affiner l'information sur ces génomes ancestraux en précisant quels groupes de gènes ont évolué ensemble, par exemple le groupe (cox2, K, atp6, cox3) chez les deusteuromes. Deuxièmement, les intervalles conservés de l'ancêtre immédiat (sommet 12) des taxa Acoe et Cnem (Mollusques privé de Ktun) conservent l'ARN de transfert P dans un groupe contenant les gènes A (tRNA-Ala) et nad6 et situé en début de génome, alors que MGR-MEDIAN rejette ce gène P en fin du génome ancestral, ce qui semble en contradiction avec la structure des génomes de Acoe et Cnem, dans lesquels P est en début de génome, proche de A et nad6. La solution proposée par notre algorithme, qui ne repose pas sur l'optimisation d'un scénario en termes de réarrangements mais sur une mesure de similarité, semble alors plus logique du point de vue de la structure de génomes de Acoe et Cnem.

En conclusion, cette étude préliminaire semble indiquer que les notions d'intervalles conservés et de distance d'intervalles sont pertinentes dans le cadre de l'inférence de phylogénies basées sur l'ordre des gènes, notamment pour la reconstruction de génomes ancestraux, et son étude devrait être étendue à des méthodes autres que Neighbor-Joining.

5. Bibliographie

[BER 02] BERGERON A. C. C. H. T., STONGE K., Properties of Sequences of Reversals that Sort a Signed Permutations, *Proceedings of JOBIM 2002*, 2002, p. 99-108.

[BER 03] BERGERON A., STOYE J., On the Similarity of Sets of Permutations and its Applications to Genome Comparison, *Proceedings of COCOON 2003*, vol. 2697 de *Lecture Notes in Comput. Sci.*, 2003, p. 68-79.

[BLA 99] BLANCHETTE M. K. T., SANKOFF D., Gene Order Breakpoint Evidence in Animal Mitochondrial Phylogeny, *J. Mol. Evol.*, vol. 49, n^o 2, 1999, p. 193-203.

[BOU 02] BOURQUE G., PEVZNER P., Genome-Scale Evolution : Reconstructing Gene Orders in the Ancestral Species, *Genome Res.*, vol. 12, n^o 1, 2002, p. 26-36.

[FEL 03] FELSENSTEIN J., *Inferring Phylogenies*, Sinauer Associates, Inc., 2003.

[MOR 02] MORET B. T. J. W. L., WARNOW T., Steps toward accurate reconstruction of phylogenies from gene-order data, *J. Comput. Syst. Sci.*, vol. 65, n^o 3, 2002, p. 508-525.

[SAN 98] SANKOFF D., BLANCHETTE M., Multiple Genome Rearrangement and Breakpoint Phylogeny, *J. Comput. Biol.*, vol. 5, n^o 3, 1998, p. 555-570.

[SAN 00] SANKOFF D., NADEAU J. E., *Comparative Genomics. Empirical and Analytical Approaches to Gene Order Dynamics, Map Alignement and the Evolution of Gene Families*, Kluwer Academic Publishers, 2000.

Algorithme génétique de pondération d'attributs pour une classification non supervisée

Alexandre Blansché, Pierre Gançarski et Jerzy J. Korczak

LSIIT, UMR 7005 CNRS-ULP
Parc d'Innovation, Boulevard Sébastien Brant
67412 ILLKIRCH
{blansche,gancarski,jjk}@lsiit.u-strasbg.fr
http ://lsiit.u-strasbg.fr/afd/

RÉSUMÉ. *La classification d'objets composés de nombreux attributs est problématique car lorsque les attributs se font plus nombreux, ils sont généralement plus bruités, certains sont corrélés entre eux, d'autres enfin, n'apportent pas d'informations pertinentes. Il est nécessaire de choisir correctement quels attributs utiliser pour classifier les données. Nous proposons une méthode de pondération locale des attributs non supervisée. Pour cela nous utilisons un extracteur par classe à trouver, chaque extracteur utilisant des poids différents. L'apprentissage, consistant à chercher ces pondérations, se fait par un algorithme génétique coévolutif.*

MOTS-CLÉS : *Classification Non Supervisée, Pondération d'Attributs, Algorithmes Génétiques Coévolutifs*

1. Introduction

La classification d'objets composés de nombreux attributs est problématique car lorsque les attributs se font plus nombreux, ils sont généralement plus bruités, certains sont corrélés entre eux, d'autres n'apportent pas d'informations pertinentes. Il est nécessaire de choisir correctement quels attributs utiliser pour classifier les données.

Il existe de nombreuses méthodes de sélection d'attributs [JOH 94, WET 95, KOH 95]. De nombreuses méthodes qui ont été proposées ont pour but de chercher un sous-ensemble de l'ensemble des attributs (c'est-à-dire une pondération binaire sur les attributs). Nous avons choisi de chercher une pondération sur les attributs. De plus, la plupart des méthodes se contentent de chercher une pondération globale, utilisée par l'ensemble des données. Il nous a semblé plus judicieux de chercher une pondération spécifique à chaque classe cherchée. Très peu de recherche s'intéressent à cette approche. Nous pouvons cependant citer [HOW 97]. Enfin, nous travaillons principalement sur des données pour lesquelles aucune connaissance de l'expert n'est fournie. Nous utilisons ainsi des méthodes de classification non supervisée, de même, la pondération d'attribut se fait de manière non supervisée. Ici aussi, on ne retrouve que très peu de recherche dans ce domaine [DY 00, MOD 03].

Nous proposons d'utiliser plusieurs extracteurs différents (utilisant une pondération différente), chaque extracteur permettant d'obtenir une classe à partir de l'ensemble des données. Ainsi, l'apprentissage consistera à trouver des classes complémentaires les unes aux autres, chacune de ces classes étant obtenue indépendamment.

Nous proposons de mettre un place un mécanisme d'apprentissage par coévolution génétique dans lequel les pondérations utilisées par les extracteurs vont « évoluer » de façon à ce que la répartition des données dans les classes soit la meilleure possible. Pour pouvoir définir la processus d'apprentissage, il est nécessaire d'éclaircir les quatre points suivants :

– Comment extraire une classe (cf. 2) ?

– Quel est le critère de qualité à utiliser pour l'algorithme génétique (cf. 3) ?
– Comment faire évoluer les extracteurs pour que les pondérations soient les meilleures possibles (cf. 4) ?
– Comment unifier les résultats (cf. 5) ?

2. Extraction d'une classe

Formellement, un extracteur est une fonction qui renvoie un sous-ensemble d'un ensemble. On peut définir un extracteur comme un triplet $X = (M, w, r)$, où M est une méthode de classification, w un ensemble de poids $w_1 \dots w_n$, $0 < w_i < 1$ (n étant le nombre d'attributs des éléments de S), et r un critère de qualité d'une classe. Pour calculer la classe extraite $X(S)$, où S est l'ensemble des données à classifier, on effectue tout d'abord une classification en utilisant la méthode M et les poids w sur les attributs des éléments de S. On obtient ainsi un ensemble de classes $\{C_1, \dots, C_{m_X}\}$. La classe C_e telle que $r(C_e) = max\{r(C_k), k = 1, \dots, m_X\}$ est sélectionnée.

Comme critère de qualité d'une classification, nous proposons d'utiliser la compacité d'une classe, comme elle est définie ci-dessous.

$$r(C_k) = \begin{cases} 0, \text{ si } \frac{1}{n_k} \sum_{l=1}^{n_k} \frac{d(x_{k,l}, g_k)}{d(x_{k,l}, g)} > 1 \\ 1 - \frac{1}{n_k} \sum_{l=1}^{n_k} \frac{d(x_{k,l}, g_k)}{d(x_{k,l}, g)}, \text{ sinon} \end{cases}$$

avec n_k le nombre d'objets appartenant à la classe C_k, $x_{k,l}$ le l-ième élément de C_k, g_k le centre de gravité de C_k et g le centre de gravité différent de g_k le plus proche de $x_{k,l}$ (c'est donc le centre de gravité de la classe la plus proche de la classe C_k).

Nous travaillons principalement avec les algorithmes K-means et Cobweb . Pour K-means, les coefficients sur les attributs seront utilisés pour le calcul de la distance globale. D'une manière similaire le calcul de la prédictivité pour la méthode Cobweb est effectué en utilisant ces pondérations.

3. Qualité d'un extracteur

On définit une répartition comme un ensemble $D = \{D_1, \dots, D_m\}$ de m parties de l'ensemble des N objets observés $S = \{o_1, \dots, o_N\}$. Le critère de qualité de l'algorithme génétique (fonction de fitness) doit dépendre de la qualité de la répartition que forment les classes obtenues. Ce critère doit tenir compte du partitionnement obtenu (cf. 3.1) et de l'homogénéité des tailles des classes qui la composent (cf. 3.2).

3.1. *Qualité de partitionnement*

Nous définissons un objet K-classifié comme un objet appartenant à la classe extraite de K extracteurs et nous définissons $S_K(D)$ l'ensemble de tous les objets K-classifiés d'une partition D. D est une partition si et seulement si $S_1(D) = S$, S étant l'ensemble des objets à classifier. On donne alors une qualité de partitionnement maximal à D. Au contraire, moins il y a d'objets 1-classifiés dans D moins la qualité de partition de D sera élevée.

Pour définir un tel critère, nous avons d'abord défini une qualité pour chaque objet o_k dans une répartition D par $Q_o(o_k, D) = 1 - \left|Card\{D_i \mid o_k \in D_i, D_i \in D\} - 1\right|$.

Ainsi, si un objet o_k appartient à un et un seul ensemble de P alors $Q_o(o_k, D) = 1$ et o_k est dit *1-classifié*, sinon $Q_o(o_k, D) \leq 0$.

Nous pouvons alors définir la qualité de partitionnement d'une répartition D par :

$$Q_P(D) = \max\left(0, \frac{1}{N}\sum_{k=1}^{N} Q_o(o_k, D)\right)$$

3.2. *Qualité d'homogénéité*

Nous avons défini ce second critère pour que les classes obtenues soient de tailles comparables et ainsi éviter qu'une classe contiennent une trop grande partie des objets. Il nous a semblé plus pertinent de définir ce critère en utilisant uniquement les objets 1-classifiés. On définit l'homogénéité d'une répartition par :

$$Q_H(D) = \frac{\prod_{j=1}^{m} nb_j}{\left(\frac{Card\left(S_1(D)\right)}{m}\right)^m}$$

où nb_j est le nombre d'objets 1-classifiés extrait par le j^{e} extracteur ($nb_j = Card\{o_k \mid o_k \in D_j \cap S_1(D)(P)\}$).

Cependant, ce critère est trop restrictif et biaise les résultats. Sa maximisation permet bien qu'une classe n'englobe pas la quasi-totalité des données. Par contre, il peut engendrer une dégradation des résultats dans le cas où une petite classe doit être mise en évidence. Il faudra donc définir un autre critère qui, à notre avis, dépendra du domaine d'application.

3.3. *Qualité totale d'une répartition*

Il nous est maintenant possible de définir un critère de qualité d'une répartition à partir des définitions précédentes par : $Q_t(P) = p_P Q_P(D) + p_H Q_H(D)$, avec p_P, *coefficient de partitionnement*, et p_H, *coefficient d'homogénéité*, tels que $p_P + p_H = 1$. p_P et p_H sont donnés par l'expert en fonction des données à traiter.

4. Évolution des extracteurs

Nous proposons une approche par coévolution [PAR 97, POT 00]. Chaque individu représente un extracteur. Un chromosome est constitué des poids $w_i \in [0, 1]$ sur chaque caractéristique pour l'extracteur qu'il représente. Le but est de faire évoluer plusieurs individus en collaboration afin que les classes obtenues par un ensemble d'individus forment une bonne classification de l'ensemble des données.

Pour cela, nous utilisons une population par classe cherchée. Les répartitions se construiront en prenant un extracteur de chaque population. La méthode consiste alors à faire évoluer plusieurs populations en collaboration.

Cependant, trouver le meilleur ensemble d'extracteurs en étudiant toutes les combinaisons possibles comportant un extracteur de chaque population pose un problème important. Si l'on cherche m classes avec p individus dans chaque population, p^m calculs de qualité sont effectués à chaque génération. Le temps de calcul devient rédhibitoire pour un nombre important de données. Nous proposons de définir la qualité d'un individu par rapport à une répartition de référence.

Une répartition de référence se définit à une génération g ($g \neq 1$) donnée comme la meilleure répartition trouvée au cours des générations précédentes, elle est notée $\Delta(g) = \{\Delta_1(g), \ldots, \Delta_i(g), \ldots, \Delta_m(g)\}$, où $\Delta_i(g)$ correspond à la i-ème classe.

$\Delta(g+1)$ est obtenue en prenant la meilleure répartition possible à partir des $\Delta_i(g)$ et du meilleur extracteur de chaque population de la génération g.

On peut alors donner une définition de la qualité d'un extracteur X_k^i, spécialisé dans la i-ème classe, en calculant la qualité de la répartition obtenue en remplaçant la i-ème classe de $\Delta(g)$ par $X_k^i(S)$. Ainsi, $Q\left(X_k^i\right) = Q_t\left(D_k^i\right)$, où $D_k^i = \left\{\Delta_1(g), \ldots, X_k^i(S), \ldots, \Delta_m(g)\right\}$.

Ainsi, seulement $m \times p + 2^m$ calculs de qualité sont effectués à chaque génération. L'évaluation à la première génération se fait en prenant un échantillon choisi au hasard car il n'y a pas encore d'ensemble de référence.

L'évolution se fait ensuite de manière classique au sein de chaque population, en appliquant des opérateurs de croisements (deux « enfants » ont chacun une moitié du génome de chaque « parent ») et de mutations (des valeurs aléatoires sont affectées à un nombre aléatoire de gènes).

5. Unification des classes

Pour chaque objet o_k, on calcule pour chaque extracteur $X_i = (C_i, w_i, r_i)$ le rapport $\rho = \frac{d_{w_i}(o_k, g_e)}{d_{w_i}(o_k, g)}$, où g_e est le centre de la classe extraite par X_i et g le centre de la classe de C_i différent de g_e le plus proche de o_k.

On affecte alors o_k à l'extracteur qui produit le rapport le plus bas. On voit facilement que si o_k est 1-classifié, il sera affecté à l'unique extracteur qui l'a extrait.

6. Application

Nous avons principalement appliqué notre méthode sur des images de la télédétection dans le cadre d'une collaboration avec le LIV[1] et du projet de recherche européen TIDE[2]. Ces résultats sont très prometteurs mais il reste encore des limites à notre méthode :

- la fonction de qualité ne modélise pas parfaitement la qualité réelle, un critère flou serait plus approprié ;
- le nombre de classes doit être défini a priori alors que cette information n'est pas toujours disponible.

7. Bibliographie

[DY 00] DY J. G., BRODLEY C. E., Feature Subset Selection and Order Identification for Unsupervised Learning, *Proc. 17th International Conf. on Machine Learning*, Morgan Kaufmann, San Francisco, CA, 2000, p. 247–254.

[HOW 97] HOWE N., CARDIE C., Examining Locally Varying Weights for Nearest Neighbor Algorithms, *ICCBR*, 1997, p. 455–466.

[JOH 94] JOHN G., KOHAVI R., PFLEGER K., Irrelevant Features and the Subset Selection Problem, *International Conference on Machine Learning*, 1994, p. 121–129.

[KOH 95] KOHAVI R., SOMMERFIELD D., Feature subset selection using the wrapper method : Overfitting and dynamic search space topology, *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, 1995, p. 192–197.

[MOD 03] MODHA D., SPANGLER S., Feature Weighting in k-Means Clustering, *Machine Learning*, vol. 52, n^o 3, 2003, p. 217–237.

[PAR 97] PAREDIS J., Coevolving Cellular Automata : Be aware of the Red Queen !, *7th Int. Conference on Genetic Algorithms (ICGA 97)*, 1997, p. 393–400.

[POT 00] POTTER M., DE JONG K., Cooperative coevolution : an architecture for evolving coadaptative subcomponents, *Evolutionary Computation*, vol. 8, 2000, p. 1–29.

[WET 95] WETTSCHERECK D., AHA D., Weighting Features, VELOSO M., AAMODT A., Eds., *Case-Based Reasoning, Research and Development, First International Conference*, Berlin, 1995, Springer Verlag, p. 347–358.

1. Laboratoire Image et Ville, ULP/CNRS UMR 7011
2. Tidal Inlets Dynamics Environment

Pertinence d'un sous-ensemble d'attributs en classification non supervisée

Lydia Boudjeloud, François Poulet

ESIEA Recherche
38, rue des docteurs Calmette et Guérin
Parc Universitaire de Laval-Changé
53000 Laval
{boudjeloud|poulet}@esiea-ouest.fr

RÉSUMÉ. L'objectif de ce travail est de proposer une nouvelle méthode capable de qualifier et de sélectionner des sous-ensembles d'attributs pertinents en classification non supervisée. L'idée principale consiste à utiliser les indices de validité des algorithmes de clustering pour valider le clustering sur un sous-ensemble d'attributs. On utilise une heuristique de recherche d'un sous-ensemble d'attributs optimal ou pertinent (approchant la solution optimale), optimal au sens de l'inertie inter et intra groupe de l'ensemble des données (selon les différents indices). Les performances de cette nouvelle approche de sélection d'attributs sont évaluées à partir de simulations sur des ensembles de données de grande taille.

MOTS-CLÉS : Sélection d'attributs, Classification non supervisée, Attribut pertinent, Attribut optimal, Grands ensembles de données

1. Introduction

Valider les résultats d'un algorithme de classification non supervisée (clustering) se fait en général en le comparant avec les résultats d'autres algorithmes de clustering ou en comparant les résultats obtenus par le même algorithme en faisant varier ses propres paramètres. On peut aussi valider les résultats obtenus par un algorithme de clustering en utilisant des indices de validité qui sont, pour la plupart, basés sur la minimisation de l'inertie intra-classe et la maximisation de l'inertie inter-classe [BER 02]. L'objectif de tout algorithme de clustering est de maximiser la distance entre les clusters et minimiser la distance entre les objets de chaque groupe ; en d'autres termes, déterminer la répartition optimale de l'ensemble de données. L'idée présentée dans cet article est d'utiliser ces critères pour valider les résultats d'un même algorithme dans un sous-espace, utilisant un sous-ensemble d'attributs.

Les données collectées dans le monde sont tellement importantes en taille qu'elles deviennent de plus en plus difficiles à appréhender par l'utilisateur. La dimension des données est une des difficultés majeures rencontrées en fouille de données [SCH 99].

Notre objectif est de pouvoir enfin qualifier un sous-ensemble d'attributs d'optimal ou de pertinent, si on arrive à comparer les résultats obtenus dans un sous-ensemble d'attributs à ceux obtenus sur tout l'ensemble de données et à en tirer des conclusions intéressantes. Dans ce qui suit, nous allons commencer par décrire les critères de validité utilisés, nous décrirons ensuite la méthodologie utilisée pour qualifier des sous-ensembles d'attributs, une fois ce sous-ensemble trouvé nous allons visualiser ses résultats pour pouvoir les expliquer au mieux.

2. Les indices

Nous comparons quelques indices de validité du clustering, décrits par Muligan et Cooper [MUL 85]. Ces indices représentent des critères internes qui peuvent -pour certains- être calculés indépendamment de l'algorithme de clustering. Le choix du nombre de clusters est souvent fait à partir de ces indices en étudiant le maximum max_n (ou le minimum min_n) de la valeur i_n (où n représente le nombre de clusters et i_n la valeur de

l'indice correspondant à *n* clusters). La solution la plus intuitive est de prendre le nombre de clusters qui maximise (ou minimise) l'indice. Le tableau 1 décrit quelques indices de validité où :

W : matrice somme des dispersions de chaque groupe, *K* : nombre de clusters,

SSW : représente l'inertie intra groupes, *SST :* représente l'inertie totale de l'ensemble,

SSB : représente l'inertie inter groupes, *B* : matrice de dispersion des centres des clusters,

T : matrice de dispersion des données, *N* : nombre de points de l'ensemble des données,

Indices	Formules correspondantes
Calinski & Harabasz (1974)	*(SSB/(k-1))/(SSW/(n-k))*
Hartigan (1975)	*Log (SSB/SSW)*
Ratkowsky & Lance (1978)	*Mean(* $\sqrt{\text{var}\,SSB / \text{var}\,SST}$ *)*
Ball & Hall (1965)	*SSW/k*
Trace Cov W (1985)	*TraceCovW*
Trace W (1970)	*TraceW*

Tab 1- Description des indices de validité

DB-index (1979)

$$R = (1/n)\sum_{i=1}^{n} R_i$$

R_i représente la valeur maximale de $R_{ij} = (SSW_i + SSW_j) / DC_{ij}$, pour $i \neq j$ et DC_{ij} représente la distance entre les centres des clusters *i* et *j*.

3. Méthodologie

Nous allons essayer de calculer les indices décrits précédemment et de comparer les valeurs obtenues en les calculant sur tout l'ensemble de données ainsi que sur des ensembles de données réduits (pour des sous-ensembles d'attributs). Nous allons utiliser pour nos tests l'algorithme *k-means* [HAR 79] sur les ensembles de données Colon tumor (2000 attributs, 62 éléments), Segmentation (19 attributs, 2310 éléments) et Shuttle (9 attributs, 42500 éléments).

Pour chaque indice nous allons essayer de retrouver les valeurs maximales et le *k* correspondant.

Répéter

Pour k entre 2 et une « valeur maximale pour *k* »

Calculer l'indice

Jusqu'à avoir une valeur maximal de l'indice → *k* optimal !!

Après avoir calculé ce *k* optimal sur tout l'ensemble de données, on essaye de retrouver ou d'approcher la valeur de l'indice correspondant au *k* optimal sur des sous-ensembles d'attributs.

4. Tests et commentaires

Nous avons recherché la valeur optimale de *k* pour tous les ensembles de données cités précédemment en utilisant l'algorithme *k-means* [HAR 79].

Indices	**Calinski**	**Hartigan**	**Ratkowski**	**Ball**	**TracCovW**	**DB**	**tracW**
***k*=2**	21.36	-1.03	0.36	$8.42\ 10^9$	$1.27\ 10^{16}$	$2.82\ 10^4$	$1.75\ 10^{10}$
***k*=7**	8.57	-0.06	0.25	$1.68\ 10^9$	$0.60\ 10^{16}$	$2.55\ 10^4$	$1.33\ 10^{10}$

Tab 2- les indices obtenus sur l'ensemble Colon Tumor

A titre d'exemple, nous présentons dans le tableau 2, les résultats observés sur l'ensemble de données *Colon Tumor* pour les valeurs des *k* testées (*k*= 2 à 8). Les valeurs décrites dans le tableau représentent la moyenne des résultats obtenus pour 10 *k-means*. Par manque de place, nous ne présentons que les valeurs obtenues pour k = 2 et k = 7. Nous décrivons plus loin l'évolution des indices par rapport à *k*, dans la figure 2 (à titre d'exemple : l'évolution de l'indice de Calinski par rapport à *k*). La courbe de la figure 1 montre bien l'évolution de l'indice de Calinski par rapport à la valeur de *k* et que la valeur optimale de l'indice est atteinte pour *k*=2 (ce qui est

cohérent, puisque les ensembles de données utilisés ont des classes connues a priori, respectivement 7, 7 et 2 pour les ensembles de données Shuttle, Segmentation et Colon Tumor).

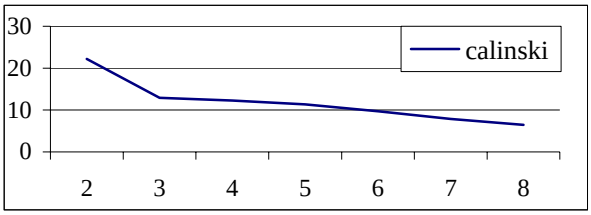


Fig 1- Evolution de l'indice de Calinsky

Nous allons, par la suite, essayer de retrouver une combinaison d'attributs optimale à l'aide d'un algorithme génétique [GOL 89] dont l'individu est représenté génétiquement par une combinaison d'attributs et la fonction fitness représente la valeur des différents indices cités précédemment. Notre objectif est de retrouver un sous-ensemble d'attributs qui représente au mieux la configuration de l'ensemble de départ, et que l'on puisse retrouver par la même occasion la configuration du clustering (c'est-à-dire : taille, nombre, contenu, …pour chaque cluster). Dans ce qui suit nous allons décrire les résultats obtenus par l'algorithme génétique.

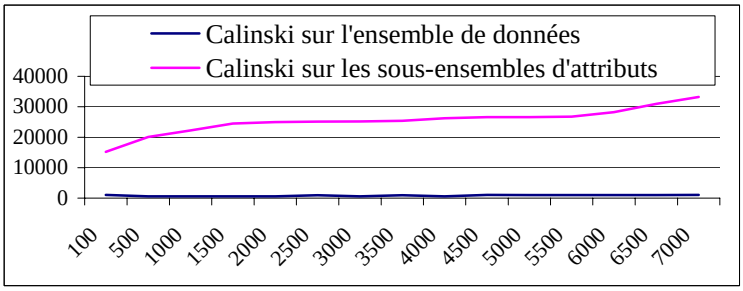


Fig 2- Evolution de l'AG sur l'ensemble de donnée Segmentation

Nous avons testé l'AG [GOL 89] avec des sous-ensembles d'attributs de taille 4 pour les ensembles de données Colon Tumor et Segmentation et des sous-ensembles de taille 3 pour l'ensemble de données Shuttle. La figure 2 montre l'évolution de l'algorithme génétique (indice de Calinsky sur l'ensemble de données Segmentation) au fil des générations, en comparaison avec l'indice de Calinsky calculé sur tout l'ensemble de données.

		SS ensemble optimal
Indices	**valeur sur ens tt**	**1089- 890-1506-1989**
Calinski	2,14E+01	8,85E+01
Ball	8,42E+09	6,46E+05
DB	2,82E+04	1,25E+02
Hartigan	-1,03E+00	3,89E-01
Ratkowsky	3,64E-01	4,95E-01
TrCovW	1,27E+16	1,54E+11
TraceW	1,75E+10	1,35E+06

Tab 3- Résultats de l'AG sur l'ensemble Colon Tumor

On remarque finalement un grand écart entre l'indice calculé sur l'ensemble de données et les indices calculés sur des sous-ensembles d'attributs. En effet, les résultats des indices sur les sous-ensembles d'attributs sont meilleurs que ceux sur tout l'ensemble des données, ceci peut être expliqué par le fait que les données peuvent être bruitées et supprimer certains attributs permet de diminuer l'effet du bruit et donc mieux regrouper les données. On peut en conclure aussi que certaines combinaisons d'attributs ont une meilleure configuration des données que celle de l'ensemble de données, et permettent une meilleure classification non supervisée. Nous présentons dans le tableau 3 la solution optimale de l'AG, c'est-à-dire le sous-ensemble d'attributs pour lequel nous avons obtenu les valeurs les plus optimales de tous les indices de validité de l'ensemble de données Colon Tumor. La figure 3 montre la visualisation de la classification non supervisée pour l'ensemble de données Colon tumor, sur le sous-ensemble d'attributs obtenu par l'AG.

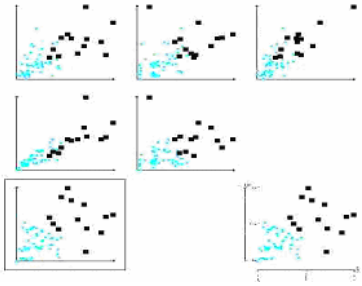


Fig 3-Visualitation des résultats AG sur Colon Tumor

5. Conclusion et perspectives

Dans cet article, nous avons voulu présenter une manière de juger un sous-ensemble d'attributs en classification non supervisée (clustering). Pour cela, nous avons utilisé les indices de validité d'un algorithme de clustering non pas, pour juger un algorithme de clustering, mais pour juger un sous-ensemble représentatif ou pertinent d'attributs par rapport à l'ensemble de données. Nous avons utilisé l'algorithme de clustering *k-means* [HAR 79] et sept sur treize indices de validité répertoriés par Mulligan et Cooper dans [MUL 85], (les six autres étant difficiles à calculer vu l'importance des données), ainsi qu'un algorithme génétique pour la sélection d'attributs, dont la fonction objectif est la valeur des indices de validité de l'algorithme *k-means*. Notre objectif était d'essayer de retrouver les valeurs des indices pour un sous-ensemble d'attributs identiques à celles obtenues sur tout l'ensemble de données. Les résultats obtenus montrent que les valeurs des indices sur des sous-ensembles d'attributs ont été meilleurs que ceux obtenus sur tout l'ensemble des données, on peut expliquer ceci par le fait que les données peuvent être bruitées sur certains attributs et que sélectionner quelques attributs dans des sous-ensembles nous permet d'éviter ces bruits, et donc obtenir de meilleurs résultats. Nous pensons par la suite, tester cette méthode sur des algorithmes de classification non supervisée de façon à pouvoir comparer les résultats obtenus par l'algorithme avec le sous-ensemble d'attributs et le résultat global de la classification non supervisée. Si les résultats sont très proches, le sous-ensemble est pertinent, sinon, il l'est moins. Nous pensons aussi à faire évoluer la méthode de façon à pouvoir juger de la pertinence ou de l'optimalité d'un sous-ensemble d'attributs en classification non supervisée, et essayer de trouver un critère convaincant (non pas une comparaison) pour pouvoir qualifier un sous-ensemble d'attributs d'optimal ou de pertinent.

6. Bibliographie

[BER 02] BERKHIN P. "Accrue Sotware: Survey Of Clustering Data Mining Techniques", 2002.

[GOL 89] GOLDBERG D.E. "Genetics Algorithms in Search", *Optimisation and Machine Learning*. Addison-Wesley, 1989.

[HAR 79] HARTIGAN, J.A., WONG, M.A. "A K-means clustering alrorithm: Algorithm AS 136." *Applied Statistics*, 28, 126-130, 1979.

[MUL 85] MULLIGAN G., COOPER M., "An examination of procedures for determining the number of clusters in a data set", *Psychometrika*, vol.52, n°2, 1985. p 159-179.

[SCH 99] SCHEUNDERS P., DE BACKER S.,"High-Dimensional Clustering using Frequency Sensitive Competitive Learning",*Pattern Recognition*, V32; n°2, pp193-202, 1999.

Les ultramétriques faibles

François Brucker

GET-ENST Bretagne, Département LUSSI
Pointe Diable CS 83818 29238 BREST CEDEX 3, FRANCE
mél : francois.brucker@enst-bretagne.fr

RÉSUMÉ. *Nous présentons dans cet article une caractérisation métrique des dissimilarités dont les classes forment une hiérarchie faible et montrons qu'il existe une bijection entre icelles et les hiérarchies faibles strictement indicées. Nous montrons également quelques propriétés topologiques et combinatoires de ces dissimilarités.*

MOTS-CLÉS : *Classification, dissimilarités, quasi-ultramétriques, hiérarchies faibles*

1. Introduction

Le but des modèles traditionnels en classification que sont les partitions et les hiérarchies de parties est de permettre de discriminer sans ambiguïté, et donc de produire des classes non empiétantes. Cependant, l'exigence de non ambiguïté peut conduire à occulter de l'information, dans le cas des plantes hybrides en biologie ou de l'étude de réseaux sociaux par exemple.

Pour pallier ce problème, de nouveaux modèles de classification en classes empiétantes ont été développés, dont les hiérarchies faibles (Bandelt et Dress, 1989 [BAN 89]).

Dans la suite de cet article, nous travaillerons sur un ensemble fini d'objets X, et on appellera *système de classes* de X un sous-ensemble $\mathcal{K}$ de 2^X tel que :
– $\phi \notin \mathcal{K}$,
– $X \in \mathcal{K}$,
– pour tout $x \in X$, $\{x\} \in \mathcal{K}$.
Une *hiérarchie* est alors un système de classes $\mathcal{K}$ tel que pour tous $A, B \in \mathcal{K}$, $A \cap B \in \{\phi, A, B\}$, et une *hiérarchie faible* est un système de classes tel que pour tous $A, \quad B, C \in \mathcal{K}$, $A \cap B \cap C \in \{A \cap B, B \cap C, A \cap C\}$. Une hiérarchie faible fermée par intersection est appelée *quasi-hiérarchie* (Diatta et Fichet, 1998 [DIA 98]).

Tout système de classes $\mathcal{K}$ peut être muni d'un *indice strict* f. Un indice strict est une fonction f de $\mathcal{K}$ dans $\mathbb{R}^+$ tel que :

1. $f(\{x\}) = 0$ pour tout $x \in X$,
2. pour tous $A, B \in \mathcal{K}$ tels que $A \subsetneq B$, $f(A) < f(B)$.

Une fonction de $\mathcal{K}$ dans $\mathbb{R}^+$ vérifiant la première condition et telle que $f(A) \leq f(B)$ pour tous $A \subsetneq B$ est simplement appelé *indice*.

Le couple $(\mathcal{K}, f)$ où $\mathcal{K}$ est un système de classes et f un indice strict (*resp.* un indice) sur $\mathcal{K}$ est appelé système de classes strictement indicé (*resp.* système de classes indicé).

Lorsque les données sont décrites par une dissimilarité d, on a coutume, dans la lignée de Jardine et Sibson (1971 [JAR 71]), de lui associer un système de classes $\mathcal{K}_d$ contenant les cliques maximales de ses graphes seuils ($G_h = (X, E_h)$ est un graphe seuil de d pour le seuil h, si et seulement si $xy \in E_h \Leftrightarrow d(x, y) \leq h$). Le couple

$(\mathcal{K}_d, \text{diam}_d)$ (où $\text{diam}_d(A) = \max\{d(x,y) \mid x,y \in A\}$ pour tout $A \subset X$) forme alors un système de classes strictement indicé.

Des théorèmes de bijection permettent de mettre en correspondance des types de sytèmes de classes strictement indicés et des types particuliers de dissimilarités *via* $(\mathcal{K}_d, \text{diam}_d)$ (*cf.* Batbedat 1988 [BAT 88] et Bertrand 2000 [BER 00] pour une bijection générale entre les dissimilarités et les systèmes de classes indicées).

Les *ultramétriques* sont ainsi en bijection avec les hiérarchies strictement indicées (Benzécri 1973 [BEN 73], Johnson 1967 [JOH 67]), et les *quasi-ultramétriques* avec les quasi-hiérarchies strictement indicées (Diatta et Fichet 1998 [DIA 98]). Les ultramétriques sont les dissimilarités u telles que :

$$\forall x, y, z \in X, u(x,y) \leq \max\{u(x,z), u(y,z)\}$$

et les quasi-ultramétriques les dissimilarités d telles que [BAN 92, DIA 98] :

$$\forall x, y, z \in X, \max\{d(z,x), d(z,y)\} \leq d(x,y) \Rightarrow \forall t \in X, d(z,t) \leq \max\{d(t,x), d(t,y), d(x,y)\}.$$

Nous montrons dans la suite de cet article qu'il existe une bijection entre un type particulier de dissimilarités, appelées *ultramétriques faible* et les hiérarchies faibles strictement indicées, et nous présenterons quelques propriétés de cette nouvelle "race" de dissimilarité.

2. Ultramétriques faibles

On définit les *ultramétriques faibles* comme étant les dissimilarités telles que $\mathcal{K}_d$ est une hiérarchie faible. La proposition 1 les caractérise, et la proposition 2 en donne une interprétation métrique.

À toute dissimilarité d peut être associée sa réalisation Δ_d (Brucker, 2003 [BRU 03] et Barthélemy 2003 [BAR 03]). Δ_d est l'ensemble de tous les $\delta[d](x,y)$, $x, y \in X$ où :

$$\delta[d](x,y) = \cap\{A | x, y \in A, A \in \mathcal{K}_d\}$$

Pour que $\mathcal{K}_d$ soit une hiérarchie faible, il faut et il suffit que quelles que soient $A, B, C \in \mathcal{K}_d$ il n'existe pas :
– $x \in A$ et $x \notin B$, $x \notin C$
– $y \in B$ et $y \notin A$, $y \notin C$
– $z \in C$ et $z \notin A$, $z \notin B$
De là découle la proposition suivante :

Proposition 1 *Pour une dissimilarité d sur X, $\mathcal{K}_d$ est une hiérarchie faible si et seulement si quels que soient $x, y, z \in X$:*

$$\delta[d](x,y) \cap \delta[d](y,z) \cap \delta[d](x,z) \cap \{x,y,z\} \neq \phi$$

Si l'on définit pour une dissimilarité d une boule de centre $x \in X$ et de rayon α comme étant l'ensemble $B(x,\alpha) = \{y | d(x,y) \leq \alpha\}$, on montre que :

$$\delta[d](x,y) = \cap_{z \in X}\{B(z, \max\{d(x,z), d(y,z), d(x,y)\})\}$$

Cette propriété permet d'établir la proposition 2 qui caractérise de façon métrique les ultramétriques faibles.

Proposition 2 *une dissimilarité d sur X est une ultramétrique faible si et seulement si pour tous $x, y, z, t_x, t_y, t_z \in X$ les trois inégalités suivantes ne sont pas satisfaites simultanément :*
– $d(t_x, x) > \max\{d(y,z), d(t_x,y), d(t_x,z)\}$

– $d(t_y, y) > \max\{d(x, z), d(t_y, x), d(t_y, z)\}$
– $d(t_z, z) > \max\{d(x, y), d(t_z, x), d(t_z, y)\}$

d_1 :

	x	y	z	t	u
x	0				
y	1	0			
z	2	2	0		
t	2	3	2	0	
u	3	2	2	3	0

d_2 :

	x	y	z	t	u
x	0				
y	1	0			
z	2	2	0		
t	2	3	2	0	
u	**2**	2	2	3	0

Ainsi la dissimilarité d_1 ci-avant n'est pas une hiérarchie faible car :
– $d(u, x) > \max\{d(y, z), d(u, y), d(u, z)\}$
– $d(v, y) > \max\{d(x, z), d(v, x), d(v, z)\}$
– $d(x, z) > \max\{d(x, y), d(x, x), d(x, y)\}$
alors que la dissimilarité d_2 en est une (la seule différence entre d_1 et d_2 est $d_1(x, u) = 3 > d_2(x, u) = 2$).

3. Théorème de bijection

Cette partie montre la réciproque de la proposition 1, à savoir que l'on peut associer une ultramétrique faible à toute hiérarchie faible strictement indicée.

En utilisant le fait que pour une ultramétrique faible d la fermeture par intertersection de $\mathcal{K}_d$ est égale à Δ_d, on peut monter le théorème suivant :

Théorème 1 *Pour toute hiérarchie faible strictement indicée $(\mathcal{K}, f)$ il existe une unique ultramétrique faible d telle que $(\mathcal{K}_d, diam_d) = (\mathcal{K}, f)$. Réciproquement, toute ultramétrique faible d définie une unique hiérarchie strictement indicée $((\mathcal{K}_d, diam_d)$.*

Le théorème 1 est ainsi une généralisation de celui démontré par Diatta et Fichet [DIA 98] associant une unique quasi-ultramétrique à une quasi-hiérarchie strictement indicée (et réciproquement).

Par exemple, la hiérarchie faible strictement indicée associée à la dissimilarité d ci-avant est constituée des classes et indices suivant :

– $\{x, y\}$ d'indice 1,
– $\{x, z, t\}$ d'indice 2,
– $\{x, y, z, u\}$ d'indice 2,
– $\{x, y, z, t, u\}$ d'indice 3.

Alors que le système de classes strictement indicé issu de la dissimialrité d_1 est :
– $\{x, y\}$ d'indice 1,
– $\{x, y, z\}$ d'indice 2,
– $\{x, z, t\}$ d'indice 2,
– $\{y, z, u\}$ d'indice 2,
– $\{x, y, z, t, u\}$ d'indice 3.
d_1 n'est pas une hiérarchie faible puisque $\{x, y\} \cap \{x, z, t\} \cap \{y, z, u\} \notin \{\{x\}, \{y\}, \{z\}\}$.

4. Propriétés

Cette dernière partie est destinée à montrer quelques relations qu'entretiennent les ultramétriques faibles avec les autres dissimilarités et en particulier les quasi-ultramétriques (proposition 3), ainsi que les relations qu'entretiennent les hiérarchies faibles avec les autres systèmes de classes (proposition 4).

En plongeant l'ensemble des dissimilarités sur X dans l'espace vectoriel $\mathcal{D}$ des applications d de $X \times X$ prenant des valeurs réelles, telles que $d(x,x) = 0$ et $d(x,y) = d(y,x)$ pour tous $x, y \in X$, on peut montrer la proposition suivante :

Proposition 3 *L'ensemble des ultramétriques faibles forme un fermé de $\mathcal{D}$ et est exactement la fermeture topologique de l'ensemble des quasi-ultramétriques.*

La propositon suivante (proposition 4) montre le lien intime qu'entretiennent les hiérarchies faibles dans la bijection générale reliant dissimilarités et systèmes de classes strictement indicées.

Proposition 4 *Si E est un ensemble de systèmes de classes tel que quel que soit $\mathcal{K} \in E$ et quel que soit f un indice strict sur $\mathcal{K}$, il existe une dissimilarité d telle que $(\mathcal{K}, f) = (\mathcal{K}_d, diam_d)$, alors E est un sous-ensemble de l'ensemble des hiérarchies faibles.*

Ainsi, si un système de classes $\mathcal{K}$ sur X n'est pas une hiérarchie faible, il existe un indice strict f sur $\mathcal{K}$ tel que pour toute dissimilarité d sur X, $(\mathcal{K}, f) \neq (\mathcal{K}_d, \text{diam}_d)$.

5. Bibliographie

[JOH 67] Johnson, S. C., Hierarchical Clustering Schemes, vol. 32, 1967, , p. 241-254.

[JAR 71] Jardine, N. et Sibson, R., *Mathematical Taxonomy*, Wiley, 1971.

[BEN 73] Benzécri, J. P., *L'analyse des données (Volume 1 : Taxonomie)*, Masson, 1973.

[BAT 88] Batbedat, A., Les isomorphismes HTS et HTE (après la bijection de Benzécri-Johnson), vol. 46, 1988, , p. 47-59.

[BAN 89] Bandelt, H.-J. et Dress, A. W. M., Weak Hierarchies Associated with Similarity Measures – an Additive Clustering Technique, vol. 51, 1989, , p. 133-166.

[BAN 92] Bandelt, H.-J., Four point characterization of the dissimilarity functions obtained from indexed closed weak hierarchies, *Mathematisches Semminar, Universität Hamburg, Germany.*, 1992.

[DIA 98] Diatta, J. et Fichet, B., Quasi-ultrametrics and Their 2-balls Hypergraphs, vol. 192, 1998, , p. 87-102.

[BER 00] Bertrand, P., Set Systems and Dissimilarities, vol. 21, 2000, , p. 727-743.

[BAR 03] Barthélemy, J. P., Classifications binaires, *Actes eds renconres de la société frncophone de classification*, p. , 67-69.

[BRU 03] Brucker, F., Réalisations de dissimilarités, *Actes eds renconres de la société frncophone de classification*, p. , 7-10.

Contribution à l'analyse d'un corpus de réseaux de galeries chez les fourmis

J. Buhl[1], L. Lasserre[2], P. Kuntz[2], G. Théraulaz[1], I. Kojadinovic[2]

[1] *Centre de Recherches sur la Cognition Animale*
CNRS UMR 5169
118, rte de Narbonne
31062 Toulouse Cedex 4

[2] *Ecole Polytechnique de l'Université de Nantes*
Laboratoire Informatique de Nantes-Atlantique
La Chantrerie
44306 Nantes cedex 3

RÉSUMÉ. Cette communication présente les premiers résultats issus d'une analyse d'un corpus de représentations planes de graphes planaires issus d'expérimentations réelles et de simulations relatives au comportement constructeur de certains insectes sociaux –ici des fourmis creusant des réseaux de galeries-. Nous étudions en particulier la distribution des variables numériques sélectionnées pour décrire les propriétés combinatoires, topologiques et fonctionnelles des graphes considérés, et la structuration de leurs interactions par des approches hiérarchiques.

MOTS-CLÉS : Graphes planaires, Variables numériques, Classification hiérarchique, Propriétés fonctionnelles

1. Introduction

De nombreuses espèces de fourmis construisent leurs nids en utilisant des mécanismes d'excavation qui aboutissent à la formation de réseaux constitués de chambres interconnectées par des galeries [RAS 99]. Les entomologistes émettent l'hypothèse d'un mécanisme de construction « auto-organisé » basé sur un mode décentralisé où la structure globale qui apparaît en fin de processus est une conséquence de l'application répétée de règles locales. Plus précisément, le lien souvent observé chez les insectes sociaux entre la structure d'un pattern émergeant et les performances fonctionnelles du collectif [BON 99] conduisent, dans le cas des réseaux, à deux questions majeures. L'efficacité de la colonie pour certaines tâches, par exemple ici la régulation du trafic et le maintien de la sécurité du nid dépend-elle de l'organisation topologique des galeries ? Et, quels sont les mécanismes sous-jacents qui permettent d'expliquer l'émergence des structures observées ?

La caractérisation des lois de croissance et d'organisation spatiale des réseaux de galeries associées aux mécanismes de production collective de ces structures repose, dans la méthodologie adoptée, sur une analyse combinant trois niveaux d'observations :

(i)- au niveau macroscopique, il s'agit de caractériser les propriétés spatiales et combinatoires des réseaux observés en laboratoire ;

(ii) – au niveau mésoscopique, on cherche à déterminer les lois qui régissent les modalités de croissance des galeries, en particulier celles concernant la création de nouvelles galeries et leur évolution spatio-temporelle ;

(iii) – au niveau microscopique, le problème consiste à découvrir les règles comportementales individuelles qui régulent le processus de creusement.

Dans cette communication, nous nous focalisons sur les deux premiers niveaux d'observations, les conclusions obtenues étant un préalable à la définition d'un modèle individuel robuste expérimentalement. Et, le travail

présenté porte sur l'analyse d'un corpus de réseaux de deux types : ceux déduits des observations *in vivo* au laboratoire, et ceux calculés par un simulateur d'un modèle de loi de croissance basé sur les paramètres observés.

2. Les données

Les données considérées peuvent être, par construction, modélisées par des représentations planes dans le plan euclidien de graphes planaires [BER 70].
Le premier processus de construction de ces graphes est issu d'une expérimentation en laboratoire. Des fourmis sont disposées initialement sur la circonférence d'un disque de sable humidifié de faible profondeur et, des images du comportement de ce collectif sont prises toutes les trois heures pendant trois jours (figure 1).

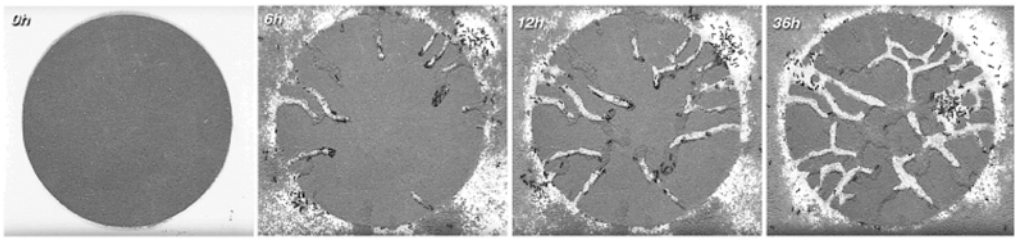


Figure 1. Evolution d'un réseau de galeries dans les conditions expérimentales

Nous ne considérons ici que le réseau final, qui vu les conditions expérimentales spécifiques, peut être considéré comme bi-dimensionnel. La squelettisation de l'image binarisée nous a permis de modéliser ce réseau de galeries par un graphe planaire (figure 2). Le corpus ainsi obtenu comprend *44* réseaux dont les conditions de construction dépendent uniquement du nombre de fourmis $f \in \{100, 200, 450\}$ et de la taille du disque $d \in \{14, 20, 30\}$.

Figure 2. Carte planaire associée à un réseau de galeries

Le deuxième processus de construction des graphes est un modèle de simulation « par croissance d'arcs » [BUH 03]. L'algorithme de construction comprend trois phases principales. La création d'un nouvel arc est fonction d'une probabilité et de contraintes qui dépendent de sa position initiale – initialisation sur la périphérie du disque ou à partir d'un arc déjà existant-. La croissance d'un arc dépend d'un angle engendré à partir d'une loi normale centrée autour de directions spécifiques selon les cas, et d'une vitesse constante. La croissance s'arrête lorsque l'arc atteint un sommet ou un arc préalablement construits ou la périphérie. Les paramètres du modèle sont basés sur des statistiques obtenues sur les données d'observation.

3. Méthodologie

Si, suite notamment à l'explosion de la Toile, de nombreux travaux ont été consacrés ces dernières années à l'étude de grands graphes pris dans leur unicité, l'analyse d'une base de graphes, en particulier de graphes planaires topologiques, pose à notre connaissance des questions méthodologiques encore largement ouvertes.
La démarche que nous avons privilégiée est basée sur la caractérisation des graphes par des variables associées à des propriétés fonctionnelles recherchées dans la problématique initiale des biologistes. L'intérêt apporté depuis peu à l'étude des « petits mondes » et des réseaux sans échelle a conduit au développement de nombreux indices

(voir [ALB 02] pour une synthèse récente). Nous avons ainsi calculé une quinzaine d'indices associés à quatre types de propriétés :

- les caractéristiques combinatoires basiques (degré des sommets, composantes connexes, …) et géométriques (fréquence des plus proches voisins adjacents, …) ;
- les caractéristiques de robustesse : indices associés à des procédures aléatoires de déconnexion des arêtes ou des sommets ;
- les caractéristiques d' « efficacité » [CRU 02] prenant en compte les distances entre les sommets ; on distingue les indices combinatoires basés sur la distance du plus court chemin en nombre de sommets des indices topologiques basés sur la distance réelle sur le plan euclidien ;
- les coefficients de « clustering » basés sur la distribution des degrés en chaque sommet.

4. Premiers résultats et perspectives

En analysant la distribution des différents indices sur chacune des deux familles de réseaux décrites dans le paragraphe précédent nous avons retrouvé des propriétés attachées aux graphes de « petits mondes ». La distribution des degrés suit une loi puissance $P(k) \sim k^{-\gamma}$, dont l'exposant varie selon la famille. La connectivité résiste bien aux déconnexions aléatoires des sommets et des arêtes ; en revanche, elle est très vulnérable aux déconnexions préférentielles qui privilégient les sommets avec des degrés élevés. Sur le plan fonctionnel, cela semble mettre en évidence le rôle dans les graphes de sommets spécifiques qui reste à expliquer. Contrairement à d'autres graphes planaires, la robustesse est ici une propriété prépondérante à la minimisation de la longueur moyenne des chaînes entre les sommets et la connectivité locale moyenne (définie pour chaque sommet par le ratio de son degré sur la taille d'un graphe complet d'ordre équivalent à son degré).
Si l'utilisation des indices définis pour la caractérisation des graphes de « petits mondes » s'avère effectivement intéressante pour la mise en évidence de propriétés fonctionnelles, les graphes considérés ici sont beaucoup plus petits (mais en revanche plus nombreux) que ceux sur lesquels reposent les caractéristiques annoncées dans la littérature, et la comparaison des valeurs intrinsèques est donc délicate.

Une première analyse de la structure des interactions entre les variables a été effectuée par une classification hiérarchique avec le critère de corrélation des rangs de Pearson. Les résultats mettent en évidence une quasi-absence de corrélation entre l'ordre, la taille, le diamètre moyen et les autres variables ; ce qui nous conduit à nous interroger sur une certaine indépendance des structures par rapport à des facteurs d'échelles. En revanche, les classes de variables obtenues semblent dépendre sensiblement des conditions expérimentales. Cependant, une partie des variables étudiées étant des variables numériques continues, nous envisageons actuellement une analyse plus adaptée basée sur une classification hiérarchique associée à l'information mutuelle [KOJ 03], qui en tant que mesure de dépendance stochastique, permet de détecter des dépendances non homogènes.

5. Bibliographie

[ALB 02] ALBERT R., BARABASI A. L. Statistical mechanics of complex networks, *Rev. Mod. Phys.*, vol. 74, n° 1, 2003, p. 47-97.

[BER 70] BERGE C. *Graphes et hypergraphes*, Dunod, 1973.

[BON 99] BONABEAU E., DORIGO M., THERAULAZ G. *Swarm intelligence – From natural to artificial systems*, Oxford Univ. Press, 1999.

[BUH 03] BUHL J., GAUTRAIS J., DEUNEUBOURG J.-L., KUNTZ P., THERAULAZ G. Simple rules of growth can account for the complexity of tunnelling networks in the ant Messor Sancta, *Proc. of the 2nd Int. Workshop on the Mathematics and Algorithms of Social Insects*, Georgia Institute of technology, 2003, p. 33-40.

[CRU 02] CRUCITTI P., LATORA V., MARCHIORI M., RAPISARDA A.. Efficiency of scale-free networks: error and attack tolerance, *Physica A*, vol. 320, 2003, p. 622-642.

[KOJ 03] KOJADINOVIC I. Agglomerative hierarchical clustering of continuous variables based on mutual information, *Computational Statistics and Data Analysis*, 2004, à paraître.

[RAS 99] RASSE P., DEUNEUBOURG J.-L. Dynamics of nest excavation and nest size regulation of Lasius Niger, *J. of Insect. Behav.*, vol. 14, 1999, p. 433-449.

Règles d'association et codage flou des données

Martine Cadot — **Amedeo Napoli**

LORIA, UMR 7503 CNRS
BP 239 - 54506 - Vandœuvre-lès-Nancy Cedex
martine.cadot@loria.fr

RÉSUMÉ. *L'extraction de jeux de règles d'association à partir de matrices sujets×propriétés se fait avec des algorithmes performants sur des données binaires [BAS 00]. Pour les autres types de données, le recodage habituel [STU 02] entraîne une grande perte d'efficacité de ces algorithmes. Nous proposons d'y remédier pour certains types de données en les recodant différemment. Nous définissons ainsi des règles floues d'association et nous les positionnons par rapport aux règles classiques de la fouille de de données et par rapport à la théorie des ensembles flous de Zadeh [ZAD 65], Dubois et Prade [DUB 88].*

MOTS-CLÉS : *fouille de données, prétraitement des données, codage flou des données, règles d'association floues, $\alpha-$coupe, treillis de motifs flous*

1. Codage flou des propriétés

Les règles d'association sont des règles du type "si A alors B" où A et B sont des conjonctions de propriétés vérifiées simultanément par un certain nombre de sujets. Ces règles sont obtenues en appliquant des algorithmes de fouille de données [HAN 01] à des matrices sujets×propriétés constituées de 0 et de 1. Le domaine qui a fait la notoriété de ces techniques de fouille de données est celui des tickets de caisse des supermarchés. Il est naturel de coder par 1 ou 0 le fait a_{ij} qu'un client c_i ait ou non acheté une marchandise m_j. Pour pouvoir appliquer ces techniques d'extraction de règles à d'autres types de données, il est nécessaire de procéder à un recodage. C'est le cas par exemple des données obtenues en interrogeant des personnes sur la force de leur adhésion à une opinion (pas du tout d'accord, désapprouve, indifférent, approuve, tout a fait d'accord) ou l'intensité d'un sentiment (pas du tout, un peu, assez, beaucoup, énormément), ou même la couleur préférée d'un objet (rouge, vert, bleu, jaune, mauve). Un premier codage se fait pour faciliter la saisie, par exemple par des nombres entiers successifs, et on obtient une matrice ayant autant de lignes que de sujets, et de colonnes que de propriétés (ce sont les opinions, sentiments ou couleurs) chaque valeur à l'intersection d'une ligne et d'une colonne étant représentée par un nombre entier, compris entre 1 et 5 pour ces trois propriétés ayant chacune cinq modalités. Le recodage le plus courant [STU 02] consiste à faire une dichotomisation, c'est-à-dire à remplacer chaque propriété de départ par autant de propriétés qu'il y a de modalités, puis on applique sur la matrice booléenne ainsi obtenue les algorithmes d'extraction de règles. Quand cela a un sens, sens que nous allons préciser par la suite, nous proposons une façon différente de procéder. Chaque propriété de départ est modifiée, ses valeurs pour chaque sujet étant remplacées par des valeurs normalisées sur une échelle allant de 0 à 1, puis nous extrayons de cette matrice "floue" des règles d'association floues au lieu des règles d'association classiques extraites d'une matrice booléenne.

Par exemple, si on prend comme propriété a la réponse à la question

Avez-vous peur des araignées : 1 :pas du tout, 2 :un peu, 3 :assez, 4 :beaucoup, 5 :énormément

on peut voir dans le tableau 1 comment sont codées les réponses de 7 sujets à cette question. Le codage par dichotomisation garde son sens quel que soit le type de mesure de la propriété a. Par contre, dans la mesure où nous utilisons les définitions (voir ci-dessous) issues de la théorie des ensembles flous de Zadeh [ZAD 65] citées dans l'ouvrage de D. Dubois et H. Prade [DUB 88], notre *codage flou* n'a de sens pour une propriété que si la

Réponses	
sujet	a
s1	beaucoup
s2	un peu
s3	assez
s4	beaucoup
s5	énormément
s6	assez
s7	pas du tout

Codage classique	
sujet	a
s1	4
s2	2
s3	3
s4	4
s5	5
s6	3
s7	1

Codage par dichotomisation					
sujet	$a1$	$a2$	$a3$	$a4$	$a5$
s1	0	0	0	1	0
s2	0	1	0	0	0
s3	0	0	1	0	0
s4	0	0	0	1	0
s5	0	0	0	0	1
s6	0	0	1	0	0
s7	1	0	0	0	0

Codage flou	
sujet	a
s1	0,75
s2	0,25
s3	0,50
s4	0,75
s5	1
s6	0,50
s7	0

TAB. 1. *Les codages les plus courants d'une propriété a et le codage flou que nous proposons.*

valeur attribuée à un sujet peut être interprétée comme un degré d'appartenance à cette propriété. Si on reprend les exemples précédents, c'est le cas pour l'échelle d'opinion, pour l'intensité d'un sentiment, mais pas pour le choix de la couleur d'un objet.

Définition 1.1. *Ensemble flou (selon Zadeh, cité par [DUB 88])*
Un ensemble flou F est la donnée d'un référentiel Ω et d'une application μ_F de Ω dans [0,1], cette application étant interprétée comme le degré d'appartenance des éléments de Ω à F.

En prenant comme référentiel Ω l'ensemble $\mathcal{S}$ des sujets, et comme application μ_F la propriété a telle que nous l'avons codée dans le dernier codage du tableau 1, l'ensemble flou F associé à ($\mathcal{S}$, a) est ce que nous appelons l'*extension [GUI 86]* de la propriété a, et nous la notons a'. Chaque propriété de $\mathcal{P}$ étant ainsi codée, la matrice sujets$\times$propriétés exprime maintenant une *relation floue* entre l'ensemble $\mathcal{S}$ des sujets et l'ensemble $\mathcal{P}$ des propriétés. Par la suite, nous appelons *propriété floue* toute propriété de cette relation floue, les propriétés classiques (binaires) en sont un cas particulier. Et on dit qu'un élément de $\mathcal{S}$ vérifie la propriété P dès que sa valeur est strictement positive.

Définition 1.2. *Opérations sur les ensembles flous*
Si F et G sont deux ensembles flous de fonctions d'appartenance μ_F et μ_G sur le référentiel Ω, on pose :

$$
\begin{array}{lll}
Cardinal: & & card(F) = \sum_{\omega \in \Omega} \mu_F(\omega) \\
Egalité: & F = G & si\ \forall \omega \in \Omega,\ \mu_F(\omega) = \mu_G(\omega) \\
Inclusion: & F \subseteq G & si\ \forall \omega \in \Omega,\ \mu_F(\omega) \leq \mu_G(\omega) \\
Complémentation & \overline{F}: & \forall \omega \in \Omega,\ \mu_{\overline{F}}(\omega) = 1 - \mu_F(\omega) \\
Intersection & F \cap G: & \forall \omega \in \Omega,\ \mu_{F \cap G}(\omega) = min(\mu_F(\omega),\ \mu_F(\omega)) \\
Réunion & F \cup G: & \forall \omega \in \Omega,\ \mu_{F \cup G}(\omega) = max(\mu_F(\omega), \mu_F(\omega))
\end{array}
$$

Nous choisissons ces définitions car, d'après D. Dubois et H. Prade [DUB 88], parmi toutes les définitions possibles, ce sont les seules qui permettent de mettre une structure de treillis sur l'ensemble $[0,1]^{\Omega}$ des ensembles flous muni des opérations de complémentation, d'intersection et de réunion, comme celle qui existe sur l'ensemble $\{0,1\}^{\Omega}$ des ensembles classiques muni de ces mêmes opérations. Cette structure de treillis est exploitée par les algorithmes de recherche de motifs et de règles d'associations. Ce qui nous permet de faire une "fuzzification" de toutes les étapes de l'extraction des règles d'association. Afin de rester brefs, nous allons montrer sur quelques points théoriques seulement comment se fait ce passage du classique au flou.

2. Motif flou, Règle floue

A partir de maintenant, chaque fois qu'une opération sur des ensembles est utilisée, s'il s'agit d'ensembles de propriétés on prend l'opération des ensembles classiques, alors que pour les ensembles de sujets, notamment pour

les extensions des propriétés, on prend l'opération correspondante des ensembles flous, telle qu'indiquée dans la définition 1.2.

Définition 2.1. *Motif flou*
Un motif flou sur un ensemble $\mathcal{S}$ est une réunion de propriétés floues sur cet ensemble. L'extension de ce motif est obtenue en faisant l'intersection des extensions des propriétés le composant. Pour le motif vide, chaque élément de $\mathcal{S}$ a une valeur de 1. On dit qu'un élément de $\mathcal{S}$ vérifie le motif flou s'il vérifie toutes les propriétés de ce motif, ou si ce motif est le motif vide.

Définition 2.2. *Support d'un motif flou M sur un ensemble $\mathcal{S}$.*
Si toutes les valeurs des sujets de $\mathcal{S}$ pour le motif M sont inférieures ou égales à 0,5, le support de M est 0, sinon il est égal au cardinal de l'extension M' du motif M, c'est-à-dire à la somme des valeurs des sujets.

Définition 2.3. *Règle d'association floue*
On appelle règle d'association floue A→B sur un ensemble $\mathcal{S}$ un couple formé de 2 parties complémentaires A et B d'un motif flou sur $\mathcal{S}$ de support non nul. On dit qu'un élément de $\mathcal{S}$ vérifie la règle d'association floue s'il vérifie la partie gauche et la partie droite de la règle, donc le motif flou A∪B.

Maintenant que nous avons défini ce qu'est une règle d'association, il faut nous donner les moyens d'extraire les motifs puis les règles. Les algorithmes incrémentaux (comme par exemple "a priori" [BAS 00]) peuvent être utilisés de la même façon pour les motifs flous que pour les motifs classiques grâce à la propriété suivante :

Propriété 2.1. *Propriétés des motifs flous emboîtés.*
Si deux motifs flous A et B, B étant de support non nul, sont tels que A⊆B, alors on a l'inclusion B'⊆A' et l'inégalité support(B)≤support(A), l'égalité A'=B' étant vérifiée si et seulement si les supports de A et de B sont égaux.

Les algorithmes d'extraction des règles classiques à partir des motifs utilisent souvent des seuils de support et de confiance des règles [STU 02]. Pour les utiliser de façon identique, il nous suffit de définir le support et la confiance de la règle floue de façon similaire à ceux des règles classiques : le support de la règle est le support du motif, et la confiance de la règle est le quotient entre les supports de sa partie droite et de sa partie gauche. Comme pour les règles exactes, la confiance d'une règle floue A→B varie entre 0 et 1 et n'atteint 1 que quand A'⊆B'. Maintenant que nous avons donné un aperçu de la cohérence de nos définitions de règles floues d'association avec les règles d'association classiques en fouille de données, nous allons donner un aperçu de leur cohérence avec quelques éléments proches issus de la théorie du flou de Zadeh.

3. Règles obtenues par $\alpha-$coupe et règles floues

Définition 3.1. *α-coupe[1] d'une propriété floue :*
Un nombre réel α étant donné, on appelle α-coupe d'une propriété floue la propriété obtenue en remplaçant toutes les valeurs supérieures ou égales au seuil α par 1, et les valeurs inférieures à α par 0.

Si on reprend les 7 sujets du tableau 1, et si on choisit pour α une valeur de $0,8$, la propriété obtenue est vérifiée seulement pour le sujet s5, car lui seul a une valeur supérieure à 0,8. Et plus on diminue le seuil, plus le nombre de sujets vérifiant l'α-coupe augmente. Avec un seuil de 0,2, tous les sujets sauf s7 vérifient la propriété.

Une valeur α étant fixée, on étend sans problème cette définition au motif et on obtient la compatibilité suivante entre les supports des motifs et de leurs α-coupes :

Propriété 3.1. *L'intervalle de R ayant pour extrémités les valeurs extrêmes des supports des α-coupes d'un motif flou quand α décrit]0,1] contient le support du motif flou. De plus, si ce support n'est pas nul, il n'atteint les bornes de l'intervalle que lorsqu'elles sont confondues.*

1. Cette définition est reprise de la définition de [DUB 88] pour les ensembles flous.

La comparaison des règles d'associations issues des alpha-coupes des motifs et les règles d'associations floues produit une propriété équivalente pour les indices de confiance.

Nous voyons donc que notre jeu de règles floues d'association réalise une sorte de moyenne entre tous les jeux de règles d'association que nous pourrions obtenir par $\alpha-$coupes quand α varie entre 0 et 1, les jeux extrêmes étant éliminés.

4. Conclusion

Notre but en proposant ces règles d'association floues était de corriger l'explosion combinatoire des jeux de règles due à la dichotomisation des propriétés à nombreuses modalités, tout en modifiant le moins possible le processus d'extraction des règles. La construction de ces techniques diffère de l'extraction des règles d'association classique sur les points suivants :

- Comme nous l'avons dit précédemment, un choix de ce type ne peut pas se faire sans une connaissance experte des données, l'échelle devant être au minimum ordinale. De plus, les règles obtenues avec cette méthode floue sont plus générales que les règles obtenues par dichotomisation, car elles ne contiennent plus de références aux diverses modalités des propriétés. A notre avis, la transformation des propriétés par codage flou ne peut se faire sans la présence de l'expert.
- Le choix des définitions que nous avons fait pour construire des règles floues ne nous permet pas d'obtenir des treillis de Galois [GOD 95] de concepts flous. Certains auteurs [BEL 99] ont fait d'autres choix afin d'obtenir des treillis de concepts flous pour lesquels la dualité sujets/propriétés est respectée. Nous justifions notre choix par une plus grande lisibilité des concepts trouvés, et par leur nombre inférieur[2], d'où une plus grande efficacité des algorithmes.
- Par rapport à la dichotomisation, les algorithmes d'extraction des motifs flous doivent maintenant stocker en plus des libellés des sujets concernés, les coefficients correspondants. Cela multiplie le temps de traitement par un facteur constant. Par contre, le nombre de motifs potentiels est diminué de façon exponentielle. Toutefois, le gain est variable selon les bases de données, et les seuils de support choisis pour l'extraction, car le support des motifs augmente quand leur nombre diminue et inversement.

Le but principal nous parait atteint. Le jeu de règles floues obtenu sur une matrice booléenne est identique au jeu de règles classique. Et celui extrait à partir de propriétés à plusieurs modalités ordonnées selon la technique que nous venons d'exposer nous semble de meilleure qualité que celui obtenu en faisant des $\alpha-$coupes et plus synthétique que celui obtenu par dichotomisation.

5. Bibliographie

[BAS 00] Bastide Y., Data mining : algorithmes par niveau, techniques d'implantation et applications, Thèse d'informatique, Université Blaise Pascal, Clermont-Ferrand, 2000.

[BEL 99] Belohlavek R. Fuzzy Galois connections, *Mathematical logic quaterly, 45*, p. 497-504, 1999.

[DUB 88] Dubois D., Prade H.,*Théorie des possibilités*, Paris, Masson, 1988.

[GOD 95] Godin R., Mineau G., Missaoui R., Mili H., "Méthodes de classification conceptuelle basées sur les treillis de Galois et applications", *Revue d'Intelligence Artificielle, 1995, 9(2)*, p.105-137

[GUI 86] Guigues J.L. et Duquenne V. (1986) Familles minimales d'implications informatives résultat d'un tableau de données binaires, *Math. Sci. Hum. n°95*, p. 5-18

[HAN 01] Han J. and Kamber M., *Data Mining : Concepts and Techniques*, Morgan Kaufmann Publishers, 2001.

[STU 02] Stumme G., Taouil R., Bastide Y., Pasquier N., Lakhal L. Computing Iceberg Concept Lattices with Titanic, *Data & knowledge Engineering, vol 42, n°2*, p. 189-222. 2002

[ZAD 65] Zadeh, L.A. (1965). Fuzzy Sets, *Information & control., 8*, 338-353.

2. Nous en trouvons 12 selon notre définition avec les données de l'article de [BEL 99] alors qu'il en trouve 38 selon la sienne.

Comparaison d'une classification hiérarchique factorielle de variables avec des méthodes classiques(*)

Sergio Camiz, Valerio de Patta Pillar

Dipartimento di Matematica «Guido Castelnuovo»
Università di Roma «La Sapienza», Piazzale Aldo Moro, 2 - I 00185 Roma Italie
sergio.camiz@uniroma1.it

Departemento de Ecologia
Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, 91540-000, Brazil.
vpillar@ecologia.ufrgs.br

RÉSUMÉ. Pour tester la qualité d'une classification hiérarchique de variables, donnant pas à pas un plan factoriel représentant les unités statistiques, on a comparé la hiérarchie obtenue avec les hiérarchies obtenues par les méthodes du lien simple et du lien complet. On a appliqué ces méthodes sur des jeux de données simulées, de structure définie, et on a compare les résultats avec une mesure de cohérence, basée sur l'entropie, entre les partitions espérées et celles trouvées par les méthodes.

MOTS-CLÉS : Classification, Hiérarchie, Analyse factorielle, Comparaison, Données simulées.

1. Introduction

Pour l'étude d'un tableau de données quantitatives, l'approche exploratoire traditionnelle consiste en une analyse des composantes principales suivie d'une classification ascendante hiérarchique des unités statistiques, dont les classes vont être interprétées à partir des variables originales. Dans cette procédure il manque toute référence à la classification des variables, qui ne sont que des accessoires pour interpréter les facteurs. Pour ce but spécifique, tout en restant dans un cadre exploratoire, on a récemment présenté ([DEN 01] ; [CAM 02]) une méthode de classification hiérarchique factorielle de variables, qui produit à chaque pas un plan factoriel de représentation soit des variables intéressées, soit des unités, vues par ces variables seulement. La méthode se révèle particulièrement utile, car elle permet aussi de proposer une segmentation des unités associée à la hiérarchie [CAM 03] ce qui facilite l'interprétation des résultats de la part de l'utilisateur. Pour comprendre mieux le comportement de la méthode, vis-à-vis des classes de variables existantes, on a voulu la comparer avec des méthodes bien connues, celles du lien simple et du lien complet. Dans ce but on a utilisé un ensemble de 320 jeux de données simulées, dont la structure était connue afin de voir si les méthodes retrouvaient les groupes existants. Ensuite on a comparé les résultats à l'aide d'un indice de cohérence, basé sur l'entropie, afin de voir la capacité des méthodes à rendre la structure sous-jacente.

2. Les méthodes

2.1. *La classification hiérarchique factorielle*

On rappelle ici l'idée de la méthode de classification développée par Denimal ([DEN 00] ; [DEN 01] ; [CAM 03]) qu'on a étudiée : on part d'une classification ascendante hiérarchique des variables, qui, grâce à la technique utilisée, produit à chaque pas un plan factoriel où représenter soit les variables soit les unités statistiques. Ainsi, à chaque pas, on peut montrer la distribution sur le plan factoriel associé au nœud de la hiérarchie soit des variables appartenant au nœud soit des unités *vues seulement par ces variables.* On obtient ainsi une description

(*) Ce travail a été financé par les Universités d'appartenance des auteurs qui on permis leur rencontres dans le cadre des professeurs visiteurs.

plus synthétique des relations entre le nœud de la hiérarchie et les axes factoriels, qui rend plus facile à l'utilisateur le travail d'interprétation.

Plus précisément, à partir de variables centrées et réduites, chacune représentative d'elle même, à chaque pas on considère l'analyse des composantes principales (*ACP*) non-normée de tout couple de variables représentatives des groupes existants : on agrègera les deux groupes dont les variables représentatives diffèrent le moins, la première composante principale étant la variable représentative du nouveau groupe. On utilise comme mesure de cette dissimilarité, voire comme indice de la hiérarchie, la deuxième valeur propre de l'*ACP* choisie, qu'on peut démontrer être non décroissante durant le processus d'agglomération [DEN 00]. Les représentations factorielles bidimensionnelles qui en dérivent permettent aussi de classifier les unités statistiques par rapport aux groupes formés à chaque niveau, et d'interpréter les classifications en termes des similarités et dissimilarités décrites par les mêmes plans factoriels. On signale que toutes les variables représentatives sont des combinaisons linéaires des variables originales, elles sont donc dans le même espace vectoriel, mais normalement ne sont pas orthogonales entre elles. Il faut aussi noter que les groupes de variables ainsi formés sont des *dipôles*, le signe de la corrélation ne figurant pas dans le critère d'association.

2.2. *La simulation des données*

Pour tester la méthode on a utilisé des jeux de données simulées. La structure des groupes dans les données simulées a été définie par des matrices ayant différents niveaux de corrélation entre variables. La procédure de génération des données simulées ayant une corrélation définie est décrite dans [GAN 90] et a été déjà utilisée par [PIL 99]. Elle consiste en la décomposition de Cholesky d'une matrice de corrélation *C* dans le produit d'un matrice triangulaire par sa transposée : $C = L'L$; si on multiplie à gauche *L* par une matrice *U* unitaire (dans notre cas, une matrice de coordonnées résultant d'une *ACP* d'un tableau de données aléatoires), on obtient le tableau de données simulées désiré $S = UL$, car sa matrice de corrélation est $S'S = L'U'UL = L'L = C$.

2.3. *L'évaluation des résultats*

Pour tester la capacité des méthodes de rendre les groupes espérés dans tout jeu de données, les partitions espérées ont été comparées avec les partitions obtenues en appliquant les méthodes sur les jeux de données simulées. Dans ce but on a utilisé l'analyse de la variance avec test de randomisation [PIL 86]. L'accord a été mesuré avec un coefficient de cohérence basé sur l'information [ORL 91] calculé sur le tableau de contingence croisant les groupes de variables dans le deux partitions, à savoir:

$$\rho_{ik} = \sqrt{1-((H_{i+k} - H_{ik})/H_{i+k})^2}$$

où $H_{ik} = \sum_{j=1}^{s_i} \sum_{h=1}^{s_k} p_{jh} \ln[p_{jh}/(p_{j.}p_{.h})]$ est l'entropie mutuelle et $H_{i+k} = H_{ii} + H_{kk} - H_{ik}$ est l'entropie conjointe des partitions *i* et *k*, H_{ii} et H_{kk} étant l'entropie de Shannon dans chaque partition. Effectivement, ce test permet une évaluation directe de sa qualité, son étendue étant l'intervalle [0,1] et vaut 1 quand les partitions sont identiques. A l'indice on a donc associé sa probabilité, calculée à l'aide d'une méthode de randomisation.

3. Les applications

On a comparé la méthode de classification hiérarchique factorielle avec deux méthodes classiques, à savoir le lien simple et le lien complet, utilisant la matrice des valeurs absolues des corrélations, pour homogénéité avec la méthode hiérarchique factorielle où le signe de la corrélation n'a aucun intérêt, sinon dans l'identification des dipôles.

On a utilisé dans notre expérimentation un ensemble de données simulées. On a construit 4 groupes de 8 matrices de corrélations ainsi organisés : les matrices appelées *S1*, *S2* et *S3* contiennent 12 variables : dans les matrices de type *S1*, 3 groupes contiennent 4 variables chacun, la corrélation parmi les variables de groupes différent étant 0 ; les matrices de type *S2* sont semblables aux *S1* mais la corrélation parmi les 3 groupes de 4 variables est de 0.3 ; dans les matrices de type *S3*, les 3 groupes contiennent 5, 4 et 3 variables, les groupes ayant une corrélation nulle. Les matrices de type *S4* contiennent 32 variables, formant 6 groupes de 16, 8, 4, 2, 1 et 1 variables, ces groupes ayant aussi corrélation nulle. Les 8 matrices de chaque groupe sont caractérisées par une corrélation intra groupes variant entre 0.1 et 0.8, par pas de 0.1. Tous les jeux de données ont été engendrés avec 1000 unités qui reproduisaient exactement les corrélations choisies. De tout jeu on a tiré au hasard 10 échantillons de 30 unités et on a utilisé les trois méthodes de classification sur les mêmes 320 jeux de données.

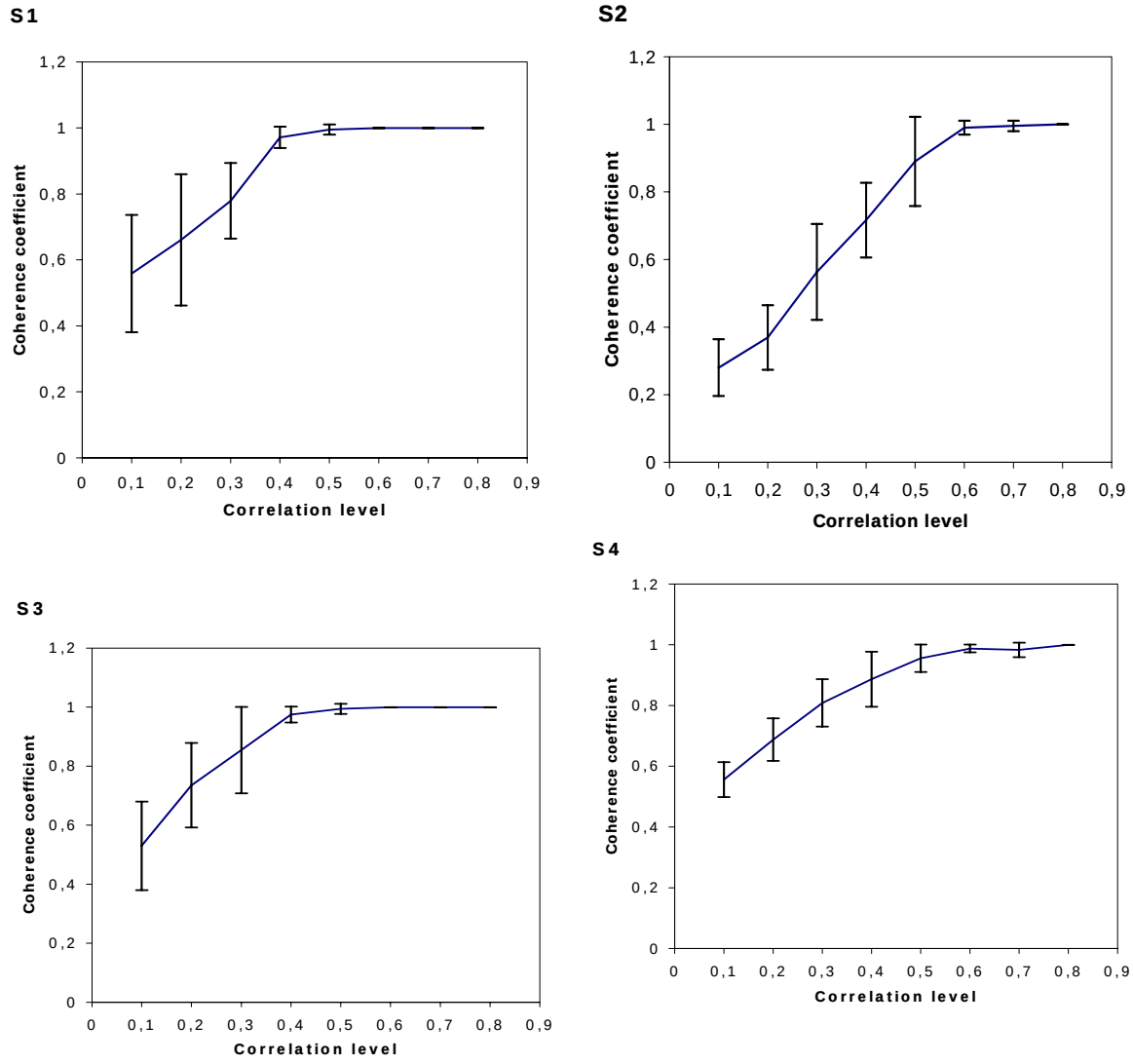


Figure 1. Variation du coefficient de cohérence moyen et de son écart type selon les niveaux de corrélation entre les variables du même groupe de la classification hiérarchique factorielle des jeux de données simulées, suivant les différentes structures. Un coefficient égal à 1 signifie agrément parfait. Ces graphiques sont à comparer avec leurs homologues issus des méthodes du lien simple et complet.

4. Les résultats

Dès que le niveau de corrélation entre les groupes simulés augmente, toutes les méthodes de classification des variables expérimentés ont été capables de révéler les partitions avec un agrément parfait avec les partitions espérées. Dans la Figure 1 on voit la moyenne et son écart type du coefficient de cohérence suivant la variation de la corrélation entre variables du même groupe, dans le cas de la classification hiérarchique factorielle étudiée. Ces graphiques sont à comparer avec leurs homologues issus des méthodes du lien simple et du lien complet. Dans les jeux avec une structure plus nette (*S1* et *S3*) l'agrément parfait a été rejoint a un niveau de corrélation plus basse que dans les jeux avec une structure moins nette (*S2* et *S4*). Les trois méthodes de classification n'ont pas été différentes dans leur performance, sinon dans le cas de groupes non bien définis dans *S2* avec une corrélation intra groupes de 0.4 (p = 0.048) et, dans une mesure mineure, dans *S1* et *S3* encore avec une corrélation intra groupes de 0.4 (p = 0.067, p = 0.058 respectivement). Dans ces cas, contrastes par paires ont indiqué que la performance de la classification hiérarchique factorielle était meilleure de celles des liens simple et complet dans *S2* et meilleure de celle du lien complet seulement dans *S1* et *S3*. Les performances des liens simple et complet n'ont pas été significativement différentes entre elles.

On peut en conclure que la méthode hiérarchique factorielle est aussi puissante que les méthodes des liens simple et complet, voire légèrement meilleure, avec l'avantage de la représentation factorielle des variables et des unités à chaque pas de la construction de la hiérarchie. On pense donc continuer l'expérimentation, en la comparant avec d'autres méthodes de la littérature, telles que la *VARCLUS* de *SAS*, les méthodes de Lerman et celles de Qannari.

5. Bibliographie

[CAM 02] CAMIZ S., DENIMAL J.J., PILLAR V.D., " Nouvelles méthodes de classification et d'aides à l'interprétation en analyse de la végétation ", Actes du IXème Congrès de la Société Francophone de Classification, Université de Toulouse Le Mirail, 2002, pp. 153-156.

[CAM 03] CAMIZ S., DENIMAL J.J., " Nouvelle technique de segmentation associée à une classification de variables ", XXXVèmes Journées de Statistique, Lyon, 2-6- juin 2003, Société Française de Statistique, Université Lumière Lyon 2, Tome 1, 2003, pp. 293-296..

[DEN 00] DENIMAL J.J., " Hierarchical Factorial Analysis ". Soumis à *Statistica Applicata.* Dipartimento di Matematica Guido Castelnuovo, Università di Roma La Sapienza, 2000, preprint n. 56/2000.

[DEN 01] DENIMAL J.J., " Hierarchical Factorial Analysis ", *Actes du 10th International Symposium on Applied Stochastic Models and Data Analysis*, Compiègne, 12-15 Juin 2001.

[GAN 90] GANENSHNANDAM S., KRZANOWSKI W.J., " Error-rate Estimation in Two-group Discriminant Analysis using Linear Discriminant Function ", *Journal of Statistical Computation and Simulation*, vol. 36, 1990, pp. 157-175.

[ORL 91] ORLOCI L., *Entropy and Information*, SPB Academic Publishing, The Hague, 1991.

[PIL 96] PILLAR V.D., ORLOCI L., " On *randomization* testing in vegetation science: multifactor comparisons of relevé groups ", *Journal of Vegetation Science*, vol. 7, 1996, pp. 585-592.

[PIL 99] PILLAR V.D., " The bootstrapped ordination reexamined ", *Journal of Vegetation Science* vol. 10, 1999, pp. 895-902.

Classification et sélection automatique de caractéristiques de textures

Marine Campedel et Eric Moulines

Ecole Nationale Supérieure des Télécommunications
Laboratoire de Traitement du Signal et des Images
46, rue Barrault, 75013 Paris
Marine.Campedel@enst.fr

RÉSUMÉ. *les outils de classification d'images utilisent des modèles variés pour représenter les textures. Nous proposons de choisir les modèles de texture les plus pertinents à l'aide d'une procédure automatique de sélection de caractéristiques. Nous comparons pour cela l'efficacité de plusieurs algorithmes à travers les performances de différents classificateurs. Nous démontrons l'intérêt d'une telle procédure de sélection à partir des images de Brodatz.*

MOTS-CLÉS : *Classification, Sélection de caractéristiques, Machines à vecteurs de support, Textures*

1. Introduction

Face à l'accroissement rapide des tailles des bases de données, en particulier des bases d'images, il est nécessaire de développer de nouveaux algorithmes de traitement facilitant à la fois le stockage et l'indexation de ces données. Nous nous intéressons dans ce travail aux algorithmes de sélection de caractéristiques (appelées aussi descripteurs) supervisés, qui permettent d'extraire une information non redondante et pertinente, en vue d'une exploitation efficace des bases de données. Ces algorithmes font l'objet d'une littérature abondante depuis une dizaine d'années [GUY 03] . Les algorithmes dits 'filters' exploitent les propriétés intrinsèques des caractéristiques utilisées, sans référence à une quelconque application. Ceux appelés 'wrappers', au contraire, définissent la pertinence des caractéristiques par l'intermédiaire d'une prédiction de la performance du système final (classification par exemple).

Nous avons choisi d'étudier quatre algorithmes : RELIEFF [KIR 92, ROB 03], FISHER (appelé aussi LDA ou analyse linéaire discriminante), RFE [GUY 02] et L0 [WES 03]. Ces trois derniers algorithmes calculent la pertinence de chaque caractéristique à l'aide des poids estimés par un classificateur linéaire (Fisher ou SVM). Nous nous intéressons aux machines à vecteurs de support (SVM) car elles limitent le risque de surapprentissage du fait de leur capacité de régularisation (ce risque étant particulièrement important lorsque le nombre de caractéristiques, i.e. la dimension, est grand face au nombre de données). Les quatre algorithmes étudiés reposent sur l'estimation de poids (scores) correspondant à chaque caractéristique. Ces poids sont utilisés pour ordonner puis sélectionner les K (parmi D) descripteurs les plus pertinents (K est fixé par l'utilisateur). Le problème du choix des bons descripteurs pour la classification d'images est un problème récurrent dans la littérature [SEB 00, RUI 01]. Nous proposons donc d'y répondre à l'aide de techniques de sélection automatiques.

Nous appliquons nos différentes procédures de sélection sur des images de textures issues de la base Brodatz, afin de déterminer les caractéristiques les plus discriminantes, parmi un ensemble calculé sur des matrices de cooccurrence (coefficients dits d'Haralick [HAR 73]), des filtres de Gabor et diverses ondelettes. L'ensemble de nos simulations s'effectue à l'aide de l'outil Matlab Spider développé par Elisseeff et Weston [WES 04]. Les textures

d'images sont calculées à partir des implantations de Boland [BOL 98], de la librairie d'ondelettes de Pyr [SIM 01] et des 'contourlets' de [DO 03].

2. Les algorithmes de sélection

Soient les données $x_i, i = 1, .., N$ et les étiquettes associées y_i. Nous ne traitons que des étiquettes discrètes. Les données sont numériques et multivaluées sur un espace initial de dimension D ($x_i \in R^D$). Nous notons w_d le score associé à la dième caractéristique. Lorsque les procédures de sélection sont relatives à un problème de discrimination linéaire, ces poids correspondent aussi aux poids du classificateur.

- RELIEFF
  Cet algorithme, introduit sous le nom de Relief dans [KIR 92] puis amélioré et adapté au cas multi-classes par Kononenko sous le nom de ReliefF, ne se contente pas d'éliminer la redondance mais définit un critère de pertinence. Ce critère mesure la capacité de chaque caractéristique à regrouper les données de même étiquette et discriminer celles ayant des étiquettes différentes. L'analyse approfondie de ReliefF est effectuée dans [ROB 03].
- FISHER
  Le deuxième algorithme choisi repose sur l'analyse discriminante linéaire (LDA) de Fisher. Nous utilisons l'implantation de l'algorithme fournie par Spider [WES 04].
- RFE
  Cet algorithme de sélection est présenté dans [GUY 02]. Il repose lui-aussi sur l'estimation de poids relatifs à l'optimisation d'un problème de discrimination linéaire, ce problème étant résolu à l'aide d'une machine à vecteurs de support (SVM). Il est montré dans [GUY 02] que le coût de suppression d'une caractéristique est de l'ordre de w_d^2. La procédure de sélection est décrémentale et élimine donc progressivement les caractéristiques de faible poids, obtenus par apprentissage d'une SVM linéaire. Nous utilisons l'implantation faite dans l'outil Spider. La procédure est grandement accélérée lorsque plusieurs caractéristiques sont éliminées simultanément et lorsque l'on stoppe la boucle d'élimination dès obtention du nombre désiré de caractéristiques.
- L0
  L'algorithme L0 présenté dans [WES 03] utilise lui-aussi les poids estimés par un classificateur SVM. L'idée générale est cependant très différente de SVM-RFE, puisqu'il s'agit dans ce cas de favoriser la mise à zero du plus grand nombre de poids. Les auteurs proposent de trouver l'ensemble minimal de caractéristiques ayant un poids non nul, en minimisant la norme L0 de ces poids. Le problème est résolu par une procédure itérative (convergeant vers un minimum local) utilisant l'apprentissage d'une SVM ainsi que la multiplication des données d'apprentissage par les poids de la SVM. Nous utilisons l'implantation Matlab des auteurs de l'algorithme.

3. Simulations

3.1. *Procédure d'évaluation*

Afin de comparer les résultats des quatre algorithmes présentés ci-dessus, nous avons choisi d'évaluer les performances en terme d'erreur de classification à l'aide de trois algorithmes classiques (Knn, Fisher, SVM). En pratique, nous effectuons une validation croisée : les quatre cinquièmes des données sont utilisées pour la sélection des caractéristiques et l'apprentissage des classificateurs, la partie restante étant utilisée pour l'évaluation. La performance est mesurée par l'erreur de classification moyennée sur les cinq ensembles (disjoints) de test.
Nous avons évalué les potentialités de l'ensemble de la procédure sur un problème synthétique (deux caractéristiques sont pertinentes, 6 autres sont des versions bruitées, les 42 restantes étant purement du bruit uniformément distribué sur [0 1]. Nous disposons de 400 données réparties en deux classes). Les résultats sont présentés figure 1(gauche). Les principales remarques sont :

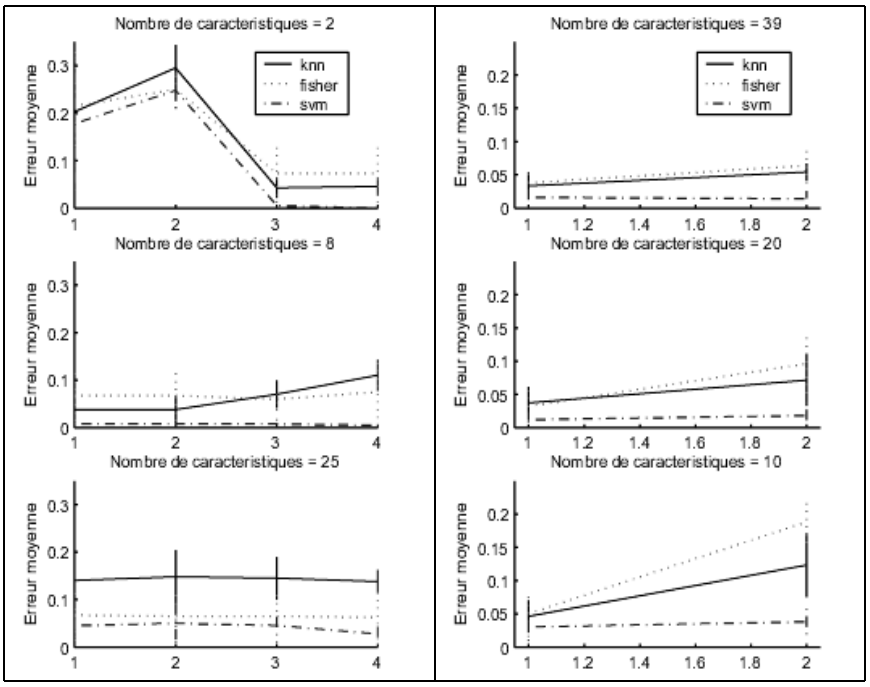


FIG. 1. *Valeurs moyennes et écarts types (barres verticales) obtenus sur 5 tirages. A gauche, le problème est synthétique, les méthodes de sélection sont désignées par un index correspondant à :1=RELIEFF 2=FISHER 3=RFE 4=L0. A droite, sur les images de Brodatz, les méthodes sont 1=FISHER 2=RFE*

- Le classificateur SVM obtient toujours les meilleures performances. Ceci est dû à sa capacité intrinsèque à gérer les éléments bruités ;
- RELIEFF et FISHER ne parviennent pas à sélectionner les deux descripteurs pertinents, contrairement à L0 et RFE ;
- Lorsque 8 descripteurs sont sélectionnés, les deux pertinentes sont toujours dedans. Par contre, RELIEFF et FISHER privilégient l'ajout de caractéristiques redondantes, alors que L0 et RFE prépondèrent les dimensions de bruit ;
- La performance dégradée pour 25 descripteurs met bien en évidence la nécessité de réduire les dimensions redondantes ou bruitées ;
- L0 et RFE ont des performances semblables, mais L0 présente l'inconvénient d'être beaucoup plus lent.

3.2. *Modèles de texture*

L'objectif de cette simulation n'est pas de rendre un verdict définitif sur le choix des modèles de textures. Il s'agit de décrire une méthodologie de sélection automatique. Bien souvent les experts ont des préférences différentes quant au modèle de texture à utiliser. Estimer simultanément tous ces modèles introduit une redondance préjudiciable au niveau du stockage des caractéristiques comme au niveau de la performance de classification. Nous mettons donc en évidence l'intérêt d'une sélection automatique des caractéristiques, parmi un ensemble présélectionné. Comme base de travail nous avons tiré aléatoirement 20 images de textures de Brodatz. Nous les avons décomposées en 25 imagettes disjointes de taille 128x128. Chaque imagette est ensuite décrite par un vecteur de caractéristiques, résultant de la concaténation de plusieurs vecteurs de textures. Les modèles de textures utilisés sont : les coefficients d'Haralick (13 types de statistiques calculées sur la matrice de co-occurrence), des coefficients de Gabor et divers coefficients d'ondelettes (moyennes et variances sur chaque sous-bande). Au total, nous disposons de 234 caractéristiques normalisées (moyenne nulle et variance unitaire).

La meilleure performance de classification de cet ensemble de caractéristique est de $2.4\% \pm 1.8\%$ pour un classificateur SVM. Lorsque seules les caractéristiques d'Haralick sont évaluées, on descend à une erreur moyenne de $1.6\% \pm 1.7\%$. Nous appliquons deux des procédures de sélection, FISHER et RFE, aux descripteurs d'Haralick (cf figure 1). Les mêmes performances de classification sont obtenues lorsqu'on ne conserve que 20 des 78 caractéristiques. En outre, il est intéressant de constater dans ce cas que les deux algorithmes ne sélectionnent pas les mêmes caractéristiques, bien que la performance SVM soit constante, ce qui signifie que l'information utile n'est pas perdue. Enfin, nous constatons que RFE tend à être plus sélective sur le type de statistique appliqué à la matrice de co-occurrence.

4. Conclusion

Nous avons montré dans cette étude la nécessité d'appliquer une procédure de sélection automatique de caractéristiques en vue d'une tâche de classification. Nous avons comparé différents algorithmes de sélection (supervisés) récents à l'aide des erreurs de classification induites. Nous avons montré leur efficacité à l'aide d'un problème synthétique ainsi que d'un problème de classification d'images de textures.
Nous préconisons l'usage d'une procédure de sélection, non seulement pour réduire l'espace de stockage et améliorer les performances de classification, mais aussi pour justifier le choix d'un modèle donné de caractéristiques. Dans l'étude présentée ci-dessus, nous avons appliqué cette stratégie à la sélection de modèles de textures.
Dans la plupart des systèmes impliquant un grand nombre de données et de caractéristiques, les étiquettes associées aux données ne sont pas connues. Nous nous intéressons donc maintenant aux algorithmes de sélection non supervisés, qui exploite la capacité à clusteriser des données multivaluées ainsi qu'aux heuristiques permettant d'évaluer ces algorithmes.

5. Bibliographie

[BLU 97] BLUM A., LANGLEY P., Selection of relevant features and examples in machine learning, *Artif. Intell.*, vol. 97, n^o 1-2, 1997, p. 245–271, Elsevier Science Publishers Ltd.

[BOL 98] BOLAND M., Programmation en C des coefficients d'Haralick, 1998.

[DO 03] DO M., VETTERLI M., Contourlets, *Beyond Wavelets*, Academic Press, 2003.

[GUY 02] GUYON I., Gene Selection for Cancer Classification using Support Vector Machines, *Journal of Machine Learning Research*, vol. 46, 2002, p. 389-422.

[GUY 03] GUYON I., ELISSEEFF A., An introduction to feature and variable selection, *Journal of Machine Learning Research*, vol. 3, 2003, p. 1157-1182.

[HAR 73] HARALICK R., SHANMUGAM K., DINSTEIN I., Textural features for image classification, *IEEE Transactions on Systems, Man, and Cybertinetics*, vol. 6, 1973, p. 610-621.

[KIR 92] KIRA K., RENDELL L., A practical approach to feature selection, *Proceedings of the International Conference on Machine Learning*, vol. 1, 1992, p. 249-256.

[KOH 97] KOHAVI R., JOHN G., Wrappers for feature subset selection, *Artif. Intell.*, vol. 97, n^o 1-2, 1997, p. 273-324, Elsevier Science Publishers Ltd.

[ROB 03] ROBNIK-SIKONJA M., KONONENKO I., Theoretical and Empirical Analysis of ReliefF and RReliefF, *Journal of Machine Learning Research*, vol. 53, n^o 1-2, 2003, p. 23-69, Kluwer Academic Publishers.

[RUI 01] RUI Y., HUANG T., CHANG S., Image Retrieval : current techniques, promising directions and open issues, *Journal of Visual Communication and Image Representation*, vol. 10, 2001, p. 39-62.

[SEB 00] SEBE N., LEW M., Wavelet Based texture Classification, *IEEE International Conference on Pattern Recognition*, vol. 3, 2000.

[SIM 01] SIMONCELLI E., MatLab tools for multi-scale image processing, 2001.

[WES 03] WESTON J., ELISSEEFF A., SCHOLKOPF B., TIPPING M., Use of the Zero-Norm with Linear Models and Kernel Methods, *Journal of Machine Learning Research*, vol. 3, 2003, p. 1439-1461.

[WES 04] WESTON J., ELISSEEFF A., BAKIR G., SINZ F., The Spider for Matlab - v1.4, 2004.

Sur la normalisation pour la classification de données intervalles

Marie Chavent

Mathématiques Appliquées de Bordeaux, UMR 5466 CNRS
Université Bordeaux1 - 351, Cours de la liberation
33405 Talence Cedex
chavent@math.u-bordeaux.fr

RÉSUMÉ. *L'objectif de ce travail est de proposer plusieurs mesures de dispersion d'une variable décrite par des intervalles. On pourra en particulier utiliser ces mesures de dispersion pour normaliser le tableau de données intervalles ou encore de manière équivalente la distance utilisée dans l'algorithme de classification.*

MOTS-CLÉS : *Données symboliques intervalles, standardisation, distance normalisée, classification.*

1. Introduction

Dans un tableau de données $(x_i^j)_{n\times p}$ où n individus $\{1, ..., i, ..., n\}$ sont décrits par p variables quantitatives, lorsque toutes les variables ont des unités de mesures différentes, l'utilisation des variables telles quelles dans le calcul de la distance donnera de façon implicite plus de poids aux variables de plus forte dispersion, anihilant presque complètement l'effet des autres variables.
Une approche classique pour obtenir une classification ne privilégiant pas uniquement les variables de forte dispersion est de standardiser les données variable par variable. Or, une fois les variables centrées par la moyenne $\bar{x}^j$ (ou encore la médiane) et réduite par l'écart-type σ^j (ou encore l'étendue, l'intervalle interquartile...), la distance euclidienne entre i et i' s'écrit :

$$\sqrt{\sum_{j=1}^{p}(\frac{x_i^j-\bar{x}^j}{\sigma^j}-\frac{x_{i'}^j-\bar{x}^j}{\sigma^j})^2}=\sqrt{\sum_{j=1}^{p}\frac{1}{(\sigma^j)^2}(x_i^j-x_{i'}^j)^2} \qquad [1]$$

Les classifications obtenues sur les données centrées-réduites (ou même simplement réduites) avec la distance euclidienne simple et les classifications obtenues sur les données brutes avec la distance euclidienne normalisée par l'inverse de la variance sont équivalentes (cf. [1]). D'une manière plus générale, pour une distance basée sur des différences (écarts au carré, en valeur absolue...), la classification des données brutes ou centrées est la même. Nous ne nous intéressons donc pas ici au problème du centrage de variables intervalles. D'autre part, la classification obtenue avec les données brutes et la distance normalisée (i.e. distance "pondéré") est généralement la même que celle obtenue avec les données normalisées et la distance "simple" (non pondérée).
Dans ce travail, nous nous sommes intéressés au problème de la normalisation d'un tableau de données intervalles où chaque individu i est décrit sur chaque variable j par un intervalle

$$x_i^j=[a_i^j,b_i^j]\in I=\{[a,b]\mid a,b\in\Re\ ,\ a\leq b\}$$

Chaque individu i est ainsi décrit par un hyper-rectangle de $\mathbb{R}^p$:

$$x_i=\prod_{j=1}^{p}[a_i^j,b_i^j]$$

Il s'agit d'un cas particulier de données symboliques [DID 88], [BOC 00].
Le problème de la normalisation de données symboliques avait déjà été posé dans [CHA 97]. L'écriture de la variance comme une double somme pondérée des écarts était utilisée afin de définir la mesure de dispersion suivante :

$$\sigma^j = \frac{1}{2n} \sum_{i=1}^{n} \sum_{i'=1}^{n} d^2(x_i^j, x_{i'}^j) \quad [2]$$

où d est une fonction de comparaison entre deux descriptions symboliques quelconques. Cette mesure de dispersion y était utilisée pour définir une distance normalisée du type :

$$(\sum_{j=1}^{p} \frac{1}{(\sigma^j)^\alpha} d(x_i^j, x_{i'}^j)^\alpha)^{1/\alpha} \quad [3]$$

Le principal inconvénient de cette mesure de dispersion est la double somme qui peut rendre ce critère long à calculer pour de volumineuses bases de données.
On retrouve cette question de la normalisation de données intervalles dans [DEC 03] où les mesures de dispersion utilisées sont basées sur la dispersion des centres, des bornes supérieures ou inférieures des intervalles ou encore sur le maximum des bornes supérieures et le minimum des bornes inférieures.
Une approche parfois utilisée dans les algorithmes de classification de données intervalles est de considérer chaque intervalle comme un point de $\Re^2$, de comparer ces deux points avec la distance L_1 ou la distance L_2 en utilisant pour comparer deux hyper-rectangles :

$$\sum_{j=1}^{p} (d(x_i^j, x_{i'}^j)^\alpha)^{1/\alpha} \quad [4]$$

avec $\alpha = 1$ pour la distance L_1 et $\alpha = 2$ pour la distance L_2. Cette distance (avec $\alpha = 1$ et la distance L_1) est utilisée par exemple dans [DES 04] pour définir un algorithme de classification de type Nuées Dynamiques. Finalement, cela revient à créer dans le tableau de données initial deux colonnes indépendantes pour les bornes inférieures et les bornes supérieures des intervalles. La notion d'intervalle n'est donc pas vraiment prise en compte avec ce type de distance et cela revient à un recodage du tableau de données intervalles. On retrouve ainsi un tableau de données quantitatives classiques et des mesures de dispersion connues.
L'idée ici est donc double : utiliser une distance plus "spécifique" à la notion d'intervalle et s'affranchir de la double somme dans le calcul de la mesure de dispersion [2]. Afin de répondre à ce double objectif et dans la continuité des articles de [BOC 01], [CHA 02], [CHA 04], la distance de Hausdorff a été choisie pour définir des mesures de dispersion autour d'un centre optimal.

2. Mesures de dispersion autour d'un centre optimal

Pour une variable quantitative classique, la variance mesure la dispersion autour de la moyenne $\bar{x}^j$ qui est la solution optimale $\hat{y}$ du problème de minimisation suivant :

$$\min_{y \in \mathbb{R}} \sum_{i=1}^{n} (x_i^j - y)^2 \quad [5]$$

La variance s'écrit donc (à un coefficient près) :

$$f(\hat{y}) = \min_{y \in \mathbb{R}} \sum_{i=1}^{n} d^2(x_i^j, y) \quad [6]$$

De même, l'écart moyen à la médiane mesure la dispersion autour de la médiane x_M^j qui est la solution optimale $\hat{y}$ du problème de minimisation suivant :

$$\min_{y \in \mathbb{R}} \sum_{i=1}^{n} |x_i^j - y| \quad [7]$$

L'écart moyen à la médiane s'écrit donc (à un coefficient près) :

$$f(\hat{y}) = \min_{y \in \mathbb{R}} \sum_{i=1}^{n} d(x_i^j, y) \quad [8]$$

Si l'on se place maintenant dans le cas où $x_i^j = [a_i^j, b_i^j]$ et $y = [\alpha, \beta]$, plusieurs mesures de dispersion autour d'un centre optimal $\hat{y} = [\hat{\alpha}, \hat{\beta}]$ peuvent être définies selon la distance d choisie pour comparer deux intervalles et la fontion f choisie pour mesurer cette dispersion. Ici, la distance choisie pour comparer deux intervalles est la distance de Hausdorff. Cette distance d_H définie pour deux ensembles quelconques se simplifie dans le cas de deux intervalles [CHA 97] :

$$d_H([a_i^j, b_i^j], [a_{i'}^j, b_{i'}^j]) = \max(|a_i^j - a_{i'}^j|, |b_i^j - b_{i'}^j|) \quad [9]$$

La distance de Hausdorff est donc le maximum des écarts entre les bornes supérieures et les bornes inférieures des intervalles soit la distance L_∞ entre les vecteurs (a_i^j, b_i^j) et $(a_{i'}^j, b_{i'}^j)$.

2.1. *Mesure de "l'étoile"*

On se place dans le cas où la mesure de dispersion autour de $\hat{y}$ est :

$$f(\hat{y}) = \min_{y \in I} \sum_{i=1}^{n} d_H(x_i^j, y) \quad [10]$$

On retrouve ici une formulation proche de la mesure d'homogénéité d'une classe C dite mesure de "l'étoile" :

$$\min_{i \in C} \sum_{j \in C} d_{ij}$$

Dans [CHA 02] on démontre par un recodage des intervalles $[a_i^j, b_i^j]$ en fonction de leur milieu m_i^j et de leur demi-longueur l_i^j que l'intervalle $\hat{y}$ qui minimise $\sum_{i=1}^{n} d_H(x_i^j, y)$ a pour milieu $\hat{\mu}$ et pour demi-longueur $\hat{\lambda}$ avec :

$$\hat{\mu} = mediane\{m_i^j \mid i = 1, ..., n\} \quad [11]$$

$$\hat{\lambda} = mediane\{l_i^j \mid i = 1, ..., n\} \quad [12]$$

On définit alors la mesure de dispersion σ^j suivante :

$$\sigma^j = \sum_{i=1}^{n} \max(|a_i^j - \hat{\mu} + \hat{\lambda}|, |b_i^j - \hat{\mu} - \hat{\lambda}|) \quad [13]$$

2.2. *Mesure du "rayon"*

On se place dans le cas où la mesure de dispersion autour de $\hat{y}$ est :

$$f(\hat{y}) = \min_{y \in I} \max_{i=1...n} d_H(x_i^j, y) \quad [14]$$

On retrouve ici une formulation proche de la mesure d'homogénéité d'une classe C dite mesure du "rayon" :

$$\min_{i \in C} \max_{j \in C} d_{ij}$$

Dans [CHA 04], on démontre que l'intervalle $\hat{y}$ qui minimise $\max_{i=1\ldots n} d_H(x_i^j, y)$ a pour borne inférieure et supérieure :

$$\hat{\alpha}^j = \frac{\max_{i=1\ldots n} a_i^j + \min_{i=1\ldots n} a_i^j}{2} \quad [15]$$

$$\hat{\beta}^j = \frac{\max_{i=1,\ldots,n} b_i^j - \min_{i=1,\ldots,n} b_i^j}{2} \quad [16]$$

On définit alors la mesure de dispersion σ^j suivante :

$$\sigma^j = \max_{i=1\ldots n} \max(|a_i^j - \hat{\alpha}^j|, |b_i^j - \hat{\beta}^j|) \quad [17]$$

3. Application en classification

Ces deux mesures de dispersion [13] et [17] peuvent être utilisées en classification. Par exemple dans les deux méthodes de type Nuées Dynamiques proposées dans [CHA 02] et [CHA 04] la classification obtenue avec le tableau des intervalles normalisés $z_i^j = [\frac{a_i^j}{\sigma^j}, \frac{b_i^j}{\sigma^j}]$ et la classification obtenue avec la distance normalisée (i.e. pondérée par l'inverse de σ^j) sont les mêmes. En effet,

$$d_H(z_i^j, z_{i'}^j) = \max(|\frac{a_i^j}{\sigma^j} - \frac{a_{i'}^j}{\sigma^j}|, |\frac{b_i^j}{\sigma^j} - \frac{b_{i'}^j}{\sigma^j}|) = \frac{1}{\sigma^j} d_H(x_i^j, x_{i'}^j) \quad [18]$$

Dans [CHA 02], la distance d entre deux hyper-rectangles est la somme sur toutes les variables des distances de Hausdorff entre les intervalles et on a donc :

$$d(z_i, z_{i'}) = \sum_{j=1}^{p} \frac{1}{\sigma^j} d_H(x_i^j, x_{i'}^j) \quad [19]$$

Dans [CHA 04], la distance d entre deux hyper-rectangles est le maximum sur toutes les variables des distances de Hausdorff entre les intervalles et on a donc :

$$d(z_i, z_{i'}) = \max_{j=1\ldots p} \frac{1}{\sigma^j} d_H(x_i^j, x_{i'}^j) \quad [20]$$

4. Bibliographie

[BOC 00] BOCK H.-H., DIDAY E., Eds., *Analysis of symbolic data. Exploratory methods for extracting statistical information from complex data*, Studies in classification, data analysis and knowledge organisation, Springer Verlag, Heidelberg, 2000.

[BOC 01] BOCK H.-H., Clustering algorithms and kohonen maps for symbolic data, *ICNCB Proceedings*, Osaka, 2001, p. 203–215.

[CHA 97] CHAVENT M., Analyse des données symboliques. Une méthode divisive de classification, PhD thesis, Université Paris-IX Dauphine, 1997.

[CHA 02] CHAVENT M., LECHEVALLIER Y., Dynamical clustering of interval data. Optimization of an adequacy criterion based on hausdorff distance, JAJUGA K., SOKOLOWSKI A., BOCK H.-H., Eds., *Classification, Clustering, and Data Analysis*, Berlin, 2002, Sringer Verlag, p. 53–60.

[CHA 04] CHAVENT M., An Hausdorff distance between hyper-rectangles for clustering interval data, *IFCS Proceedings*, Chicago, 2004.

[DEC 03] DE CARVALHO F. A. T., BRITO P., BOCK H.-H., Une méthode Type Nuées Dynamiques pour les données symboliques quantitatives, DODGE Y., MELFI G., Eds., *Méthodes et Perspectives en Classification*, Presses Académiques Neuchatel, 2003, p. 79-81.

[DES 04] DE SOUZA R. M. C. R., DE CARVALHO F. A. T., Clustering of interval data based on city-block distances, *Pattern Recognition Letters*, vol. 25, 2004, p. 353-365.

[DID 88] DIDAY E., The symbolic approach in clustering and related methods of data analysis : The basic choices, BOCK H.-H., Ed., *Classification and related methods of data anlysis*, Amsterdam, 1988, North Holland, p. 673–684.

La Classification Pyramidale Symbolique : Sélection de paliers et de variables

Mohamed Cherif Rahal, Edwin Diday

CEREMADE, Université Paris Dauphine
Place du Maréchal de Lattre de Tassigny,
75775 Paris Cedex, {rahal, diday}@ceremade.dauphine.fr

RÉSUMÉ. Le but de ce travail est de faciliter l'interprétation d'une pyramide. On sélectionne les " meilleures " classes de la pyramide en utilisant les "sauts" d'un palier au suivant, et on les décrit par des variables que l'on sélectionne également en utilisant le "degré de généralité", qui les "expliquent" le mieux possible. Une simulation montre ensuite comment évoluent ces sélections quand le nombre de paliers et de variables croit.

MOTS-CLÉS : Classification pyramidale, Classification Hiérarchique, Interprétation d'une classification.

1. Introduction

En 1984, Diday a proposé l'algorithme CAP (**C**lassification **A**scendante **P**yramidale) [DID 84], [BER 86] par analogie à l'algorithme CAH [BEN 73] et a montré que dans le cas hiérarchique une représentation de groupes non disjoints est plus fidèle et riche en information par rapport aux données initiales qu'une représentation en classes disjointes. L'introduction de l'analyse de données symboliques [DID 87], [BOC 00], a conduit à étendre les méthodes d'analyse de données à variables numériques ou qualitatives à des données mieux adaptées à la description de concepts dont l'extension est formée d'individus décrits de façon standard à l'aide de variables dites "symboliques" ; par exemple à valeur intervalle, ensemble de valeurs parfois pondérées et munies de règles et de taxonomies. Pour la classification pyramidale symbolique (ie. à chaque palier de la pyramide est associé un objet symbolique), Brito [BRI 91] propose en utilisant le **degré de généralités** (ie. mesure de dissimilarité de type symbolique) une généralisation de l'algorithme CAP en gardant les mêmes principes d'agrégation. Nous pouvons également citer les algorithmes CAPS (Classification Ascendante Pyramidale Symbolique) et CAPSO (Classification Ascendante Pyramidale Symbolique avec Ordre donnée) présentés par Rodriguez dans [ROD 00]. D'autres travaux ont été élaborés dans le cadre de la sélection des variables sur un ensemble d'objets symboliques [ZIA 96], [VIG 91] et [LEB 91].
Le but principal de ce travail est de faciliter l'interprétation des pyramides. Pour cela d'une part nous réduisons le nombre de palier à étudier et à visualiser en sélectionnant ceux qui se distinguent le plus du reste de la population. De plus nous facilitons l'interprétation de ces derniers car on sélectionne les variables les plus significatives. Enfin, une simulation montre le comportement de ces sélections quand la taille augmente.

2. Algorithme de sélection des paliers

La représentation pyramidale induit souvent un grand nombre de classes, d'ordre $n \times (n-1)/2$, où n est le nombre d'individus à classer. L'objectif ici est de pouvoir restreindre le nombre de paliers en choisissant les plus significatifs. Dans ce qui suit, on dit que $P_{père}$ est le père de P_{fils} (resp. P_{fils} est le fils de $P_{père}$) si et seulement si $P_{fils} \subset P_{père}$ (inclusion stricte) et qu'il n'existe pas de paliers intermédiaires. Dans la figure FIG. 1 : Le palier *P7* est le père de *P5* et *P6*. On appelle *saut*, la différence de hauteur entre 2 paliers fils et père (par exemple, on voit FIG.1 que le saut entre P9 et P8 est 0.1111-0.1085= 0.0026). Enfin palier est dit significatif si le saut entre ce dernier et son plus bas père est grand, ou plus précisément dépasse la moyenne des sauts.
L'algorithme que nous présentons a pour but la recherche de ce type de paliers. Cet algorithme utilise en entrée

un ensemble appelé Pyramide, qui contient tous les paliers de la pyramide dans l'ordre de leur construction, les paliers les plus significatifs sont ajoutés à un deuxième ensemble appelé P_int. Cette insertion se fait en comparant les sauts entre chaque palier et son plus bas père (PBP) (Fig.1). Si ce dernier est supérieur ou égal au à la moyenne des sauts, ce palier est ajouté à l'ensemble des paliers intéressants (P_int). Autrement dit : P_int = {∀ Pi ∈ Pyramide / Sp(Pi) ≥ Moy(Sp(Pk))} où Sp(Pi) est le saut du palier Pi par rapport à son plus bas père.
Exemple : Soit le tableau de données symbolique suivant :

	Y_1	Y_2	Y_3	Y_4
ω1	[1, 2]	[2, 3]	[1, 2]	[0, 1]
ω2	[7, 9]	[7, 10]	[0, 6]	[11, 20]
ω3	[4, 5]	[0, 7]	[-6, 3]	[-30, 30]
ω4	[-1, 0]	[-7, 2]	[-3, 3]	[10, 100]

TAB 1 - Tableau de données symboliques

Après le déroulement de l'algorithme de Pak [PAK 04] de classification pyramidale symbolique, nous obtenons la pyramide de la figure FIG.1 – *a*, et après sélection des paliers nous obtenons la pyramide de la figure FIG.1 - *b*:

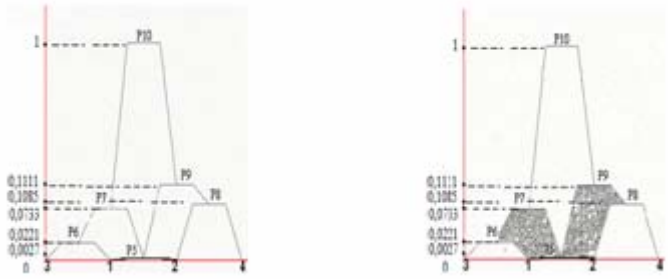


FIG. 1 - a - Pyramide symbolique b - Pyramide après sélection de paliers.

Pour cet exemple l'ensemble *Pyramide* contient tous les paliers de la pyramide : *{1, 2, 3, 4, P5, P6, P7, P8, P9, P10}*. Après le déroulement de l'algorithme de sélection on ne garde que les paliers *{P5, P7, P9}* qui ont des sauts minimums supérieurs à la moyenne des sauts.

3. Algorithme de sélection des variables

Vu que le nombre de variables, pouvant expliquer un objet ou un individu à classer peut être très grand, le second objectif de ce travail est de réduire l'ensemble des variables descriptives pour chaque classe, autrement dit, trouver les variables qui "*expliquent*" le mieux possible un palier.

3.1. *Degré de généralité d'un palier*

Soit *S(P)* l'objet symbolique associé au palier *P*, qui est décrit par la conjonction $\bigwedge_{j=1}^{k} e_j$, où e_j est un événement élémentaire ($e_i = [y_j = v_j]$, v_j est la valeur prise par la variable y_j dans son espace d'observation O_j). Le degré de généralité [BRI 91] de *S(P)* est défini par : $DG(S(P)) = \prod_{j=1}^{k} g(e_i)$, ($\in[0,1]$). Où $g(e_i) = C(v_j)/C(O_j)$, *C* est : 1) le cardinal si y_j est un ensemble de valeurs discrètes, 2) la longueur si y_j est une variable à valeur intervalle, 3) la moyenne des fréquences si y_j est une variable à valeur histogramme.
Toutes les variables y_i sont parcourues, pour chaque palier de la liste *P_int*, on calcule la différence du degré de généralité de l'événement élémentaire associé à cette dernière (dans le palier père et le palier fils) comme suit : $R(y_j) = DG_{père}(e_j) - DG_{fils}(e_j)$ où $R(y_j) \in [0,1]$, où $DG_{fils}(e_j)$ est le degré de généralité de l'événement élémentaire associé à la variable y_j dans le palier P_{fils}. On détermine un critère de sélection de variables pour chaque palier intéressant, $Sc(P_{fils})$, en considérant que seules les variables qui satisfont $R(y_j) \geq Sc(P_{fils})$ seront sélectionnées, où $Sc(P_{fils})$ est défini par : $Sc(P_{fils}) = \frac{\sum_{j=1}^{k} R(y_j)}{k}$.

La contribution de la variable y_j dans la variation entre le palier en question (P_{fils}) et son père ($P_{père}$) est égale au rapport entre $R(y_j)$ et la différence entre les degrés de généralité globaux (des deux paliers). Cette dernière est calculée par $contr(y_j) = \frac{R(y_j)}{DG(P_{père}) - DG(P_{fils})}$. Le dénominateur $DG(P_{père}) - DG(P_{fils})$, n'est jamais nul car on ne prend en considération que les paliers de grands sauts.

Après avoir calculé les contributions de toutes les variables pour un palier, il est facile de calculer le pourcentage de contribution pour chacune, par : $\%contr(y_j) = \frac{contr(y_j)}{\sum_{i=1}^{k} contr(y_i)} \times 100\%$.

3.2. *Exemple récapitulatif*

Nous calculons pour chaque classe et pour chaque variable son degré de généralité ($DG(y_j)$), le rapport R ainsi que son pourcentage de contribution ($\%C$), ces calculs sont résumés dans le tableau ci-dessous :

	Y1			*Y2*			*Y3*			*Y4*			
	DG	R	*%C*	*DG*	R	*%C*	*DG*	R	*%C*	*DG*	R	*%C*	*Seuil*
P5	*0,3*	*0,1*	*9,25*	*0,24*	***0,35***	*32,63*	*0,5*	***0,5***	*22,56*	*0,08*	***0,38***	*35,56*	*0,33*
P7	*0,4*	*0,1*	*9,52*	*0,59*	***0,41***	*39,21*	*1*	*0*	*0*	*0,46*	***0,54***	*51,27*	*0,26*
P9	*0,4*	*0,1*	*8,92*	*0,53*	***0,47***	*41,99*	*0,75*	*0,25*	*22,31*	*0,7*	***0,3***	*26,77*	*0,28*

TAB 2 - Tableau résumant les calculs de l'algorithme 2.

Dans le tableau Tab.2, on résume les calculs de l'algorithme précédemment présenté, en effet pour le palier P_5 nous gardons les variables Y_2, Y_3 et Y_4 car *R(Y3)> R(Y4)>R(Y2)> 0.33 (seuil)*. Donc l'ensemble *Vars* pour le palier *P5* est *: Vars*(P_5*)= {*Y_2*, *Y_3*, *Y_4*}*. Idem pour les autres paliers de l'ensemble *P_int : Vars (*P_7*)= {*Y_2*, Y4}, Vars (*P_9*)= {*Y_2*, Y4}*. Avant la sélection des paliers intéressants (figure FIG.2 - *a*) nous distinguons six (6) classes correspondant au différents paliers de la pyramide. Remarquons que le plus bas palier de la pyramide $P5 = [y_1 = [1,9]] \wedge [y_2 = [7,10]] \wedge [y_3 = [0,6]] \wedge [y_4 = [0,20]]$ regroupe les deux objets les plus proches (ω1, ω2) et s'écarte par rapport au reste de la population par un saut minimum égal à 2.58 10^{-1} (PBP = *P7*) qui est supérieur à la moyenne des sauts (2.49 10^{-1}), donc ce palier est significatif. La représentation, en terme d'objets symboliques, des trois classes significatives obtenues, est : $P5 = [y_1 = [1,9]] \wedge [y_2 = [7,10]] \wedge [y_3 = [0,6]] \wedge [y_4 = [0,20]]$, $P7 = [y_1 = [1,9]] \wedge [y_2 = [0,10]] \wedge [y_3 = [-6,6]] \wedge [y_4 = [-30,30]]$, $P9 = [y_1 = [-1,9]] \wedge [y_2 = [-7,10]] \wedge [y_3 = [-3,6]] \wedge [y_4 = [0,100]]$.

Après avoir choisi les variables significatives pour chaque palier *Pi* (on notera *P'i* les objets symboliques (assertions) relatifs aux paliers après sélection des variables), nous pouvons représenter les paliers précédemment décrits comme suit:

$P'5 = [y_2 = [7,10]] \wedge [y_3 = [0,6]] \wedge [y_4 = [0,20]]$, $P'7 = [y_2 = [0,10]] \wedge [y_4 = [-30,30]]$, $P'9 = [y_2 = [-7,10]] \wedge [y_4 = [0,100]]$.

Nous calculons les extensions des objets symboliques obtenus après sélection de variables (TAB.2), et nous vérifions que nous obtenons les mêmes résultats qu'avec les objets symboliques relatifs aux paliers *P5, P7, P9,* obtenus avant sélection des variables.

Ext (P5) = Ext(P'5)= {ω1, ω2}, Ext(P7) = Ext(P'7)= {ω1, ω2, ω3}, Ext(P9)=Ext(P'9)={ω1, ω2, ω4}.

4. Résultats expérimentaux

Afin de tester les méthodes précédemment présentées, nous avons utilisé un jeu de test, ce dernier contient plus de 200 pyramides construites à partir de données tirées au hasard (bases de données simulées), les pyramides sont construites à partir de 4 jusqu'à 20 individus à classer. Pour la sélection des paliers significatifs, nous remarquons d'après les courbes de la figure FIG.2-gauche, que le nombre de paliers à visualiser (les paliers significatifs) représente entre 6 et 14% du nombre total des paliers construits par l'algorithme de classification pyramidale. Par exemple pour 20 individus de départ nous avons obtenu en moyenne 204 classes dont 30 significatives. Pour la sélection des variables, nous avons utilisé un autre jeu de test en faisant varier le nombre de variables explicatives pour chaque sous-ensemble de ce dernier. Dans la figure FIG.2-droite, nous remarquons que le nombre de variables sélectionnées est nettement plus petit que le nombre total des variables (d'ordre 15 à 20 %), car pour la plupart des paliers construits il suffit qu'une variable change pour que la hauteur entre un palier et son père croit.

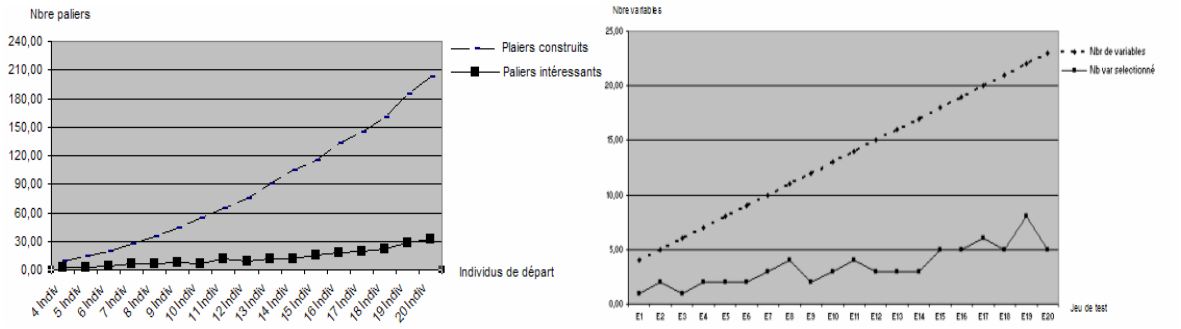


FIG. 2 - résultats des tests - gauche - sélection des paliers -droite – sélection des variables

On remarque enfin que la pente du nombre de paliers ou du nombre de variables est plus grande que celle du nombre de paliers sélectionné ou du nombre de variables retenues.

5. Conclusion

Dans cet article, nous avons présenté deux méthodes pour l'aide à l'interprétation des pyramides, la première consiste à réduire le nombre de classes en choisissant parmi l'ensemble celles qui s'écartent le plus par rapport au reste de la population. La seconde est une méthode de sélection de variables qui permet de trouver le sous-ensemble de variables qui fait qu'une classe s'écarte par rapport aux autres. Il s'est avéré, dans la pratique, que les deux méthodes sont utiles, elles permettent une meilleure interprétation que les méthodes actuelles, d'une part par rapport au nombre de paliers à visualiser, et d'autre part par rapport au nombre de variables explicatives.

6. Bibliographie

[BEN 73] J.P.BENZECRI. *L'analyse de données*. Dunod, Paris, 1973

[BER 86] P.BERTRAND. *Etude de la représentation pyramidale.* Thèse de doctorat, Université Paris IX-Dauphine, 1986.

[BOC 00] H.H.BOCK ET E.DIDAY. Exploratory methods for extracting statistical information from complex data. Springer, Verlag 2000.

[BRI 91] P.BRITO. *Analyse de données symboliques: Pyramides d'héritage.* Thèse de doctorat, Université Paris IX Dauphine, 1991.

[DID 84] E.DIDAY *Une représentation visuelle des classes empiétant*. Rapport INRIA n-291. 1984.

[DID 87] E.DIDAY. Introduction à l'approche symbolique en Analyse des Donnés. *Premières Journées Symbolique-Numérique*. Université Paris IX Dauphine. Décembre 1987.

[LEB 91] J.LEBBE. *Représentation des concepts en biologie et en médecine*. Thèse de doctorat, Université Pierre et Marie Curie.

[PAK 04] K.K.PAK. *La classification pyramidale symbolique : planaires et spatiales.* Thèse de doctorat, CEREMADE, Université Paris Dauphine, à paraître en Juin 2004.

[ROD 00] O.RODRIGUEZ. *Classification et modèles linéaires en analyse de données symboliques*, Thèse de 3 cycle, Université Paris IX- Dauphine, 2000.

[VIG 91] R.VIGNES. *Caractérisation automatique de groupes biologiques.* Thèse de doctorat, Université Pierre et Marie Curie.

[ZIA 96] D.ZIANI. *Sélection de variables sur un ensemble d'objets symboliques*, Thèse de doctorat, Université Paris IX- Dauphine, 1996.

DynaSpat : Plate-forme d'intégration de méthodes statistiques et évolutionnistes pour l'analyse des dynamiques spatiales

Chettouh R.*, Coelho S., Duthen Y.* et Kettaf F.-Z.***

*{chettouh,duthen,kettaf}@irit.fr
IRIT, UMR CNRS 5505
118 Route de Narbonne
31062 Toulouse Cedex 4

***scoelho@univ-tlse1.fr*
UT1 Sciences sociales, place Anatole
Manufacture du tabac.
31042 Toulouse cedex.

RÉSUMÉ. DynaSpat est une plate-forme de développement d'applications pour la résolution des problèmes d'optimisation dans l'espace. Il s'agit d'applications décisionnelles telles que le géomarketing, l'analyse de la criminalité ou encore l'analyse des risques. Elle fusionne plusieurs logiciels et bibliothèques dans le but d'identifier des variables pertinentes dans un jeu de données hétérogènes et lance ensuite un système de simulation basé sur les algorithmes génétiques. Cette plate-forme est le fruit d'un projet de recherche pluridisciplinaire, financé par la région Midi-Pyrénées. Il réunit trois laboratoires de recherche : L'IRIT (UT1-UPS), le laboratoire de mathématiques GREMAQ(UT1) et le laboratoire d'economie LEREPS (UT1), ainsi qu'un partenaire industriel CS (Communication et Systèmes).

MOTS-CLÉS: Optimisation multicritères, Fouille de données, Algorithmes génétiques, Analyse exploratoire, Analyse spatiale.

1. Introduction

De nos jours, la quantité de données stockées augmente considérablement et l'information pertinente devient de plus en plus difficile à obtenir. Cette difficulté freine considérablement le processus de prise de décision. C'est pourquoi, une automatisation des tâches d'exploration et d'exploitation des données est nécessaire, tant pour la rapidité du processus que pour sa qualité. Au cours de ces dix dernières années, plusieurs disciplines (et plus particulièrement la statistique et l'apprentissage artificiel) ont servi à la résolution de ces problèmes. Des systèmes variés ont vu le jour, ils sont efficaces mais leur utilisation est, souvent, limitée à leurs domaines d'application. La résolution de problèmes complexes faisant appel à différentes disciplines nécessite une intégration judicieuse des différents modules. L'objectif de ce projet est de mettre au point une plate-forme pluridisciplinaire (DynaSpat) basée sur la notion du territoire. Elle constitue un environnement de développement d'applications et de résolution de problèmes ayant des contraintes spatiales. DynaSpat est aussi un outil global mettant à la disposition de l'utilisateur un système d'information géographique (GéoConcept), un ensemble de techniques d'analyse exploratoire de données géoréférencées (GéoXP) et un outil de simulation évolutionniste et d'optimisation (AGMC). Ces trois composants communiquent entre eux afin de permettre une analyse minutieuse et flexible. Le principal

apport de cette démarche est d'introduire, à tous les niveaux de la réflexion, le territoire comme facteur déterminant.
La figure1 montre la structure globale de la plate-forme [COE 04], elle est constituée de trois modules principaux : le stockage des données, le traitement de données et la visualisation. La section suivante détaille chacun de ces modules.

2. DynaSpat

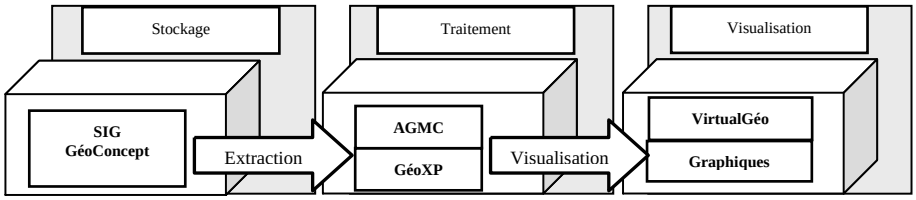


Figure 1: Structure de la plate-forme

2.1. *Stockage des données*

Le stockage et l'accès aux différentes données sont assurés par le SIG « GéoConcept ». Un SIG est un système informatique permettant, à partir de diverses sources d'informations, de rassembler et d'analyser des informations localisées géographiquement afin de contribuer, notamment, à la gestion de l'espace. D'une manière générale, un SIG doit répondre à quatre fonctionnalités [BUR 86]: la saisie, le stockage, l'analyse et la visualisation des données. GéoConcept permet, non seulement, le stockage de données qui servent à l'analyse statistique et spatiale et à l'optimisation d'une problématique, mais aussi la représentation 2D du territoire sur lequel sera effectué l'ensemble des analyses spatiales.

2.2. *Traitement des données*

2.2.1 *Fouille de données*

Hand et al. [HAN 01] donnent la définition suivante de la fouille de données : « Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner ». Rappelons que les tâches principales de la fouille de données peuvent être : descriptives (visualisation, réduction de la dimension de l'espace de représentation, etc.), prédictives (classification de nouvelles données), orientées vers la régression comme outils d'analyse de la dépendance entre variables, ou tournées vers la découverte de règles d'association [FRE 02] [FAY 96] [FRE 98]. De plus, pour atteindre ces objectifs, les données sont soumises à des pré-traitements [PYL 99] et les résultats d'analyse à des post-traitements. Certains post-traitements (souvent des règles) sont basés sur des critères subjectifs et sur l'interaction avec l'utilisateur [LIU 97], d'autres sont purement objectifs [GEB 91].
La sélection d'attributs est l'un des traitements les plus exécutés d'un processus de découverte des connaissances. Dans le cadre de la classification, ce problème consiste en la sélection, parmi tous les attributs, d'un sous-ensemble d'attributs pour la prédiction de la classe.
Dans le cadre de ce projet, nous nous intéressons plus particulièrement à la réduction des données, à la découverte de règles prédictives et d'association ainsi qu'à la fusion des données pour résoudre des problèmes d'analyse de dynamiques spatiales complexes.

2.2.2 *Analyse statistique et spatiale*

L'analyse exploratoire de données géoréférencées [WIS 01] doit prendre en compte leur dimension spatiale. Malheureusement, les systèmes d'information géographique ne disposent pas d'outils statistiques très sophistiqués adaptés aux données spatiales [CRE 93]. GéoXP (développé par le GREMAQ) offre une panoplie d'outils d'analyse statistique spatiale permettant d'intégrer une carte et un ensemble de graphiques statistiques [HEB 03]. Elle incorpore un bon nombre d'outils d'analyse exploratoire (géostatistique ou économétrie spatiale) et permet de lier à la carte un ensemble de graphiques tels que des histogrammes, des

diagrammes de dispersion, des nuages de variogramme, des diagrammes de Moran, des courbes de Lorenz, etc. Ce composant inclut également des techniques purement statistiques telles que l'étude des tendances de variables, des phénomènes d'auto-corrélation spatiale ou encore la détection de points aberrants.

2.2.3 *Optimisation*

L'optimisation multi-objectifs est un problème difficile. L'intégration de plusieurs critères permet de diriger la décision vers un compromis judicieux plutôt qu'un optimum souvent désuet [SCS 84]. Les algorithmes génétiques et les méthodes multi-agent au sein de la librairie AGMC constituent une bonne alternative à la résolution de ce type de problèmes. Cette bibliothèque permet d'optimiser un problème et de prendre en compte les contraintes spatiales. Les problématiques pouvant être intégrées sont nombreuses et très intéressantes dans des domaines variés tels que l'optimisation de placements, la simulation comportementale, les phénomènes d'agglomération urbaine et périurbaine, l'analyse spatiale des marchés, etc. De plus, l'intégration d'un moteur de visualisation 3D (VirtualGéo), développé par le partenaire industriel du projet (CS), permet de gérer l'ensemble de la visualisation et de l'interaction sur les résultats 2D ou 3D obtenus.

2.3. *Visualisation*

L'utilisation de la cartographie pour la visualisation est limitée à certains types de données. En effet, GéoConcept ne prend en compte que des données bidimensionnelles. L'intégration d'un moteur de visualisation 3D « VirtualGéo » permet donc à l'utilisateur de survoler un territoire et de découvrir un ensemble d'objets correspondant soit à des données brutes, soit à des données issues de traitements DynaSpat. L'utilisateur accède donc à une forme d'information beaucoup plus abordable et immédiate.

3. Applications

La distribution spatiale d'une trentaine de casernes sur la Haute Garonne est un problème d'optimisation qui a été résolu par cette plate-forme. Une représentation génétique appropriée a été établie [COE 04] sur les données fournies par les pompiers (des informations sur des milliers de sinistres). Le résultat de l'optimisation, obtenu grâce à la bibliothèque AGMC, a permis de trouver les meilleurs emplacements des casernes ainsi que les capacités et les rayons d'action associés. Ces résultats sont visualisés en 3D par VirtualGéo. La figure 2 donne un aperçu sur cette représentation. Chaque caserne est représentée par un cône dont le rayon est proportionnel au rayon d'action. La couleur représente la capacité de gestion des sinistres.

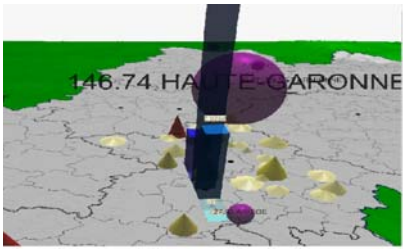


Figure 2 - Représentation graphique des centres des pompiers.

4. Conclusion et perspectives

DynaSpat est une plate-forme de simulation et de visualisation pour la description, l'analyse et la modélisation des comportements économiques. Le premier prototype constitue un fort potentiel pour l'identification de données pertinentes et pour la simulation de stratégies. Des premières applications ont validé cette approche et ont fait émerger de nouvelles idées. La fouille de données, et plus particulièrement, la sélection des attributs et la découverte de règles d'association couplées aux méthodes génétiques nous semblent être des voix prometteuses.

5. Bibliographie

[BUR 86] BURROUGH P.A., “Principles of geographical information systems for land resources assessment”, Oxford, Clarendon Press, 1986.

[COE 04] COELHO S., DUTHEN Y., THOMAS-AGNAN C. « La plate-forme DynaSpat : Les Dynamiques Spatiales. EGC'2004, 2e Atelier Visualisation et extraction de connaissances. Site : http://visu.egc.free.fr/

[CRE 93] CRESSIE N., ”Statistics for spatial data”, John Wiley & Sons, New York, 1993.

[FAY 96] FAYYAD U.-M., PIATETSKY-SHAPIRO G., SMYTH P., “From data mining to knowledge discovery: an overview, Advances in Knowledge Discovery \& Data Mining”, 1-34, AAAI/MIT, 1996.

[FRE 98] FREITAS A.-A., LAVINGTON S.-H., “Mining Very Large Databases with Parallel Processing”. Kluwer, 1998.

[FRE 02] FREITAS A.-A., “A survey of evolutionary algorithms for data mining and knowledge discovery”, Advances in Evolutionary Computation, A. Ghosh and S. Tsutsui, 819-845, Springer-Verlag, 2002.

[GEB 91] GEBHARDT F., “Choosing among competing generalizations”, Knowledge Acquisition 3, 1991, 361-380.

[HAN 01] HAND D., MANNILA H., PADHRAIC S., “Principles of Data Mining”, MIT Press, Cambridge, Massachusetts, London, England, 2001.

[HEB 03] HÉBA I., MALIN E., THOMAS-AGNAN C., “Exploratory Spatial Data Analysis with GeoXp”, submitted to Geographical Analysis, 2003.

[HOL 75] HOLLAND J.H., ”Adaptation in Natural and Artificial Systems”, University of Michigan Press, Ann Arbor, 1975. Republished by the MIT Press, 1992.

[LIU 97] LIU B., HSU W., CHEN S., ”Using general impressions to analyze discovered classifications rules”. Proc. 3rd Int. Conf. Knowledge Discovery & Data Mining, 31-36. AAAI Press, 1997.

[PYL 99] PYLE D., “Data Preparation for Data Mining”. Morgan Kaufmann, 1999.

[SCS 84] SCHAFFER, J.D., "Some experiments in machine learning using vector evaluated genetic algorithms", Vanderbilt University, 1984, (Dissertation Abstracts International, 46; University, Microfilms No. 85-22492).

[WIS 01] WISE S., HAINING R., MA J., “Providing spatial statistical analysis functionality for the GIS user : the SAGE project”, Int. J. Geographical Information Science, pp. 239-254, 2001.

Consensus moyen et consensus en norme infinie : deux méthodes pour les hiérarchies indicées

Guy Cucumel

École des sciences de la gestion
Université du Québec à Montréal
CP 8888, Succursale Centre-Ville
Montréal (Québec)
H2T 1M5
Canada
courriel : cucumel.guy@uqam.ca

RÉSUMÉ. Les méthodes de recherche de consensus sont largement employées pour combiner des hiérarchies obtenues à partir de plusieurs jeux de données portant sur les mêmes observations. De nombreux algorithmes, dont le consensus moyen ont été proposés au cours des dernières décennies pour construire des hiérarchies consensus. Nous proposons ici de comparer cette approche avec le consensus en norme infinie.

MOTS-CLÉS : classification hiérarchique, consensus moyen, consensus en norme infinie

1. Introduction

La recherche d'un consensus en classification hiérarchique consiste à associer à un profil de k hiérarchies $P=\{H_1, H_2, \ldots, H_k\}$ définies sur un même ensemble Ω de m objets une hiérarchie H_c représentative (en un certain sens) des hiérarchies initiales ([LEC 87] et [LEC 98]). Depuis le premier algorithme proposé par Adams [ADA 72], l'utilisation de hiérarchies consensus a largement augmenté et a donné naissance à de nombreux algorithmes dont certains s'appliquent à des profils de hiérarchies indicées ([NEU 83], [STI 84], [FIN 85], [BAR 86], [CUC 90], [LAP 97] et [CHE 00]). Nous proposons de comparer deux approches : le consensus moyen (consensus en norme L_2) et le consensus en norme infinie.

2. Consensus moyen

Le consensus moyen, originalement proposé par Cucumel [CUC 90], consiste à minimiser la somme des carrés des distances (au sens de la distance de Hartigan [HAR 67]) entre chacune des hiérarchies indicées du profil P et la hiérarchie consensus. Ce problème est équivalent à trouver l'ultramétrique la plus proche (au sens de la norme L_2) de la dissimilarité obtenue en calculant la moyenne terme à terme des ultramétriques associées aux hiérarchies du profil initial P [LAP 97]. Il s'agit d'un problème NP-complet qui ne peut être résolu qu'en utilisant un algorithme de type « branch and bound » [CHA 84]. Cet algorithme, qui est une généralisation de l'algorithme du lien moyen, conduit à une solution qui n'est pas nécessairement unique. Lorsque le nombre d'objets à classifier est grand, il faut avoir recours à une solution approchée. L'ultramétrique associée à la hiérarchie du lien moyen, à laquelle conduit une des branches de l'algorithme développé par Chandon [CHA 84], est une solution approchée possible.

3. Consensus en norme infinie

Une nouvelle approche applicable à des profils de hiérarchies indicées a été développée par Chepoi et Fichet [CHE 00]. Ils proposent un algorithme dont la complexité est en m^4 (où m est le nombre d'objets sur lesquels

sont définies les hiérarchies indicées) qui conduit à une solution unique : le consensus en norme infinie. Nous en présentons les étapes ci-dessous[1].

Notons d_1, d_2, …, d_k les k ultramétriques associées aux k hiérarchies indicées du profil P et $||.||$ la norme infinie.

Étape 1 : calculer $u = \inf(d_1, d_2, \ldots, d_k)$, minimum terme à terme, $v = \sup(d_1, d_2, \ldots, d_k)$, maximum terme à terme et u* la sous-dominante de u
calculer $e = ||v - u||/2$ et calculer u_1 en retranchant e à chaque terme de v et v_1 en ajoutant e à chaque terme de u*
poser n =1

Étape 2 : calculer $e_n = ||u_n - v_n||/2$
si $e_n = 0$, fin de l'algorithme, $u_n = v_n$ est le consensus en norme infinie

Étape 3 : calculer t_n en retranchant e_n à tous les terme de v_n
calculer $u_{n+1} = \sup(u_n, t_n)$
calculer s_n en ajoutant e_n à tous les terme de u_n
calculer v_{n+1} la sous-dominante de $\inf(u_n, t_n)$
poser n=n+1 et retour à l'étape 2

L'algorithme converge en au plus m^2 étapes. Comme la complexité du calcul des sous-dominantes est en m^2 également, l'algorithme a une complexité en m^4.

4. Exemple

Les deux méthodes sont appliquées pour la recherche d'un consensus entre les trois hiérarchies H_1, H_2 et H_3 de la figure 1[2]. Nous montrons également une solution approchée du consensus moyen, obtenue par l'algorithme du lien moyen.
Le consensus en norme infinie a la même structure que le consensus majoritaire et ne retient que deux paliers $\{x_1, x_2\}$ et Ω et donc n'accepte que peu de compromis. L'indice associé à Ω (2.5) est plus faible que ceux qui lui sont associés dans les hiérarchies initiales ce qui a comme conséquence que certains objets comme x_4 et x_6 ou x_2 et x_5 se trouvent beaucoup plus proches dans le consensus que dans les hiérarchies du profil initial.
Le consensus moyen et son approximation ont tous deux la même structure. Il est intéressant de noter que dans ce cas particulier l'approximation est excellente car seul l'indice associé au palier Ω diffère légèrement d'un consensus à l'autre. Le consensus moyen est nécessairement binaire par construction, ce qui présente l'inconvénient de forcer certains regroupements. Comme pour le consensus en norme infinie, la proximité de x_1 et x_2 dans les hiérarchies initiales ressort bien. La proximité relative entre x_1 et x_3 d'une part et x_5 et x_6 d'autre part dans les hiérarchies initiales est également bien représentée dans le consensus. Par contre, la structure binaire du consensus moyen fait apparaître x_2 et x_3 assez proches, ce qui est questionnable compte tenu de la position relative de ces deux objets dans les hiérarchies initiales. Il en est de même pour x_4 et x_5.

[1] Nous remercions Bernard Fichet pour nous avoir communiqué une version détaillée de l'algorithme.
[2] Cet exemple a été utilisé par Chepoi et Fichet pour le calcul d'un consensus en norme infinie.

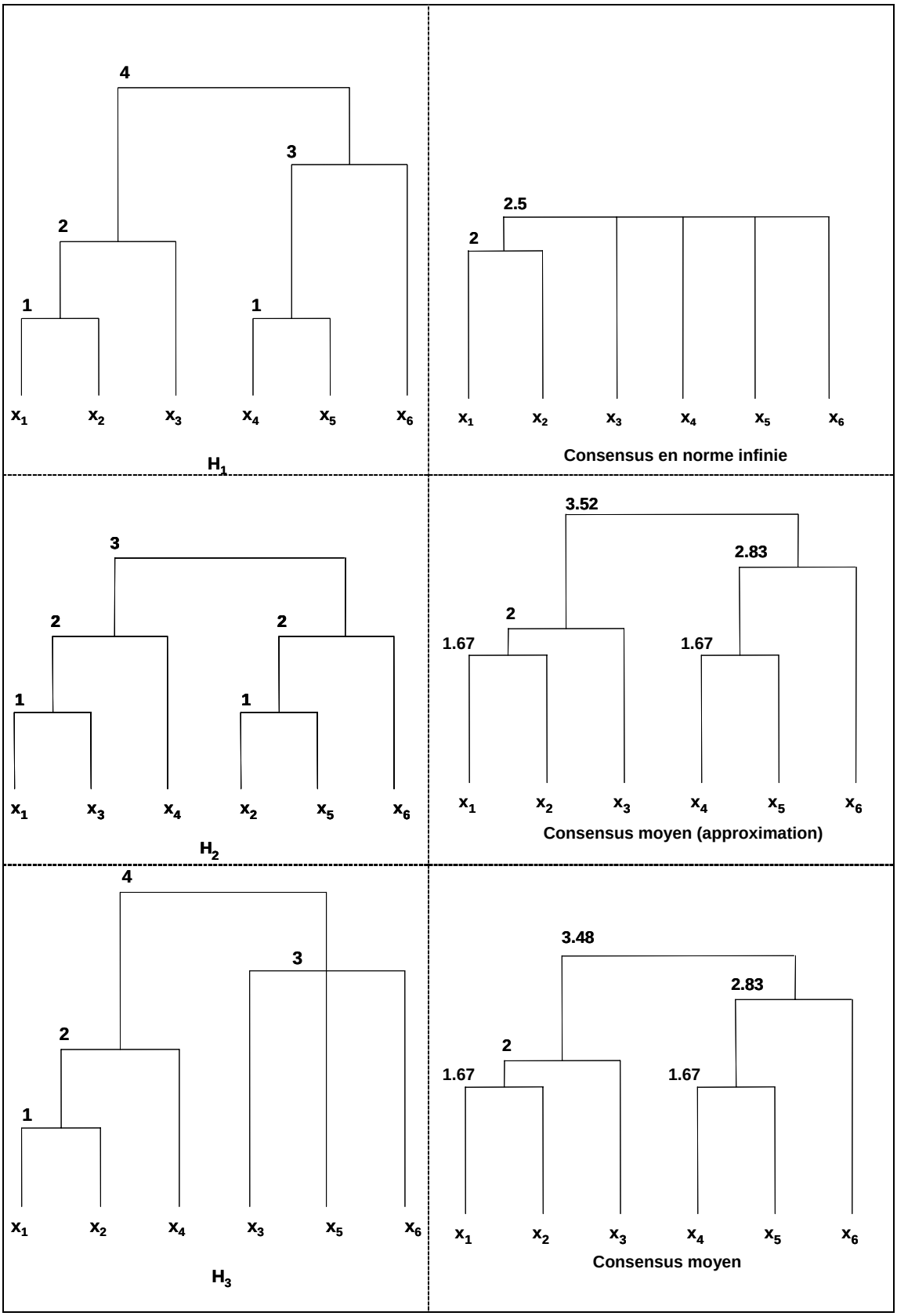


Figure 1. Hiérarchies H_1, H_2 et H_3 et consensus en norme infinie et moyen associés

Sur cet exemple, le consensus en norme infinie semble un peu drastique alors que le consensus moyen apparaît pour sa part un peu trop permissif. L'algorithme de construction du consensus en norme infinie présente

néanmoins l'avantage incontestable d'être polynomial. Une étude empirique sur des jeux de données réelles est nécessaire afin de mettre en évidence les avantages et inconvénients de chacune des méthodes.

5. Bibliographie

[ADA 72] ADAMS E. N. III., "Consensus Techniques and the Comparison of Taxonomic Trees", Systematic Zoology, vol. 21, 1972, p. 390-397.

[BAR 86] BARTHÉLEMY J.-P., MCMORRIS F. R., "The Median Procedure for n-Trees", Journal of Classification, vol. 3, 1985, p. 229-334.

[CHA 84] CHANDON J.-L., DE SOETE G., "Fitting a Least Squares Ultrametric to Dissimilarity Data: Approximation versus Optimisation", Data Analysis and Informatics III, E. Diday et al. (eds.), 1984, p. 213-219, Elsevier Science Publishers B.V., Amsterdam.

[CHE 00] CHEPOI, V., FICHET, B., "L_∞-Approximation via Subdominants". Journal of Mathematical Psychology, 44, 2000, p. 600-616.

[CUC 90] CUCUMEL, G., "Construction d'une hiérarchie consensus à l'aide d'une ultramétrique centrale", Recueil des textes des présentations du colloque sur les méthodes et domaines d'application de la Statistique 1990, 1990, p. 235-243, Bureau de la Statistique du Québec, Québec.

[FIN 85] FINDEN C. R., GORDON A. D., "Obtaining Common Pruned Trees", Journal of Classification, vol. 2, 1985, p. 225-276.

[HAR 67] HARTIGAN J. A., "Representation of Similarity Matrices by Trees", Journal of the American Statistical Association, vol. 62, 1985, p. 1140-1158.

[LAP 97] LAPOINTE F.-J., CUCUMEL G., "The Average Consensus Procedure : combination of Weighted Trees Containing Identical or Overlapping Sets of Objects", Systematic Zoology, vol. 46, 1984, p. 306-312.

[LEC 87] LECLERC B., CUCUMEL G., "Consensus en classification : une revue bibliographique", Mathématiques et sciences humaines, n° 100, 1987, p. 109-128.

[LEC 98] LECLERC B., "Consensus of Classification : the Case of Trees", Data Analysis, Classification and Related Methods, H. A. L. Kiers et al. (eds.), 1998, p. 81-90, Springer-Verlag, Berlin.

[NEU 83] NEUMANN D. A., "Faithful Consensus Methods for n-Trees", Mathematical Biosciences, vol. 63, 1983, p. 271-287.

[STI 84] STINEBRICKNER R., "An Extension of Intersection Methods from Trees to Dendrograms", Systematic Zoology, vol. 33, 1984, p. 381-386.

Décomposition d'un tournoi en classes acycliques.

Jean-Francois Culus et Bertrand Jouve

Équipe GRIMM
Université Toulouse Le Mirail
UFR SES, Département de Math-Info
31058 Toulouse Cedex
culus@univ-tlse2.fr et jouve@univ-tlse2.fr

RÉSUMÉ. *La coloration d'un graphe est un problème bien connu, recherchant à regrouper des individus dans des classes "stables" (sans arêtes dans une classe). Mais le caractère symétrique du graphe ne nous permet pas de modéliser un certain nombre d'intéractions qui sont naturellement antisymétriques. En particulier, nous pouvons souhaiter regrouper les individus dans des ensembles ordonnés tels que tous les individus d'une même classe aient mêmes intéractions avec les indivudus hors cette classe. Nous étudions dans ce travail une notion de décomposition d'un graphe orienté permettant de regrouper les individus dans des classes prenant en compte ces demandes. Nous présentons ici des aspects algorithmiques et de complexité combinatoire de ce problème.*

MOTS-CLÉS : *décomposition d'un tournoi, ordre, coloration, complexité.*

1. Introduction

Lorsque l'on cherche à généraliser la notion classique de coloration d'un graphe non-orienté à des graphes orientés, nous trouvons dans la littérature deux approches différentes : l'une, due à [SOP 01], consiste à considérer la décomposition d'un graphe orienté G en classes stables (sans arcs entres les sommets d'une même classe) mais en imposant l'unidirection entre les différentes classes monochromatiques (tous les arcs entres deux classes monochromatiques doivent avoir la même direction). L'autre, due à [NEU 82] ou (indépendamment) à [BOK 03], consiste à trouver la plus petite partition (en nombre de classes) des sommets du graphe orienté G en classes acycliques, sans conditions sur les arcs entre les classes monochromatiques.
Nous proposons alors l'introduction de la notion de k-décomposition d'un graphe orienté G comme étant la partition de l'ensemble des sommets G en k classes telle que : (1)- les classes monochromatiques soient acycliques et (2)- les classes monochromatiques sont en unidirection les unes par rapport aux autres (ie : tous les arcs entre deux classes de colorations sont dans le même sens).
La k-décomposition d'un graphe orienté G proposée ici est donc une décomposition médiane entre les deux généralisations précédement citées.

2. Exemple

Un zoo fictif dispose de n primates à répartir de telle manière à reconstituer des clans. Par l'observation du comportement des primates, il est aisé de connaître, étant donné deux individus A et B, leur relation hierarchique : soit *A domine B,* soit *B domine A* soit *A et B sont indifférents l'un à l'autre*. Une première necéssité consiste à former des clans hierarchisés (indépendamment les uns des autres) : il est donc nécessaire que le graphe des relations entre les individus d'un même clan soit acyclique. Mais, lors de la mise en présence des différents clans dans un même espace, une seconde condition pour s'assurer de la stabilité des clans est que les différents membres d'un même clan aient une attitude similaire face aux individus d'un autre clan. Ainsi, par exemple, tout individu du

clan C_1 devra dominer ou être indifférent aux individus du clan C_2 si l'un des membres de C_1 domine l'un des membres de C_2.
Rechercher à trouver la répartition des primates optimale afin d'obtenir le plus petit nombre de clans possible revient à trouver le plus petit entier k tel qu'il existe une k-décomposition du graphe des relations entre les différents primates.

3. Complexité

Soit k un entier. Nous pouvons montrer qu'il existe une réduction de Turing du problème de k-décomposition au problème de k'-coloration orienté dû à [SOP 01]. De plus, il existe une réduction (similaire à la première) de ce problème au problème classique de k''-coloration d'un graphe non orienté. Ce dernier étant NP-complet, la NP-complétude des deux premiers s'ensuit.

4. Le cas des tournois

Remarquons déjà que pour T tournoi, les classes monochromatiques obtenues par une k -décomposition sont des ordres.
Soit x un sommet de T. Nous appelons (s'il existe) *plus grand successeur de* x le sommet x^+ de T défini par : $\Gamma^+(x^+) = \Gamma^+(x) \setminus \{x\}$, où $\Gamma^+(x)$ désigne l'ensemble des successeurs du sommet x. Notons que, dans le cas des tournois, le plus grand successeur de x est nécessairement unique s'il existe.
Nous montrons que si x^+ n'existe pas, alors le sommet x est nécessairement le plus petit sommet dans l'ordre induit par sa classe monochromatique.
De plus, nous pouvons montrer que, si le sommet x^+ existe, nous pouvons le supposer être dans la même classe monochromatique que x sans augmenter le nombre de couleurs k de la solution.
Cela nous conduit à démontrer qu'il y a unicité de la k-décomposition minimale étant donné un tournoi T.
Il existe des propriétés symétriques pour le plus petit prédécesseur x^- de x, défini par : $\Gamma^-(x^-) = \Gamma^-(x) \setminus \{x^-\}$.
Ces considérations nous amènent à introduire l'algorithme suivant :

Entrée Un tournoi T.
Sortie L'entier k minimum tel qu'il existe une k-décomposition de T ainsi que cette k-décomposition, donnée par les marques $\{M(x), x \in V(T)\}$.
Initialisation : Pour tout sommet $x \in T$, $M(x) = 0$.
Soit k un entier, initialement nul.
Tant qu'il existe un sommet x de marque $M(x) = 0$, faire :
$k \leftarrow k + 1$
$v \leftarrow x$.
$M(v) \leftarrow k$.
Tant que v^+ existe faire :
$v \leftarrow v^+$
$M(v) = k$.
Tant que v^- existe :
$v \leftarrow v^-$
$M(v) = k$.

Ainsi obtenons-nous un algorithme polynomial permettant de donner la k-décomposition minimale d'un tournoi T. La famille des tournois est donc une classe polynomiale pour ce problème.

5. Tournois indécomposables.

Par un argument probabiliste, nous pouvons démontrer que la probabilité qu'un tournoi aléatoire ayant n sommets soit indécomposable tend vers 1 quand n tend vers $+\infty$. A titre d'exemple, les tournois réguliers sont indécomposables (car quelque soit le sommet $x \in T$, les sommets x^+ et x^- ne peuvent exister).
C'est pourquoi nous nous intéressons plus spécifiquement à ces tournois. En nous inspirant de [NEU 84], nous définissons parmi ceux-ci les tournois sommets-critiques indécomposables (SCI) comme étant les tournois T indécomposables tels que pour tout sommet x de T, le sous-tournoi $T \setminus \{x\}$ soit k'-décomposable, avec $k' \leq |V(T)| - 2$.
Si T est SCI, alors pour u sommet de T, le sous-tournoi $T \setminus \{u\}$ est par définition décomposable, donc il existe deux sommets $\{i_u, j_u\}$ de $T \setminus \{u\}$ ayant mêmes successeurs et mêmes prédécesseurs dans $T \setminus \{u\}$.
A ce tournoi T SCI, nous associons alors le graphe non orienté G défini par :
• Les sommets de G sont les sommets de T.
• Les arêtes de G sont les $\{i_u, j_u\}_{u \in V(T)}$.
Rappelons la définition d'un tournoi circulant : $C_{2j+1}(1, 2, 3, ..., j)$ est le $(2j + 1)$-tournoi dont l'ensemble des sommets est $\{1, 2, 3, ..., 2j + 1\}$ et dont l'ensemble des arcs est $\{uv; 0 \leq v - u \leq j \ \ mod[2j + 1]\}$.
Nous proposons la caractérisation des tournois SCI suivante :
T est SCI si et seulement si le graphe G associé à T est union de tournois circulants.

6. Bibliographie

[BOK 03] BOKAL D., FIJAVZ G., JUVAN M., KAYLL P. M., MOHAR B., The circular chromatic number of a digraph, *Journal of Graph Theory*, , 2003.

[NEU 82] NEUMANN-LARA V., The dichromatic Number of a digraph, *Journal of combinatorial Theory, Series B*, vol. 33, 1982, p. 265-270.

[NEU 84] NEUMANN-LARA V., Vertex critical r-dichromatic tournaments, *Discrete Mathematics*, vol. 498, 1984, p. 83-87.

[SOP 01] SOPENA E., Oriented graph coloring, *Discrete Mathematics*, vol. 229, 2001, p. 359-369.

Un noyau pour séquences inspiré de modèles markoviens

Marco Cuturi, Jean-Philippe Vert

Groupe de Biologie Computationnelle
Ecole des Mines de Paris
35 rue Saint-Honoré 77305 Fontainebleau cedex, France
marco.cuturi@ensmp.fr,jean-philippe.vert@ensmp.fr

RÉSUMÉ. *Nous proposons dans cet exposé un nouveau noyau pour séquences. Etant donné deux chaînes X et Y, ce dernier est construit en considérant les vraisemblances de X, Y, puis de leur concaténation XY évaluées sur une large famille de modèles paramètriques markoviens. Ces dernières sont ensuite moyennées selon une approche bayésienne sur les divers paramètres de nos modèles, une approche inspirée de la théorie du codage universel et de la compression. Son calcul rapide, d'une complexité linéaire en temps et espace mémoire, permet de mener des expériences sur des données biologiques, à travers des méthodes à noyaux telles que les machines à vecteur de support.*

MOTS-CLÉS : *Noyau d'information mutuelle, Codage Universel, Méthodes à Noyaux*

1. Introduction

Les *méthodes à noyaux* [SCH 02] désignent un ensemble de méthodes de classification et de regression pouvant opérer sur des objets de nature complexe, notamment des chaînes de caractères [JAA 00, VER 04], des arbres [VER 02], des ensembles de vecteurs [KON 03] ou des graphes [VER 03]. On peut citer parmi ces méthodes les machines à vecteur de support (SVM, [BOS 92, VAP 98]), l'analyse en composante indépendantes [BAC 02] et principales [SCH 98] sur l'espace des caractéristiques associées à un noyau (kICA et kPCA), ou encore les processus gaussiens pouvant être interprétés comme des lois a prioris sur des espaces de fonctions[SEE 03]. Ces méthodes ont en commun la connaissance préalable d'une mesure de similarité, un noyau, sur les objets qu'elles se proposent de traiter. Plus précisément, étant donnée une tâche de classification portant sur des objets contenus dans un ensemble $\mathcal{X}$, étant donné un ensemble d'apprentissage $x_1, ..., x_n$, les méthodes à noyaux n'utilisent pas directement des vecteurs de caractéristiques $\Phi(x_i)_{i=1..n} \in \mathbb{R}^d$ (à l'image des réseaux de neurones par exemple) plus aisément manipulables ; ces méthodes fondent leurs décisions sur les similarités mesurées deux-à-deux des points de $\mathcal{X}$ via un noyau k, la décision de classification d'un nouveau point x ne dépendant alors que de sa similarité avec un sous-ensemble I de points d'apprentissage $k(x, x_i)_{i \in I}$.

Dans le contexte méthodologique proposé par ces méthodes, il apparaît primordial de pouvoir proposer à la fois : des noyaux spécialisés permettant de cerner un type précis de similarité pour une tâche de classification bien spécifiée ; des noyaux capables d'embrasser plusieurs notions de similarité suffisamment générales pour être employés dans une démarche plus exploratoire. Idéalement, le noyau est supposé résumer toute la connaissance préalable dont nous disposons sur nos éléments $\mathcal{X}$ pris dans une perspective de classification ou de regression donnée. Cette subjectivité est ce qui laisse aujourd'hui la place à un vaste champ de proposition de noyaux pour objets complexes, parmi lesquels les chaînes de caractères.

Cet exposé s'articule autour des sections suivantes : dans la section 2 nous effectuons un très court rappel théorique sur les noyaux et les SVM. S'ensuit en section 3 une présentation du cadre des noyaux d'information mutuelle avant d'en exposer en section 4 une adaptation théorique et une implémentation pratique sur des chaînes de charactères. Faute de place pour les exposer, nous renvoyons le lecteur interessé par les détails de ces

implémentations et résultats pratiques (en bioinformatique, pour la détection d'homologies parmi des séquences de protéines) à [CUT 04].

2. Rappels sur les noyaux

Un noyau k sur un ensemble $\mathcal{X}$ est une fonction *symétrique* de deux variables prises dans cet ensemble, à valeur complexe (nous nous limiterons au cas réel dans la pratique), semi-définie positive : ce qui équivaut à dire que pour tout nombre arbitraire n d'éléments de $\mathcal{X}$, $(x_1, ..., x_n)$, et toute famille $c_1, ..., c_n$ de complexes,

$$\sum_{i=1}^{n} c_i \bar{c_j} k(x_i, x_j) \geq 0.$$

Cette dernière propriété est capitale afin de considérer k comme un produit scalaire dans un espace de caractéristiques $\mathcal{F}$. La construction d'un noyau sur $\mathcal{X}$ s'avère ainsi immédiate si l'on dispose d'un espace de Hilbert (muni d'un produit scalaire $\langle , \rangle_{\mathcal{F}}$) et d'une application $\Phi : \mathcal{X} \rightarrow \mathcal{F}$ en posant, pour tout $x, y \in \mathcal{X}$, que $k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{F}}$. Il est en effet aisé de vérifier que la dernière quantité vérifie l'inégalité précédente (notons que la proposition inverse est également vraie : étant donné un noyau k, l'existence d'un Hilbert $\mathcal{F}$ satisfaisant les propriétés voulues est assurée via notamment la théorie des espaces de noyaux auto-reproduisants). Ces propriétés permettent de rendre convexe l'optimisation des coefficients des machines de vecteurs de supports [VAP 98] : étant donnés un ensemble de points $x_1, ..., x_n$ munis de labels binaires $y_1, ..., y_n$ (pris dans $\{-1, +1\}$, on pourra se référer à [WES 98] pour le cas multiclasse), ces dernières se proposent de sélectionner un sous-ensemble I de $1, .., n$ tel que la classification d'un nouvel élément x soit déterminée par :

$$d(x) = \text{sgn}\left(\sum_{i \in I} y_i \lambda_i k(x_i, x) + b \right),$$

la définition de ces constantes λ_i et b étant obtenue à travers une optimisation quadratique calculable quand la fonction k est semi-définie positive [SCH 02].

3. Noyaux d'information mutuelle sur objets de taille variable

Etant donnée une famille de lois paramétriques issues de modèles susceptibles de générer nos éléments $\mathcal{X}$ (chaînes de Markov, chaînes de Markov cachées dans le cas où $\mathcal{X}$ est un espace de chaînes de caractères par exemple), e.g. une famille $\{p_\theta, \theta \in \Theta\}$ où $\Theta \subset \mathbb{R}^k$, l'approche des noyaux d'information mutuelle introduite par [SEE] consiste à considérer deux séquences sous la représentation de toutes leurs vraisemblances selon l'ensemble des lois précédemment citées. Ceci consiste en pratique en une transformation d'un élément x de $\mathcal{X}$ en un vecteur de charactéristiques $\Phi(x) = (p_\theta(x))_\theta \in \Theta$ qui détaille toute la capacité de la famille p_θ à capter la complexité de x. En supposant également connue un loi a priori sur les paramètres, i.e. une densité ω définie sur Θ, nous choisissons d'utiliser ces quantités dans un cadre de mélange Bayésien :

$$k(x, y) = \int_\Theta p_\theta(x) p_\theta(y) \omega(d\theta).$$

Cette intégrale existe (les vraisemblances étant bornées) et notre fonction k est par construction un noyau valable. Deux objets sont ainsi considérés comme étant similaires (valeur de k élevée) s'ils admettent en même temps de fortes vraisemblances selon les mêmes distributions, à condition que ces distributions soient elles-mêmes vraisemblables au sens de ω.

Nous pouvons généraliser l'approche précédente en considérant : des familles de modèles paramétriques, i.e. des modèles m contenus dans un espace mesurable $\mathcal{M}$, définissant une famille *doublement indexée* de mesures de probabilité $\{P_{m,\theta} | m \in \mathcal{M}, \theta \in \Theta_m\}$ où Θ_m désigne l'espace des paramètres associé à un modèle m ; des

vraisemblances normalisées par un facteur de taille, afin de pouvoir comparer deux objets de taille variable sans que celle-ci n'intervienne sur l'ordre des grandeur de ces quantités. Dans le cas de modèles exponentiels, auxquels nous nous restreindrons dans cet exposé, cela peut être formalisé par un changement de notre espace de caractéristiques, en pondérant les vraisemblances d'un objet x par sa taille $|x|$ et en utilisant un paramètre σ de manière telle qu'à x soient associées les vraisemblances $\Phi(x) = (p_{m,\theta}(x)^{\frac{\sigma}{|x|}})_{m\in\mathcal{M},\theta\in\Theta_m}$. Nous pouvons alors définir le double-mélange normalisé suivant :

$$k(x,y) = \sum_{m\in\mathcal{M}} \pi(m) \int_{\Theta_m} p_{m,\theta}(x)^{\frac{\sigma}{|x|}} p_{m,\theta}(y)^{\frac{\sigma}{|y|}} \omega(d\theta).$$

4. Définition et implémentation d'un noyau d'information mutuelle pour séquences

Nous proposons dans cette section une application directe du type de noyau présenté précédemment pour des chaînes de caractères prises sur un alphabet E (de taille d) en faisant pour cela appel à une classe de modèles markoviens, les arbres-contextes adaptatifs [WIL 95].

4.1. *Modèles d'arbres-contextes*

Dans le cadre d'un modèle de chaîne de Markov d'ordre D et étant donnée une chaîne m, la probabilité d'apparition d'un nouveau symbole e après cette chaîne m ne dépend que du suffixe de taille D de m (i.e. de ses D dernières lettres). La taille du suffixe pris en compte par les arbres-contextes est en revanche adaptative et dépend des dernières lettres observées. Ces derniers s'appuient plus précisément sur la définition d'un dictionnaire de suffixes complet (c.s.d). Un c.s.d $\mathcal{D}$ est un ensemble fini de mots sur E tel que toute séquence infinie à gauche m a un unique suffixe dans $\mathcal{D}$, que nous notons $\mathcal{D}(m)$, mais qu'aucun des mots de $\mathcal{D}$ n'a de suffixe strict dans $\mathcal{D}$ (i.e. non-égal à lui-même). Nous noterons $\mathcal{F}_D$ l'ensemble des c.s.d. $\mathcal{D}$ de mots de taille inférieure ou égale à D. Afin de définir une loi s'appuyant sur ce dictionnaire, nous associons à chaque mot s d'un c.s.d. $\mathcal{D}$ une distribution multinomiale $\theta_s \in \Sigma_d$ (où Σ_d est le simplexe canonique de dimension d). Etant donnée une séquence m dont le contexte dans $\mathcal{D}$ est $s = \mathcal{D}(m)$, la probabilité de voir apparaître après cette séquence une lettre e est donnée par $\theta_s(e)$. La vraisemblance de m sous le modèle défini par un c.s.d $\mathcal{D}$ et un ensemble de paramètres associés $\theta = (\theta_s)_{s\in\mathcal{D}}$ est définie comme le produit des vraisemblances de chaque D-transition de m, soit : $p_{\mathcal{D},\theta}(m) = \prod_{i=1}^{|m|} \theta_{\mathcal{D}(m^i)}(m'^i)$, où nous avons énuméré via l'indice i l'ensemble des transitions observées dans m entre un contexte de taille D et la lettre qui le suit. En pratique $|m|$, le nombre de ces transitions (qui nous servira de facteur de renormalisation), est égal à $l(m) - D$. Ainsi, pour un chaîne $m_x = ABCDE$ et une profondeur maximale $D = 2$ nous considèrerons les 2-transitions contenues dans $X = \{(AB, C); (BC, D); (CD, E)\}$ pour calculer la vraisemblance de m_x.

4.2. *Lois a priori sur $\mathcal{D}$ et θ*

Le cadre de mélange bayésien que nous nous sommes donnés nous impose de proposer des lois a priori sur les distributions paramétriques possibles, en spécifiant $\omega(\mathcal{D}, \theta) = \pi(\mathcal{D})\omega(\theta|\mathcal{D})$. Pour plus de détails sur cette démarche voir [CUT 04]. Suivant les travaux de [WIL 95] et [CAT ar] nous utilisons une loi a priori inspirée des processus de branchement pour la génération de c.s.d (par analogie avec la génération d'arbres pour lesquels chaque noeud n'a soit pas de descendance soit une descendance complète), cette loi étant paramétrée par un réel ε. Pour le choix de $\omega(\theta|\mathcal{D})$ nous utilisons des lois a priori de Dirichlet, prises indépendamment sur chacun des paramètres $\theta_s, s \in \mathcal{D}$, utilisant dans nos expériences alternativement des lois simples (avec les d paramètres pris à $\frac{1}{2}, ..., \frac{1}{2}$, communément dénommée *prior* de Jeffrey ou de Krichesvky-Trofimov) ou des mélanges additifs (de k lois de dirichlets ω_i, chacune pondérées par une coefficient $\gamma_i, i = 1..k$) préestimés dans le cadre d'applications biologiques, voir [BRO 93]. Dans ce dernier cas, notre noyau prend la forme suivante :

$$k_\sigma(m,m') = \sum_{\mathcal{D}\in\mathcal{F}_D} \pi(\mathcal{D}) \int_{\Theta_\mathcal{D}} p_{\mathcal{D},\theta}(m)^{\frac{\sigma}{|m|}} p_{\mathcal{D},\theta}(m')^{\frac{\sigma}{|m'|}} \prod_{s\in\mathcal{D}} \sum_{i=1}^{k} \gamma_i \omega_i(d\theta_s).$$

4.3. *Implémentation pratique*

Etant données les lois a priori sur nos distributions décrites dans la section précédente, le calcul du noyau proposé semble ardu, puisqu'il nécessite de mener à la fois un mélange discret sur les modèles $\mathcal{D}$, continu sur les paramètres $\theta \in \Theta_\mathcal{D}$ avec éventuellement des mélanges additifs de lois de Dirichlet. Ce calcul est cependant réalisable grace à une adaptation de l'ingénieux algorithme récursif dit du *Context-tree weigthing*, proposé dans [WIL 95] et analysé dans [CAT ar], que nous ne pouvons détailler ici faute de place. Le lecteur interessé par cette implémentation ainsi que par des résultats pratiques menés en bioinformatique pourra se référer à [CUT 04]. Notre noyau est donc calculable en temps et espace mémoire linéaire en la longueur des séquences, pouvant ainsi être utilisé dans des applications de classification sur de larges bases de données.

5. Bibliographie

[BAC 02] BACH F., JORDAN M., Kernel Independent Component Analysis, *Journal of Machine Learning Research*, vol. 3, 2002, p. 1–48.

[BOS 92] BOSER B. E., GUYON I. M., VAPNIK V. N., A training algorithm for optimal margin classifiers, *Proceedings of the 5th annual ACM workshop on Computational Learning Theory*, ACM Press, 1992, p. 144–152.

[BRO 93] BROWN M. P., HUGHEY R., KROGH A., MIAN I. S., SJÖLANDER K., HAUSSLER D., Using Dirichlet Mixture Priors to Derive Hidden Markov Models for Protein Families, *Proc. of First Int. Conf. on Intelligent Systems for Molecular Biology*, Menlo Park, CA, 1993, AAAI/MIT Press, p. 47–55.

[CAT ar] CATONI O., *Statistical learning theory and stochastic optimization, Saint-Flour lecture notes*, Springer Verlag, *to appear*.

[CUT 04] CUTURI M., VERT J.-P., A Mutual Information Kernel for Sequences, *IEEE International Joint Conference on Neural Networks*, 2004.

[JAA 00] JAAKKOLA T., DIEKHANS M., HAUSSLER D., A Discriminative Framework for Detecting Remote Protein Homologies, *Journal of Computational Biology*, vol. 7, n^o 1,2, 2000, p. 95–114.

[KON 03] KONDOR R., JEBARA T., A Kernel between Sets of Vectors, *Proceedings of the ICML*, 2003.

[SCH 98] SCHÖLKOPF B., SMOLA A., MÜLLER K., Nonlinear Component Analysis as a Kernel Eigenvalue Problem, *Neural Computation*, vol. 10, n^o 5, 1998, p. 1299–1319.

[SCH 02] SCHÖLKOPF B., SMOLA A. J., *Learning with Kernels : Support Vector Machines, Regularization , Optimization, and Beyond*, MIT Press, Cambridge, MA, 2002.

[SEE] SEEGER M., Covariance Kernels from Bayesian Generative Models, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, MIT Press.

[SEE 03] SEEGER M., Gaussian Processes for Machine Learning, rapport, 2003, University of California, Berkeley, UK.

[VAP 98] VAPNIK V. N., *Statistical Learning Theory*, Wiley, New-York, 1998.

[VER 02] VERT J.-P., A tree kernel to analyze phylogenetic profiles, *Bioinformatics*, vol. 18, 2002, p. S276-S284.

[VER 03] VERT J.-P., KANEHISA M., Graph-driven features extraction from microarray data using diffusion kernels and kernel CCA, MIT Press, 2003.

[VER 04] VERT J.-P., SAIGO H., AKUTSU T., Local alignment kernels for protein sequences, SCHÖLKOPF B., TSUDA K., VERT J.-P., Eds., *Kernel Methods in Computational Biology*, MIT Press, 2004.

[WES 98] WESTON J., WATKINS C., Multi-class support vector machines, 1998.

[WIL 95] WILLEMS F. M. J., SHTARKOV Y. M., TJALKENS T. J., The context-tree weighting method : basic properties, *IEEE Transancations on Information Theory*, , 1995, p. 653–664.

Collections hiérarchiques et faiblement hiérarchiques de fermés de Galois d'un contexte

Jean Diatta

IREMIA, Université de la Réunion
15 avenue René Cassin - BP 7151
97715 Saint-Denis messag cedex 9, France
Jean.Diatta@univ-reunion.fr

RÉSUMÉ. *Nous nous intéressons aux fermés de Galois d'un contexte où des entités sont décrites dans un inf-demi-treillis. Nous donnons une condition nécessaire et suffisante pour qu'une collection particulière de ces fermés soit hiérarchique. De plus, nous spécifions la collection maximum de ces fermés, composée de classes fortes associées à une mesure de dissimilarité mutuelle ou de classes faibles associées à une mesure de dissimilarité k-voies.*

MOTS-CLÉS : *Contexte, Demi-treillis, Dissimilarité, Fermé de Galois, Hiérarchie*

1. Introduction

La classification a pour objectif de regrouper des données dans des classes de sorte que le degré d'association soit fort entre membres d'une même classe et faible entre membres de différentes classes. Elle est à ce titre un outil de découverte qui peut révéler dans des données des associations et de la structure qui, bien que pas évidentes, peuvent s'avérer utiles une fois trouvées. Il existe plusieurs structures de classification, les plus connues allant des hiérarchies aux hiérarchies faibles ([BEN 73],[DID 84],[BAN 89],[BER 02]). Ces structures sont souvent obtenues par une approche dite classique, fondée sur des mesures de (dis)similarités. Par ailleurs, la classification conceptuelle a pour objectif de regrouper des données dans des classes représentant certains concepts descriptifs ([MIC 80],[MIC 83],[CHE 85],[FIS 85]). Le cadre de la classification conceptuelle est un contexte composé d'un ensemble d'entités, d'un langage conceptuel pour décrire les entités (e.g., des conjonctions de motifs descriptifs), et d'une description de chaque entité dans ce langage de description.

Cette communication se situe dans le cadre d'un contexte où des entités d'un ensemble fini sont décrites dans un inf-demi-treillis ([BRI 94],[GAN 01],[DIA 02],[DIA 04]). Le descripteur qui à chaque entité associe sa description induit alors une correspondance de Galois entre l'ensemble des parties de l'ensemble des entités et l'espace de description de ces entités, donc un opérateur de fermeture sur chacun de ces deux ensembles ([BIR 67],[BAR 70]). Nous nous intéressons aux fermés selon (i.e. points fixes de) l'opérateur de fermeture induit sur l'ensemble de parties de l'ensemble des entités. Nous donnons une condition nécessaire et suffisante pour qu'une collection particulière de ces fermés soit hiérarchique. De plus, nous spécifions la collection maximum de ces fermés, composée de classes fortes associées à une mesure de dissimilarité mutuelle ou de classes faibles associées à une mesure de dissimilarité k-voies. Au-delà de ces résultats, ces collections sont potentiellement utiles dans des applications où des structures de classification sont construites en utilisant des approches aussi bien classiques que conceptuelles ([DEU 98],[PER 00]).

2. Fermés de Galois dans un contexte de descriptions ordonnées

Considérons un contexte constitué d'un ensemble fini d'entités décrites dans un inf-demi-treillis. Nous désignerons un tel contexte par un triplet $\mathbb{K} = (E, \underline{D}, \delta)$ où E représente l'ensemble des entités, $\underline{D} := (D, \leq)$ l'espace de description des entités, et δ un descripteur qui applique E dans $\underline{D}$. Le descripteur δ induit une correspondance de Galois entre les ensembles ordonnés $(\mathcal{P}(E), \subseteq)$ et $\underline{D}$ par le biais des applications

$$f : X \mapsto \inf \{\delta(x) \, : \, x \in X\}$$

et

$$g : I \mapsto \{x \in E \, : \, I \leq \delta(x)\},$$

pour $X \subseteq E$ et $I \in \underline{D}$. Ainsi les applications $\phi_\delta := g \circ f$ et ${\phi_\delta}' := f \circ g$ sont des opérateurs de fermeture dans $(\mathcal{P}(E), \subseteq)$ et $\underline{D}$, respectivement [BIR 67]. Les points fixes de ces opérateurs de fermetures sont connus sous le nom de fermés de Galois [BAR 70]. Dans cette communication, nous nous intéresserons particulièrement aux points fixes de ϕ_δ. On notera que lorsque $\underline{D}$ est un sup-demi-treillis le descripteur δ induit une correspondance de Galois entre $(\mathcal{P}(E), \subseteq)$ et le dual d'ordre de $\underline{D}$ par le biais des applications

$$f^\partial : X \mapsto \sup \{\delta(x) \, : \, x \in X\}$$

et

$$g^\partial : I \mapsto \{x \in E \, : \, I \geq \delta(x)\},$$

pour $X \subseteq E$ et $I \in \underline{D}$. Dans ces conditions, par le principe de dualité, nos résultats concernant les points fixes de ϕ_δ restent valides pour ceux de $\phi_\delta^\partial := g^\partial \circ f^\partial$. On notera que la plupart des données traitées en analyse de données symboliques peuvent être représentées sous forme de tels contextes [BOC 00]. On notera aussi que les premières projections des éléments de ce qui est appelé treillis de Galois de l'union dans [POL 98] ne sont rien d'autres que les points fixes de ϕ_δ^∂.

Dans tout ce qui suit, E désignera un ensemble fini non vide d'entités, $\underline{D}$ un inf-demi-treillis, et δ un descripteur appliquant E dans $\underline{D}$.

3. Caractérisation des points fixes de ϕ_δ

Il existe diverses caractérisations des points fixes de ϕ_δ. Dans cette section, nous en proposons une fondée sur la notion de valuation. Une *valuation* sur un ensemble ordonné $(P, \leq)$ est une application $h : P \to \mathbb{R}_+$ telle que $h(x) \leq h(y)$ lorsque $x \leq y$ [BAR 70]. On dira que h est une valuation *stricte* lorsque $x < y$ implique $h(x) < h(y)$.

Pour toute partie X de E, $\delta(X)$ désignera l'ensemble des descriptions des entités appartenant à X, et pour toute valuation h sur $\underline{D}$, $X^h_\wedge$ désignera le sous-ensemble de E défini par

$$X^h_\wedge = \{x \in E : h(\inf \delta(X \cup \{x\})) = h(\inf \delta(X))\}.$$

Alors nous avons le résultat suivant :

THÉORÈME 1 *Une partie X de E est un point fixe de ϕ_δ si et seulement si $X^h_\wedge = X$ pour n'importe quelle valuation stricte h sur $\underline{D}$.*

REMARQUE 1 *Le Théorème 1 fournit un moyen simple pour à la fois (1) décider si une partie X de E est un point fixe de ϕ_δ ou pas, et (2) construire la fermeture par ϕ_δ de toute partie Y de E. En effet, il suffit pour (1) de vérifier si $X^h_\wedge = X$ et pour (2) de calculer $Y^h_\wedge$, pour une certaine valuation stricte h sur $\underline{D}$.*

Le résultat ci-après est une conséquence immédiate du théroème précédent.

COROLLAIRE 1 *Une partie X de E est un point fixe de ϕ_δ si et seulement si pour toute valuation stricte h sur $\underline{D}$ et pour tout $Y \subseteq X$, $Y^h_\wedge \subseteq X$.*

4. Collections de points fixes de ϕ_δ

Dans cette section, nous considérons des collections de points fixes non vides de ϕ_δ, à la lumière de deux structures de classification : les hiérarchies et les hiérarchies faibles. Cela est principalement motivé par d'une part un résultat de [DIA 04] prouvant la coïncidence entre ces points fixes et les classes faibles associées à certaines mesures de dissimilarité (multivoies) et, d'autre part, le fait que ces classes faibles forment une hiérarchie (k-) faible ([BAN 94],[DIA 97]).

Les classes faibles introduites dans [BAN 89] en termes de mesures de similarité sont dite faibles par opposition à des classes dites fortes. Une partie non vide X de E est dite être une *classe forte* associée à une mesure de dissimilarité mutuelle d_2 (ou *classe d_2-forte*), si son *indice d'isolation d_2-forte*

$$\mathbf{i}^s_{d_2}(X) := \min_{\substack{x,y\in X\\ z\notin X}} \{d_2(x,z) - d_2(x,y)\}$$

est strictement positif ; elle est dite être une *classe faible* associée à d_2 (ou *classe d_2-faible*), si son *indice d'isolation d_2-faible*

$$\mathbf{i}^w_{d_2}(X) := \min_{\substack{x,y\in X\\ z\notin X}} \{\max\{d_2(x,z), d_2(y,z)\} - d_2(x,y)\}$$

est strictement positif. Ces deux types de classes se généralisent naturellement aux mesures de dissimilarité k-voies, $k \geq 2$. Une mesure de dissimilarié k-voies sur E est une application d_k de $E^*_{\leq k}$ à valeurs réelles positives ou nulles telle que $d_k(X) \leq d_k(Y)$ lorsque $X \subseteq Y$, où pour toute partie non vide $X \subseteq E$, $X^*_{\leq k}$ désigne l'ensemble des parties non vides de X ayant au plus k éléments. Ainsi, par exemple, une partie non vide X de E est dite être une *classe faible* associée à d_k (ou *classe d_k-faible*), si son *indice d'isolation d_k-faible*

$$\mathbf{i}^w_{d_k}(X) := \min_{\substack{Y\in X^*_{\leq k}\\ z\notin X}} \{\max_{Z\in Y^*_{\leq k-1}} d_k(Z+z) - d_k(Y)\}$$

est strictement positif.

On notera que pour une mesure de dissimilarité k-voies d_k, toute classe d_k-forte est aussi d_k-faible. De plus, on montre facilement que les classes fortes (resp. faibles) associées à une mesure de dissimilarité mutuelle (resp. k-voies) forment une collection hiérarchique (resp. k-faiblement hiérarchique) ([BAN 89],[BAN 94],[DIA 94],[DIA 97]).

Une collection $\mathcal{C}$ de parties de E sera dite *hiérarchique* lorsque deux quelconques C, C' de ses éléments sont toujours soit disjoints, soit emboîtés, c'est-à-dire, $C \cap C' \in \{\varnothing, C, C'\}$; elle sera dite *k-faiblement hiérarchique* lorsque l'intersection de $k+1$ quelconque $C_1, \ldots, C_k, C_{k+1}$ de ses éléments est toujours égale à l'intersection de k parmi ces $k+1$, i.e. il existe $i \in \{1, \ldots, k+1\}$ tel que $\bigcap_{j\neq i} C_j \subseteq C_i$.

Dans tout ce qui suit, h désignera une valuation stricte sur $\underline{D}$. Le résultat suivant donne une condition suffisante pour qu'une collection particulière de points fixes de ϕ_δ soit hiérarchique.

PROPOSITION 1 *S'il n'y a pas trois éléments distincts x, y, z de E tels que $\delta(x) \leq \delta(z)$ et $\delta(y) \leq \delta(z)$, alors la collection $(\{x\}^h_\wedge)_{x\in E}$ de points fixes de ϕ_δ est hiérarchique.*

La condition ci-après est nécessaire et suffisante pour que la collection $(\{x\}^h_\wedge)_{x\in E}$ soit hiérarchique.

PROPOSITION 2 *La collection $(\{x\}^h_\wedge)_{x\in E}$ est hiérarchique si et seulement s'il n'y a pas trois éléments distincts x, y, z de E tels que $\delta(x)$ et $\delta(y)$ ne sont pas comparables, $\delta(x) \leq \delta(z)$, et $\delta(y) \leq \delta(z)$.*

Les notions de boule et de k-boule sont essentielles dans la spécification de collections de points fixes de ϕ_δ ci-après. Pour $k \geq 2$, soit d_k une mesure de dissimilarité k-voies sur E. Soit $X \in E^*_{\leq k-1}$ et $r \geq 0$. La *d_k-boule* de centre X et de rayon r est l'ensemble $B^{d_k}(X, r)$ défini par $B^{d_k}(X, r) = \{y \in E : d_k(X \cup \{y\}) \leq r\}$. Soit

maintenant $X \in E^*_{\leq k}$. Alors la (d_k, k)-*boule* (ou k-*boule* relative à d_k) engendrée par X est l'ensemble $B_X^{d_k}$ défini par $B_X^{d_k} = B(X, d_k(X))$ lorsque $|X| \leq k-1$, et $B_X^{d_k} = \underset{x \in X}{\cap} B(X \setminus \{x\}, d_k(X))$ autrement.

On montre sans difficulté que toute classe d_2-forte est une boule de la forme $B(x, d_2(x,y))$ [DIA 94]. De même, on montre que toute classe d_k-faible est une k-boule [DIA 97]. De plus, on a le résultat suivant.

THÉORÈME 2

(i) L'ensemble $\mathcal{H}^c_{d_2} := \{B(x, d_2(x,y)) : x, y \in E, B(x, d_2(x,y))^h_\wedge = B(x, d_2(x,y)), \mathbf{i}^s_{d_2}(B(x, d_2(x,y))) > 0\}$ *est la collection maximum de points fixes de* ϕ_δ*, composée de classes* d_2*-fortes.*

(ii) L'ensemble $\mathcal{W}^c_{d_p} := \{B_X^{d_p} : X \in E^*_{\leq p}, (B_X^{d_p})^h_\wedge = B_X^{d_k}, \mathbf{i}^w_{d_p}(B_X^{d_p}) > 0\}$ *est la collection maximum de points fixes de* ϕ_δ*, composée de classes* d_p*-faibles.*

5. Bibliographie

[BIR 67] G. Birkhoff, *Lattice theory*, 3rd edition, Coll. Publ., XXV, American Mathematical Society, Providence, RI, 1967.

[BAR 70] M. Barbut and B. Monjardet, *Ordre et classification*, Hachette, Paris, 1970.

[BEN 73] J-P. Benzécri, *L'Analyse des données : la Taxinomie*, Dunod, Paris, 1973.

[MIC 80] R. Michalski, Knowledge Acquisition Through Conceptual Clustering. A Theoretical Framework and an Algorithm for Partioning Data into Conjunctive Concepts, vol. 4, 1980, , p. 219–244.

[MIC 83] R. Michalski and R. E. Stepp, Automated construction of classifications : conceptual clustering versus numerical taxonomy, vol. 5, 1983, , p. 396–410.

[DID 84] E. Diday, rapport, n^{o}291, 1984, INRIA, France.

[FIS 85] D. Fisher and P. Langley, Approaches to Conceptual Clustering, *Proccedings of the International Joint Conference on Artificial Intelligence*, Los Angeles, CA, 1985, p. , 691–697.

[CHE 85] Y. Cheng and K. S. Fu, Conceptual clustering in knowledge organisation, vol. 7, 1985, , p. 592–598.

[BAN 89] H-J. Bandelt and A. W. M. Dress, Weak hierarchies associated with similarity measures : an additive clustering technique, vol. 51, 1989, , p. 113–166.

[DIA 94] J. Diatta and B. Fichet, , , *New Approaches in Classification and Data Analysis*, Springer-Verlag, p. , 111–118.

[BRI 94] P. Brito, Order Structure of Symbolic Assertion Objects, vol. 6, n^o 5, 1994, , p. 830–835.

[BAN 94] H-J. Bandelt and A. W. M. Dress, An order theoretic framework for overlapping clustering, vol. 136, 1994, , p. 21–37.

[DIA 97] J. Diatta, Dissimilarités multivoies et généralisations d'hypergraphes sans triangles, vol. 138, 1997, , p. 57–73.

[DEU 98] A. van Deursen and T. Kuipers, rapport, n^{o}SEN-R981, 1998, CWI, Amsterdam, The Netherlands.

[POL 98] G. Polaillon, , , *Advances in Data Science and Classification*, Springer-Verlag, p. , 433–440.

[BOC 00] *Analysis of Symbolic Data*, Springer-Verlag, 2000.

[PER 00] M. Perkowitz and O. Etzioni, Towards adaptative Web sites : Conceptual framework and case study, vol. 118, 2000, , p. 245–275.

[GAN 01] B. Ganter and S. O. Kuznetsov, Pattern Structures and Their Projections, vol. 2120, 2001, , p. 129–142.

[DIA 02] J. Diatta and H. Ralambondrainy, The conceptual weak hierarchy associated with a dissimilarity measure, vol. 44, 2002, , p. 301–319.

[BER 02] P. Bertrand, Set systems for which each set properly intersects at most one other set - Application to cluster analysis, Personal communication, 2002.

[DIA 04] J. Diatta, Concept Extensions and Weak Clusters Associated with Multiway Dissimilarity Measures, *Concept Lattices*, Lecture Notes in Artificial Intelligence 1910, Spinger-Verlag, p. , 236–243.

Introduction à la classification spatiale : le cas des hiérarchies et pyramides spatiales

Edwin Diday

CEREMADE Université Paris Dauphine
Place du Maréchal de Lattre de Tassigny
75775 Paris Cedex 16

RÉSUMÉ. On étend les hiérarchies et pyramides classiques de support linéaire à des m -hiérarchies et m/k-pyramides à support multidimensionnel basé sur un "m/k-maillage", autrement dit, un maillage où m arêtes définissant m-1 angles égaux, partent de chaque nœud et dont les plus petits cycles contiennent k arêtes de même longueur. Ainsi, au lieu de représenter les individus classifiés sur une droite comme en classification hiérarchique ou pyramidale classique, on peut représenter les individus sur un plan ou dans un volume de façon plus fidèle aux données initiales. L'objectif de cet article est de donner quelques éléments de base pour comprendre la classification spatiale et ses vastes perspectives de recherches et d'applications

MOTS-CLÉS : Classification automatique, Hiérarchies, Pyramides

1. Introduction

On est passé des hiérarchies et leur bijection avec les ultramétriques [JOH 67], [BEN73] aux pyramides classifiantes et leur bijection avec les dissimilarités robinsoniennes [DID 84] puis des pyramides classifiantes aux pyramides spatiales et leur bijection avec les dissimilarités yadidiennes [DID 04]. Les pyramides bâties sur un support linéaire ont donné lieu à plusieurs thèses ([BER 86], [BRI 91], [AUD 96], [ROD 00],…). Les pyramides spatiales sont bâties comme les hiérarchies et les pyramides classifiantes en associant à chaque individu de la population à classifier un individu d'un maillage qui peut être de dimension quelconque au lieu d'être linéaire comme dans le cas des hiérarchies et pyramides classifiantes standards. On voit ainsi dans la figure 1 une hiérarchie et une pyramide classifiante fournissant une structure classifiante de 5 individus et bâties sur une suite de 5 points x1,…, x5. Ces points sont à égale distance sur une droite formée de segments égaux dont les extrémités représentent ces individus. On obtient des supports plus riches en utilisant une tessellation, autrement dit un recouvrement du plan par des figures jointes régulières comme indiquées, par exemple en figure 2a. Ainsi, dans la figure 1 c, on représente une pyramide sur une grille 3x3 dont les 9 sommets sont associés à une population de 9 individus. On trouvera dans [DID 04] un algorithme de construction ascendante d'une pyramide spatiale général dans ces journées dans un article de K. Pak et E. Diday, un algorithme de construction détaillé d'une hiérarchie spatiale.

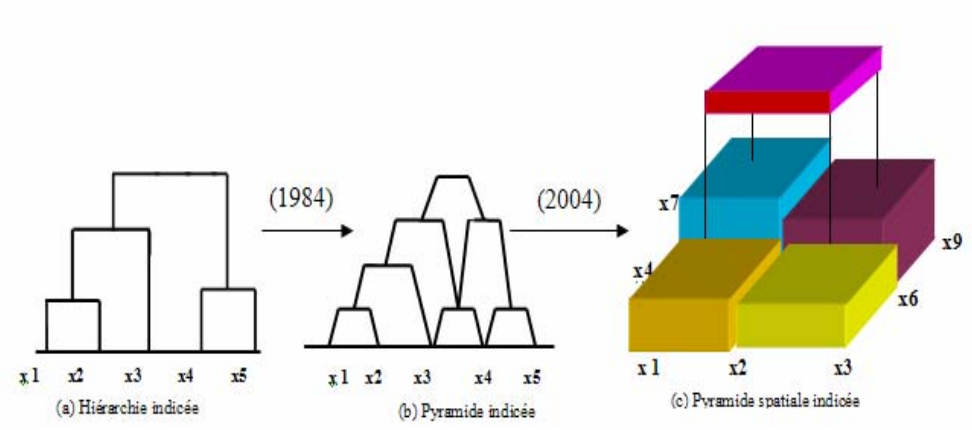
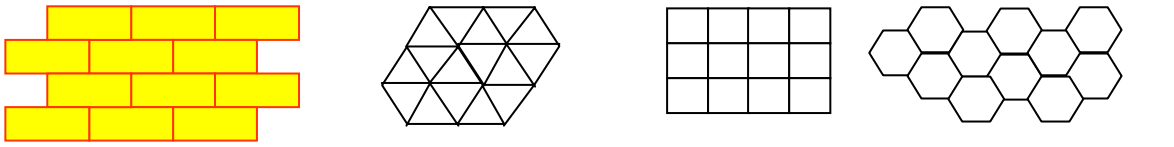


FIG 1 : Des hiérarchies aux pyramides indicées puis aux pyramides spatiales indicées.

2. Définition d'un m/k-maillage

Un m/k-maillage est un graphe dont chaque sommet est au point de rencontre de m arêtes au maximum formant m angles consécutifs égaux strictement positifs, dont les plus petits cycles (i.e. ceux qui contiennent le minimum de sommets) contiennent k arêtes de même longueur et forment des "cellules" de surface non nulle qui partitionnent et couvrent l'espace dans lequel il est plongé. Plutôt que maillage, on pourrait dire aussi: "tessellation". On peut démontrer (par exemple, dans un rapport CEREMADE de l'auteur, à paraître) que dans le plan il n'existe que trois possibilités indiquées figure 2: les (6, 3)-maillages dont la cellule de base est un triangle équilatéral, les (4, 4)-maillages ou "grille" dont les cellules sont des carrés, les (3, 6)-maillages dont les cellules sont des hexagones. Il existe bien sûr de tels maillages en volume comme le (6, 12)-maillage correspondant à une grille cubique dont les sommets peuvent être aussi associés aux individus d'une population sur laquelle on cherche une structure classifiante par exemple, une pyramide dont les paliers sont des volumes (plutôt que des surfaces comme celles indiquées dans la pyramide à trois dimensions de la figure 1(c)).



(a) Une Tessellation qui n'est pas un (m, k)-maillage

(b) Un (6, 3)-maillage.

(c) Un (4, 4)-maillage.

(d) Un (3, 6)-maillage.

FIG 2 : Exemples de tessellations et de (m, k)-maillages

3. La classification spatiale

La classification spatiale de données qui nous intéresse ici, s'adresse à tout type de données où n individus sont caractérisés par p variables descriptives (qu'elles soient classiques ou symboliques). Elle a pour objectif d'associer ces individus à chaque nœud d'un (m, k)-maillage et d'en extraire simultanément une structure classifiante "compatible" avec ce maillage. Les pyramides spatiales que nous introduisons en 4 ont cet objectif. Il ne faut pas confondre la classification de données spatiales et la classification spatiale de données. La classification de données spatiales a pour but de faire une classification automatique d'individus (par exemple, des lieux géographiques) qui se trouvent dans l'espace réel à trois dimensions.

4. Parties convexes, connexes, maximales et dissimilarités induites d'un maillage

On assimile un (m, k)-maillage à un graphe dont les nœuds sont les nœuds du graphe et les arêtes, les côtés des cellules du maillage. On considère par la suite que la longueur d'un chemin dans ce graphe (i.e. dans le maillage) est le nombre de ses arêtes. Soit Ω un ensemble d'individus, une partie C de Ω est dite convexe dans un maillage si et seulement si le chemin de plus courte longueur qui relie deux quelconques de ses sommets est dans C. Une partie C du maillage est dite connexe si et seulement si deux quelconques de ses sommets sont joignables par une chaîne reliant des sommets contigus du maillage.

La dissimilarité induite par un maillage M notée d_M entre deux nœuds a, b de M est définie par $d_M(a, b)$ = {la longueur d'un plus court chemin de M reliant a et b}. On utilise d_M pour définir une partie dite "maximale" du

maillage. Une partie C de Ω est dite maximale dans un maillage M ssi tout élément c de C a une dissimilarité avec tout élément c' de C inférieure à une longueur α et tout noeud c du maillage qui a une dissimilarité à tout élément c' de C inférieure à α est dans C.

5. Définition d'une hiérarchie et d'une pyramide spatiale indicée

Définition d'une Sm-hiérachie

Une Sm-hiérarchie convexe (resp. connexe, maximal, autre) est un ensemble de parties H (appelés paliers) d'un ensemble Ω satisfaisant aux propriétés suivantes :

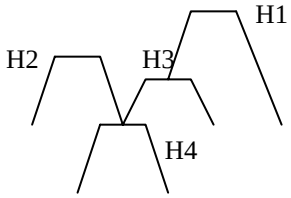
1) $\Omega \in H$. 2) $\forall w \in \Omega$, $\{w\} \in H$. 3) $\forall P_1,\ldots,P_m \in H$, $P_1 \cap \ldots \cap P_m = \phi$ ou $\exists j \in \{1,\ldots,m\}$: $P_1 \cap \ldots \cap P_m = P_j$ .4) Il existe un m/k-maillage de Ω pour lequel tout élément de H est convexe (resp. connexe, maximal, autre).

Une hiérarchie classique est le cas particulier d'une S2-hiérarchie dont les paliers (à deux dimensions : hauteur, longueur) sont à la fois convexes, connexes, maximaux et pour tout intervalle d'un "maillage linéaire" défini par les sommets d'une suite de segments égaux successifs sur une droite qui forment ses arêtes.

Une S4- hiérarchie convexe est une hiérarchie spatiale dont chaque palier (à trois dimensions : hauteur, longueur, largeur) est un convexe pour une grille à deux dimensions. Une S6-hiérarchie convexe est une hiérarchie spatiale dont chaque palier (à quatre dimensions : hauteur (degré d'agrégation), longueur, largeur, profondeur) est un convexe pour une grille à trois dimensions. Une S3-hiérarchie est une pyramide particulière. En effet (voir figure 3 b), c'est une pyramide dont trois paliers d'intersection non vide est nécessairement l'un d'entre eux, ce qui n'est pas toujours le cas.

	x_{11}	x_{12}	x_{13}	x_{21}	x_{22}	x_{23}	x_{31}	x_{32}	x_{33}
x_{11}	0	4	6	4	4	6	5	6	8
x_{12}		0	5	4	4	5	6	6	7
x_{13}			0	6	5	5	8	7	5
x_{21}				0	1	6	1	1	7
x_{22}					0	3	1	1	3
x_{23}						0	7	3	1
x_{31}							0	1	3
x_{32}								0	3
x_{33}									0

$Z_{2\,3}(d)$ = 1 1 7 / 1 1 3 / 7 3 1



(a) **(b)**

FIG 3. (a) La partie supérieure d'une matrice yadidienne associée à une grille 3x3 et la matrice bloc $Z_{2\,3}(d)$.
(b) Une S3-hiérarchie .

Définition d'une Sm-pyramide

Une Sm-pyramide convexe (resp. connexe, maximal, autre) est un ensemble de parties P d'un ensemble Ω satisfaisant aux propriétés suivantes : *1) $\Omega \in P$. 2) $\forall w \in \Omega$, $\{w\} \in P$.3) $\forall P_1,\ldots,P_m \in P$, $P_1 \cap \ldots \cap P_m = \phi$ ou $P_1 \cap \ldots \cap P_m \in H$. 4) Il existe un m/k-maillage de Ω pour lequel tout élément de P est convexe (resp. connexe, maximale, autre) Remarquons qu'une pyramide classique est une S2-pyramide.*

6. Dissimilarités Yadidiennes et leurs propriétés

Considérons le cas d'une grille M (i.e. un (m, k)-maillage) à n lignes et p colonnes. La dissimilarité d_M induite de cette grille satisfait deux propriétés remarquables. Considérons la matrice $Z_{ij}(d_M)$ (ordonnée en lignes et en colonnes selon l'ordre des lignes de la grille), des dissimilarités d_M entre chaque élément de la ième colonne et chaque élément de la jème colonne (voir $Z_{23}(d)$ dans la figure 3a) . Une première propriété est que $Z_{ij}(d_M)$ est symétrique et que ses termes croissent en lignes et colonnes à partir de sa diagonale principale dont les termes sont nuls si i = j (ainsi, par définition les matrices $Z_{ii}(d_M)$ sont robinsoniennes). La seconde propriété est que la matrice $Y(d_M)$ dont le terme générique est une matrice $Z_{ij}(d_M)$ pour i et j variant de 1 à p, est symétrique et croissante en lignes et en colonnes à partir de sa diagonale principale (dont les termes sont des matrices robinsoniennes). Une matrice est dite yadidienne si elle satisfait ces deux propriétés. Une dissimilarité d est dite yadidienne si Y(d) est yadidienne. On donne un exemple de dissimilarité yadidienne en 3.a). Une matrice yadidienne n'est pas robinsonienne puisque ses termes (les $d(x_{ik}, x_{jm})$ pour i, j ∈ {1,…,p} et k, m ∈ {1,…,n}) ne

croissent nécessairement pas à partir de sa diagonale principale. D'autre part, le nombre maximum de termes différents dans une telle matrice peut être inférieur au nombre maximal de termes différents dans une matrice robinsonienne de même taille. On a ainsi démontré (voir [DID 04]) que dans le cas d'une S4-pyramide convexe, le nombre de termes différents dans une matrice yadidienne tend à devenir deux fois moins grand que le plus grand nombre possible de termes différents d'une matrice robinsonienne de même taille. On a aussi démontré différents théorèmes de bijection entre différents types de yadidiennes et hiérarchies ou pyramides spatiales indicées strictement ou largement [DID 04]. Il en résulte que l'on peut mesurer la qualité d'une pyramide spatiale par l'écart entre la yadidienne qui lui est associée par la bijection et la dissimilarité initiale. Cette qualité peut aussi se mesurer par le pourcentage de triplets ayant le même couple de dissimilarité maximale pour la dissimilarité d_M induite du maillage et pour la dissimilarité initiale. Remarquons au passage que cet indice est une extension du taux de Kendall qui se réduit à la recherche des couples au lieu des triplets.

7. Conclusion

Nous avons ainsi donné quelques éléments de base pour la classification spatiale offrant un large champ de recherche et d'applications. Cette approche peut par exemple, enrichir les cartes de Kohonen en obtenant d'abord des centres par des nuées dynamiques débarrassées de la contrainte imposée par la carte (donc plus rapide avec plus de chance d'obtenir une meilleure solution en répétant les tirages) puis en obtenant une grille sur ces centres induite par une classification spatiale hiérarchique ou pyramidale dont la yadidienne associée s'écarte le moins possible de la dissimilarité initiale et dont les paliers peuvent enrichir l'interprétation de la carte. Remarquons qu'en considérant une grille volumétrique (i.e. un (6, 12)-maillage) et en associant à chaque palier d'une pyramide spatiale connexe un concept modélisé par un objet symbolique [BIL 03], on obtient des concepts "volumétriques" qui peuvent être organisés selon un treillis de Galois stochastique [DID 03].

8. Bibliographie

[AUD 99] J.C.AUDE. *Analyse de génomes microbiens, apports de la classification pyramidale*. Thèse de Doctorat Université Paris IX Dauphine 75016 Paris. France, 1999.

[BIL 03] L.BILLARD, E.DIDAY, From the statistics of data to the statistic of knowledge: Symbolic Data Analysis. *JASA . Journal of the American Statistical Association.* Juin 2003, Vol. 98, N° 462.

[BEN 73] J.P.BENZECRI, l'Analyse des données: la Taxinomie, Vol. 1, 1973, Dunod, Paris.

[BER 86] P.Bertrand. *Etude de la représentation pyramidale.* Thèse de doctorat, Université Paris IX- Dauphine, 1986.

[BRI 91] P.Brito. *Analyse de données symboliques: Pyramides d'héritage.* Thèse de doctorat, Université Paris IX Dauphine, 1991.

[DID 03]Diday.E , Emilion.R (2003) "Maximal and stochastic Galois Lattices" . *Journal of Discrete Applied Mathematics*. 127, 271-284.

[DID 04] E.DIDAY. Spatial Pyramidal Clustering Based on a Tessellation. Proc.IFCS'2004,Chicago. Spri Verlag.

[DID 84] E.Diday. *Une représentation visuelle des classes empiétantes*. Rapport INRIA n- 291. Rocquencourt 78150, France, 1984.

[JOH 67] JOHNSON S.C. Hierarchical clustering schemes, Psychometrika 32 pp. 241-254, 1967.

[ROD 00] O.Rodriguez. *Classification et modèles linéaires en analyse de données symboliques*, Thèse de 3 cycle, Université Paris IX- Dauphine, 2000.

Sur le consensus des systèmes de classification

Florent Domenach et Bruno Leclerc

Institute of Policy and Planning Sciences
Tsukuba University
1-1-1 Tenno-Dai
Tsukuba, Ibaraki 305-8573, Japon
domenach@sk.tsukuba.ac.jp

École des Hautes Études en Sciences Sociales
Centre d'Analyse et de Mathématique Sociales
54 boulevard Raspail
75270 Paris cedex 06, france
leclerc@ehess.fr

RÉSUMÉ. On considère le problème de l'agrégation de systèmes de classification. Après avoir signalé les résultats obtenus dans le cadre de la théorie des treillis, on reprend l'approche d'Adams (dans le cas des hiérarchies) en s'intéressant aux emboîtements associés aux systèmes de classification. Un résultat d'unicité est obtenu, qui permet d'étendre les propriétés déjà connues du consensus d'Adams

MOTS-CLÉS : Système de fermeture, Système de classification, Implication, Emboîtement, Treillis, Hiérarchie

1 Introduction

Soit S un ensemble fini. On s'intéresse ici à l'agrégation d'un profil $\lambda^* = (\lambda_1, \lambda_2, \ldots, \lambda_k)$ de classifications sur S en une classification consensus $\lambda = c(\lambda^*)$. Par classification, on entend ici une familles de parties (classes) comprenant S et toutes les parties de S à un élément (singletons), et stable par intersection. En d'autres termes, ce sont les systèmes de fermeture contenant tous les singletons.
Ce modèle permet de donner une formalisation commune à des informations structurelles (ordre, arbre,…) fournies par des variables de type divers. Elle contient aussi, comme cas particulier, des problèmes d'agrégation devenus classiques, comme celui des partitions ([RÉG 65], [MIR 75], [BL 95]) ou celui des hiérarchies ([MM 81], [ADA 86]), avec la possibilité d'utiliser les résultats connus sur l'agrégation des systèmes de fermeture ([RAD 01], [LEC 03], [MR 04]).

2 Classifications et systèmes de fermeture

2.1 Définition et cas particuliers

Un *système de classification* sur S est un ensemble $\lambda \subseteq \wp(S)$ de classes (parties) de S. Une classe $C \in \lambda$ est interprétée comme un ensemble d'éléments regroupés en vertu de propriétés communes, ou d'un certain type de proximité entre les éléments de C. On a alors des conditions naturelles :

(C1) $S \in \lambda$ (la classe "universelle") ;
(C2) $C, C' \in \lambda \Rightarrow C \cap C' \in \lambda$ (la classe des éléments appartenant à C et à C') ;
(C3) pour tout $s \in S$, $\{s\} \in \chi$(les éléments sont particularisables).

Alors (C2) et (C3)) entraînent que la partie vide est une classe ; cette propriété n'est pas habituelle mais permet de travailler dans des structures appropriées. Si λ satisfait simplement (C1) et (C2), on dit que c'est un *système de fermeture* (ou *famille de Moore*)

Les modèles classificatoires usuels correspondent fréquemment à des systèmes de classification, parfois avec une légère modification. Ainsi, on obtient un tel système, noté $\mathcal{L}_\wedge$, en ajoutant la partie vide à une hiérarchie $\mathcal{L}_\wedge$*, ou en ajoutant S, les singletons et la partie vide à une partition. Les pyramides (ou quasi-hiérarchies) et les hiérarchies faibles, dans leur variante préservant l'intersection, fournissent d'autres exemples.

2.2 Structures équivalentes

Parmi de nombreuses formalisations équivalentes à celle de système de fermeture (cf. [CM 03]), nous en considérons trois. Une *fermeture* φ sur $\mathcal{P}(S)$ est une application satisfaisant les trois propriétés d'*isotonie* (pour tous $A, B \subseteq S$, $A \subseteq B$ entraîne $\varphi(A) \subseteq \varphi(B)$), d'*extensivité* (pour tout $A \subseteq S$, $A \subseteq \varphi(A)$) et d'*idempotence* (pour tout $A \subseteq S$, $\varphi(\varphi(A)) = \varphi(A)$). Les éléments de l'image $\mathcal{L}_\varphi = \varphi(\mathcal{P}(S))$ de $\mathcal{P}(S)$ par φ sont les *fermés* de φ, et $\mathcal{L}_\varphi$ est un système de fermeture sur S. Réciproquement, une fermeture $\varphi_\mathcal{L}$ sur $\mathcal{P}(S)$, donnée par $\varphi_\mathcal{L}(A) = \bigcap\{F \in \mathcal{L} : A \subseteq F\}$ (i.e., par (C2), la plus petite classe de $\mathcal{L}$ contenant A), est associée à tout système de fermeture $\mathcal{L}$ sur S.

Un *système implicatif complet* (SIC) sur S, noté I, $\rightarrow_I$ ou simplement $\rightarrow$, est une relation binaire sur $\mathcal{P}(S)$ vérifiant :

(I1) pour tous $A, B \subseteq S$, $B \subseteq A \Rightarrow A \rightarrow B$;

(I2) pour tous $A, B, C \subseteq S$, $A \rightarrow B$ et $B \rightarrow C \Rightarrow A \rightarrow B$;

(I3) pour tous $A, B, C, D \subseteq S$, $A \rightarrow B$ et $C \rightarrow D \Rightarrow A \cup C \rightarrow B \cup D$.

Une *relation d'emboîtement* (RE) sur S est aussi une relation binaire $\sqsubset$ sur $\mathcal{P}(S)$, vérifiant :

(E1) pour tous $A, B \subseteq S$, $A \sqsubset B \Rightarrow A \subset B$;

(E2) pour tous $A, B, C \subseteq S$, $A \subset B \subset C \Rightarrow [A \sqsubset C \iff A \sqsubset B$ ou $B \sqsubset C]$;

(E3) pour tous $A, B \subseteq S$, $A \sqsubset A \cup B \Rightarrow A \cap B \sqsubset B$.

Les ensembles de systèmes de fermeture, de fermetures, de SICs et de REs sur S sont respectivement notés $\mathbf{O}$, $\mathbf{X}$, $\mathbf{//}$ et $\mathbf{Y}$. Ils sont deux à deux en correspondance biunivoque. Outre la correspondance entre les fermetures et leurs systèmes rappelée plus haut, on donne ci-dessous deux correspondances, la première due à Armstrong [ARM 74], la seconde établie dans [DL 04] :

$$A \rightarrow B \iff B \subseteq \varphi(A)$$

$$A \sqsubset B \iff [A \subset B \text{ et } \varphi(A) \subset \varphi(B)]$$

Pour des systèmes de classification, $A \rightarrow B$ signifie que toute classe qui contient A contient aussi B, tandis que $A \sqsubset B$ signifie que B contient A et qu'il y a au moins une classe contenant A et non B.

Des conditions supplémentaires correspondent à des systèmes particuliers. Par exemple, une RE correspond à un système de classification si et seulement si elle vérifie la condition (ES) ci-dessous. Elle correspond à une hiérarchie si, de plus, la condition (EH) est substituée à (E3) ([ADA 86], [DL 04]) :

(ES) pour tous $A \subseteq S$, $s \in S$, $A \notin \{\emptyset, \{s\}\}$ entraîne $\emptyset \sqsubset \{s\} \sqsubset A \cup \{s\}$;

(EH) pour tous $A, B, C \subseteq S$, $A \sqsubset C$ et $B \sqsubset C$ entraînent $A \cup B \sqsubset C$ ou $A \cap B = \emptyset$.

3 Consensus dans le treillis des systèmes de fermeture

Chacun des ensembles $\mathbf{O}$, $\mathbf{X}$, $\mathbf{//}$ et $\mathbf{Y}$ est naturellement ordonné : $\mathbf{O}$ par l'inclusion sur $\mathcal{P}(\mathcal{P}(S))$, $\mathbf{//}$ et $\mathbf{Y}$ par l'inclusion sur $\mathcal{P}(\mathcal{P}(S) \times \mathcal{P}(S)) = \mathcal{P}((\mathcal{P}(S))^2)$, $\mathbf{X}$ par l'ordre usuel sur les applications : $\varphi \leq \varphi'$ si $\varphi(A) \subseteq \varphi'(A)$ pour tout $A \subseteq S$. Ces ordres sont isomorphes ou duaux : si φ, I et $\sqsubset$ (resp. φ', I' et $\sqsubset'$) sont la fermeture, le SIC et la RE associés à $\mathcal{L}$ (resp. à $\mathcal{L}'$), on a $\mathcal{L} \subseteq \mathcal{L}' \iff \varphi' \leq \varphi \iff I' \subseteq I \iff \sqsubset \subseteq \sqsubset'$ (cf. [CM 03], et [DL 04] pour les emboîtements).

Les ensembles $\mathbf{O}$ et $\mathbf{//}$ sont stables pour l'intersection, et $\mathbf{Y}$ pour l'union. Les plus grand éléments de $\mathbf{O}$, $\mathbf{//}$ et $\mathbf{Y}$ sont respectivement $\mathcal{P}(S)$, $(\mathcal{P}(S))^2$ et $\{(A, B) : A, B \subseteq S, A \subset B\}$, et les plus petits, $\{S\}$, $\{(A, B) : A, B \subseteq S, B \subseteq A\}$ et la relation vide sur $\mathcal{P}(S)$. Alors, $\mathbf{O}$ et $\mathbf{//}$ sont des systèmes de fermeture, respectivement sur $\mathcal{P}(S)$ et sur $(\mathcal{P}(S))^2$.

Un système de fermeture ordonné par l'inclusion est un treillis ($\curlywedge$, $\vee$, $\cap$) avec $F \vee F' = \varphi(F \cup F')$, pour tous F, $F' \in \curlywedge$. Cette structure latticielle a des conséquences pour le problème de l'agrégation d'un profil $\curlywedge^* = (\curlywedge_1, \curlywedge_2, \ldots, \curlywedge_k)$ de systèmes de fermeture en un système $\curlywedge = c(\curlywedge^*)$. Des résultats sur le consensus dans les treillis ont été obtenus, entre autres, dans [MON 90], [BJ 91] et [LEC 94], avec des conséquences dans des cas particuliers comme ceux des hiérarchies [BLM 86], des partitions [BL 95] ou des ordres [LEC 02]. Des résultats particuliers aux systèmes de fermeture sont donnés dans [RAD 01] et [MR 04].

Une *fédération* sur K est une famille $\wr$ de parties de K vérifiant $[L \in \wr, L' \supseteq L] \Rightarrow [L' \in \wr]$. On lui associe la fonction de consensus $c_\wr$ définie par $c_\wr(\curlywedge^*) = \vee_{L \in \wr} (\cap_{i \in L} \curlywedge_i)$. En particulier, $\wr$ est une *oligarchie* si $\wr = \{L \subseteq K: L \supseteq L_0\}$ pour une partie L_0 fixée de K. Une autre classe de fonctions de consensus est celle des *règles de quota* $c_q = c_\wr$, avec $\wr = \{L \subseteq K: |L| \geq q\}$ pour un nombre donné q ($0 \leq q \leq k$). De façon équivalente, $c_q(\curlywedge^*) = \vee\{A \subset S : |\{i \in K: A \in \curlywedge_i\}| \geq q\}$, le système de fermeture engendré par les classes présentes dans au moins q des systèmes $\curlywedge_i$. En particulier, pour $q = k$, on retrouve l'oligarchie correspondant à $L_0 = K$. On note que si les éléments de $\curlywedge^*$ sont tous des systèmes de classification, il en est de même de $c_\wr(\curlywedge^*)$, quel que soit $\wr$.

Une approche axiomatique (cf. [DM 03]) du problème du consensus sur $\bigcirc$ conduit aux règles oligarchiques ([RAD 01]), tandis que l'approche métrique sur $\bigcirc$, fondée sur la distance de la différence symétrique ∂ définie par $\partial(\curlywedge, \curlywedge') = |\curlywedge \Delta \curlywedge'|$ mène au résultat suivant : toute *médiane* du profil $\curlywedge^*$, c'est-à-dire tout élément $\bigcirc$ de $\bigcirc$ minimisant $\rho(\bigcirc, \curlywedge^*) = \sum_{1 \leq i \leq k} \partial(\bigcirc, \curlywedge_i)$ (cf. [BM 81]), vérifie $\bigcirc \subseteq c_{k/2}(\curlywedge^*)$ [LEC 94].

4 Un résultat d'unicité basé sur les emboîtements

Les fonctions $c_\wr$ ne prennent en compte que la présence ou l'absence de classes dans un nombre suffisant d'éléments du profil. On a observé dans le cas des hiérarchies qu'il y a là une limitation qui peut empêcher la reconnaissance de caractères communs, même évidents. De plus, un consensus basé sur les classes risque d'être trivial, ou presque. Par exemple, si aucune classe (autre que S, la partie vide et les singletons), n'apparaît dans une majorité des éléments du profil, les approches ci-dessus mènent à un consensus peu informatif, car ne contenant que ces classes obligées. Pour cette raison, Adams [ADA 86] a développé une méthode de consensus de hiérarchies par intersection de classes et caractérisé cette méthode à partir des REs des hiérarchies considérées. Ici, nous formulons un résultat général sur l'ajustement d'une relation d'emboîtement à une relation quelconque Ξ sur $\wp(S)$. La seule condition sur Ξ est : $(A, B) \in \Xi$ entraîne $A \subset B$. Considérons alors les deux propriétés suivantes pour un système de fermeture $\curlywedge$ et sa relation d'emboîtement $\sqsubset$:

(AΞ1) $\Xi \subseteq \sqsubset$, (préservation de Ξ)

(AΞ2) pour tous $F, G \in \curlywedge$, $F \subset G$ entraîne $(F, G) \in \Xi$. (emboîtements certifiés)

Théorème. *Il y a au plus un système de fermeture vérifiant simultanément les conditions* (AΞ1) *et* (AΞ2).

Preuve. Supposons qu'il y a deux systèmes de fermeture $\curlywedge$ et $\curlywedge'$, avec les fermeture φ et φ' et les relations d'emboîtement $\sqsubset$ et $\sqsubset'$ associées, vérifiant tous deux (AΞ1) et (AΞ2). On sait que S est dans $\curlywedge$ comme dans $\curlywedge'$. Si $\curlywedge \neq \curlywedge'$, la différence symétrique $\curlywedge \Delta \curlywedge'$ n'est pas vide. Soit F un élément maximal de $\curlywedge \Delta \curlywedge'$, que l'on peut supposer être dans $\curlywedge$. Comme F n'est pas égal à S, il y a au moins un élément G de $\curlywedge$ qui lui est immédiatement supérieur. Par (AΞ2), on a $(F, G) \in \Xi$ et, par (AΞ1), $F \sqsubset' G$. Posons $F' = \varphi'(F)$. Nous avons $F \subset F'$, car $F \notin \curlywedge'$, et $F' \sqsubset' G$, car $F' = \varphi'(F) = \varphi'(F') \subset \varphi'(G) = G$. Mais, selon les hypothèses, $F \subset F'$ entraîne $F' \in \curlywedge$, avec $F \subset F' \subset G$, ce qui contredit l'hypothèse selon laquelle G est immédiatement supérieur à F dans $\curlywedge$.

Lorsque $\curlywedge_1, \curlywedge_2, \ldots$ et $\curlywedge_k$ sont des hiérarchies sur S (et, pour tout i, $\sqsubset_i$ est l'emboîtement associé à $\curlywedge_i$), on retrouve, avec $\Xi = \cap_{1 \leq i \leq k} \sqsubset_i$, la caractérisation par Adams de sa méthode de consensus. On a en fait un résultat plus fort :

Corollaire. *Avec la relation Ξ définie ci-dessus, la hiérarchie obtenue par la méthode d'Adams est le seul système de fermeture vérifiant simultanément les conditions* (AΞ1) *et* (AΞ2)*.*

Un point important est que, avec les résultats d'Adams, nous avons un cas où il existe effectivement un emboîtement vérifiant les conditions (AΞ1) et (AΞ2). Un autre cas, assez proche, a été produit par Semple et Steel ([SS 00]) pour la recherche d'un "super-arbre". Un problème est alors de déterminer pour quelles relations Ξ une telle solution existe.

5 Bibliographie

[ADA 86] ADAMS III E.N., " N-trees as nestings: complexity, similarity and consensus ", Journal of Classification, vol. 3, 1986, p. 299–317.

[ARM 74] ARMSTRONG W.W., " Dependency structures of data base relationships ", Information Processing, vol. 74, 1974, p. 580–583.

[BJ 91] BARTHÉLEMY J.P., JANOWITZ M.F., " A formal theory of consensus ", *SIAM J. Discr. Math.* , vol. 4, 1991, p. 305-322.

[BL 95] BARTHÉLEMY J.P., LECLERC B., " The median procedure for partitions ", *in* I.J. Cox, P. Hansen, and B. Julesz, eds., *Partitioning data sets*, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science 19*, Amer. Math. Soc., Providence, RI, 1995, p. 3-34.

[BLM86] BARTHÉLEMY J.P., LECLERC B., MONJARDET B., " On the use of ordered sets in problems of comparison and consensus of classifications ", *Journal of Classification*, vol. 3, 1986, p. 187-224.

[CM 03] CASPARD N., MONJARDET B., " The lattices of Moore families and closure operators on a finite set: a survey ", *Discrete Applied Math.* , vol. 127, 2003, p. 241–269.

[DM 03]DAY W.H.E., McMORRIS F.R., *Axiomatic Consensus Theory in Group Choice and Biomathematics*, SIAM monograph,forthcoming.

[DL 04] DOMENACH F., LECLERC B., " Closure Systems, Implicational Systems, Overhanging Relations and the case of Hierarchical Classification ", *Mathematical Social Sciences*, vol. 47, 2004, p. 349-366.

[LEC 94] LECLERC B., " Medians for weight metrics in the covering graphs of semilattices ", *Discrete Applied Math.*, vol. 49, 1994, p. 281-297.

[LEC 03] LECLERC B., " The median procedure in the semilattice of orders ", Discrete Applied Math., vol. 127, 2003, p. 285–302.

[MM 81] MARGUSH T., McMORRIS F.R., " Consensus *n*-trees ", *Bull. Mathematical Biology*, vol. 43, 1981, p. 239-244.

[MIR 75] MIRKIN B., " On the problem of reconciling partitions ", In *Quantitative Sociology, International Perspectives on mathematical and Statistical Modelling*, New York, Academic Press, 1975, p. 441-449.

[MON 90] MONJARDET B., 1990b), " Arrowian characterization of latticial federation consensus functions ", *Math. Soc. Sci.* , vol. 20, 1990, 51-71.

[MR 04] MONJARDET B., RADERANIRINA V., " Lattices of choice functions and consensus problems ", *Social Choice and Welfare*, 2004, à paraître.

[RAD 01] RADERANIRINA V., *Treillis et agrégation de familles de Moore et de fonctions de choix*, Thèse, Université Paris 1, 2001.

[RÉG 65] RÉGNIER S., " Sur quelques aspects mathématiques des problèmes de classification automatique ", *ICC Bulletin*, vol. 4, 1965, p. 175-191 (*Mathématiques et Sciences humaines*, vol. 82, 1983, p. 13-29.

[SS 00] SEMPLE C., STEEL M.A., " A supertree method for rooted trees ", *Discrete Applied Mathematics*, vol. 105, 2000, p. 147-158.

Analyse statistique de la contamination des poissons par le mercure en Guyane

Gilles Durrieu — Régine Maury-Brachet — Alain Boudou

Laboratoire LEESA
UMR EPOC CNRS 5805 & Université Bordeaux 1
Place du Dr Peyneau, 33120 Arcachon
g.durrieu@epoc.u-bordeaux1.fr

RÉSUMÉ. La pollution par le mercure en Guyane française est principalement liée aux activités d'orpaillage, qui rejettent des quantités importantes de mercure dans les systèmes aquatiques. Dans cet article, nous présentons plusieurs approches statistiques complémentaires (analyse multivariée, estimation non paramétrique de lois et modèles stochastiques de biométrie et d'accumulation) afin de décrire les niveaux de bioaccumulation du mercure dans les poissons. A partir de ces résultats, nous discutons les possibilités de considérer l'espèce carnivore/piscivore Hoplias aimara comme un bioindicateur de l'accumulation du métal toxique le long des chaînes alimentaires conjointement au risque de toxicité envers les populations humaines dont l'alimentation est basée sur une forte consommation de poissons.

MOTS-CLÉS : analyse des correspondances multiples, environnement, estimateur de type noyau, orpaillage, Guyane française

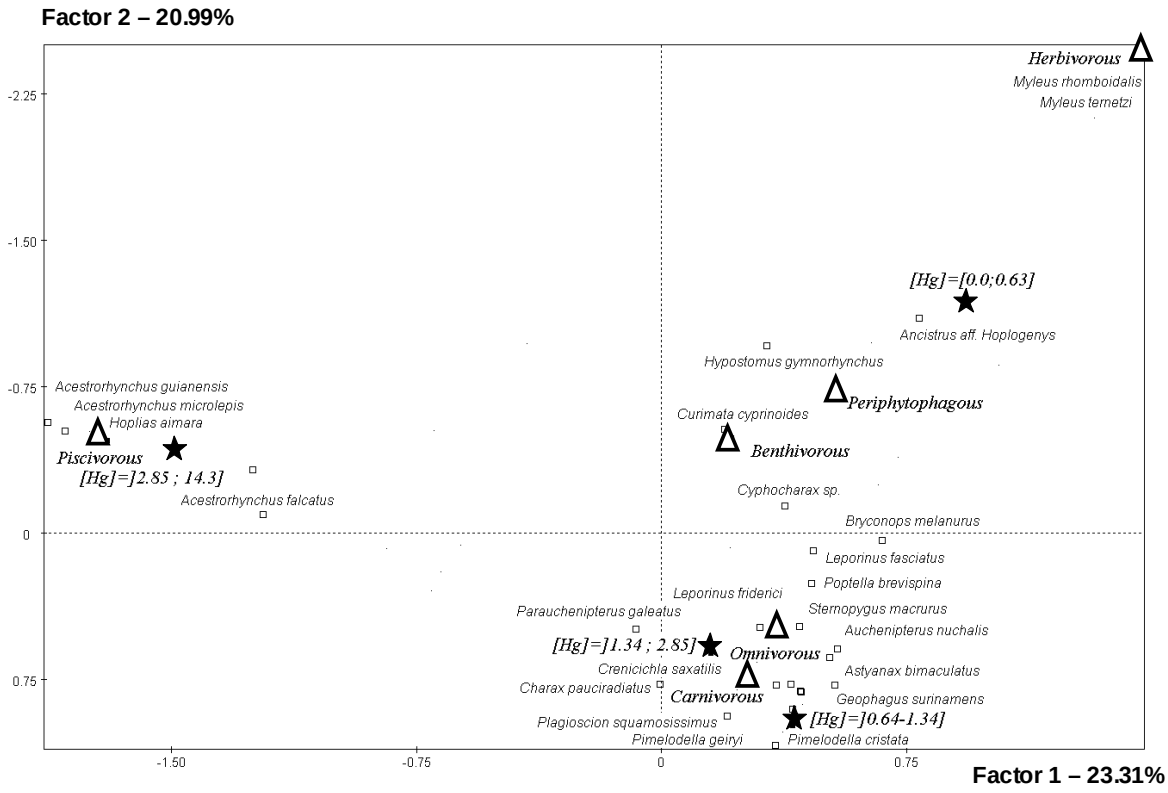
1. Introduction

La pollution par le mercure (Hg) en Amazonie brésilienne et en Guyane française est principalement liée aux activités d'orpaillage, qui rejettent des quantités importantes de mercure élémentaire (Hg°) lors des étapes d'amalgamation et amplifient les apports de Hg inorganique (HgII) *via* l'érosion des sols et des sédiments, naturellement riches en métal. Dans le cadre du programme de recherche interdisciplinaire "Mercure en Guyane", initié par le CNRS/PEVS [BOU 2002], une étude approfondie a été menée sur la contamination de la composante biologique des hydrosystèmes guyanais. Les résultats présentés dans cet article reposent sur l'analyse statistique de données recueillies à propos de l'étude de la contamination de 35 espèces de poissons représentatives des principaux niveaux trophiques (approche globale), avec ensuite une étude limitée à une espèce carnivore/piscivore (*Hoplias aimara*), largement répandue dans les cours d'eau de Guyane et fortement consommée par la population humaine. A partir de ces résultats, nous discuterons la possibilité de considérer cette espèce en tant que bioindicateur de la contamination des milieux aquatiques par le mercure, à l'échelle de l'ensemble des hydrosystèmes guyanais.

2. Résultats

Afin de clarifier les relations entre les régimes alimentaires des poissons et leurs niveaux de contamination par le mercure, une analyse multivariée a été effectué dans les différents sites de pêche (zone du barrage hydroélectrique de Petit-Saut). 42 espèces de poissons ont été capturées pendant les différentes campagnes de pêche en 1999/2000. Pour chacune d'elle, les régimes alimentaires ont été définis à partir des données disponibles dans la bibliographie et des informations fournies par des laboratoires spécialisés en hydrobiologie et ichtyologie générale (IRD, Cayenne et MNHN, Paris). Une analyse des correspondances multiples, prenant en compte les variables «concentration du Hg dans le muscle» (après découpage de cette variable quantitative en 4 classes d'effectifs égaux), «espèces » (35 espèces, celles ayant un effectif inférieur à 3 n'ayant pas été retenues – 986 poissons) et « régimes alimentaires » (5 types de régimes alimentaires) a montré que les régimes

alimentaires et les espèces de poissons qui leur sont associées sont étroitement caractérisés par des niveaux de contamination en Hg très différenciés, le long d'un gradient de concentration allant des espèces herbivores aux espèces piscivores (Figure ci-dessous).



Ainsi, les écarts entre les concentrations du mercure dans le muscle des poissons en fonction des niveaux trophiques sont très importants, fournissant une parfaite illustration de la bioamplification du métal. Les mécanismes géochimiques et écotoxicologiques mis en jeu, notamment la méthylation du Hg(II), la biodisponibilité de cette forme organique à l'égard des espèces aquatiques situées à la base des réseaux trophiques et ensuite les transferts cumulatifs entre les proies et les prédateurs, sont à l'origine des très fortes concentrations chez les espèces aquatiques carnivores, les niveaux d'accumulation étant maximaux chez les organismes piscivores, quelle que soit leur taille. On peut aussi noter de cette analyse que le poisson *Hoplias aimara*, fortement apprécié pour ses propriétés gustatives et abondant dans la quasi-totalité des cours d'eau guyanais, est associé à la classe de concentration la plus forte]2.85; 14.3 µgHg/g, poids sec].

141 *Hoplias aimara* ont été capturés lors de l'ensemble des missions. L'analyse des données biométriques (poids et longueur des poissons) met en évidence un très bon ajustement du modèle puissance poids = 0,010 $Long^{3,15}$ (R^2 = 97%, $p < 0.05$), sans discrimination significative des différents sites étudiés.

Une analyse descriptive réalisée sur l'ensemble des zones de pêche a montré d'une part, que le niveau moyen de contamination, pour l'ensemble des poissons analysés, est environ deux fois supérieur à la norme de consommation de 2,5 µg/g (poids sec), adoptée sur le continent Nord et Sud-américain [WHO 1990] ; et d'autre part, que les concentrations moyennes dans les aymaras capturés dans la zone du Haut-Maroni, à proximité des villages amérindiens au niveau desquels les niveaux d'imprégnation des populations humaines, estimés *via* le dosage du Hg dans les cheveux, sont les plus élevés de Guyane (57,4 % des 235 échantillons de cheveux analysés sont supérieurs à la valeur seuil de 10 µg/g [FRE 2001]), sont environ deux fois moins contaminés que ceux provenant du site de Petit-Saut.

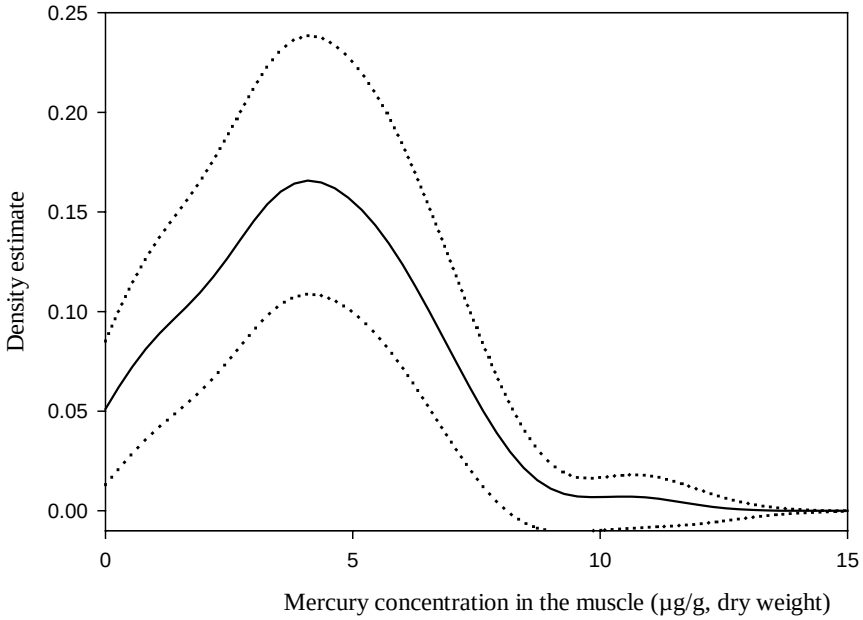
Afin de quantifier de manière probabiliste les risques de toxicité envers la population humaine, nous avons estimé la densité de probabilité de la variable concentration en mercure [Hg] dans le muscle de *Hoplias aimara* sur l'ensemble des sites de pêche par l'estimateur de type noyau

$$\hat{f}_h(x) = \frac{1}{n\,h}\sum_{i=1}^{n} K\left(\frac{x-[Hg]_i}{h}\right),$$

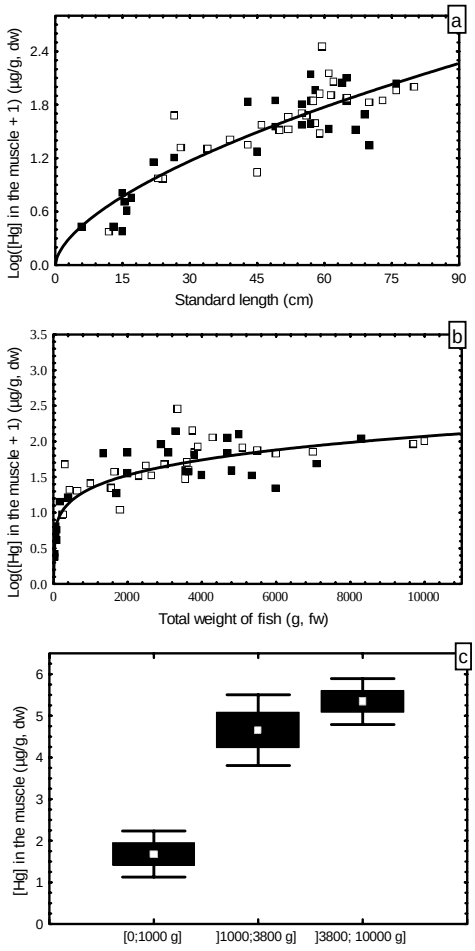
ainsi que l'intervalle de confiance asymptotique de la densité théorique *f* à (1-α)%

$$\left[\hat{f}_h(x) - z_{1-\alpha/2}\sqrt{\frac{\hat{f}_h(x)\|K\|_2^2}{nh}},\ \hat{f}_h(x) + z_{1-\alpha/2}\sqrt{\frac{\hat{f}_h(x)\|K\|_2^2}{nh}}\right]$$

où K est la fonction noyau (Gaussien), h le paramètre de lissage estimée par la méthode de validation croisée (=0.82), n la taille de l'échantillon et $\|.\|_2$ la norme euclidienne. Ainsi, les résultats obtenus dans la Figure ci-dessous



ont montré une bande de confiance étroite au niveau de confiance 95% et ainsi une bonne estimation de la probabilité de dépasser, pour l'ensemble des sites étudiés, le niveau de contamination toléré par la norme de l'OMS (Pr{[Hg] > 2.5 µg/g (poids sec)}= 0.93) obtenue par intégration numérique – algorithme de la quadrature de Gauss en considérant uniquement des poids > 1 kg, qui correspondent aux aymaras habituellement consommés en Guyane. Cette probabilité, extrêmement élevée, souligne donc clairement le risque lié à la consommation de cette espèce, en terme de santé publique. Pour cette raison, nous nous sommes alors intéressés à la mise en place d'un modèle d'accumulation sur une large gamme de poids (de quelques grammes à 10 kg). Ce modèle révèle clairement une très forte accumulation en fonction du poids des individus pour les poissons pesant moins de 1 kg ; au-delà de ce poids, l'accroissement des concentrations du mercure dans le tissu musculaire est très faible et non significatif (analyse de la variance non paramétrique de Kruskall-Wallis, avec pour variable indépendante la variable "poids", discrétisée en 3 classes d'effectifs égaux) (Figure a-c ci-dessous et [DUR 2004]).



3. Conclusion

Cette étude réalisée en Guyane française, dans le cadre d'un programme de recherche pluridisciplinaire [CHA 2002], montre que l'espèce *Hoplias aimara* est associée à de fortes concentrations de mercure dans le muscle. Pour l'ensemble des sites de pêche dans la zone du barrage de Petit-Saut, la probabilité de capturer un *H. aimara* au dessus de la norme fixée par l'OMS est supérieure à 90%. Ainsi, ce poisson peut être utilisé comme un bioindicateur d'évaluation de risque pour la population humaine puisque ces poissons sont présents dans la quasi-totalité des cours d'eau de Guyane, facilement identifiables, se déplaçant peu le long des rivières et étant de ce fait représentatifs de secteurs géographiques définis.

4. Bibliographie

[BOU 2002] BOUDOU A., DURRIEU G., MAURY-BRACHET R. 2002. Bioamplification du mercure et risques à l'égard des populations humaines, Programme mercure en Guyane, CNRS/PEVS, Paris, France, pages 35-51.

[CHA 2002] CHARLET L., BOUDOU A. 2002. Cet or qui file un mauvais mercure, La Recherche, 359, 52-59.

[DUR 2004] DURRIEU G., MAURY-BRACHET R., BOUDOU A., "Goldmining and mercury contamination of the piscivorous fish *Hoplias aimara* in French Guiana (Amazon basin)", *Ecotoxicology and Environmental Safety*, sous presse.

[FRE 2001] FRERY N., MAURY-BRACHET R., MAILLOT E., DEHHEGER M., de MERONA B., BOUDOU A. 2001. Goldmining activities and mercury contamination of native amerindian communities in french Guiana: key role of fish in dietary uptake. *Environmental Health Perspectives* 109, 449-456.

[WHO 1990] WHO (World Health Organization), 1990. Environmental health criteria,101:methylmercury. WHO/IPCS, Geneva, Switzerland.

Une application des cartes topologiques auto-organisatrices à l'analyse des fichiers Logs

Aïcha El Golli*, Brieuc Conan-Guez*, Fabrice Rossi*, Doru Tanasa, Brigitte Trousse**, Yves Lechevallier***

Projet AxIS

**INRIA-Rocquencourt*
Domaine De Voluceau, BP 105
78153 Le Chesnay Cedex, France

***INRIA-Sophia*
2004 route des Lucioles, BP 93
06902 Sophia Antipolis,France

Aicha.Elgolli, Brieuc.Conan-guez, Fabrice.Rossi, Doru.Tanasa, Brigitte.Trousse, Yves.Lechevallier@inria.fr

RÉSUMÉ. *Dans ce travail nous présentons une classification des rubriques visitées par des internautes grâce à une approche classificatoire utilisant une adaptation des cartes topologiques auto-organisatrices aux tableaux de dissimilarités*

MOTS-CLÉS : *Classification, cartes topologiques auto-organisatrices, Web Usage Mining*

1. Introduction

Les cartes topologiques auto-organisatrices de Kohonen [KOH 97] sont parmi les méthodes de classification non supervisées les plus utilisées. En effet, outre leur faculté à regrouper les données similaires au moyen de prototypes comme en quantification vectorielle et/ou en classification, elles autorisent la conservation de la topologie, d'où leur capacité à produire des représentations ordonnées, qu'on appelle prototypes ou vecteurs référents, sur une carte. Le calcul de ces vecteurs référents se base sur la notion de centre de gravité et malheureusement ce concept n'est pas applicable aux données complexes [GAN 04] [1]. Une adaptation de la version batch des cartes topologiques aux tableaux de dissimilarités a été proposée dans [ELG 03], [ELG 04] afin de permettre son application à différents types de données. Dans cette adaptation seule la définition d'une mesure de dissimilarité est nécessaire au déroulement de la méthode.
Soit d la mesure de dissimilarité choisie, rappelons les principales étapes de l'algorithme itératif des cartes topologiques de Kohonen sur tableaux de dissimilarités, à savoir l'étape d'affectation et l'étape de représentation. Chaque neurone c appartenant à la carte C est représenté par un ensemble $a_c = \{z_1 ... z_q\}$ d'élements de Ω de cardinal fixe q et appelé individu référent. Durant la phase d'affectation chaque élément $z_i \in \Omega$ est associé à un neurone gagnant c. Ce neurone est défini comme le neurone qui minimise la fonction d'adéquation d^T entre son individu référent a_c et l'élément z_i.

$$f(z_i) = \min_{c \in C} d^T(z_i, a_c) = \min_{c \in C} \big(\sum_{r \in C} K^T(\delta(r,c)) \sum_{z_j \in a_r} d^2(z_i, z_j) \big)$$

$K^T(\delta(r,c))$ étant une fonction de voisinage qui dépend de la distance $\delta(r,c)$ entre le neurone gagnant c et le neurone r sur la carte. Après affectation de tous les éléments $z_i \in \Omega$, la phase de représentation permet de chercher

1. Un compte rendu rédigé par Brigitte Trousse est disponible à l'URL : http ://www-sop.inria.fr/axis/fdc-egc04/fdc-cr.html

le système des individus référents minimisant une fonction coût E. Pour cela, à chaque neurone r de la carte C on associe l'individu référent a_r minimisant la fonction E_r suivante :

$$E_r = \sum_{z_i \in \Omega} K^T(\delta(f(z_i), r)) \sum_{z_j \in a_r} d^2(z_i, z_j)$$

2. Données et problème

Le développement du Web a entraîné au cours de ces dernières années une explosion des données liées à son activité. Pour analyser ce nouveau type de données, de nouvelles méthodes d'analyse sont apparues sous le terme du Web Mining. Dans cet article nous présentons une partie d'une étude de l'activité du site Web de l'Institut National de Recherche en Informatique et Automatique (INRIA). Le premier objectif de cette étude est l'analyse de la perception de l'activité de l'INRIA par les internautes via celle de son site. Le deuxième objectif est d'apporter des éléments significatifs en vue de l'amélioration de la qualité du site, et de la réponse qu'il apporte aux besoins des utilisateurs. La principale source d'information des visiteurs d'un site Web provient des fichiers Logs listant toutes les requêtes HTTP des clients dans l'ordre de leurs visites. La grande quantité de données et la faible qualité de l'information se trouvant dans ces fichiers nécessitent leurs prétraitements.
Pour notre application, nous avons pris les fichiers Logs HTTP du serveur Web national de l'INRIA et ceux du serveur de Sophia Antipolis issus des 15 premiers jours de l'année 2003. Un utilisateur qui recherche de l'information, navigue parmi tous les serveurs de l'INRIA d'une façon relativement transparente car les pages des différents serveurs Web sont fortement liées entre elles. Il y a de grandes chances que le visiteur ne remarque même pas que le serveur Web a changé. Pour l'analyste du Web Usage Mining, ce changement est très important car il permet d'analyser le comportement de l'utilisateur dans sa recherche de l'information. Ayant un fichier Log Web par serveur, l'analyste doit donc reconstituer le chemin suivi par l'utilisateur sur les différents serveurs sur lesquels ce dernier a navigué. Notre solution est de fusionner tous ces fichiers Logs Web, puis de reconstituer les visites des internautes [TAN 03], [TAN 04].
Deux grandes étapes constituent le prétraitement des fichiers Logs, à savoir la transformation des données et le nettoyage des données. Le résultat du prétraitement est une base de données relationnelle [ARN 03].
L'étape de transformation des données consiste à fusionner les fichiers Logs, rendre anonymes les Ip (ou les noms des domaines) dans le fichier Log obtenu et à grouper les requêtes par session (même Ip, même Agent). Ensuite, les sessions sont divisées en navigations en choisissant un seuil $\Delta t = 30min$.
Le nettoyage des données pour les fichiers Logs consiste à supprimer les requêtes pour les ressources Web qui ne font pas l'objet de l'analyse (les fichiers images par exemple) et les requêtes ou visites provenant des robots Web.
La structure des sites (graphe des liens hypertextes) et l'information sur les utilisateurs des sites (leurs profils) constituent des sources d'information supplémentaires à la base de données relationnelle obtenue suite au prétraitement. A partir de la base de données obtenue grâce au prétraitement, nous avons décidé de sélectionner les navigations d'une durée supérieure à soixante secondes. Nous avons aussi éliminé les pages dont le code statut représente une erreur. Dans nos traitements, nous avons choisi d'analyser les navigations des sites du siège (www) et de Sophia (SOP), l'équivalent de 300 000 pages visitées. A ces 300 000 pages visitées correspondent 3969 navigations visitant donc les pages du siège et aussi celles de Sophia [LEC 03]. A chaque page visitée correspond une rubrique 1 et une rubrique 2.

$$http://\underbrace{www-sop}.inria.fr/\underbrace{axis}/\underbrace{personnel}/Brigitte.Trousse/bri-eng.html$$

Site rubrique 1 rubrique 2

Nous avons créé une taxonomie sur les "rubriques 1". En effet, chaque rubrique 1 appartient à une rubrique sémantique. Par exemple : les rubriques "axis", "sinus", "sloop", sont des projets de l'INRIA Sophia et donc appartiennent à la rubrique sémantique "projet". Nous avons donc créé une table relative aux rubriques sémantiques qui à chaque rubrique fait correspondre sa rubrique sémantique.

3. Traitements et analyses

Nous avons choisi de nous intéresser à la classification des rubriques afin de trouver des corrélations. L'approche que nous adoptons s'appuie sur les navigations des internautes. Pour cela, ayant les 3969 navigations grâce à la base de données relationnelle, nous avons construit un tableau décrivant chaque navigation par la liste des "rubriques 1" consultées. A partir de ce tableau on construit un tableau binaire dont les individus sont les 196 "rubriques 1" et les variables sont les navigations : une navigation N_i visitant la rubrique R_j et pas la rubrique R_k sera codée respectivement par 1 pour R_j et 0 pour R_k dans le tableau (voir tableau 1). Ayant deux vecteurs binaires R_1

Navigations / Rubriques	N_1	N_2	...	N_{3969}
R_1	0	1	...	0
R_2	1	0	...	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
R_{196}	0	0	...	0

TAB. 1. *Tableau binaire décrivant les 196 rubriques visitées (1) ou pas (0) par une navigation*

et R_2, pour définir une similarité ou une dissimilarité spécifique, il est nécessaire d'introduire les quatres quantités suivantes :

- soit a le nombre de fois où $R_1^j = R_2^j = 1$;
- soit b le nombre de fois où $R_1^j = 0$ et $R_2^j = 1$;
- soit c le nombre de fois où $R_1^j = 1$ et $R_2^j = 0$;
- soit d le nombre de fois où $R_1^j = R_2^j = 0$;

La similarité choisie dans notre cas entre les rubriques est la suivante : $S(R_1, R_2) = \frac{a}{a+b+c}$
Ceci correspond à l'indice de similarité de Jaccard. Cet indice indique la probabilité de visite de la rubrique R_1 et de la rubrique R_2 sachant qu'on a visité au moins l'une des deux.

Ayant donc, le tableau de dissimilarités entre les 196 rubriques, nous avons appliqué la méthode des cartes topologiques sur tableau de dissimilarités [ELG 03]. Les paramètres utilisés pour le déroulement de notre algorithme sont les suivants :

Paramètres	valeurs
Dissimilarité	$1 - S(R_1, R_2) = 1 - \frac{a}{a+b+c}$
Ensemble d'apprentissage	196
Nombre de neurones	$12 : 4 \times 3$
cardinal individus référents : q	1

Les résultats obtenus sont assez intéressants. Dans la classification obtenue on s'est intéressé à la rubrique sémantique "projet" et les classes obtenues sont relativement fidèles à l'organisation des sites des projets de l'INRIA. En effet, avant le 1^{er} Avril 2004 les projets de l'INRIA sont groupés par "Thème", il existait 4 thèmes à savoir :

- Thème 1 : Réseaux et systèmes
- Thème 2 : Génie logiciel et calcul symbolique
- Thème 3 : Interaction homme-machine, images, données, connaissances
- Thème 4 : Simulation et optimisation de systèmes complexes

En prenant les individus référents des classes et en se référant aux rubriques sémantiques correspondantes, on obtient la carte de la figure 1.

Et pour mieux voir les associations et les corrélations entre les projets, voici le détail des classes obtenues pour la rubrique sémantique "projet". On représente les individus référents en gras :

manifestation	Projet(Thème 1)	Projet(Thème 3)	inria
manifestation	Projet(Thème 1)	Projet(Thème 4)	Projet(Thème 2)
Projet(Thème 2)	Projet(Thème 4)	Projet(Thème 4)	Projet(Thème 4)

FIG. 1. *La carte (4× 3) obtenue : représentation de la correspondance sémantique des individus référents (pour les projets on représente le thème auquel ils sont attachés)*

Thème 1 Thème 2 Thème 3 Thème 4	meije Koala, croap odyssee Opale	Thème 1	SOP-mistral [2], SOP-Mimosa, SOP-sloop, SOP-rodeo, rodeo, mascotte, **SOP-mascotte**, sloop, SOP-planete, SOP-oasis	Thème 3	robovis, epidaure, ariana, acacia, orion, aid, SOP-robovis , SOP-epidaure, SOP-odyssee, SOP-acacia, **SOP-orion**, SOP-ariana, SOP-aid, SOP-axis, SOP-visa		
Classe 9		**Classe 10**		**Classe 11**		**Classe 12**	
Thème 1 Thème 3 Thème 4	tropics reves Omega	Thème 1	Mimosa, tick, **SOP-tick**	Thème 4	**comore**, mefisto, miaou, SOP-mefisto, SOP-smash	Thème 2	Prisme, SOP-Prisme, SOP-lemme, **SOP-galaad**, SOP-cafe, SOP-saga, SOP-safir
Classe 5		**Classe 6**		**Classe 7**		**Classe 8**	
Thème 2 Thème 4	cafe, lemme, **certilab** Chir, Fractales, opale	Thème 1 Thème 2 Thème 4	Mistral, planete, SOP-meije oasis, saga, safir, SOP-Koala **caiman**, sinus	Thème 4	icare, SOP-sinus, **SOP-icare**, SOP-miaou, SOP-caiman	Thème 1 Thème 2 Thème 3 Thème 4	SOP-tropics SOP-certilab SOP-reves **SOP-Omega**, SOP-sysdys
Classe 1		**Classe 2**		**Classe 3**		**Classe 4**	

Les classes 6 et 10 par exemple sont composées exclusivement de projets appartenant aux mêmes thèmes. Dans la classe 11, on constate la présence simultanée du projet *Aid* et du projet *Axis*. En effet, le projet *Axis* a remplacé le projet *Aid* et la visite de l'un entraine souvent la visite de l'autre, car il y a un lien mutuel sur les deux pages. On retrouve le même comportement pour le projet *Odyssee* et le projet *Robovis*. La classe 12 ne contient aucun projet.
Remerciement *: Nous tenons à remercier Mihai Jurca (Axis Sophia) pour son aide dans le développement de l'outil de prétraitement Axis LogMiner*[3].

4. Bibliographie

[ARN 03] ARNOUX M., LECHEVALLIER Y., TANASA D.AND TROUSSE B., VERDE R., Automatic Clustering for the Web Usage Mining, PETCU D., ZAHARIE D., NEGRU V., JEBELEANU T., Eds., *Proceedings of the 5th Intl. Workshop on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC03)*, Editura Mirton, Timisoara, 1-4 October 2003, p. 54 – 66.

[ELG 03] EL GOLLI A., CONAN-GUEZ B., Adaptation des cartes topologiques auto-organisatrices aux tableaux de dissimilarités, *Xèmes Rencontres de la Société Francophone de Classification,* , 2003, p. 99-102.

[ELG 04] EL GOLLI A., CONAN-GUEZ B., ROSSI F., a self organizing map for dissimilarity data, accepté à IFCS (International Federation of Classification Societies), 2004.

[GAN 04] GANÇARSKI P., TROUSSE B., Actes de l'atelier : Fouille de données complexes dans un processus d'extraction de connaissances, EGC, 2004.

[KOH 97] KOHONEN T., *Self-Organizing Maps*, Springer Verlag, New York, 1997.

[LEC 03] LECHEVALLIER Y., TANASA D., TROUSSE B., VERDE R., Classification automatique : Applications au Web Mining, *Méthodes et Perspectives en Classification, Xèmes Rencontres de la Société Francophone de Classification,* , 2003, p. 157-160.

[TAN 03] TANASSA D., TROUSSE B., Le prétraitement des fichiers log Web dans le Web Usage Mining Multi-sites, *journée Francophones de la toile*, , 2003.

[TAN 04] TANASA D., TROUSSE B., Advanced Data Preprocessing for Intersites Web Usage Mining, *IEEE Intelligent Systems*, vol. 19, n^o 2, 2004, p. 59–65.

2. Le prefixe SOP- signifie que le projet a été consulté à partir du site de Sophia
3. Description de l'outil disponible à l'URL : http ://www-sop.inria.fr/axis/axislogminer/

La méthode graphique : deux exemples de l'influence de Georg von Mayr en France et en Suisse

A. de Falguerolles[1], R. Ostermann[2]

[1]*Université Paul Sabatier (Toulouse III)*
Laboratoire de statistique et probabilités
118 route de Narbonne
F-31062 Toulouse

[2]*Fachhochschule Münster*
Fachbereich Pflege
D-48149 Münster

*RÉSUMÉ. Le nom de Georg von Mayr (1841-1925) est assez peu connu des statisticiens contemporains. Pourtant, ce statisticien allemand a significativement contribué au développement méthodologique de la statistique officielle et de la statistique graphique. Dans ce dernier domaine, les écrits de Mayr ont reçu un accueil favorable en France et en Suisse notamment. Maurice Block (1816-1901) dans son traité théorique et pratique de statistique (1878) reprend, avec quelques réserves prudentes, les enseignements de base que Mayr a publié en 1877. Plus tardif, l'atlas graphique et statistique de la Suisse (1897) reproduit en version bilingue (allemand-français) le noyau dur des enseignements de Mayr. Ces deux textes attestent l'influence certaine de Mayr sur la méthode graphique en France et en Suisse durant la deuxième moitié du XIX*e *siècle. Les journées de classification de Bordeaux sont l'occasion de faire redécouvrir des textes toujours d'actualité*

MOTS-CLÉS : méthode graphique, cartogramme, diagramme, Maurice Block, Georg von Mayr, Atlas graphique et statistique.

1. Introduction

L'article de H. Gray Funkhouser [FUN 37] est, encore de nos jours, une référence incontournable pour l'étude historique de la méthode graphique. En ce qui concerne la seconde moitié du XIXe siècle, Funkhouser reconnaît trois maîtres ou pionniers : Georg von Mayr (1841-1925), Pierre Emile Levasseur (1828-1911) et Emile Cheysson (1836-1910). Aujourd'hui, seule la biographie de Mayr figure dans l'ouvrage collectif *Statisticians of the centuries (2001)* [HEY 01]. Toutefois, Cheysson et Levasseur restent assez connus des géographes [PAL 96]. Funkhouser indique dans son article que Mayr a publié en 1877 la doctrine graphique qu'il avait exprimée lors de nombreux congrès [MAY 77]. C'est ce document qui a servi de base pour de nombreux travaux, en tout cas nombre de ceux que nous évoquons ou discutons dans cet article.

En 1878, l'Exposition Universelle se tient à Paris. Comme le rappelle Funkhouser, de telles expositions fournissaient des occasions rêvées de montrer l'efficacité de la méthode graphique en statistique dont on débattait aussi dans les congrès internationaux de statistique. Cet engouement est attesté par le rapport de circonstance présenté par Cheysson à la Société de Statistique de Paris : *Les méthodes de la statistique graphique à l'exposition universelle de 1878*. Mais l'année 1878 voit aussi la publication en France de deux ouvrages généraux où le rôle de la méthode graphique est souligné : le *traité théorique et pratique de statistique* de Maurice Block et *la méthode graphique dans les sciences expérimentales et principalement en physiologie et en médecine* d'Etienne-Jules Marey (1830-1904).

Certes, ces deux livres n'ont pas le même objectif. Block consacre à la méthode graphique un chapitre et s'intéresse à la grammaire du graphique ; il reprend essentiellement les enseignements de Mayr publiés en 1877 qu'il cite explicitement (et ne présente aucun graphique !). Marey consacre à la méthode graphique la première partie (106 pages) de son volumineux ouvrage ; il s'intéresse au graphique en tant qu'instrument privilégié de synthèse des données, et, plus particulièrement, des données spatio-temporelles (et illustre abondamment la méthode !) . Rappelons que Etienne-Jules Marey et Eadweard Muybridge (1830-1904) sont considérés comme les pères fondateurs de l'étude du mouvement du vivant. A ce titre, ils sont mieux connus des médecins ou des vétérinaires que des statisticiens.

Que vaut la « grammaire » du graphique selon Georg von Mayr ? Son noyau dur est reproduit en version bilingue (allemand et français) dans l'avant-propos de *l'Atlas Graphique et Statistique de la Suisse / Graphisch-statistischer Atlas der Schweiz de 1897.* C'est ce texte que nous évoquons dans la section 2. Nous reprenons ensuite quelques questions soulevées par Block dans son ouvrage de 1878 : en effet, elles ne sont pas sans relation avec des questions classiquement soulevées lors d'exposés contemporains.

2. L'avant-propos de l'atlas de 1897

L'*Atlas Graphique et Statistique de la Suisse / Graphisch-statistischer Atlas der Schweiz de 1897* est un volume spécial d'une série de publications officielles annuelles commencée en 1891 (*Annuaire statistique de la Suisse / Statistisches Jahrbuch der Schweiz).* L'atlas commence par un exposé de la méthode graphique tiré des bonnes pages de l'ouvrage de Mayr publié en 1877. On peut donc y lire le niveau de formation souhaité pour tout lecteur d'un ouvrage destiné au grand public. Hélas, la traduction française n'est pas à la hauteur du texte allemand ! Elle donne lieu notamment à un amusant quiproquo entre cubes nécessaires à la construction de stéréogrammes et dés. Dans la partie française seule, la motivation mise en avant pour la publication de l'atlas est encore l'engouement pour la statistique, et plus particulièrement, pour la statistique graphique dont auraient fait preuve les visiteurs de l'Exposition nationale de Genève de 1896 !

Les circonstances de la publication de cet atlas et l'analyse de trois de ses planches sont étudiées dans un article soumis à publication par les auteurs. L'avant-propos bilingue de l'atlas est aussi disponible auprès de ces auteurs dans une version élargie de cet article.

3. Le Traité de Maurice Block

Comme il a été déjà dit dans l'introduction, Maurice Block (1841-1925) consacre, dans la partie pratique de son *traité théorique et pratique de statistique*, son chapitre XIII à la « méthode graphique ». Celui-ci s'articule en quatre parties.

3.1. *Un aperçu historique*

Maurice Block rappelle que William Playfair est l'inventeur incontesté de la méthode graphique pour les statistiques. (L'ouvrage de Playfair a été traduit en français par François Donnant et publié en France en 1802.) En ce qui concerne la cartographie statistique, Block évoque les contributions de Charles Dupin (1784-1873, professeur du Conservatoire National des Arts et Métiers, chaire de Géométrie appliquée aux arts et statistiques de 1839-1873) et de Charles Minard (1781-1870) Pour l'état de l'art de la méthode graphique, il renvoie au travail de Mayr [MAY 77], qu'il considère « ce qu'il y a de mieux jusqu'à présent » (p 383). Il en reprend donc les grandes lignes pour son exposé sommaire en deux parties : diagrammes et cartogrammes.

3.2. *Un exposé sommaire de la méthode*

L'exposé de Block n'a rien de vraiment original. En ce qui concerne les diagrammes, son évocation de l'utilisation des triangles (comme d'ailleurs celle figurant dans le texte de l'atlas) montre que les représentations des profils ou des données de compositions ternaires fondées sur les propriétés géométriques des triangles équilatéraux ne sont pas encore exploitées par les statisticiens. Il ne mentionne pas non plus l'usage des solides. Rappelons que l'article fondateur des « stéréogrammes » date de 1880 [PER 80].

En ce qui concerne les cartogrammes, il distingue les cartes où l'on représente un seul caractère et celles où l'on tente d'en représenter plusieurs. Pour les premières, suivant en cela l'exemple d'Adolf Ficker (1860), il recommande l'emploi d'une seule couleur, les « groupes se distinguant par des nuances plus ou moins claires » (p 389). Mais il ne présente pas de procédé permettant de relier valeur et nuance (voir un exemple dans [TOB 79] de fonction continue très utilisée en cartographie). De façon générale, Block reste plus succinct que l'atlas suisse qui distingue trois sortes de cartogrammes**.**

3.3. *Le problème des groupements*

Maurice Block étudie deux des difficultés de la méthode graphique : le degré de regroupement des catégories (« Combien formera-t-on de groupes ? ») et la manière de constituer les groupes (« Comment former les groupes ? »). Certains aspects de ces questions, toujours d'actualité, sont étudiées dans la version étendue de ce document.

3.4. *Une mise en garde*

Block termine son chapitre en rappelant que ce serait « flatter les représentations graphiques que de les élever au rang d'une méthode ; il serait plus juste de n'en faire qu'un procédé auxiliaire, dont l'utilité est incontestable dans certaines limites ». Sa mise en garde est sans doute dictée par l'émergence de méthodes plus

mathématiques : Paul Henri imaginera la droite de Henry à Toulouse en 1880 [CRE 93]. Block pressent que la normalisation internationale des graphiques comme celle des nomenclatures, souvent évoquée dans les congrès de cette époque, ne sera bientôt plus d'actualité. Si « certains principes généraux [de la méthode graphique] s'imposent à tous », elle n'est sans doute pas appelée à devenir un langage universel. Il écrit même, « pourquoi défendrions-nous au goût du rédacteur de s'exercer ? ».

4. Conclusion

Dans l'introduction de son ouvrage de 1878, Marey précisait que son intention était « d'exposer les ressources présentes de la méthode graphique et de faire pressentir les développements qu'elle peut prendre sans qu'on puisse assigner de limites à sa bienfaisante extension ». Quelles bienfaisantes extensions constate-t-on maintenant ? Il est clair que l'utopie du graphique 1/1 permettant de représenter toutes les données de base a vécu. Le graphique statistique est devenu plus « théorique » du point de vue mathématique. Il est devenu plus ambitieux aussi : dans sa forme ainsi apurée, il vise soit à représenter les statistiques exhaustives des données, soit à visualiser les résultats d'un modèle (parfois implicite).

Commencée en 1879 et réalisée sous l'autorité de Cheysson, la fameuse série des *Albums de Statistique Graphique* publiée annuellement par le ministère des travaux publics français s'arrêta en 1899 [PAL 96]. L'atlas suisse connut un successeur en 1914, reposant sur la même technologie graphique que celui de 1897. Il reste à effectuer un recensement international des documents de ce type publiés dans la période qui précède la première guerre mondiale. Mais l'espèce n'est pas éteinte, voire connaît un renouveau : *L'atlas statistique des pyrénées / atlas estadístico del pireneo / atles estadístic dels pirineus, pirinioetako atlas estatistikoa (2002), l'atlas de l'aire urbaine de Toulouse (2002)* … en sont de dignes représentants.

De nombreuses publications (voir par exemple [VAL 00]) ne relevant pas du « genre atlas » présentent aussi des graphiques très parlants : « cordonnées parallèles » « diagrammes en boîtes », « biplots », « mosaiques »… Il faudrait aussi parler des nouvelles possibilités (représentations 3D, interactives, dynamiques…) offertes par l'informatique : « crayons lexis », « grand tour »…

En contrepartie, la complexité d'emploi et, partant, la toute puissance aveugle des options par défaut des logiciels nuisent à la qualité des réalisations. Le choix d'une couleur, d'une graduation de teintes, d'une échelle sont trop importants pour être confiés à un logiciel. Une formation de base n'est pas sans utilité et la lecture des formidables ouvrages de Jacques Bertin [BER 73, BER 77] et de Leeland Wilkinson [WIL 99] ne peut qu'être fortement conseillée. En complément pédagogique, on peut essayer de faire de faire revivre certains tâtonnements de la méthode graphique en statistique. C'est l'objet cet exposé. Une visite du site développé par Michael Friendly s'impose aussi. On y verra une présentation très soignée et innovante de l'histoire de la méthode graphique :

http://www.math.yorku.ca/SCS/Gallery/milestone/index.html

5. Bibliographie

[BER 73] BERTIN J., *Sémiologie Graphique. Les diagrammes, les réseaux, les cartes*. 2° édition Gauthier-Villars, Paris, 1973.

[BER 77] BERTIN J., *La graphique et le traitement graphique de l'information*. Flammarion, Paris, 1977.

[BLO 78] BLOCK M., *Traité théorique et pratique de statistique*. Guillaumin et C^{ie}, Paris,1878.

[CRE 93] CREPEL P., Henri et la droite de Henry. *Matapli* 36 : 19-22.

[FUN 37] FUNKHOUSER H. G., Historical development of the graphical representation of statistical data. *Osiris*, 3 : 269-404, 1937.

[HEY 01] *Statisticians of the Centuries*. CC. Heyde and E. Seneta (Editors). Springer, New York, 2001.

[MAR 78] MAREY E.-J., *La méthode graphique dans les sciences expérimentales et principalement en physiologie et en médecine*. Masson, Paris, 1878.

[MAY 77] MAYR G. VON, *Die Gesetzmäßigkeit im Gesellschaftsleben*. Munich, Oldenburg. 1877

[PAL 96] PALSKY G., *Des chiffres et des cartes, naissance et développement de la cartographie quantitative française au XIX^{e} siècle*. CTHS, Paris,1996.

[PER 80] PEROZZO L., Della rappresentazione grafica di una collettività di individui nella successione del tempo, e in particolare dei diagrammi a tre coordinate. *Annali di Statistica* 12 : 1–16,1880.

[TOB 79] TOBLER W.R., Smooth pycnophylactic interpolation for geographic regions. *Journal of the American Statistical Association*,74, 519-530.

[VAL 00] VALOIS J.-P., L'approche graphique en analyse des données (avec discussions). *Journal de la Société Française de Statistique*. 141(4) : 3 - 107, 2000.

[WIL 99] WILKINSON L., *The grammar of graphics*. Springer, New York, 1999.

Qualité de prédiction de performances des algorithmes de classification de données

Edwige Fangseu Badjio, François Poulet

ESIEA Recherche,
38, rue des Docteurs Calmette et Guérin
Parc Universitaire de Laval-Changé
53000 Laval
{fangseubadjio | poulet}@esiea-ouest.fr

RÉSUMÉ. Du besoin de prédire les performances des algorithmes de classification de données sont nés de nombreux travaux. Ces travaux utilisent des mesures de comparaison des ensembles de données et s'appuient sur la classification supervisée. Partant de la présentation de l'algorithme des plus proches voisins utilisé pour la prédiction d'algorithmes de classification de données, nous avons procédé à une évaluation de la qualité des prédictions obtenues par cet algorithme. Forts des observations de cette évaluation, nous présentons une nouvelle approche de prédiction, améliorant ainsi les résultats obtenus par la méthode basée sur les plus proches voisins.

MOTS-CLÉS : prédiction d'algorithmes, classification supervisée, k plus proches voisins, mesures de comparaison de données.

1. Introduction

Les progrès du domaine de la fouille de données ont permis d'intégrer plusieurs algorithmes de classification de données dans un même environnement de fouille. Pour un problème soumis en entrée de l'environnement, se pose alors le problème de sélection de l'algorithme le plus approprié pour la résolution de ce problème.

La boucle de fouille prévoit un cycle permettant à l'utilisateur d'effectuer de nombreux essais de résolution des algorithmes de l'environnement sur les données du problème qu'ils ont à résoudre afin de choisir l'algorithme le plus approprié à cet effet. Etant donné que le temps nécessaire à l'exécution d'un algorithme de classification de données peut s'avérer long, le temps total mis dans la boucle de fouille avant le choix de l'algorithme adéquat à exécuter peut être considérable.

La prédiction qui consiste à estimer une valeur future, sachant que des valeurs connues sont mémorisées permet d'apporter des solutions à ce problème. Cependant, l'utilisation de la prédiction nécessite la mise en œuvre de contraintes afin que la prédiction puisse se faire de façon adéquate.

Partant d'une présentation du problème auquel nous nous attaquons, nous exposerons une des méthodologies utilisées en prédiction d'algorithmes de classification de données, puis nous évaluerons cette méthodologie. Enfin, nous présenterons l'approche que nous proposons et ses résultats avant de conclure sur les perspectives.

2. Description du problème

Etant donné :

- un ensemble *A* d'algorithmes candidats pour une tâche de classification,
- un ensemble *D* d'ensembles de données dont la performance sur l'ensemble d'algorithmes *A* est connue,
- un nouvel ensemble de données *N* relatif au problème d'un utilisateur,

Il s'agit de :

- sélectionner de l'ensemble *D* un sous-ensemble *S* d'ensembles de données tel que chaque élément de *S* soit similaire à *N*,
- retrouver les informations concernant les performances des algorithmes de *A* sur les ensembles de données de *S*,
- prédire la performance des algorithmes de *A* sur *N* en fonction des performances des algorithmes de *A* sur les données de S.

3. Eléments de décision

Mesures de comparaison des ensembles de données

Des mesures statistiques permettent de définir la similarité entre ensembles de données. Les comparaisons des ensembles de données sont possibles grâces à ces mesures. L'algorithme des plus proches voisins a servi à la recherche des similarités entre ensemble de données.

Algorithme des plus proches voisins

L'algorithme des plus proches voisins a été utilisée par [BRA, 00, 03], [KOF, 02], [KAL, 99]. L'idée de cette méthode est la prise de décisions basée sur la recherche de un ou plusieurs cas similaires déjà résolus et dont les résultats ont été mémorisés.

En effet, l'algorithme cherche les k plus proches voisins du nouveau cas et prédit la réponse la plus fréquente de ces k plus proches voisins. La méthode utilise deux paramètres : le nombre k et la fonction de similarité pour comparer le nouveau cas aux cas déjà classés.

4. Evaluation des prédictions

Pour évaluer les capacités prédictives de l'approche des plus proches voisins, nous avons cherché à mesurer la pertinence de la liste classée par performance décroissante d'algorithmes délivrée par cet algorithme. Pour ce faire, nous avons :

Un ensemble *A* constitué de 23 algorithmes de classification supervisée.

$$A = \left\{ \begin{array}{l} Ac2, Alloc80, Backprop, Bayes, BayesTree, C4.5, CART, Cal5, Cascade, Castle, Cn2, \\ Default, Dipol92, Discrim, ITrule, IndCART, KNN, Kohonen, LVQ, LogDisc, RBF, \\ SMART \end{array} \right\}$$

Un ensemble *D* constitué de 19 ensembles de données de UCI [BLA, 98] dont les valeurs des mesures de comparaison sont connues,

$$D = \left\{ \begin{array}{l} australian, Belgian, BT, Credit, Chromosone, Cut, Diabetes, Digits, DNA, Faults, \\ German, Head, Kldigit, NewBelgian, SatImage, Segment, Shuttle, TseTse, Vehicle \end{array} \right\}$$

Un ensemble de données du problème à résoudre : *N = {heart}*

Les performances des algorithmes de *A* sur *N* sont connues.

4.1. *Application de l'algorithme des plus proches voisins*

Nous nous sommes limités à k = 5.

	1	2	3	4	5
N	TseTse	NewBelgian	SatImage	Credit	Australian

Tableau 1 Les cinq plus proches voisins de *N*

Les ensembles de données les plus similaires à Heart sont TseTse, NewBelgian, SatImage, Credit, Australian.

4.2. *Performances des algorithmes*

Pour la performance des algorithmes, nous avons choisi de travailler avec les cinq meilleurs algorithmes de chaque ensemble de données.

	1	2	3	4	5
TseTse	Cn2	IndCART	NewId	CART	Smart -Ac2
NewBelgian	Smart	IndCART	NewId	C4.5	Ac2
SatImage	KNN	LVQ	Dipol92	RBF	Alloc80
Credit	C4.5	IndCART	Cal5	Smart	Castle
Australian	Cal5	IRrule	Discrim	Logdiscr	Dipol92

Tableau2 - Classement des cinq algorithmes les plus performants sur les ensembles de données proches de *N*

L'ensemble de données TséTsé est le plus similaire à *N*, les algorithmes les plus performants ayant servi au traitement de TséTsé devraient être les plus performants pour *N*.

4.3. *Performance effective de d*

Après avoir obtenu les prédictions de performances des algorithmes de *A* sur *N*, nous avons cherché à savoir quelle était la pertinence de cette prédiction. Pour cela, nous avons recherché la performance effective des algorithmes de *A* sur *N*.

	1	2	3	4	5
d	NaiveBayes	Discrim	Logdiscr	Alloc80	Quadisc

Tableau3 - Performance des algorithmes de *A* sur *N*

Le meilleur algorithme obtenu par l'exécution de tous les algorithmes de *A* sur *N* est NaiveBayes, cet algorithme n'apparaît pas dans la liste des 5 meilleurs algorithmes prédits de TséTsé et des autres voisins de *N*.

Discrim, Logdisc, respectivement deuxième et troisième algorithme plus performant pour le traitement de *d* sont retrouvés en troisième et quatrième position du classement des performances d'algorithmes du cinquième plus proche voisin de *N*.

Alloc80 qui occupe la quatrième position du tableau3 est le cinquième algorithme plus performant du troisième plus voisin de *N*, Quadisc n'apparaît pas dans les prédictions. L'explication relative à ces résultats est la suivante : malgré le fait que les ensembles de données TséTsé, NewBelgian, SatImage, Credit, Australian soient les plus proches voisins de *N*, leur similarité avec *N* est négligeable.

4.4. *Synthèse des résultats : vers une nouvelle approche de comparaison*

Ces résultats rejoignent l'un des inconvénients majeurs du raisonnement à partir de cas. Il s'agit de la pertinence de l'ensemble de cas traités sur lequel les décisions sont basées. Pour contourner la difficulté relative à la pertinence de la base de cas, nous avons fixé à partir de simulations un **seuil de similarité** en deçà duquel on ne considère plus deux ensembles de données comme similaires. L'application de cette nouvelle approche permet pour le cas présenté ci-dessus, et pour d'autres cas que nous n'avons pas pu présenter, après calcul des similarités d'aboutir à un résultat stipulant qu'aucun ensemble de données n'est similaire aux données du problème à résoudre. L'exécution effective l'ensemble des données à traiter sur les algorithmes de l'ensemble *A* en parallèle sur un réseau nous permet d'obtenir dans un délai raisonnable des valeurs exactes de performances des algorithmes de *A* sur l'ensemble de données à traiter. Durant sa phase de maintenance, la base de cas traités est mise à jour avec des connaissances relatives à la performance des algorithmes non erronés.

5. Conclusion

Partant d'une évaluation de la qualité de prédiction des algorithmes de classification des données, nous avons présenté une approche permettant d'obtenir de meilleurs résultats de prédiction d'algorithmes de classification de données. Tout comme l'algorithme des plus proches voisins, cette approche s'appuie sur des cas traités en utilisant le calcul des paires de similarités. Les résultats obtenus sont satisfaisants. Nous apportons ainsi une esquisse de solution au problème de sélection d'algorithmes dans un environnement de fouille de données.

Nous nous intéressons à présent à un environnement destiné à des utilisateurs spécialistes des données, qui ont un bagage de base en analyse de données. Nous comptons étendre les travaux décrits ici pour les appliquer à ce problème tout en tenant compte des préférences des utilisateurs.

6. Bibliographie

[BLA, 98] BLAKE C., MERZ C., UCI Repository of machine learning databases, [www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, University of California, Department of Information and Computer Science, 1998.

[BRA, 00] Brazdil P., Soares C., A Comparison of Ranking Methods for Classification Algorithm Selection, Machine Learning: ECML 2000, 11th European Conference on Machine Learning, R. López de Mántaras and E. Plaza (Eds.), LNAI 1810, Springer Verlag, 2000.

[BRA, 03] Brazdil P., Soares C., Costa J., Ranking Learning Algorithms Machine Learning: Using IBL and Meta-Learning on Accuracy and Time Results. Machine Learning 50(3): 251-277, 2003.

[KAL, 99] Kalousis A., Theoharis T., NOEMON: Design, implementation and performance results of an intelligent assistant for classifier selection. Intelligent Data Analysis. volume 3. number 5, pages 319-337, 1999.

[KOF, 02] Köfp C., Iglezakis I., Combination of Task Description Strategies and Case Base Properties for Meta-Learning, Proc of the 2nd Intl. Workshop on Integration and Collaboration Aspects of Data Mining, Decision Support and Meta-Learning (IDDM), pages 65-76, 2002.

[MIC 94] MICHIE D., SPIEGELHALTER D.J., TAYLOR C.C. (eds.), Machine Learning, Neural and Statistical Classification Ellis Horwood, 1994.

Classification croisée entre quasi-hiérarchies

Bernard Fichet

Laboratoire de biomathématiques, 27 bd J. Moulin 13385 Marseille (France)
bernard.fichet@medecine.univ-mrs.fr

RÉSUMÉ. *Répondant à la recherche d'une classification croisée sur le produit cartésien de deux ensembles, il fut établi dans le passé que le produit cartésien de deux hiérarchies était une quasi-hiérarchie. Nous étendons ici ce résultat, au produit cartésien de r λ-quasi-hiérarchies.*

MOTS-CLÉS : *hiérarchie, λ-quasi-hiérarchie, ultramétrique, quasi-ultramétrique, produit cartésien*

1. Introduction

Introduire et développer des structures croisées de classification, répond à une motivation très naturelle. La raison réside dans l'origine de l'analyse des données, qui requiert souvent en amont un tableau de données, généralement rectangulaire, croisant deux ensembles. L'analyse de l'un d'entre-eux repose alors sur le choix d'une distance, voire d'une dissimilarité, et d'une méthode de traitement. Mais lorsqu'il y a double analyse, les résultats sur les deux ensembles sont liés par le tableau. Il est donc naturel de chercher à les comparer.
La représentation simultanée, chère aux méthodes dites factorielles comme l'analyse des correspondances, répond à ce souci. Toutefois elle revient par nature à l'analyse de l'union des deux ensembles. Mais, comme l'avait déjà démontré B. Escofier, 1969, pour l'analyse des correspondances, voir aussi Benzécri, 1973, on pouvait obtenir les mêmes résultats en travaillant sur le produit cartésien des deux ensembles ligne et colonne. D'un point de vue métrique, cela revenait à réaliser le produit L_2 des deux espaces métriques euclidiens, donnés par les métriques du χ^2.
Le produit cartésien de structures de classification évoqué dans cet exposé répond au même objectif. Mathématiquement, étant donnés r systèmes de classes $\mathcal{H}_1, ...\mathcal{H}_r$ sur r ensembles $I_1, ..., I_r$, le produit cartésien $\mathcal{H} = \mathcal{H}_1 \times ... \times \mathcal{H}_r$ sur $I = I_1 \times ... \times I_r$ est le système $\mathcal{H}$ des classes de la forme $H = H_1 \times ... \times H_r$, $H_k \in \mathcal{H}_k$, $k = 1, ...r$. C'est effectivement, à une bijection près, un produit cartésien. Nous avons dans le passé, voir Fichet, 1998, établi que le produit cartésien de deux hiérarchies était une quasi-hiérarchie. Une analyse précise de la démonstration laissait entrevoir une extension aisée au produit de r hiérarchies pour peu que l'axiome de base des quasi-hiérarchies soit généralisé à ce que Diatta, 1997, nomme λ-quasi-hiérarchies. Mais de fait, un résultat encore plus fort peut être établi. Nous montrons ici que le produit de r λ-quasi-hiérarchies, elles-mêmes produits ou non de hiérarchies, est encore une λ-quasi-hiérarchie. Enfin, nous discutons, sans toutefois apporter réponse, d'une possible dualité en termes de produit d'espaces quasi-ultramétriques, lorsque les λ-quasi-hiérarchies sont indicées. En passant, nous établissons que la somme de deux ultramétriques est une quasi-ultramétrique, faisant apparaître comme corollaire le produit direct de deux espaces ultramétriques.

2. Des hiérarchies aux λ-quasi-hiérarchies

L'axiome de base des quasi-hiérarchies a été introduit séparément par Batbedat, 1989, sous le vocable propriété médinclus, en référence à la médiane ensembliste, et par Bandelt et Dress, 1989, avec le concept de hiérarchie faible. C'est le suivant :

$$\forall H_1, H_2, H_3 \in \mathcal{H}, H_1 \cap H_2 \cap H_3 \in \{H_1 \cap H_2, H_2 \cap H_3, H_3 \cap H_1\} \,. \quad (1)$$

Avec la fermeture sous intersection et deux axiomes classiques, à savoir l'ensemble total I appartient à $\mathcal{H}$ et les éléments minimaux de $\mathcal{H}$ partitionnent I, on a ce que l'on appelle une quasi-hiérarchie, voir Diatta et Fichet, 1994. Muni d'un indice de niveau f vérifiant : $H \subset H^{'} \implies f(H) < f(H^{'})$ et (H minimal) $\implies f(H) = 0, (\mathcal{H}, f)$ forme une quasi-hiérarchie indicée.

Il existe une bijection entre quasi-hiérarchies indicées et quasi-ultramétriques, voir Diatta et Fichet, 1994,1998, ainsi que, via une condition de quatre points, Bandelt et Dress, 1994. Rappelons qu'une quasi-ultramétrique est une dissimilarité vérifiant la condition d'inclusion : $\forall k, l \in B^d_{ij}, B^d_{kl} \subseteq B^d_{ij}$ et la condition du diamètre, $\forall k, l \in B^d_{ij}$, $d(k,l) \leq d(i,j)$, où $B^d_{ij} = B(i, d(i,j)) \cap B(j, d(i,j))$ est l'intersection des boules de centre i et j et de même rayon $d(i,j)$.

Les quasi-hiérarchies généralisent les pseudo-hiérarchies indicées, avec leur diagramme pyramidal, en bijection avec les dissimilarités fortement de Robinson, et donc par là les hiérarchies indicées en bijection avec les ultramétriques. Encore, les quasi-ultramétriques contiennent les (semi)-distances de type arboré. D'autres petites propriétés, comme la suivante en réponse à une question de B. Van Cutsem, agrémentent le décor quasi-ultramétrique : l'union de deux hiérarchies, fermé par intersection, est une quasi-hiérarchie ; c'est une pseudo-hiérarchie si et seulement si les deux hiérarchies ont un ordre compatible commun, i.e. tel que les classes soient des intervalles.
L'axiome (1) admet une généralisation naturelle, introduite par Bandelt et Dress, 1994, et Diatta, 1997, voir aussi Bertrand et Janowitz, 2003. Fixant un entier $\lambda \geq 2$: $\cap_i H_i \in \{\cap_{i \neq j} H_i\}_{j=1,..,(\lambda+1)}$ pour toute collection de classe $H_1, .., H_{\lambda+1}$. Pour λ quelconque on une une λ-quasi-hiérarchie, et donc une quasi-hiérarchie correspond à $\lambda = 2$. Par convention, pour énoncer un résultat général, nous conviendrons d'appeler une hiérarchie, une 1-quasi-hiérarchie (alors que pour $\lambda = 1$, l'axiome précédent donne un système de classes emboitées).

3. Résultats

Nous avons montré, Fichet, 1998, que le produit cartésien de deux hiérarchies est une quasi-hiérarchie, offrant ainsi une structure de classification croisée. Un coup d'oeil précis sur la démonstration, montre à l'évidence que celle-ci peut être étendue au produit de r hiérarchies, pour obtenir une r-quasi-hiérarchie. En conséquence, le produit de r λ-quasi-hiérarchies est une μ-quasi-hiérarchie, dès que les λ-quasi-hiérarchies sont elles-mêmes produit de hiérarchies. Avec le théorème suivant, on peut se soustraire de cette dernière condition.

Théorème. *Pour k variant de 1 à r, soit $\mathcal{H}_k$ une λ_k-quasi-hiérarchie sur I_k. Alors $\mathcal{H} = \mathcal{H}_1 \times ... \times \mathcal{H}_r$ est une λ-quasi-hiérarchie sur $I = I_1 \times ... \times I_r$, avec $\lambda = \lambda_1 + ... + \lambda_r$.*

Comme dans Fichet, 1998, on peut munir $\mathcal{H}$ d'une indice de niveau f, dès que les $\mathcal{H}_k$ sont munis d'indices f_k, en posant : $f\,(H) = f_1\,(H_1) + ... + f_r\,(H_r)$, où $H = H_1 \times ... \times H_r$, $H_k \in \mathcal{H}_k$. Dans la référence précédente, un diagramme cohérent est proposé, en montrant que l'espace quasi-ultramétrique associé à $(\mathcal{H}, f)$ est le produit direct des deux espaces ultramétriques originels. Au passage, ce résultat n'est autre qu'un corollaire de la propriété générale suivante.

Proposition. *Si d'et d'' sont deux ultramétriques sur I, alors $d = d' + d''$ est quasi-ultramétrique.*

Peut-on, pour les produits de λ-quasi-hiérarchies, espérer un diagramme similaire ? Encore faudrait-il faire usage d'une bijection entre λ-quasi-hiérarchies indicées et une classe de dissimilarités particulières. Certes, une telle bijection a été établie par Bandelt et Dress, 1994, et Diatta, 1997, via des dissimilarités multivoies. Ce qui est hors de notre propos ici. Et la bijection proposée par Bertrand et Janowitz, 2003, pour riche qu'elle soit, fait malheureusement intervenir des λ-quasi-hiérarchies faiblement indicées, et non indicées.

4. Bibliographie

[ESC 69] Escofier, B., L'analyse factorielle des correspondances, Cahiers du B.U.R.O., Université Paris VI, 13, 1969, p.25-59.

[BEN 73] Benzécri, J.-P., *L'Analyse des Données*, Tomes 1 et 2, Dunod, Paris, 1973.

[BAT 89] Batbedat, A., Les dissimilarités médas et arbas, *Statistique et Analyse des données*, 14 ,3, 1989, p.1-18.

[BAN 89] Bandelt, H.J. and DRESS A.W., Weak hierarchies associated with similarity measures : an additive clustering technique, *Bull. Math. Biology*, 51, 1989, p.133-166

[DIA 94] Diatta, J. and Fichet,B., From Apresjan hierarchies and Bandelt-Dress weak hierarchies to quasi-hierarchies, in : Diday et al., (Eds.), *New Approaches in Classification and Data Analysis*, Springer, Berlin, 1994, p.111-118.

[BAN 94] Bandelt, H.J. and Dress, A.W., An order theoretic framework for overlapping clustering, *Discrete Math*, 136, 1994, p.21-37.

[DIA 97] Diatta, J., Dissimilarités multivoies et généralisations d'hypergraphes sans triangles, *Math. Info. Sci. Hum*. 138, 1997, p.57-73.

[FIC 98] Fichet, B., The L_p-Product of ultrametric spaces and the corresponding product of hierarchies, in : Hayashi C. et al.,(Eds), *Data Science, Classification and Related Methods*, Springer, Berlin, 1998, p.145-153.

[DIA 98] Diatta, J. and Fichet, B., Quasi-ultrametrics and their 2-ball hypergraphs, *Discrete Math.*, 192, 1998, p.87-102.

[BER 03] Bertrand, P. and Janowitz, M.F., The k-weak hierarchical representations : an extension of the indexed closed weak hierarchies, *Discrete Applied Math.*, 127, 2003, p.199-220.

Significativité des niveaux d'une hiérarchie orientée en analyse statistique implicative

Régis Gras*, Pascale Kuntz*, Jean-Claude Régnier**

**Laboratoire d'Informatique de Nantes-Atlantique FRE 2729*
Site Ecole Polytechnique de l'Université de Nantes
La Chantrerie – BP 60601- 44306 Nantes cedex 3
regisgra@club-internet.fr, pascale.kuntz@polytech.univ-nantes.fr
***EA 648 Savoirs, Diversité et Professionnalisation*
86, rue Pasteur – 69365 Lyon cedex 07
Jean-Claude.regnier@univ-lyon2.fr

RÉSUMÉ. Dans le cadre de l'analyse statistique implicative développée à l'origine par R. Gras, nous avons proposé le modèle de « hiérarchie orientée » pour structurer des quasi-implications de type $a \rightarrow b$ *et des règles de règles, appelées* R-*règles, issues d'un corpus de données décrites par des attributs binaires. Dans cette communication, nous proposons un nouveau critère de significativité des niveaux de la hiérarchie orientée basé sur des comparaisons de préordres. Une application à un questionnaire d'opinions illustre l'intérêt de la démarche*

MOTS-CLÉS : régles d'association, hiérarchie, fouille de règles

1. Introduction

Introduites en Extraction des Connaissances dans les Données au début des années *90* par R. Agrawal et al [AGR 96] pour exprimer simplement des tendances implicatives $A_i \rightarrow A_j$ entre des sous-ensembles d'attributs A_i et A_j d'une table relationnelle, les règles d'associations ont rapidement connu une utilisation intensive. De nombreux algorithmes, dont le plus célèbre est certainement *A Priori*, ont été proposés dans la littérature pour leur calcul. Cependant, il est bien connu dans la pratique qu'ils engendrent un nombre prohibitif de règles pour une analyse directe *in extenso*. Il est devenu alors nécessaire de présenter les règles sous une forme organisée, la structuration pouvant être partie intégrante d'un processus de fouille, ou intervenir en seconde étape sur l'ensemble *S* des règles pré-sélectionnées par un algorithme automatique. Prolongeant un travail initié par R. Gras [GRA 79, 92] nous avons récemment développé une démarche [GRA 03], qui permet non seulement de structurer certaines règles pertinentes mais également de découvrir de nouvelles relations implicatives entre ces règles sous la forme $R \rightarrow R'$ où la prémisse *R* et la conclusion *R'* peuvent être elles-mêmes des règles ; on parle alors de *R*-règles. Le modèle proposé, appelé « hiérarchie orientée », est une extension du modèle hiérarchique classique sur l'ensemble des parties de l'ensemble *A* des attributs à un ensemble de règles : les niveaux de la hiérarchie orientée sont des règles ou des *R*-règles, et contrairement au modèle classique où l'ensemble *A* est dans la hiérarchie, une hiérarchie orientée ne contient que des règles significatives selon un critère statistique que nous avons défini.

Comme en classification hiérarchique classique, étant donné la multiplicité des niveaux de la hiérarchie orientée, il est nécessaire de dégager ceux qui sont les plus pertinents par rapport à l'intention classificatrice de l'utilisateur et eu égard aux critères de construction choisis. Cette problématique peut être envisagée selon deux points de vue complémentaires : un point de vue global qui cherche à quantifier la qualité de chacune des partitions associées à chaque niveau de la hiérarchie, et un point de vue local qui se focalise sur la qualité des *R*-règles –assimilables dans une première approche à des classes- construites à chaque niveau. Le premier point de vue, inspiré très étroitement d'une démarche proposée par I. Lerman [LER 81], a été traité par R. Gras [GRA 96]. Le critère de significativé d'un niveau de la hiérarchie orientée est défini à partir d'une préordonnance Ω induite par un indice sur $A \times A$, appelé indice de cohésion, défini pour valider la qualité implicative des *R*-règles. Il s'agit alors de comparer l'ensemble des couples de couples de $A \times A$ qui respectent la préordonnance initiale

Ω avec celui des couples de couples qui respecteraient une préordonnance aléatoire Ω^* dans l'ensemble de toutes les préordonnances de même cardinal que Ω muni d'une probabilité uniforme.

Dans cette communication, nous nous focalisons sur le second point de vue. Ainsi, au lieu de nous intéresser au préordre sur l'ensemble des couples d'attributs, nous nous intéressons au préordre défini sur les couples d'attributs « agrégés » à un même niveau de la hiérarchie orientée pour former une *R*-règle. Nous comparons le nombre d'inversions entre l'ordre observé dans la classe et celui induit d'un modèle statistique, l'intensité d'implication, au nombre d'inversions attendu avec un ordre aléatoire sur un ensemble de même cardinal.

2. Cadre méthodologique

Nous considérons un ensemble *I* d'individus décrits par un ensemble fini $A = \{a_1, a_2, \ldots.\}$ de *m* attributs binaires. On note Ω_A l'ensemble de toutes les *k*-permutations de *A*, pour *k* =1 à *m*, et l'ordre de lecture sur les attributs d'une *k*-permutation est noté <.

Définition 2.1. Une *hiérarchie orientée* H_A sur *A* est un ensemble d'éléments de Ω_A, appelés *classes*, vérifiant les trois conditions suivantes :

1. H_A contient tous les attributs de *A*, appelés classes élémentaires ;
2. Pour chaque couple *C'*, *C''* de classes de H_A, on a $C' \tilde{\cap} C'' \in \{\emptyset, C', C''\}$, où l' « intersection » $\tilde{\cap}$ entre deux séquences de Ω_A est définie comme étant la plus grande sous-séquence d'attributs contigus communs à *C'* et C'' (par exemple, $a_1a_3a_4a_2 \tilde{\cap} a_7a_3a_4 = a_3a_4$); en cas d'égalité on retient la première, selon <, sous-séquence de *C'* ;
3. Pour toute classe non élémentaire *C* de H_A, il existe un unique couple *C'*, *C''* telle que $C = C' \tilde{\cup} C''$, où l' « union » $\tilde{\cup}$ de deux séquences disjointes de Ω_A est définie par la concaténation de *C'* et C'' selon l'ordre < (par exemple, $a_1a_3a_4a_2 \tilde{\cup} a_7a_3a_4 = a_1a_3a_4a_2\ a_7a_3a_4$).

Par exemple, $H_A = \{a_1, a_2, a_3, a_4, a_5, a_2a_3, a_5a_4, a_1a_5a_4\}$ est une hiérarchie orientée sur $A = \{a_1, a_2, a_3, a_4, a_5\}$. Elle peut être représentée par un arbre dont les nœuds représentent des relations d'implications entre les attributs de *A* ; ces relations peuvent être des quasi-implications simples telle que $a_2 \rightarrow a_3$, ou bien des *R*-règles telle que par exemple $a_1 \rightarrow (a_5 \rightarrow a_4)$.

Définition 2.2. Les *R-règles* de degré 0 sont les attributs de *A*, considérés implicitement de la forme $a_i \rightarrow a_i$. Les *R*-règles de degré 1 sont les quasi-implications simples de la forme $a_i \rightarrow a_j$. Une *R*-règle de degré *i*, $1 < i \leq p$, de la forme $R' \rightarrow R''$ entre deux R-règles *R'* et *R''* de degrés respectifs *j* et *k* vérifie $j + k = i - 1$.

Chaque classe *C* d'une hiérarchie orientée peut être associée à une unique *R*-règle [GRA 03], ce qui facilite son interprétation. Ainsi, la hiérarchie H_A de l'exemple ci-dessus peut être associée à un ensemble unique de *R*-règles $\vec{H}_A = \{a_1, a_2, a_3, a_4, a_5, a_2 \rightarrow a_3, a_5 \rightarrow a_4, a_1 \rightarrow (a_5 \rightarrow a_4)\}$. Intuitivement, on voit bien ici qu'une *R*-règle de degré >0 construite à un niveau *k* résulte de l'« agrégation » de *R*-règles précédemment construites à des niveaux inférieurs ; par exemple, la *R*-règle $a_1 \rightarrow (a_5 \rightarrow a_4)$ de degré *2* associée à la classe $a_1a_5a_4$ résulte de l'agrégation de a_1 et de $a_5 \rightarrow a_4$ qui est associée à la classe a_5a_4.

La construction d'une hiérarchie orientée H_A dépend étroitement du critère d'agrégation choisi sur Ω_A. Il s'agit de découvrir des *R*-règles $R' \rightarrow R''$ avec des relations d'implications fortes entre les attributs de *R'* et ceux de *R''*. Ainsi, l'indice *c* que nous avons défini pour quantifier la « cohésion » d'une *R*-règle $R' \rightarrow R''$, où *R'* et *R''* sont respectivement associées aux permutations $a'_1, a'_2, \ldots, a'_k$ et $a''_1, a''_2, \ldots, a''_k$, est de la forme suivante :

$$c(R', R'') = \left(c(R').c(R'').\prod_{i=1,k;k=1,h} c(a'_i, a''_j)\right)^{2/r(r-1)} \text{ où } r = k + h \qquad (1)$$

Pour calculer *c*, nous nous sommes placés dans le cadre de l'analyse statistique implicative. Rappelons brièvement, que dans ce cadre, il s'agit pour évaluer la qualité d'une quasi-implication $a_i \rightarrow a_j$ de modéliser la surprise suscitée par cette règle par rapport au comportement attendu ; en d'autres termes, si $n_{a_i \wedge \neg a_j}$ est le nombre de contre-exemples de la règle et $X_{a_i \wedge \neg a_j}$ la variable aléatoire associée dans un modèle aléatoire, la mesure $\varphi(a_i, a_j)$ de la qualité de la règle est une fonction de la probabilité de l'écart entre $n_{a_i \wedge \neg a_j}$ et $X_{a_i \wedge \neg a_j}$. Ainsi, la cohésion $c(a_i, a_j)$ est mesurée par un contraste entre la valeur de l'implication observée et le désordre associé à une expérience aléatoire que nous mesurons par une entropie.

La construction de la hiérarchie orientée est itérative et obtenue par une méthode ascendante. Elle est initialisée, au niveau *0*, par les attributs. Puis, à chaque niveau on construit une nouvelle classe qui est une union au sens de la définition 2.1. de deux classes construites à des niveaux précédents qui maximisent la cohésion.

3. Critère de cohérence des niveaux

Une classe C de la hiérarchie orientée H_A formée au niveau k est considérée comme *cohérente* pour un seuil α, si il y a conformité ou quasi-conformité au seuil α entre l'ordre -ou le préordre- ω_0 dans lequel s'organisent les attributs de C selon la cohésion et l'ordre -ou le préordre- théorique ω_t défini par leurs intensités d'implication mutuelles. Pour évaluer précisément cette conformité, nous nous basons sur une propriété de l'intensité d'implication [GRA 92] : si le nombre d'occurrences de a_i est inférieur au nombre d'occurrences de a_j, alors la qualité de $a_i \rightarrow a_j$ au sens de φ est meilleure que celle de sa réciproque $a_j \rightarrow a_i$. Ainsi, l'ordre théorique ω_t défini par les intensités d'implications mutuelles coïncide avec celui défini par les occurrences des attributs. Nous comparons la conformité entre ω_0 et ω_t avec celle entre un ordre aléatoire ω^* et ω_t . Nous mesurons la conformité par le nombre d'inversions entre les différents ordres : i est le nombre d'inversions observées entre ω_0 et ω_t et I est le nombre d'inversions entre ω^* et ω_t . Le nombre d'inversions entre deux ordres est simplement défini ici par le nombre de paires d'attributs (a_i, a_j) telles que a_i est avant a_j dans le premier ordre et après dans le second. Intuitivement cela signifie que, si α est petit, la conformité entre ω_0 et ω_t est invraisemblablement très grande puisqu'il paraît exceptionnel que le hasard « fasse mieux » que ce qui est observé.

Définition 3.1. La *cohérence o(C)* d'une classe C d'une hiérarchie orientée est définie par la probabilité $Pr(I > i)$.

D'une façon générale, la mise en œuvre de la cohérence définie en 3.1. nécessite de déterminer la loi de I, que nous noterons I_m dans la suite puisqu'elle dépend du nombre m d'attributs. Notons que le recours à la variable aléatoire I_m donnant le nombre d'inversions entre deux permutations est présent dans le calcul du coefficient de corrélation des rangs τ de Kendall. Cependant, à notre connaissance, la loi de I_m n'est pas donnée [KEN 91], et nous proposons dans la suite une formule de récurrence permettant de calculer ses valeurs dans l'indice de cohérence.

Sous l'hypothèse d'équiprobabilité des permutations, nous considérons la variable aléatoire $N(I_m(k))$ donnant le nombre total de permutations aléatoires correspondant à un nombre d'inversions avec ω_t égal à k pour un nombre d'attributs égal à m. Notons que l'on a trivialement $Pr(I_m = 0) = 1/m\,!$ puisque le nombre d'inversions est nul si et seulement si ω_t coïncide avec ω^*.

Proposition 3.1. Pour tout $k < m$, on a

$$N(I_m(k)) = \sum_{j=0}^{k} N\left(I_{m-1}(j)\right) \quad (5)$$

et, pour tout $k \geq m$, on a

$$N(I_m(k)) = \sum_{j=k-m+1}^{k} N\left(I_{m-1}(j)\right) \quad (6)$$

Conséquence 3.1. Pour tout $k < m$, on a $N(I_m(k)) = N(I_{m-1}(k)) + N(I_m(k-1))$ et, pour tout $k \geq m$, on a $N(I_m(k)) = N(I_m(k-1)) + N(I_{m-1}(k)) - N(I_{m-1}(k-m))$.

On peut ainsi déduire de façon récurrente les différentes valeurs de la loi de I_m utiles pour le calcul de la cohérence.

Proposition 3.2. L'espérance de la variable aléatoire I_m vaut $E(I_m) = m(m-1)/4$ et sa variance vaut $V(I_m) = m(m-1)(2m+5)/72$. Et, la loi de I_m converge vers une loi normale quand m tend vers l'infini.

4. Vers un nouvel indice de significativité

A un niveau $k > 0$ donné, une hiérarchie orientée H_A présente plusieurs classes déjà formées -associées à des R-règles de degré > 0-, et éventuellement, quelques attributs non encore associés. Afin de restituer l'information maximale relative à l'ensemble des classes constituées, cette significativité doit intégrer deux paramètres majeurs : les cohésions des classes dont, par construction de H_A, les valeurs décroissent avec la croissance des niveaux de la hiérarchie, et les cohérences des classes qui peuvent croître ou décroître selon les niveaux en

fonction de la probabilité associée à la variable aléatoire I_m eu égard aux inversions observées et à la taille de la classe. Le concept que nous proposons pour associer ces deux paramètres satisfait des contraintes suivantes liées à la « sémantique » de la significativité.

Définitions 4.1. L'indice *co* de *cohésion-cohérence* qui mesure la significativité de la classe C_{k+1} formée au niveau $k + 1$ est défini par

$$co(C_{k+1}) = \frac{c(C_{k+1})}{c(C_k)}.o(C_{k+1}) \quad (14)$$

Et, par convention, $co(C_0) = 1$.Un niveau k de la hiérarchie orientée H_A est *significatif* si il correspond à un maximum local de l'indice de cohésion-cohérence de la classe formée à ce niveau.

En effet, l'indice *co* n'étant pas une fonction monotone, il apparaît des maxima locaux correspondant d'une part à une meilleure adéquation entre les restrictions, à la classe formée à ce niveau, des préordres théoriques ω_t et contingents ω_0, et d'autre part à une bonne cohésion.

Définitions 4.2. La *qualité* de l'ensemble des niveaux h, $0 \le h \le k$, est définie par

$$q_k(H_A) = \left(\prod_{i=1}^{k} oc(C_i) \right) \quad (15)$$

où C_i désigne la classe formée au niveau i.
La hiérarchie orientée H_A est *significative* au niveau k si sa qualité $q_k(H_A)$ admet un minimum local.

5. Conclusion

Nous avons développé une approche pour évaluer la significativité des niveaux d'une hiérarchie orientée et la qualité d'une hiérarchie orientée partielle qui tient compte du préordre défini sur les attributs de chaque *R*-règle constituée à chaque niveau de la hiérarchie. Cette approche ne nécessite pas, contrairement à une approche globale précédemment employée, la détermination d'une préordonnance sur l'ensemble des couples selon le critère de cohésion. De plus, lorsque le nombre m d'attributs « classés » devient grand les calculs du nouveau critère peuvent être simplifiés par le recours à une loi normale.

Nous avons appliqué cette méthodologie sur les résultats d'une enquête de l'Association des Professeurs de Mathématiques auprès de professeurs de mathématiques. Les niveaux significatifs avaient, à l'exclusion d'un niveau, été obtenus précédemment avec la méthode globale inspirée des travaux de I. Lerman, 1981. Cependant, ce résultat n'a pas valeur de généralité. Dans la situation expérimentale elle-même la sémantique semble bien respectée dans les deux cas.

Notre approche pourrait être généralisée à la recherche d'une mesure de distorsion entre deux permutations, prolongeant ainsi des travaux de Kendall. Mais, de nouvelles mises à l'épreuve sur des données réelles et, en particulier, des corpus de grandes tailles tels qu'on les trouve en fouille de données, permettront une comparaison plus robuste sur le plan de l'information restituée au cours des analyses que pourraient en faire des experts des domaines étudiés.

6. Bibliographie

[AGR 96] AGRAWAL R., IMILIENSKY T., SWAMI A.. Mining association rules between sets of items in large databases. Proc. of the ACM SIGMOD'93, p. 679-696, AAAI Press, 1996.

[GRA 79] GRAS L. Contribution à l'analyse expérimentale et à l'analyse de certaines acquisitions cognitives et de certains objectifs didactiques en mathématiques. Thèse d'Etat, Université de Rennes 1, 1979.

[GRA 92] GRAS R., LAHRER A. L'implication statistique, une nouvelle méthode d'analyse de données. Mathématique, Informatique et Sciences Humaines, 120 : 5-31, 1992.

[GRA 96] GRAS R., RATSIMBA-RAJOHN. Analyse non symétrique de données par l'implication statistique. RAIRO-Recherche Opérationnelle, 30(3) : 217-232, 1996.

[GRA 03] GRAS R., KUNTZ P. Discovering R-rules with an oriented hierarchy., Proc. of the 4th Int. Conf. on Knowledge Discovery and Discrete Mathematics, JIM'03, INRIA, p. 223-229, 2003.

[KEN 91] KENDALL M.G, STUART A., Kendall's advanced theory of statistics. Vol. 2, Edward Arnold, London, 1991.

[LER 81] LERMAN I.C. Classification et analyse ordinale des données. Dunod, 1981.

Clustering by density in any metric space

Alain Guénoche

IML, 163 Av. de Luminy, 13009 Marseille (France)
guenoche@iml.univ-mrs.fr

RÉSUMÉ. *Nous présentons une nouvelle méthode de classification à partir d'une distance pour extraire des classes et construire une partition d'un ensemble X muni d'une distance D. La construction repose sur une fonction de densité calculée en chaque sommet d'un graphe $G = (X, E)$ construit à partir de la matrice de distance. Les classes de X sont des parties connexes de G agglomérées autour des sommets de forte densité.*

MOTS-CLÉS : *Classes, partitions, densité, espace métrique*

1. Graph, classes and partitions

Given a distance matrix on X, establishing partitions (all the elements are clustered in disjoint classes) is generally made optimizing a criterion over the set of all the partitions with a number of classes that must be indicated. In this paper, we investigate a different approach. From the distance matrix, we first build a valued graph $\Gamma = (X, E)$, for each vertex a density function De is evaluated and we perform clustering from Γ and De. Our algorithm differs from similar approaches ([FRA 92], [H.M 99], [RIV 03], [ROU 03]) in many ways ; the graph is not a threshold graph, we use the valuation of the edges to measure a density in each vertex (instead of edge) and to perform progressive clustering. An extended version of this article has been presented at the IFCS conference ([GUE 04]).

Starting from a distance matrix, a threshold graph is generally used, keeping only edges corresponding to distance values lower than a threshold σ. The nested family of threshold graphs is obtained from a distance matrix making σ vary from 0 to $Dmax$, the largest distance value. But for many metrics, as the Sczekanowski-Dice distance on graphs, or Manhattan distance on boolean arrays, choosing a threshold value is a delicate problem. On sparse graph, most of the distance values are equal to the maximum, and a small threshold gives many singletons. So we try to keep the same number of edges from any vertex, selecting a graph which generally contains a single connected component.

1.1. *Graph*

Given a distance matrix, $D : X \times X \rightarrow \mathbb{R}$, the first operation is to select a degree δ which works as a threshold. From any element x, the distance values $D(x, y)$ are ranked in decreasing order and the δ-th value gives the σ_x threshold. Then, we take as edges all the pairs (x, y) such that $D(x, y) \leq \sigma_x$. Let $n = |X|$, $m = |E|$ and $\Gamma_\delta = (X, E)$ the corresponding graph. It is not a classical threshold graph on D, since the threshold values are not the same for all the vertices. And it is not really a regular graph with degree δ, because the edge selection process is not symmetrical ; y can be one of the δ closest element to x but it can be different for x to y. This directed relation is symmetrized in Γ_δ ; consequently, there can be more than δ vertices incident to x.

When there is no ambiguity on the δ value, the graph will simply be denoted Γ. For any part Y of X, let $\Gamma(Y)$ be the set of vertices not in Y that are adjacent to Y. Thus, the neighborhood of x is denoted $\Gamma(x)$, the degree of a vertex x is $Dg(x) = |\Gamma(x)|$.

1.2. *Density function*

For each vertex x, we evaluate a density function denoted De which would be high when the elements of $\Gamma(x)$ are close to x. We propose a density function computed from the average length of the edges from x.

$$De(x) = \frac{Dmax - \frac{1}{Dg(x)} \sum_{y \in \Gamma(x)} D(x, y)}{Dmax}$$

The dense classes are by definition connected parts in Γ sharing high density values. Our initial idea was to search for a density threshold and to consider the partial subgraph whose vertices have a density greater than this threshold. Classes would have been its connected components. This strategy does not give the expected results. Enumerating all the possible threshold values, we have observed that often none was satisfying. By decreasing the threshold, we often obtain only a single growing class, and many singletons. Since there is no straightforward way to determine a threshold, the *local maximum values* of the density function are considered.

1.3. *Classes at three levels*

We construct classes in three steps :

Kernels

A kernel, denoted K, is a connected part of Γ, obtained by the following algorithm : we first search for the local maximum values of the density function and we consider the partial subgraph of Γ reduced to these vertices.

$$\forall x \in K, \forall y \in \Gamma(x) \text{ we have } De(x) \geq De(y).$$

The initial kernels are the connected components of this graph. More precisely, if several vertices with maximum value are in the same kernel, they necessary have the same density value ; otherwise the initial kernels are singletons. Then, we assign recursively to each kernel K the vertices (i) having a density greater than or equal to the average density value over X and (ii) that are adjacent to only one kernel. Doing so, we avoid any ambiguity in the assignment, postponing the decision when several are possible.

The number of kernels is the number of classes and it remains unchanged in the following. So it is not necessary to indicate first this number, as for all the alternative methods optimizing a criterion. We shall see that it performs pretty well, when there is a small number of classes, having from 30 to 50 elements.

Extended classes

At the second level, we just assign elements that are connected to a unique kernel, whatever is their density. If an element not in a kernel is connected to several ones, the decision is again postponed.

Complete classes

Finally, to get partitions, we assign the remaining elements to one class. For x and any extended class C to which it is connected, we compute the number of edges between x and C, and also its average distance value to C. Finally there are two candidates, the majority connected class C_m and the closest one C_d. If they are identical, x is connected to it. And if they are different we apply the empiric following rule : if $\frac{|C_m|}{|C_d|} > 1.5$, class C_m is retained, because the number of links to C_m is clearly larger than to C_d ; otherwise C_d is retained. When classes are not necessary disjoint, x can be assigned to both classes.

1.4. *Complexity*

To establish graph Γ, it is necessary to order the distance values from any x. The computation of σ_x is in $O(n \log n)$ and the selection of the edges is in $O(n)$. Finally of the graph construction is in $O(n^2 \log n)$. To evaluate $De(x)$ it is sufficient to test the edges in the neighborhood of x which contains at most n vertices. The computation of the density function is thus in $O(n^2)$.

Kernel computation is in $O(n^2)$ to find the local maximum vertices, and in $O(m) = O(n^2)$ to determine the kernel elements. During the following steps, for any x we count its connections to the kernels, and then to the extended classes. Both are also in $O(n^2)$. Finally, the complexity of the clustering method is $O(n^2 \log n)$.

2. Monte Carlo simulation on binary data

In order to show that this method permits to recover existing classes, we have tested it on euclidian, boolean and graph distances. For all these metric spaces, we have generated distance arrays containing initial classes. The two main points to assess are the ability to recover the correct number of classes and their quality. Here, we only detail the results on the symmetrical difference distance on binary tables.

First, we have developed a generator of binary tables (n rows, m columns) in which there are p classes. Each class is indicated by a specific attribute taking value 1 only for its elements. For the $m - p$ other attributes, value 0 or 1 is selected at random, with .5 probability. The attributes are weighted : 1.0 for those characterizing the classes and a random number between 0 and 1 for the others. At each trial, the symmetrical difference distance between rows is computed.

2.1. *Quality of the classes compared to the initial partition*

For the three levels of classes, we would like to estimate the quality of the obtained sets of classes, and so the efficiency of the clustering process. The initial partition is denoted $P = \{C_1, ..C_p\}$. Let n'_c be the number of classified vertices at each level. They are distributed in p' classes denoted $C'_1, ..C'_{p'}$ realizing a partition P' over a subset of X for the kernels and the extended classes.

We first aim to map the classes of P' onto those of P evaluating $n_{i,j} = |C_i \bigcap C'_j|$. We define the *corresponding* class of C'_j, denoted $\Theta(C'_j)$, as the one in P, that contains the greatest number of elements of C'_j. $\Theta(C'_j) = C_k$ if and only if $n_{k,j} \geq n_{i,j}$ for all i from 1 to p.

In order to measure the accuracy of the classes, we evaluate three criteria.

– τ_c : the percentage of clustered elements in P' ($\tau_c = \frac{n'_c}{n}$).

– τ_e : the percentage of elements in one of the p' classes which belong to its corresponding class in P.

$$\tau_e = \frac{\sum_i |\Theta(C'_i) \bigcap C'_i|}{n'_c}$$

– τ_p : the percentage of pairs in the same class in P' which are also joined together in P.

The first criterion measures the efficiency of the clustering process at each level ; if very few elements are clustered, the method is inefficient. For the second criterion, we must compute, for each class in P', the distribution of the elements of any initial class to define its corresponding class in P. Thus it can be interpreted as a percentage of "well classified" elements. The third one estimates the probability for a pair in one class of P' to come from a single class in P.

Remark : The two last criteria may reach their maximum value (1.0) even when partitions P and P' are not identical. When a class in P is subdivided in two parts, they will have the same corresponding class ; all their elements will be considered as well classified, and the rate of pairs will also be equal to 1.

2.2. *Results*

We have generated 200 binary tables with 200 elements distributed in 5 classes, setting first $m = 10$ and then $m = 20$. Table 1 indicates the percentage of trials giving each computed number of classes when δ varies.

	Nb. of classes	2	3	4	5	6	7	8	9
	$\delta = 18$	0.0	.01	.04	.41	.36	.12	.04	.01
m=10	$\delta = 20$	0.0	.01	.16	.63	.15	.05	.01	0.0
	$\delta = 22$	0.1	.05	.32	.52	.09	.01	.01	0.0
	$\delta = 18$	0.0	.04	.14	.29	.25	.19	.09	.01
m=20	$\delta = 20$	0.0	.05	.27	.32	.23	.12	.01	0.0
	$\delta = 22$	.02	.11	.30	.33	.19	0.4	.01	0.0

Table 1 : Distribution of the number of classes according to the number of attributes m and the degree δ.

One can see that for $m = 10$ and $\delta = 20$ the correct number of classes is the most frequently recovered. It differs at most of one unity in 94% of the trials. It is less promising when $m = 20$ but, in that case the 5 partitioning attributes are hidden by 15 random ones. Now we evaluate the quality of the classes, using the above criteria with $\delta = 20$, because problems giving a number of classes greater than 5 are compensated by those giving a lower one.

	$m = 10$			$m = 20$		
	τ_c	τ_e	τ_p	τ_c	τ_e	τ_p
Kernels	.46	1.0	1.0	.28	.90	.83
Classes	.80	.96	.93	.47	.80	.68
Partitions	1.0	.92	.86	1.0	.77	.63

Table 2 - Average results of the quality criteria.

These two tables prove that this clustering method is able to recover classes in binary tables when some attributes possess the partitioning information. Similar simulations have been done for euclidian spaces and graphs. In each case the number of classes can be correctly predicted.

The density clustering method has many advantages over classical partitioning ones.
- It allows both to extract partial classes (that do not cover the complete set of elements) and to built a partition.
- It provides the number of classes, and the correct number can be recovered if the classes have a few ten of elements.

Finally it is a one parameter method (the initial degree) that can be used for large clustering problems.

3. Bibliographie

[FRA 92] H. de Fraissex and P. Kuntz, Pagination of large scale networks ; embedding a graph in $\mathbb{R}^n$ for effective partitioning, vol. 2, 1992, , p. 105-112.

[H.M 99] H. Matsuda, T. Ishihara, A. Hashimoto, Classifying molecular sequences using a linkage graph with their pairwise similarities, vol. 210, 1999, , p. 305-325.

[ROU 03] J. Rougemont and P. Hingamp, DNA microarray data and contextual analysis of correlation graphs, *BMC Bioinformatics*, vol. 4 :15, 2003.

[RIV 03] A.W. Rives and T. Galitski, Modular organization of cellular networks, vol. 100, 2003, , p. 1128-1133.

[GUE 04] A. Guenoche, Clustering by graph density, *Proceedings of IFCS'04 conference*, University of Chicago, 2004.

Acknowledgements

This work is supported by the "Origine de l'Homme, des Langages et des Langues" (OHLL) CNRS program.

La discrimination à plus de deux classes Comparaison de plusieurs approches issues des *Support Vector Machines* (SVM)

Emmanuel Jakobowicz

Ecole Polytechnique Fédérale de Lausanne
Institut de mathématiques
CH-1015 Lausanne (Suisse)
emmanuel.jakobowicz@epfl.ch

RÉSUMÉ. *L'apprentissage automatique est aujourd'hui en pleine expansion, suite aux découvertes de V. Vapnik, la méthode de discrimination issue de ces théories obtient des résultats extrêmement bons. Les Support Vector Machines (SVM) ont été très bien développés dans le cas de deux classes, mais pour le cas de plus de deux classes plusieurs méthodes existent mais elles n'ont pas été clairement comparées. Dans ce travail, nous avons rassemblé les principales méthodes existantes et les avons comparées sur un certain nombre de jeux de données tests. D'autre part, comme le pourcentage de bien classées n'est bien souvent pas suffisamment sensible pour juger de la qualité d'un modèle, nous avons étudié de nouveaux critères de validation.*

MOTS-CLÉS : *Support Vector Machines (SVM), multi-classes, critère de validation, apprentissage automatique*

1. Introduction

Les méthodes de Support Vector Machines ou Séparateurs à Vaste Marges (SVM) font parties aujourd'hui des méthodes de discrimination les plus efficaces. Ce sont des méthodes d'apprentissage automatique, c'est-à-dire à partir d'un échantillon d'apprentissage, on va créer un modèle applicable sur des échantillons à tester. Elles ne sont par contre pas adaptées au cas d'une discrimination à plus de deux classes. Dans ce travail, nous avons cherché à rassembler les techniques utilisées pour passer à plus de deux classes.
Tout d'abord nous présenterons les SVM bi-classes. Suivra la description des différentes méthodes à plus de deux classes. J'introduirai ensuite rapidement quelques nouveaux critères de validation des modèles. Cet article se terminera sur des remarques et conclusions issues d'un certain nombre d'applications de ces différentes techniques.

2. Les Séparateurs à vaste marges (SVM)

Les SVM sont des méthodes issues de la théorie de minimisation structurelle du risque développée par V. Vapnik [VAPNIK 98]. Leur but est de séparer linéairement les observations de l'échantillon d'apprentissage en les projetant dans un espace de très grande dimension. La séparation se fera par une maximisation de la marge. Pour des observations $\mathbf{x}_i$ de la classe $y_i \in \{-1, 1\}$, il y aura deux étapes :

1. Projection des observations dans un espace de Hilbert à noyau reproduisant (de très grande dimension)
2. Séparation linéaire des données ainsi transformée en tentant de maximiser la marge.

Ceci revient à résoudre le problème :

$$\begin{array}{ll} & \min_{\mathbf{w}} \frac{1}{2} \parallel \mathbf{w} \parallel^2 \\ s.c. & y_i(\mathbf{w}\,\Phi(\mathbf{x}_i) + b) \geq 1 \quad \forall i \end{array}$$

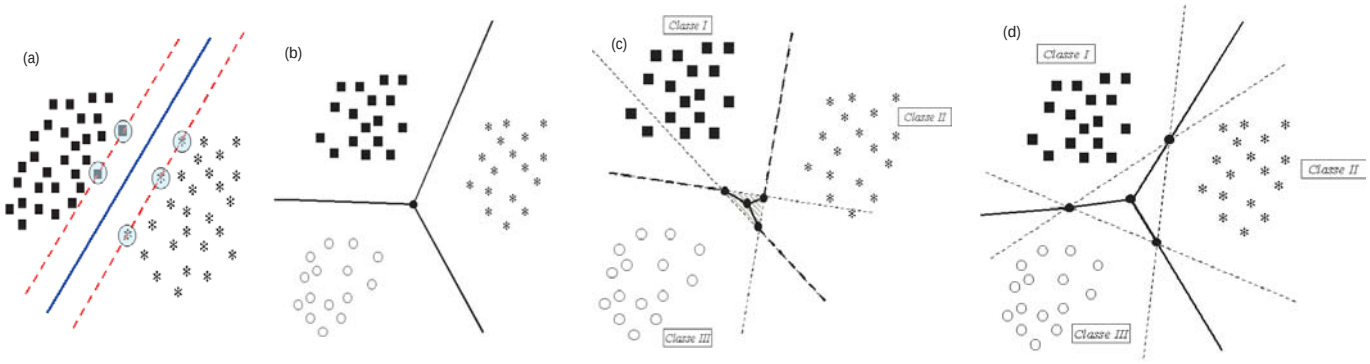


FIG. 1. *(a) Séparation par maximisation de la marge ; Méthodes M-SVM : (b) Weston & Watkins et Crammer & Singer ; (c) "un contre un" ; (d) "un contre le reste"*

La solution de son dual ne dépendant que du produit scalaire $\Phi(\mathbf{x}_i)\Phi(\mathbf{x}_j)$, nous utiliserons des fonctions appelées noyaux notées $k(\mathbf{x}_i, \mathbf{x}_j)$ qui devront satisfaire quelques propriétés (propriété de Mercer) et qui permettront de ne pas avoir à calculer de produit scalaire dans un espace de très grande dimension.

Le cas non séparable :
Bien souvent, malgré la projection, les données restent non séparables, on devra donc modifier le problème primal qui prendra la forme suivante :

$$\begin{array}{ll} & \min_{\mathbf{w},\xi_i,C} \|\mathbf{w}\|^2 + C \sum \xi_i \\ s.c. & y_i(\mathbf{w}\,\Phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \quad \forall i \end{array}$$

On a ajouté des variables d'écart et un paramètre C qu'il faudra régler afin d'équilibrer d'un côté la maximisation de la marge et de l'autre le nombre d'observations que l'on accepte de mal classer.
Ces méthodes sont adaptées pour deux classes mais leur généralisation à plus de 2 classes n'est pas directe.

3. Les SVM à plus de deux classes

3.1. *Méthodes issues directement des SVM bi-classes [*FRIEDMAN *96]*

Méthode "un contre un" : On va tester toutes les paires de classes. On aura donc $\frac{k(k-1)}{2}$ problèmes du type SVM. [FRIEDMAN 96]
Pour appliquer cette méthode, pour chaque observation, on crée une nouvelle observation à k catégories, si entre les classes i et j, l'observation $\mathbf{x}$ choisit i, alors on augmente de 1 la i^{eme} catégorie de la nouvelle observation.
On choisit pour l'observation $\mathbf{x}$ la classe j correspondant à la catégorie avec le plus de vote. En cas d'égalité, on choisit le plus petit index de classe.

Méthode "un contre le reste" : On va tester chaque classe contre les $k-1$ autres classes. On choisira la classe obtenant les meilleurs résultats.

3.2. *Méthodes multi-classes*

Plusieurs méthodes ont été mises au point afin de passer à plus de deux classes en traitant simultanément toutes les classes. Les méthodes de Weston & Watkins et de Crammer & Singer seront développées.

Méthode de Weston & Watkins[WESTON 98] : Elle traite toutes les classes simultanément en résolvant le problème suivant :

$$\begin{array}{ll} \min_{\mathbf{w}} \frac{1}{2}\sum_{j=1}^{k} \|\mathbf{w}_j\|^2 + C\sum_{i=1}^{n}\sum_{j=1}^{k}\xi_i^j \\ s.c. & \begin{cases} (\mathbf{w}_{C(\mathbf{x}_i)} - \mathbf{w}_j)^T\mathbf{x}_i + b_{C(\mathbf{x}_i)} \geq 1 - \xi_i^j \\ \xi_i^j \geq 0, \quad (1 \leq i \leq n),\ (1 \leq j \neq C(\mathbf{x}_i) \leq k) \end{cases} \end{array}$$

Méthode de Crammer & Singer[CRAMMER 01] : Elle est directement issue de la méthode précédente, quelques simplifications ont été effectuées. En effet, on a retiré le biais et les variables d'écart ont été rassemblées de manière à obtenir un dual plus simple.

4. Les critères de validation

L'utilisation d'un modèle de discrimination est motivé par la qualité de celui-ci. Or, cette qualité est estimée par des critères de validation. J'ai donc consacré une partie de mon travail à l'étude de critères qui seraient plus appropriés que le simple pourcentage de bien classés qui n'est plus suffisant.

Le Ki de KXEN : Cette notion développée par l'entreprise $KXEN$ et présentée dans [MARKADE 03], obtient de bons résultats dans le cas binaire.

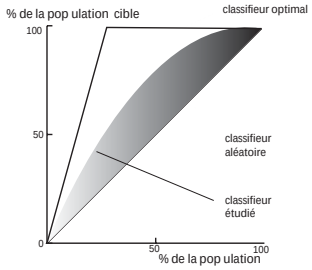


FIG. 2. *Ki : rapport de l'aire de la surface grise et de celle de la surface entre le classifieur aléatoire et le classifieur optimal*

Ce critère a l'avantage de valoir 1 pour un modèle parfait et 0 pour un modèle aléatoire. De plus, il permet de souligner des erreurs même dans le cas d'une distribution des classes très inégale. Malheureusement, le Ki n'est pas directement extensible à plus de deux classes, jusqu'alors, on a donc utilisé la moyenne ou le minimum des Ki pour chaque paire de classes. On aura donc $\frac{k(k-1)}{2}$ Ki à calculer.

Le κ de Cohen : C'est une mesure non paramétrique d'accord entre deux variables qualitatives pour des données appariées. Elle mesure l'écart à la diagonale dans la matrice de confusion. On peut l'écrire :

$$\kappa = \frac{N \sum_{i=1}^{k} z_{ii} - \sum_{i=1}^{k} z_{i.}\, z_{.i}}{N^2 - \sum_{i=1}^{k} z_{i.}\, z_{.i}}$$

où (z_{ij}) est la matrice de confusion associée au classifieur et N le nombre total d'observations.

5. Résultats et comparaisons

J'ai modifié et utilisé des logiciels mis au point par C.J. Lin, C.C. Chang et C.W. Hsu pour tester ces méthodes. [HSU]
J'ai procédé par validation croisée à 10 parts et calculé le pourcentage de bien classés ainsi que la moyenne et le minimum des Ki et le κ. J'ai d'autre part comparé ces méthodes à des discriminations par k plus proches voisins (k-NN) et par arbre de décision (ID3). J'ai utilisé un noyau radial car, par l'expérience, il obtient les meilleurs résultats et des temps de travail très courts. La recherche des paramètres optimaux (le C et l'inverse de la variance dans le noyau) a été faite par un test de chaque combinaison de paramètres possible dans des intervalles choisis. D'autre part, nous n'avons étudié que des jeux de données avec un faible nombre de classes (entre 3 et 5).
Les différentes applications donnent des résultats assez proches, on le voit dans le tableau suivant où apparaissent les κ pour chaque méthode. On obtient un léger avantage pour la méthode "un contre un" sur les trois autres. Par

contre, la comparaison faite avec les deux méthodes non SVM donne un avantage significatif aux méthodes de SVM multi-classes. D'autre part, le temps de travail est aussi une notion essentielle dans ces méthodes. En effet, ces méthodes doivent être rapides car lors de l'optimisation des paramètres, il faut appliquer un grand nombre de fois chaque méthode. Il ressort de même que la méthode "un contre un" est la plus rapide, ceci peut s'expliquer par le fait que nous n'avons traité que des cas avec peu de classes, ce qui fait peu de problèmes bi-classes à résoudre pour cette méthode. D'autre part, lorsque le C augmente, il arrive que la méthode de Crammer & Singer obtienne des temps de convergence extrêmement longs. Ceci peut être expliqué par le fait qu'on a utilisé un algorithme de décomposition du dual qui peut poser des problèmes de convergence.
Les nouveaux critères de validation des modèles ressortent comme plus sensibles que le pourcentage de bien classés. La moyenne des Ki et le κ semblent spécialement adaptés. Le fait qu'ils soient reliés à l'aire sous la courbe ROC (receiver operating characteristic) leur donne des propriétés que le pourcentage de bien classés ne possède pas.

Données	"un contre un"	"un contre le reste"	W. & W.	C. & S.	ID3	k-NN	# classes	# obs.	# var.
dna	**0.93**	0.92	0.89	0.92	0.88	–	3	2000	180
car	**0.97**	0.81	0.97	0.94	0.91	0.91	4	1728	6
wine	0.73	**0.89**	0.66	0.86	0.53	0.74	3	178	13
iris	0.89	**0.94**	0.89	0.88	0.84	0.83	3	120	4

TAB. 1. *κ pour un certain nombre de jeux de données*

6. Conclusions et ouvertures

Toutes ces observations amènent à un certain nombre de remarques. Tout d'abord, les méthodes SVM à plus de deux classes obtiennent de très bons résultats, elles surpassent les méthodes de discrimination classique aussi bien au niveau des résultats qu'au niveau du temps d'exécution. De plus, elles sont très simple d'utilisation et un processus automatisé peut assez facilement être mis en place. La méthodes qui, dans le cas d'un nombre de classes faible, semble la plus adaptée du fait de son efficacité mais aussi de sa simplicité est la méthode "un contre un". D'autre part, les critères de validation étudiés permettent d'éviter des erreurs et il serait intéressant dans bien des cas d'associer au pourcentage de bien classés soit la moyenne des Ki, soit le κ de Cohen.

Bien sûr, des approfondissements seraient possibles, ainsi on pourrait modifier la méthode de vote dans la méthode "un contre un", améliorer les décompositions proposées pour les méthodes traitant toutes les classes simultanément ou tester des jeux de données avec plus de classes (par exemple 26 pour la reconnaissance de caractères).

7. Bibliographie

[CRAMMER 01] CRAMMER K., SINGER Y., On the algorithmic implementation of multiclass kernel-based vector machines, *School of Computer Science and Engineering, Hebrew University*, , 2001.

[FRIEDMAN 96] FRIEDMAN J., Another Approach to Polychotomous Classification, *Stanford University*, , 1996.

[HSU] HSU C., LIN C., CHANG C., Logiciels LIBSVM et BSVM, `URL : www.csie.ntu.edu.tw/~cjlin/`.

[MARKADE 03] MARKADE E., Evaluating Modeling Techniques, KXEN Inc. (Knowledge Extraction Engines), 2003.

[VAPNIK 98] VAPNIK V., *Statistical learning theory*, John Wiley & Sons, Inc., N.Y., 1998.

[WESTON 98] WESTON J., WATKINS T., Multi-class Support Vector Machines, *Technical Report - Royal Holloway University of London*, , 1998.

Un survol des algorithmes génétiques dans la fouille des données

Fatima-Zohra Kettaf[*], **Jean-Pierre Asselin de Beauville**[**]

[*] *IRIT-UPS, UMR 5505, 118 route de Narbonne 31062 Toulouse cedex 04 France*
kettaf@irit.fr
[**] *Laboratoire d'Informatique, Ecole Polytechnique, Université de Tours, 64 avenue Jean Portalis 37200 Tours, Détaché à l'Agence Universitaire de la Francophonie au Canada (Montréal)*
jean-pierre.asselin@auf.org

RÉSUMÉ. *Cet article a pour objectif la présentation des travaux récents dans le domaine de la fouille de données, basés sur l'évolution génétique.*

MOTS-CLÉS : *fouille données, algorithme génétique*

1. Introduction

La fouille des données peut se définir comme étant un ensemble de méthodes permettant d'analyser des données déjà collectées dans de très grandes bases de données (les transactions bancaires, les données biologiques, ...) afin d'extraire des relations ou des structures ayant une sémantique utile pour les utilisateurs. Les Algorithmes génétiques (AG) constituent des outils incontournables dans le processus de fouille des données. On aborde dans cet article leur contribution dans les phases de pré-traitement (réduction de données), de post-traitement et dans la découverte de règles d'association et de prédiction.

2. La réduction de données

Elle englobe deux types de techniques : la sélection de variables (attributs) et la sélection de prototypes (instances) [GUE 98][LUD 98]. Le génome manipulé par l'AG est donc un sous-ensemble de variables ou de prototypes. C'est une chaîne binaire de longueur égale au nombre initial de variables (prototypes), et dans laquelle la présence de la variable (prototype) est représentée par le bit 1 et son absence par le bit 0, les variables comme les prototypes sont supposées indicées. Soit $\{v_1, \ldots, v_{12}\}$ l'ensemble des variables descriptives initiales, un génome ne retenant que les variables v_2, v_7 et v_{12} est représenté par la chaîne : 010000100001. Cette représentation offre l'avantage de pouvoir réutiliser tels quels les opérateurs classiques de croisement et de mutation. Si l'évaluation du génome se fait en dehors des procédés d'apprentissage et de généralisation des classifieurs, l'approche utilisée est désignée comme étant une approche "filter" ; elle procède comme un filtre. Si au contraire, l'évaluation se fait au sein d'un classifieur donné, l'approche est appelée "wrapper". Plus précisément, on fournit les variables retenues par ce génome à un algorithme de classification, par exemple à un arbre de décision, qui les utilisera dans ses phases d'apprentissage et de classement (exemple : Set-GEN [CHE 96]). La qualité des génomes manipulés par Set-GEN est évaluée par le taux de classement moyen de l'arbre construit (calculé par validation croisée).

3. Découverte de règles d'association

L'objectif est double et consiste à extraire des bases de données des concepts locaux ainsi que les règles permettant de les expliquer. Il existe deux approches : Pittsburgh (GABIL)[FRE 02] qui préconise le codage de la connaissance (disjonction de règles, appelée aussi hypothèse) par un seul chromosome, tandis que l'approche Michigan COGIN, REGAL)[FRE 02] associe à chaque règle un chromosome qui la représente. D'une façon plus générale, les règles d'association s'écrivent sous la forme : $\theta \Rightarrow \phi$ où θ est une conjonction de faits. Le codage des règles est souvent binaire et consiste à juxtaposer le code des faits et à le terminer par une sous-chaîne représentant la conclusion. Par exemple la décision de "jouer au tennis" dépend des valeurs prises par trois attributs : le temps, l'humidité, et le vent. Ils sont tous symboliques et prennent respectivement 3, 2 et 2 valeurs possibles. Chaque attribut sera codé par autant de bits que de valeurs possibles. Par exemple la valeur "ensoleillé" de l'attribut temps sera codée par la chaîne 100. On peut coder le fait que le temps prenne soit la valeur "ensoleillé" soit la valeur "pluvieux" par 101. Si aucune contrainte n'est spécifiée pour cet attribut, c'est la chaîne 000 qui sera utilisée. La concaténation des faits traduit la partie condition de la règle. Sa partie conclusion est codée, dans cet exemple, par un seul bit (1 signifie "jouer au tennis" et 0 son contraire). Le génome $100.10.00|0||100.01.00|1||010.00.00|1||001.00.10|0||001.00.01|1$ pourrait correspondre à un ensemble de règles de décision pour l'exemple sus-mentionné. Il constitue une seule solution potentielle dans l'approche Pittsburgh. Le génome $100.10.00|0$ constituerait à lui seul une solution dans l'approche Michigan. Dans les deux approches, le codage de la partie conclusion des règles pose un problème, surtout quand il s'agit de la classification d'instances. Soit la classe est codée explicitement dans la règle et elle est sujette à évolution, soit elle est fixée d'avance et ne change pas. Ce qui nécessite plusieurs exécutions de l'AG ainsi que le choix de la classe particulière que l'on cherche à apprendre. Soit elle n'est pas représentée et on l'estime avec les fréquences calculées à partir des ensembles d'apprentissage et de test. Pour mesurer la qualité des règles on a recours au calcul de fréquences sur les attributs ou combinaisons d'attributs des bases de données. Pour la règle ($\theta \Rightarrow \phi$), on définit $freq(\theta)$ comme étant le nombre de cas où θ est satisfait. $freq(\theta et \phi)$ est appelé support de θ. On définit la confiance d'une règle $c(\theta \Rightarrow \phi)$ comme étant la proportion d'occurrences satisfaisant ϕ parmi celles qui satisfont θ : $\frac{freq(\theta et \phi)}{freq(\theta)}$. En plus du coefficient de confiance, la qualité d'une règle dépend de sa complétude ($Fitness = CF * Comp$ où $Comp = \frac{freq(\theta \text{ et } \phi)}{freq(\theta \text{ et } \phi) + freq(\neg\theta \text{ et } \neg\phi)}$) et de sa simplicité ($Fitness = w_1(CF * Comp) + w_2 * Simpl$ où $Simpl$ est la mesure de simplicité de la règle). Pour éviter les convergences lentes causées par ces mesures, on a introduit un coefficient qui mesure l'utilité des attributs : $F = \frac{w_1 J_1 + w_2(\frac{N_{pu}}{NT})}{w_1 + w_2}$ où N_{pu} est le nombre d'attributs utiles de la partie θ. On appelle "attribut utile", un attribut apparaissant dans la règle avec une valeur donnée et tel qu'au moins une instance dans les données présente la valeur indiquée par la règle pour cet attribut ainsi que la même valeur pour l'attribut but de cette règle. NT est le nombre total d'attributs présents dans la partie condition θ. Des expérimentations ont montré que le rapport $\frac{N_{pu}}{NT}$ ne change pas à partir d'un certain nombre de générations g_n. C'est pour cette raison que Jourdan et *al.* [JOU 02] modifient la mesure F en l'adaptant aux générations de l'AG. S'agissant des opérateurs génétiques, les AG de découverte de règles devront disposer de nouveaux opérateurs génétiques dédiés à la généralisation et à la spécialisation de règles (ou d'hypothèses) [JOU 02][FRE 02].

4. Découverte de règles prédictives et post-traitement

Carvalho *et al.* [CAR 02] proposent une méthode hybride de découverte de règles prédictives. Ils utilisent un AG et un algorithme de construction d'arbre de décision (C4.5). Ils justifient leur choix (hybridation) par le fait que les arbres de décision sont plus appropriées pour la découverte de règles longues et sont moins efficaces dans la recherche de règles courtes. Ils commencent par appliquer l'algorithme C4.5 pour aboutir à un arbre de décision qu'ils élaguent et traduisent en un ensemble de règles propositionnelles. Ils séparent ensuite les règles en deux ensembles disjoints ; un ensemble de règles courtes et un ensemble de règles longues. L'AG a pour tâche la découverte de données provenant du premier groupe et l'algorithme C4.5 le classement des données du deuxième groupe. Bala *et al.* [BAL 95] proposent une hybridation de l'AG avec un autre algorithme de construction d'arbres de décision : ID3 et le baptisent GA-ID3. L'AG recherche dans l'espace des sous-ensembles de variables, le sous-ensemble optimal au regard de la qualité de la discrimination réalisée par ID3 et en se basant sur un ensemble de

données d'apprentissages décrites par ce sous-ensemble de variables. Cette contribution aurait pu faire l'objet de la section 2. En effet, ce travail a un double objectif : la sélection (de type "wrapper") d'attributs pertinents et la construction du meilleur arbre de décision sur les données d'apprentissage. Une approche originale a été proposée dans Papagelis *et al.* [PAP 01] et qui consiste à faire évoluer directement des arbres de décision (contrairement à l'hybridation) grâce à des opérateurs génétiques adaptés aux structures d'arbres.
Pour le post-traitement, on se restreint aux règles d'association et à l'élagage des arbres de décision. Le but étant d'extraire le sous-ensemble de règles le plus performant au regard des critères que l'on se fixe, et qui sont généralement ; la compréhensibilité de la règle et son intérêt. Il existe des critères subjectifs donnés par l'utilisateur et dépendants du domaine. L'utilisateur peut guider le système en lui spécifiant des patrons de règles d'association ainsi qu'une liste ou une combinaison d'attributs qu'il souhaite obtenir dans les règles [KLE 94]. On trouve par ailleurs dans [FRE 99] des exemples de critères objectifs.

5. Conclusion

Les AG s'avèrent très utiles lorsque le problème nécessite une analyse et optimisation multi-critères. Ils sont malheureusement de complexité algorithmique élevée (même si elle est atténuée par la baisse des prix et par l'augmentation des performances matérielles). Ils ont toutefois une bonne capacité à intégrer les connaissances expertes sous forme d'opérateurs génétiques et se prêtent bien à l'hybridation. Une des voies, a notre sens, prometteuse réside dans l'utilisation des AG interactifs pour le choix d'algorithmes de fouille appropriés au problème à résoudre et aux données à traiter.

6. Bibliographie

[BAL 95] BALA J., DE JONG K. D., HUANG J., VAFAIE H., WECHSLER H., Hybrid Learning Using Genetic Algorithms and Decision Trees For Pattern Classification, *14 th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 1995.

[CAR 02] CARVALHO R.-R., FREITAS A.-A., A genetic algorithm with sequential niching for discovering small-disjunct rules, *Genetic and Evolutionary Computation Conference. GECCO 2002*, Morgan Kaufmann, 2002, p. 1035-1042.

[CHE 96] CHERKAUER K. J., SHAVILK J. W., Growing Simpler Decision Trees to Facilitate Knowledge Discovery, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, Portland, 1996, AAAI Press.

[FRE 99] FREITAS A.-A., Rule interestingness Measures, *Knowledge-Based systems*, vol. 12, n^o 5-6, 1999, p. 309-315.

[FRE 02] FREITAS A.-A., A survey of evolutionary algorithms for data mining and knowledge discovery, GHOSH A., TSUTSUI S., Eds., *Advances in Evolutionary Computation*, Springer-Verlag, 2002, p. 819-845.

[GUE 98] GUERRA-SALCEDO C., WHITLEY D., Genetic Search For Feature Subset Selection : A Comparison Between CHC and GENESIS, *Proceedings of the third annual Genetic Programming Conference*, vol. 3, San Mateo, CA : Morgan Kaufman, 1998, p. 797-803.

[JOU 02] JOURDAN L., DHAENENS C., TALBI E.-G., ASGAR : un algorithme génétique pour les règles d'association , *Extraction de Connaissances et Apprentissage*, vol. 0/2002, 2002, p. 100-127.

[KLE 94] KLEMETTINEN M., MANNILA H., RONKAINEN P., TOIVONEN H., VERKAMO A. I., Finding interesting rules from large sets of discovered association rules, *3rd Int. Conf. on Information and Knowledge Management*, Gaithersburg, Maryland, Nov/Dec 1994, ACM Press, p. 401-407.

[LUD 98] LUDMILA I., BEZDEK J. C., Nearest Prototype Classification : Clustering, Genetic Algorithms, or Random Search ?, *IEEE Trans on Systems, man, and Cybernetics-Part C : Applications and Reviews*, vol. 28, n^o 1, 1998, p. 160-164.

[PAP 01] PAPAGELIS A., KALLES D., Breding Decision Trees Using Evolutionary Techniques, *ICML2001*, 2001, p. 393-400.

Proposition de classification simultanée individus-variables fondée sur une notion de SVD de matrice partitionnée.

Lafosse Roger

Lab. de Statistique et Probabilités, Université P. Sabatier, 118 rte de Narbonne, 31062 Toulouse Cedex 4, lafosse@cict.fr

RÉSUMÉ. *Une récente extension de la décomposition en valeurs singulières (SVD) a été proposée dans un contexte analyse factorielle. Elle revient à découper à l'aide de solutions successives les liens particuliers créés par les blocs d'une matrice partitionnée selon les lignes et les colonnes. Etant donnée une matrice individus-variables, on use de cet outil pour choisir des permutations entre lignes ou entre colonnes dans le but de révéler les groupes d'individus s'associant plus particulièrement à certains paquets de variables.*

MOTS-CLÉS : *Classification, Décomposition en valeurs singulières, Matrices ordonnables.*

1. Introduction

Sachant un tableau de données individus $\times$ variables, nous voulons constituer deux groupes d'individus en association avec deux paquets respectifs de variables, chacun des deux paquets devant révéler au mieux l'existence d'un fort lien entre les individus de son groupe. C'est dire que les variables d'un paquet doivent apparaître corrélées pour les individus du groupe associé. On crée ces regroupements en permutant les lignes et les colonnes du tableau. Le nouveau tableau doit alors définir une partition en quatre blocs. Chacun des deux blocs diagonaux définit un groupe d'individus très liés entre eux pour les variables correspondantes. Les deux autres blocs extradiagonaux doivent au contraire être relatifs à des liens très faibles entre individus pour les variables correspondantes. Le géographe Bertin [BER 67] effectuait ce travail à la main, chaque valeur du tableau étant portée sur un carton. Les permutations des lignes ou des colonnes de cartons de la "matrice ordonnable" étaient effectuées à l'aide de tringles. Aujourd'hui ce travail est réalisable à l'aide d'un tableur et aucune méthode automatique ne pourra jamais remplacer l'intelligence de celui qui révèle ainsi les associations simultanées entre lignes et entre colonnes, privilégiant certaines permutations en partie depuis sa compréhension des données. La nécessité de produire cependant ce type d'associations de façon automatique provient de l'existence de jeux de données de taille élevée, par exemple issus de la génétique. On se propose ici de fonder les permutations sur un critère qui détaille la contribution relative de chacun des quatre blocs. Une permutation est faite si elle provoque une augmentation du lien ligne-colonne révélé par les blocs diagonaux, supérieure à celle du lien révélé par les blocs extra-diagonaux.

La décomposition en valeurs singulières (SVD) d'une matrice permet d'analyser le lien entre les lignes et les colonnes. Dans un contexte analyse factorielle, [LAF 97] ont proposé une extension de la SVD d'une matrice, à N matrices ayant toutes le même nombre p de lignes, pouvant correspondre à des applications linéaires d'un même sous-espace de R^p engendré par toutes les colonnes, dans N sous-espaces de dimensions différentes q_i, $i = 1, ...N$, engendrés respectivement par les lignes des N matrices. La première solution de cette analyse, associant un vecteur de R^p à N vecteurs des espaces respectifs R^{q_i}, décompose le lien global entre toutes les lignes et colonnes et permet de préciser la contribution relative de chacune des N matrices à ce lien.

La démarche précédente a été prolongée dans une analyse nommée concorGM [KIS 04]. Cette fois M sous-espaces sont associés à N sous-espaces depuis la donnée de $M \times N$ matrices. Une première solution calculée

revient à définir M vecteurs et N vecteurs qui résument l'association multiple. En considérant que les $M \times N$ matrices correspondent aux $M \times N$ blocs d'une matrice individus $\times$ variables, suite à une partition des variables en N paquets et des individus en M groupes, le critère utilisé fournit une évaluation de la contribution relative de chacun des $M \times N$ blocs au lien global entre lignes et colonnes.

Dans la section suivante on rappelle ces deux extensions successives de la SVD. Dans la section 3 on indique la règle de décision d'une permutation, dans des conditions très simplifiées.

2. Deux extensions successives de la notion de SVD

2.1. *SVD d'une matrice*

On rappelle ici quelques propriétés du premier couple de vecteurs singuliers de la SVD usuelle d'une matrice. Soit A, une matrice $p \times q$, considérée comme celle d'une application linéaire entre les espaces métriques (R^p, I_p) et (R^q, I_q), plus précisément entre le sous-espace engendré par les colonnes de A et celui engendré par les lignes. Quand on écrit, pour des vecteurs normés $u \epsilon R^p$ et $v \epsilon R^q$,

$$(I_p - uu')Av = 0,$$

cela signifie que v est relié à u par A puisqu'alors et de façon équivalente, avec $s \epsilon R$,

$$Av = su. \qquad [1]$$

Mais cela signifie aussi que le lien de v avec le sous-espace orthogonal à u est nul. Le vecteur u apparaît donc comme isolé dans sa relation par A avec v, le vecteur v n'étant relié qu'à u dans le sens indiqué. Un couple (u, v) de $R^p \times R^q$ est dit couple de vecteurs singuliers s'il vérifie à la fois (1) et

$$A'su = s^2 v. \qquad [2]$$

Celui qui est associé à la plus grande valeur possible de s^2 est dit premier couple singulier. Ce couple solution (u, v) correspond à la maximisation sous contraintes de norme du critère

$$f(u, v) = (u'Av)^2, \qquad [3]$$

l'optimum valant s^2.

Les autres couples singuliers, associés à des valeurs singulières plus faibles, sont à rechercher avec ce critère en se plaçant dans l'orthogonal de u et dans l'orthogonal de v.

2.2. *SVD d'une matrice partitionnée*

On considère maintenant N matrices $A_h,\ p \times q_h,\ h = 1, ..., N$, qui correspondent à N applications linéaires entre un espace métrique R^p et des espaces métriques R^{q_h}.

On note $A = [A_1\ A_2\ \cdots\ A_N]$ la matrice $p \times q$ obtenue par concaténation des matrices A_h. On peut dire aussi que A est partitionnée par les blocs A_h. Un (N+1)-uple de $N+1$ vecteurs normés $(u,\ v_1, v_2, ..., v_N)$ est dit premier (N+1)-uple singulier de la partition de A s'il vérifie les $N + 1$ égalités

$$A_h'u = s_h v_h\ \forall h, \qquad [4]$$

$$A \begin{bmatrix} s_1 v_1 \\ s_2 v_2 \\ \vdots \\ s_N v_N \end{bmatrix} = s^2 u, \qquad [5]$$

avec $s^2 = \sum s_h^2$.

Rechercher une première solution en maximisant $\sum s_h^2$, revient à considérer sous $N+1$ contraintes de norme respectives la maximisation du critère

$$f(u,\ v_1, v_2, ..., v_N) = \sum_{h=1}^{N} (u' A_h v_h)^2. \qquad [6]$$

La solution a été apportée par [LAF 97] dans un contexte analyse factorielle nommée analyse Concor. On note (u, v) le premier couple singulier de A associée à la plus grande valeur singulière s, et b_h les N vecteurs-bloc de v ayant pour dimensions respectives q_h. Une première solution globale est obtenue pour u et $v_h = \frac{b_h}{|b_h|},\ \forall h$, chaque terme de la somme (6) vérifiant

$$(u' A_h v_h)^2 = s^2\ |b_h|^2\ ,\ \forall h.$$

Le vecteur u est alors aussi le premier vecteur singulier à gauche de la matrice $p \times N$

$$[A_1 v_1\ A_2 v_2\ ...\ A_N v_N]. \qquad [7]$$

En référence à ce qui est indiqué en section 2.1, on peut remarquer que les autres (N+1)-uples singuliers, associés à des valeurs singulières plus faibles, ne peuvent être recherchés depuis le critère (6) qu'en se plaçant dans l'orthogonal de u et,$\forall h$, dans l'orthogonal de v_h.

2.3. *SVD d'une matrice bipartitionnée*

La présente définition est une extension de la précédente et a été introduite par [KIS 04] dans un contexte analyse factorielle nommée concorGM. L'association multiple de N espaces métriques R^{q_h} avec M espaces métriques R^{p_k} correspond à la donnée de $N \times M$ matrices $A_{kh},\ k = 1, ..., M,\ h = 1, ..., N$. Ces matrices constituent en fait les sous-blocs d'une matrice A partitionnée selon les lignes et selon les colonnes (A est bipartitionnée).

Pour k fixé, on note $\left\{A_{(k)}\right\}$ la famille des N matrices $A_{kh},\ h = 1, ..., N$, ayant donc une dimension commune, représentant un bloc ligne de A. En référence à la section 2.2, on pourrait définir M premiers (N+1)-uples singuliers respectivement associés aux M blocs ligne $\left\{A_{(k)}\right\},\ k = 1, ..., M$. On définirait ainsi en particulier M respectifs N-uple de vecteurs normés $(v_1, v_2, ..., v_N)$, conçus dans des espaces respectifs de même dimensions, mais tous différents. En fait ici on veut en définir un seul, ce N-uple $(v_1, v_2, ..., v_N)$ commun aux M blocs ligne devant constituer un compromis des N-uples qui auraient pu être calculés. Un raisonnement analogue est tenu, après échange des indices h et k. Finalement, un ($M + N$)-uple $(u_1, u_2, ..., u_M,\ v_1, v_2, ..., v_N)$ est dit ici premier $(M+N)$-uple singulier de la bipartition de A, s'il est solution du critère à maximiser sous $M + N$ contraintes de norme

$$f(u_1, u_2, ..., u_M,\ v_1, v_2, ..., v_N) = \sum_{k=1}^{M} \sum_{h=1}^{N} (u_k' A_{kh} v_h)^2. \qquad [8]$$

Chaque vecteur solution u_k est alors premier vecteur singulier à gauche de la matrice $p_k \times N$

$$[A_{k1} v_1\ A_{k2} v_2\ ...\ A_{kN} v_N], \qquad [9]$$

alors même que chaque vecteur solution v_h est premier vecteur singulier à gauche de la matrice $q_h \times M$

$$[A_{1h}' u_1\ A_{2h}' u_2\ ...\ A_{Mh}' u_M]. \qquad [10]$$

Les solutions successives sont recherchées en se plaçant dans les $M + N$ sous-espaces orthogonaux respectifs.

3. Une étude de la méthode par simulations

On a travaillé avec des jeux de données très simplifiés, en prenant $M = 2$ et $N = 2$. On génère au hasard un jeu de données gaussiennes Y_{11}, $n_1 \times p_1$ de p_1 variables centrées, à partir d'une matrice de corrélations fixée sans corrélation négligeable. Un autre jeu de données Y_{22}, $n_2 \times p_2$, est généré de la même façon à partir d'une autre matrice de corrélation. Puis on génère deux matrices Y_{12} et Y_{21} de dimensions respectives $n_1 \times p_2$ et $n_2 \times p_1$ associées à des variables gaussiennes indépendantes de variance bien inférieure à 1. Un tableau Y (centré) est ainsi formé de $p = p_1 + p_2$ variables et $n = n_1 + n_2$ individus

$$Y = \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix}$$

Il est associé à une partition qui est à retrouver par la méthode. En effet, on génère maintenant au hasard des permutations des lignes et des colonnes de Y, de sorte que tout soit désordonné dans la matrice X alors obtenue. C'est cette matrice X qui constitue le jeu de données sur lequel s'applique la méthode.

Supposons la matrice X divisée en quatre blocs ($n = m_1 + m_2$ et $p = q_1 + q_2$)

$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix}$$

On calcule alors le critère concorGM associé à la première solution

$$(u_1' X_{11} v_1)^2 + (u_1' X_{12} v_2)^2 + (u_2' X_{21} v_1)^2 + (u_2' X_{22} v_2)^2.$$

Chacun des quatre termes mesure l'importance relative de chacun des quatre blocs quant au lien entre ligne et colonne de X. On effectue alors une permutation de X, réalisant un échange entre une des m_1 lignes et une des m_2 lignes, ou entre une des q_1 colonnes et une des q_2 colonnes. On recalcule alors la valeur du critère, la taille des blocs n'ayant pas changé

$$(u_1^{p'} X_{11}^p v_1^p)^2 + (u_1^{p'} X_{12}^p v_2^p)^2 + (u_2^{p'} X_{21}^p v_1^p)^2 + (u_2^{p'} X_{22}^p v_2^p)^2.$$

La permutation est acceptée si l'augmentation de la mesure du lien sur les blocs diagonaux est supérieure à l'augmentation (éventuelle) de la mesure du lien sur les blocs extra-diagonaux, c'est à dire si la quantité

$$(u_1^{p'} X_{11}^p v_1^p)^2 + (u_1^{p'} X_{12}^p v_2^p)^2 - (u_2^{p'} X_{21}^p v_1^p)^2 - (u_2^{p'} X_{22}^p v_2^p)^2$$

est supérieure à

$$(u_1' X_{11} v_1)^2 + (u_1' X_{12} v_2)^2 - (u_2' X_{21} v_1)^2 - (u_2' X_{22} v_2)^2.$$

Pour les valeurs de m_1, m_2, q_1 et q_2 fixées cette règle donne satisfaction. Mais on ne veut pas envisager toutes les permutations possibles pour chaque hypothèse de division en quatre. La règle devrait pouvoir encore fonctionner quand on passe d'une hypothèse de division en quatre à une autre. La solution obtenue dépend du choix de la première hypothèse de division en quatre sous-blocs considérée et des premières permutations tentées. Se contenter de la première solution de concorGM pour construire la règle peut être insuffisant. Quelques pistes sont alors suggérées, plutôt qu'une indication précise sur une mise en oeuvre efficace et générale de la méthode.

4. Bibliographie

[BER 67] Bertin, J., Sémiologie Graphique. Les diagrammes, les réseaux, les cartes. *http//www.sciences-po.fr/cartographie/*, Gauthiers-Villars, 1967.

[KIS 04] Kissita, G. and Cazes, P., Hanafi, M, Lafosse, R. Deux méthodes d'analyses factorielles du lien entre deux tableaux de variables partitionnés. *Revue de Statistique Appliquée* vol. 54, 3, 2004.

[LAF 97] Lafosse, R. and Hanafi, M. Concordance d'un tableau avec K tableaux : définition de K+1 uples synthétiques. *Revue de Statistique Appliquée* vol. 45, 4, 1997, p. 111-126.

Détection de phénomènes d'irritation cutanée et de bronzage

Julie Latreille*, Sophie Dheurle*, Christiane Guinot*, Laurence Ambroisine*, Emmanuelle Mauger*, Michel Tenenhaus, Sabine Guéhenneux*, Frédérique Morizot* & Erwin Tschachler*,*****

** CE.R.I.E.S., 20 rue Victor Noir, 92521 Neuilly sur Seine, France, julie.latreille@ceries-lab.com*
*** Département SIAD, HEC School of Management-Paris, Jouy en Josas, France*
**** Département de Dermatologie,Université de Vienne,Vienne, Autriche*

RÉSUMÉ. L'objectif de ce travail était d'étudier la capacité de spectrophotomètres Minolta® (CM503i et CM2600d) à détecter d'une part une rougeur induite par une faible concentration d'irritant (méthyle nicotinate à 2,5 mM), et d'autre part un bronzage induit par une irradiation aux rayons ultraviolets (UV) à dose infra-érythémale en tenant compte de la couleur de la peau des individus mesurée par l'angle typologique individuel (ITA). L'analyse discriminante PLS (PLS-DA) et deux méthodes de classification (K-means et CART) ont été utilisées afin d'en comparer les résultats. La rougeur a été détectée uniquement chez des sujets ayant une peau claire à très claire définie par ITA ≥ 40°. De même, le bronzage a pu être détecté uniquement chez les individus présentant une peau claire à foncée définie par ITA < 50°.

MOTS-CLÉS : Analyse discriminante PLS, CART, couleur de la peau, K-means, spectrocolorimétrie

1. Introduction

La spectrocolorimétrie est une technique non invasive dont les applications en dermatologie sont nombreuses : aide au diagnostic de lésions cutanées (détection de mélanomes, détection d'altération de la peau due à des maladies…), information sur la physiologie de la peau (hydratation de la peau, contenu en pigments...), ou encore étude de la réponse de la peau à différents stimuli. La spectrocolorimétrie a donc été proposée pour estimer de manière non invasive la couleur de la peau dont les principaux pigments sont l'hémoglobine et la mélanine [YOU 97, ZON 01, SOW 01, WAG 02, SHI 01, ANG 01]. Deux études ont été menées afin de répondre à chaque objectif : tester la capacité des mesures spectrales à détecter une rougeur (c'est-à-dire une augmentation du volume de sang contenu dans la peau, et donc de l'hémoglobine, induite par une faible concentration d'une substance irritante) et tester la capacité des mesures spectrales à détecter un bronzage (c'est-à-dire une augmentation du contenu en mélanine induite par une irradiation aux UV à dose infra-érythémale (sans rougeur visible à l'oeil)). Compte tenu des contraintes de place, seule l'étude sur la détection d'une rougeur est présentée ici, la méthodologie employée est strictement la même dans l'étude sur la détection d'un bronzage.

2. Objectif : détection d'une rougeur

L'objectif de cette étude est d'étudier la détection d'une rougeur cutanée (érythème dû à une augmentation du contenu en hémoglobine) induite par l'application d'une concentration faible d'un irritant (méthyle nicotinate à 2,5 mM), en tenant compte de la couleur de base de la peau mesurée par l'angle typologique individuel [CHA 91].

2.1. *Matériel et méthodes*

Une étude a été réalisée sur 27 femmes caucasiennes, âgées de 20 à 30 ans, vivant en Ile de France et de phototype I à IV [CES 77]. Le spectrocolorimètre CM503i de Minolta® a été utilisé pour mesurer la couleur de

la peau sur deux zones adjacentes de la face interne de l'avant-bras après application d'eau (témoin) et d'une solution aqueuse de méthyle nicotinate à une concentration de 2,5 mM (irritant).

Le spectrophotomètre utilisé dans l'étude fournit une mesure de la couleur de la peau via la courbe du spectre visible de la lumière réfléchie par la peau (pourcentage de réflectance de la lumière aux longueurs d'onde λ, λ allant de 400 et 700 nm, avec un intervalle de 10 nm) et les paramètres de chromamétrie définis dans l'espace CIELab [ROB 76] : luminosité (L*), intensité de rouge (a*) et intensité de jaune (b*).

L'angle typologique individuel (ITA=arctan(L*-50/b*)x180/π, [CHA 91]) qui permet d'estimer la dose érythémale minimale (DEM), c'est-à-dire la dose minimale d'irradiation provoquant une rougeur visible à l'œil et donc la sensibilité présumée aux UV, a été calculé pour chacun des sujets.

2.1.1 Discrimination des observations

Chaque mesure spectrale est exprimée par 31 pourcentages de réflectance de la lumière à différentes longueurs d'onde, ces pourcentages de réflectance étant fortement colinéaires. Pour essayer de séparer les observations « irritant » de celles « témoin » différentes méthodes ont été utilisées, tout d'abord l'analyse discriminante PLS puis deux méthodes de classification : la méthode K-means (algorithme de type nuées dynamiques) et la méthode CART (arbre de décision).

2.1.1.1 PLS-DA

Une analyse discriminante PLS (PLS-DA) a été réalisée pour tester si les pourcentages de réflectance de la lumière à certaines longueurs d'ondes λ permettent de discriminer les observations après application de l'irritant et du témoin (logiciel SIMCA® [UME 02]). PLS-DA consiste à faire une régression PLS classique qui permet de modéliser la liaison entre un bloc de variables Y, à expliquer, et un bloc de variables explicatives X. Les variables Y correspondent ici aux deux variables indicatrices décrivant l'appartenance des observations aux classes « témoin » et « irritant », et les variables X aux pourcentages de réflectance de la lumière aux 31 longueurs d'ondes. La régression PLS consiste à chercher des composantes PLS (t_h) qui soient à la fois explicatives et descriptives des variables du bloc X et explicatives des variables du bloc Y. Les composantes PLS (t_h) doivent satisfaire simultanément trois conditions : être le plus fortement possible corrélées avec les Y, restituer le plus possible la variance des X, et ne pas être corrélées entre elles.

2.1.1.2 Méthodes K-means et CART

La méthode K-means a permis d'étudier si les observations peuvent être regroupées de manière non supervisée dans les catégories « irritant » et « témoin » à partir des mesures spectrales. Cette méthode a été réalisée à l'aide du logiciel SAS® version 8.2 [SAS 99] (Proc FASTCLUS, MAXCLUSTERS = 2). La méthode CART (arbre de décision) a permis de rechercher les éventuelles pourcentages de réflectance permettant d'identifier les classes. L'arbre de décision a été construit à l'aide du logiciel Answertree® version 3.1. La méthode CART s'appuie sur la minimalisation des mesures d'impureté. La mesure d'impureté qui a été choisie est l'indice de Gini [BRE 84, ZHA 98].

2.2. *Résultats*

2.2.1 Discrimination des observations

2.2.1.1 PLS-DA

Une première analyse discriminante PLS a été réalisée sur l'ensemble des pourcentages de réflectance aux 31 longueurs d'onde. Quatre composantes PLS significatives ont été trouvées. Afin d'améliorer ce modèle, les pourcentages de réflectance montrant un faible pouvoir explicatif ont été éliminés du modèle. Finalement, trois variables explicatives ont été conservées dans le modèle avec deux composantes PLS significatives. La part de variance des variables Y expliquée par les composantes PLS significatives est de 50,6% et la part de variance des variables X expliquée par ces mêmes composantes est de 99,7%. La figure 1.a décrit la relation globale entre les variables à expliquer « témoin » et « irritant » et les trois variables explicatives « pourcentage de réflectance aux longueurs d'onde 540 nm, 570 nm et 580 nm ». Les pourcentages de réflectance « irritant » (•) sont plus faibles

aux longueurs d'onde 540 nm, 570 nm et 580 nm que ceux « témoin » (o). La figure 1.b présente les observations « irritant » (symboles noirs) et « témoin » (symboles blancs) sur le plan des deux composantes PLS significatives t1 et t2. La forme des symboles (rond, carré, losange et triangle) indique la couleur de peau de base des individus : les triangles correspondent aux observations des femmes du groupe 1, les carrés aux observations des femmes du groupe 2, les losanges aux observations des femmes du groupe 3 ; et les ronds aux observations des femmes du groupe 4. Pour les trois premiers groupes, les observations « irritant » et « témoin » sont relativement bien séparées contrairement au quatrième groupe (les ronds) dont la couleur de peau est la moins claire. Le pouvoir discriminant du modèle (R^2) est satisfaisant et indique que les pourcentages de réflectance aux longueurs d'onde 540 nm, 570 nm et 580 nm permettent effectivement de détecter une rougeur consécutive à une application de méthyle nicotinate. Toutefois, les observations des individus du groupe 4 après application de l'irritant et du témoin ne sont pas séparées visuellement sur la carte.

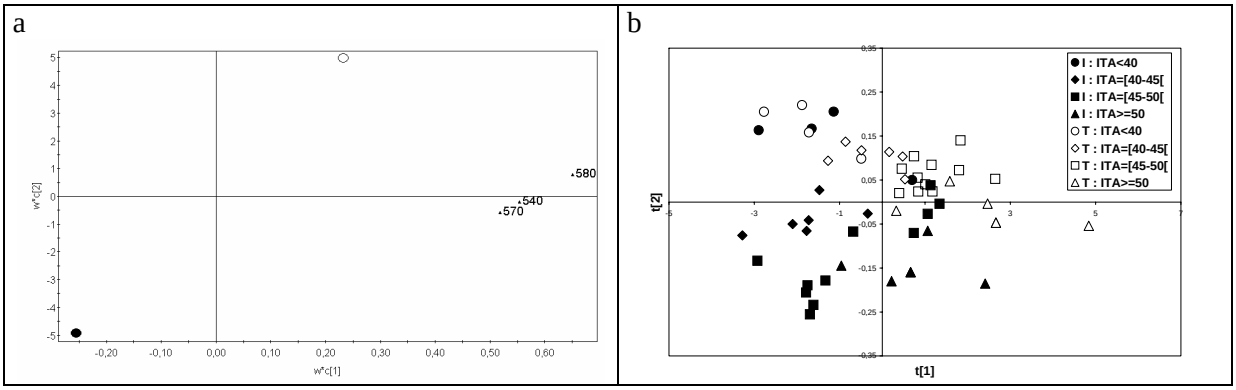


FIG 1 *a) Carte des vecteurs (w1* c1) et (w2* c2), témoin (o), irritant (•) ;*
*b) Carte des composantes (t1*t2), témoin (T), irritant (I)*

2.2.1.2 Méthodes K-means et CART

La méthode K-means n'a pas permis de classer les observations selon la catégorie « irritant » et « témoin ». En fait, les observations ont été regroupées selon la valeur de l'angle typologique individuel. La méthode CART a permis de construire un arbre de décision. Trois pourcentages de réflectance R580, R540 et R560 permettent de construire des règles de décision affectant les individus dans les classes « irritant » et « témoin » avec un risque estimé à 19%. La matrice de confusion est présentée tableau 1.

		Classe observée		
		Irritant	Témoin	Total
Classe prédite	Irritant	19	2	21
	Témoin	8	25	33
	Total	27	27	54

TAB 1 –*Matrice de confusion (effectif).*

3. Conclusion

Dans l'étude sur la détection d'une rougeur à l'aide de la PLS-DA, des variations entre les pourcentages de réflectance des observations « témoin » et « irritant » ont été mises en évidence aux longueurs d'onde correspondant aux pics d'absorption de l'oxyhémoglobine et de l'hémoglobine. Les courbes de réflectance de la peau humaine à ces longueurs d'onde forment un motif en W amplement rapporté dans la littérature [SOW 01, WAG 02, ZON 2001, SHI 01, ANG 01]. Ces variations sont liées à une modification de la couleur de la peau due à une augmentation de l'oxyhémoglobine et de l'hémoglobine en réponse à l'application du produit irritant. D'autre part la méthode CART a permis de construire des règles de décision séparant les observations « irritant » de celles « témoin » à partir des longueurs d'onde correspondant à ces mêmes pics d'absorption de l'oxyhémoglobine et de l'hémoglobine. Dans l'étude sur la détection de bronzage à l'aide de la PLS-DA, la présence d'un érythème infra-clinique (non visible à l'œil) a été identifiée. Par ailleurs, des variations en fin du spectre visible ont également été identifiées, cette région ayant été décrite dans la littérature comme étant celle

permettant d'estimer au mieux la mélanine dans le visible [WAG 02]. La spectrocolorimétrie nous a permis de détecter une rougeur induite par une faible concentration de méthyle nicotinate (2,5 mM) uniquement chez des sujets à peau claire à très claire, c'est-à-dire présentant un angle typologique individuel supérieur à 40°. L'hypothèse est que chez les sujets à peau plus foncée, le niveau en mélanine présent dans la peau pourrait masquer l'absorption de l'hémoglobine [SOW 02, ZON 01]. De même, la spectrocolorimétrie a permis de détecter une augmentation de mélanine consécutive à une série d'irradiations aux UV à dose infra-érythémale uniquement chez des individus présentant une peau claire à foncée, c'est-à-dire en dessous d'un angle typologique individuel de 50°. Ceci s'explique par le fait que chez les sujets à peaux les plus claires, la stimulation aux UV ne peut produire qu'un faible bronzage que notre étude n'a pas réussi à détecter.

4. Bibliographie

[ANG 01] ANGELOPOULOU E., MOLANA R., DANIILIDIS K., *Multispectral skin color modeling*. IEEE conference on computer vision and pattern recognition, 2001, p. 635-642, IEEE computer society press.

[BRE 84] BREIMAN L., FRIEDMAN J., OLSHEN R., STONE C., *Classification and regression trees*, Chapman & Hall, 1984.

[CES 77] CESARINI J.P., " Soleil et peau ". *J Med Esth*, 4:5-12, 1977.

[CHA 91] CHARDON A., CRETOIS I., HOURSEAU C., " Skin colour typology and suntanning pathways ", *Int J Cosm Sci*, 13:191-208, 1991.

[ROB 76] ROBERTSON A-R., " The CIE 1976 color difference formulas ", *Color Res Appl*, 2:7-11, 1976.

[SAS 99] SAS INSTITUTE INC., *SAS/STAT® User's Guide, Version 8*, SAS Institute Inc., Cary, NC, 1999.

[SHI 01] SHIMADA M., YAMADA Y., ITOH M., YATAGAI T., " Melanin and blood concentration in human skin studied by multiple regression analysis: experiments ", *Phys Med Biol*, 46:2385-2395, 2001.

[SOW 02] SOWA M.G., MATAS A., SCHATTKA B.J., MANTSCH H.H., " Spectroscopic assessment of cutaneous hemodynamics in the presence of high epidermal melanin concentration ", *Clinica Chimica Acta*, 317:203-212, 2002.

[UME 02] UMETRICS, *SIMCA-P and SIMCA-P+ 10 User Guide*. UMETRICS AB, Umeå, Suède, 2002.

[WAG 02] WAGNER J.K., JOVEL C., NORTON H.L., PARRA E.J., SHRIVER M.D., " Comparing quantitative measures of erythema, pigmentation and skin response using reflectometry ", *Pigment Cell Res*, 15:379-384, 2002.

[YOU 97] YOUNG A.R., " Chromophores in human skin ". *Phys Med Biol*, 42:789-802, 1997.

[ZHA 98] ZHANG H., CROWLEY J., SOX H.C., OLSHEN R.A., *Tree-structured statistical methods*. Dans : Armitage P, Colton T, editors. Encyclopedia of Biostatistics. Chichester : John Wiley & Sons, 4561-4573, 1998.

[ZON 01] ZONIOS G., BYKOWSKI J., KOLLIAS N., " Skin melanin, hemoglobin, and light scattering properties can be quantitavely assessed in vivo using diffuse reflectance spectroscopy ". *J Invest Dermatol*, 117:1452-1457, 2001.

Analyse de contiguïté et cartes de Kohonen

Ludovic Lebart

CNRS, ENST

46 rue Barrault,

75013, Paris

RÉSUMÉ. Les cartes de Kohonen (Self Organizing Maps) permettent de représenter les classes d' une partition selon une grille (rectangulaire, octogonale) qui donne une idée des proximités existant entre les classes. L'analyse de contiguïté réalisée à partir des mêmes variables et prenant en compte un graphe dérivé de la grille précédente fournit un opérateur projection optimal sur le plan de la grille. Lorsque les éléments à classer sont des variables (cas des mots en analyse textuelle ou en sémiométrie), cet opérateur permet projeter des réplications bootstrap du tableau de données et de tracer des ellipses de confiance pour les éléments à classer.

MOTS-CLÉS : Analyse de contiguïté, Cartes de Kohonen, Zones de confiance, Bootstrap.

1. Introduction

Pour construire une représentation visuelle d'une partition de *n* objets décrits par *p* variables, il existe trois approches possibles:

- Construire simultanément la partition et la représentation, ce qui induit des contraintes sur la partition, mais peut conduire à une représentation plus claire. Exemple : cartes auto-organisées de Kohonen [KOH 89], [COT 97], [THI 97].

- Construire la partition en s'efforçant d'optimiser un critère, puis, dans un second temps, représenter les classes dans un graphique plan. Exemple : Partition classique (k-means, nuées dynamiques, ou classification mixte : classification hiérarchique, optimisation de la coupure du dendrogramme par réaffectation du type k-means), puis représentation des classes (par leurs centres, et/ou leurs enveloppes convexes, et/ou des ellipses de densité) dans le plan (1, 2) d'une analyse en axes principaux du tableau (n, p), ou, mieux, dans le plan (1, 2) d'une analyse discriminante de la partition.

- Une variante de l'approche précédente consiste à projeter les classes (ou leurs enveloppes convexes) dans le plan (1,2) d'une analyse de contiguïté faite à partir d'un k-graphe des k plus proches voisins symétrisé [LEB 00].

- Si dans la dernière approche, on prend comme graphe entre observations un graphe induit par une carte de Kohonen, on obtient le meilleur opérateur projection sur un plan compatible avec une telle structure. Tous les calculs possibles dans le cas des axes principaux, et en particulier les calculs de zones de confiance bootstrap (lorsqu'il s'agit de classification de variables) deviennent alors possibles.

2. Graphe de contiguïté induit par une grille

2.1. *Rappels sur la contiguïté*

Soient *n* objets décrits par *p* variables, conduisant à une *(n, p)* matrice **X**. Les *n* objets seront aussi les sommets d'un graphe symétrique ***G*** dont la matrice *(n, n)* associée est **M** ($m_{ii'} = 1$ si les sommets *i* et *i'* sont joints par une arête, $m_{ii'} = 0$ sinon).

y étant une variable aléatoire prenant ses valeurs sur chaque sommet *i* de ***G***, de variance *v(y)*, la variance locale sera définie comme:

$$v^*(y) = (1/n)\sum (y_i - m_i^*)^2$$

Dans cette dernière formule, la *moyenne locale* est définie comme :

$$m_i^* = (1/n_i)\sum_k m_{ik} y_k$$

C'est la moyenne des valeurs adjacentes au sommet *i*.

Le coefficient de contiguïté *c(y)* , s'écrit : $c(y) = v^*(y) / v(y)$. Une valeur du coefficient $c(y) << 1$ indique une autocorrélation spatiale positive pour la variable *y*.

On note par **N** la *(n,n)* matrice diagonale ayant le degré de chaque sommet *i* comme élément diagonal n_i (n_i dénote ici n_{ii}). **y** est le vecteur dont la $i^{ème}$ composante est y_i.

Le coefficient *c(y)* s'écrit alors : $c(y) = \mathbf{y}'(\mathbf{I} - \mathbf{N}^{-1}\mathbf{M})'(\mathbf{I} - \mathbf{N}^{-1}\mathbf{M})\mathbf{y} / \mathbf{y}'\mathbf{y}$.

Le spectre de la matrice : **N**-**M** (*matrice Laplacienne* du graphe, [MOH 91], [CHU 97]) a d'importantes propriétés relatives à la structure du graphe, l'ordre de multiplicité de la valeur propre nulle (valeur propre 1 en AC de **M**) étant le nombre de composantes connexes du graphe. Le rapprochement avec l'opérateur de Laplace est déjà dans [BEN 73].

2.2. *Généralisation à des observations multivariées sur un graphe .*

Si **X** désigne la *(n,p)* matrice donnant les valeurs de *p* variables pour chacun des *n* sommets du graphe, décrit par sa matrice associée **M**, la matrice des covariances locales s'écrit [LEB 69] :

$$\mathbf{V}^* = (1/n)\,\mathbf{X}'(\mathbf{I} - \mathbf{N}^{-1}\mathbf{M})'(\mathbf{I} - \mathbf{N}^{-1}\mathbf{M})\mathbf{X}$$

Soit **u** un vecteur définissant une combinaison linéaire *u(i)* des *p* variables pour le sommet *i*:

Avec les notations précédentes, la variance locale de la variable *u(i)* vaut :

$$v^*(u) = \mathbf{u}'\mathbf{V}^*\mathbf{u}.$$

Le coefficient de contiguïté de cette combinaison linéaire s'écrit :

$$c(u) = \mathbf{u}'\mathbf{V}^*\mathbf{u} / \mathbf{u}'\mathbf{V}\mathbf{u},$$

où **V** est la matrice des covariances classique. La recherche de **u** qui minimise *c(u)* donne des fonctions de contiguïté minimale, dont les fonctions discriminantes de Fisher constituent un cas particulier lorsque le graphe est formé de plusieurs sous-graphes complets. C'est l'Analyse de Contiguïté.

2.3. *Graphes de contiguïté de Kohonen*

Il s'agit bien d'un graphe sur les *n* observations, qui sont par ailleurs regroupées en q^2 classes (cas d'une grille carrée (q,q)). En supposant que les individus sont rangés selon les q « lignes » de la grille mises bout à bout, la matrice symétrique (*n, n*) **M** associée au graphe sera formée de q^4 blocs (q^2 classes en ligne et les mêmes q^2 classes en colonnes). Un bloc ne contiendra que des 1, ou que des 0.

Les blocs diagonaux et les blocs correspondant à des paires de classes adjacentes sur la grille ne contiendront que des 1. Les autres blocs ne contiendront que des 0.

Les deux premiers vecteurs propres de l'analyse de contiguïté de **X** vis-à-vis du graphe associé à **M** constituent un opérateur projection optimal sur un sous-espace qui respecte autant que faire se peut la structure de la grille de Kohonen. Si l'on ne garde que les éléments diagonaux de **M**, on trouve l'analyse discriminante de Fisher, qui permet de représenter au mieux la partition dans un plan, mais sans la contrainte de proximités entre le classes.

Notons que la grille pourra être déformée, et ce d'autant plus que les contraintes impliquées dans l'algorithme de Kohonen sont en conflit avec la structure des données.

3. Zone de confiance bootstrap

Les visualisations provenant des analyses en axes principaux [décomposition aux valeurs singulières (SVD), composantes principales (ACP), correspondances (AC), Analyses de Contiguïté] n'ont de sens que si elles sont accompagnées de la confiance que l'on peut accorder à la position de chaque point. Or ces visualisations sont d'une grande complexité analytique, et il est exclu de procéder à des évaluations selon les méthodes de la statistique classique. La technique de *bootstrap* [EFR 93] va permettre de tracer des zones de confiance (ellipses ou enveloppes convexes de réplications) autour des points représentés sur les plans principaux.

Dans le cas des composantes principales, on relève plusieurs variantes de la méthode : le *bootstrap total* consiste à refaire une analyse complète pour chaque réplication. Cette procédure engendre une difficulté : les axes répliqués ne sont pas forcément homologues d'un réplication à une autre, il peut y avoir des interversions d'axes, voir des rotations. Il faut alors faire coïncider par des techniques d'*analyses procustéennes* les axes homologues [MAR 94] [MIL 95]. Le *bootstrap partiel* permet de lever cette difficulté. Il part de la constatation que le tableau initial est plus proche de la réalité observée que tous les tableaux répliqués, qui en sont des perturbations [CHA 96]. L'analyse et les plans principaux de ce tableau initial servent de référence pour la projection de tous les tableaux répliqués (lignes et colonnes) en tant qu'éléments supplémentaires. Des expériences [LEB 03] ont montré l'efficacité de cette méthode.

Lorsque la classification s'applique à des variables, ce qui est fréquent en fouille de texte, l'analyse de contiguïté permet de projeter facilement les réplications sur la carte de Kohonen reconstituée (qui devient un continuum) et donc de construire les zones de confiance pour les points-variables à partir d'enveloppes convexes ou d'ellipses d'inertie des réplications.

Les exemples présentés concerneront les différentes options possibles, pour un même jeu de données : Zones (ellipses ou enveloppes convexes de réplications bootstrap) des points variables dans le plan principal d'une analyse en composantes principales, zones dans le plan principal d'une analyse de contiguïté faite à partir du graphe des plus proches voisins, carte de Kohonen, zones dans le plan principal d'une analyse discriminante classique faite sur la partition de Kohonen, zones dans le plan d'une analyse de contiguïté induite par le graphe de la carte de Kohonen qui est la méthode présentée ici.

4. Bibliographie

[ART 82] ART D., GNANADESIKAN R., KETTENRING J.R., Data Based Metrics for Cluster Analysis, *Utilitas Mathematica*, 1982, 21 A, p 75-99.

[BEN 73] BENZECRI, J.P., *Analyse des Données: Correspondances*. 1973, Dunod, Paris.

[CHA 96] CHATEAU F. , LEBART L.: Assessing sample variability and stability in the visualization techniques related to principal component analysis; bootstrap and alternative simulation methods. *COMPSTAT 1996*, Prat A. (ed), Physica Verlag, Heidelberg (1996), p 205-210.

[CHU 97] CHUNG F.R.K., *Spectral Graph Theory*. CBMS Reg. Conf. Ser. Math. 92, American Mathematical Society, 1997.

[COT 97] COTTRELL M., ROUSSET P., The Kohonen Algorithm: a powerful tool for analysing and representing multidimensional qualitative and quantitative data. In: *Biological and Artificial Computation : From Neuroscience to Technology*. J. Mira, R. Moreno-Diaz, J. Cabestany, (eds), 1997, Springer, p 861-871.

[EFR 93] Efron B., Tibshirani R. J. *An Introduction to the Bootstrap*. 1993, Chapman and Hall, New York.

[ESC 89] ESCOFIER B., Multiple correspondence analysis and neighbouring relation. In: *Data Analysis, Learning Symbolic and Numeric Knowledge*. Diday E. (ed.), 1989, Nova Science Publishers, New York, p 55-62.

[KOH 89] KOHONEN T., *Self-Organization and Associative Memory*. 1989, Springer-Verlag, Berlin.

[LEB 69] LEBART L., Analyse Statistique de la Contiguïté, *Publ. de l'ISUP*. 1969, XVIII, p 81-112.

[LEB 03] LEBART L., PIRON M., STEINER J.-F. *La Sémiométrie*. 2003, Dunod, Paris.

[LEB 00] LEBART, L., Contiguity Analysis and Classification, In: W. Gaul, O. Opitz and M. Schader (Eds): *Data Analysis*. 2000, Springer, Berlin, p 233--244. (voir: www.lebart.org)

[MAR 94] MARKUS M.TH., Bootstrap Confidence Regions for Homogeneity Analysis; the Influence of Rotation on Coverage Percentages. *COMPSTAT 1994*, (Dutter R. and Grossmann W. (eds)) Physica Verlag, Heidelberg, 1994, p 337-342.

[MIL 95] MILAN L., WHITTAKER J., Application of the parametric bootstrap to models that incorporate a singular value decomposition. *Appl. Statist*. 44, 1995, p131-49.

[MOH 91] MOHAR B., The Laplacian Spectrum of Graphs, *Graph Theory, Combinatorics and Application*, 2, 1991, p 871-898.

[MOM 88] MOM A., *Méthodologie Statistique de la Classification des réseaux de transport*. Thèse, Université des Sciences et Techniques du Languedoc, 1988, Montpellier.

[THI 97] THIRIA S., LECHEVALLIER Y., GASCUEL O., CANU S., *Statistique et Méthodes Neuronales*, 1997, Dunod, Paris.

Nouvelle méthode de construction de variables

Gaëlle Legrand et Nicolas Nicoloyannis

Laboratoire ERIC - Université Lumière Lyon 2
5 av. Pierre Mendès-France
69 676 BRON cedex FRANCE
glegrand@eric.univ-lyon2.fr ; nicolas.nicoloyannis@univ-lyon2.fr

RÉSUMÉ. *Nous discutons du problème de la construction de nouvelles variables dans le but d'améliorer la qualité d'apprentissage supervisé. La méthode proposée construit de nouvelles variables en se basant sur l'analyse des arbres d'induction. Les nouvelles variables créées sont sous la forme de conjonction de modalités des variables initiales.*

MOTS-CLÉS : *Construction de variables, arbre d'induction.*

1. Introduction

La qualité d'apprentissage est fortement liée à la présence de variables discriminantes. Dans le cas d'une qualité d'apprentissage mediocre et en l'absence de nouvelles informations disponibles, la construction de nouvelles variables à partir des variables initiales permet de pallier ce problème. En effet, ce type de méthodes, appliquées lors de la phase de pré-traitement des données, construit des variables synthétiques pour re-décrire les données d'entrée du problème d'apprentissage.

2. État de l'art

La construction de variables augmente l'espace des variables en créant des variables supplémentaires. Cependant, aucune information extérieure à l'ensemble d'apprentissage n'est ajoutée lors du processus de construction. Il existe 4 principaux types de méthodes :

- L'induction constructive est l'application d'un ensemble d'opérateurs constructifs à un ensemble de variables, [BLO 98], [WNE 94], [MAT 89], [BLO 93].
- Les méthodes d'analyse de données.
- Les méthodes utilisant la théorie des graphes, [NGU 98].
- La construction de combinaisons booléennes de variables par analyse topologique des arbres, [OLI 94] : Ce type de méthodes s'applique initialement sur des problèmes à deux classes avec des variables exogènes booléennes.

Notre méthode appartient à l'ensemble des méthodes de construction de combinaison de variables par analyse topologique des arbres. En effet, elle utilise un arbre d'induction pour créer de nouvelles variables.

3. Formalisation et Cadre d'analyse

Nous ne travaillons qu'avec des variables qualitatives. Les variables quantitatives sont discrétisées à l'aide de Fusinter, [ZIG 96]. La méthode d'apprentissage utilisée est l'arbre de décision ID3, [QUI 86].
Soit Ω la population, chaque individu $\omega_j \in \Omega$ avec $j \in [1, ..., N]$, ($|\Omega| = N$), est décrit par p variables exogènes

$X_1, ..., X_i, ..., X_p$ et appartient à la classe C_k $avec\ k \in [1, ..., K]$. Soit D_i, le domaine d'application de la variable X_i, $|D_i| = M_i$ =Nombre de modalités de la variable X_i : $X_i : \Omega \mapsto D_i = \{x_{i1}, ..., x_{iM_i}\}$. Soit $\mathcal{C}$, l'ensemble des classes de la variable endogène $C : C : \Omega \mapsto \{C_1, ...C_K\} = \mathcal{C}$.
Notion 1 : Soit P, l'ensemble des prémisses, la prémisse $p_l \in P$ est une conjonction de plusieurs propositions logiques. Dans le cas des arbres d'induction, une prémisse est la conjonction de plusieurs modalités de différentes variables.
Notion 2 : Une règle $R_t = (p_l, C_k)$ est une combinaison unique entre une prémisse et une classe.
Notion 3 : Le regroupement de règles $R_{g\bar{k}}$ est l'ensemble $\{R_t = (p_l, C_{\bar{k}})\}$. $R_{g\bar{k}}$ rassemble toutes les règles qui ont pour conclusion la classe $C_{\bar{k}}$
Notion 4 : La base de construction d'un ensemble de données est le support d'information à partir duquel seront construites les nouvelles variables.
Notion 5 : Les données de construction sont constituées d'une partie des données initiales qui est utilisée dans le processus de construction.

4. Nouvelle méthode de construction

4.1. *Point de départ*

Nous commençons par donner les remarques constituant les bases de notre méthode.
Remarque 1 : Si une variable est choisie, à un moment donné du processus d'apprentissage, c'est parce qu'elle apporte le plus d'information. Or, ce fait est le plus souvent favorisé par la présence d'une ou plusieurs modalités (et rarement par la totalité des modalités) de la variable.

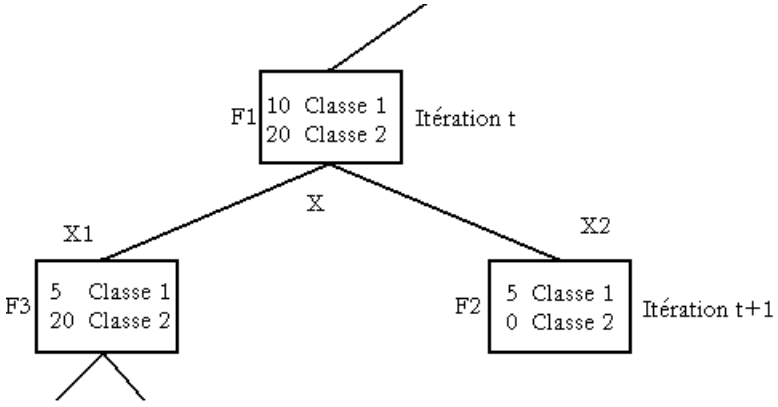


FIG. 1. *Fragment d'un arbre de décision - Exemple.*

Pour illustrer cette remarque, voici un exemple : la figure 1 nous montre le fragment d'un arbre de décision. A l'itération t+1, on suppose que la variable qui apporte le plus d'information est la variable X. X possède deux modalités X_1 et X_2. La nouvelle partition entraîne la disparition de la feuille F_1 et l'apparition des deux feuilles F_2 et F_3. Si l'on calcule le gain d'incertitude, $\mathcal{J}_{t+1}$, on obtient : $\mathcal{J}_{t+1} = \mathcal{I}_t - \mathcal{I}_{t+1}$ avec $\mathcal{I}_t$, l'incertitude sur la partition obtenue en t. Ici, à l'itération t, $\mathcal{I}_t = 0.98$. Et avec $\mathcal{I}_{t+1}$, l'incertitude sur la partition obtenue en t+1 :

$$\mathcal{I}_{t+1} = \mathcal{I}_{t+1}(F_2, F_3) = \sum_{k=2}^{3} f_{.k}(-\sum_{i=1}^{2} \frac{f_{ik}}{f_{.k}} \log_2 \frac{f_{ik}}{f_{.k}})$$

$$= \underbrace{-\frac{25}{30}(\frac{5}{25} \log_2 \frac{5}{25} + \frac{20}{25} \log_2 \frac{20}{25})}_{F_3} \underbrace{-\frac{5}{30}(\frac{5}{5} \log_2 \frac{5}{5} + \frac{0}{5} \log_2 \frac{0}{5})}_{F_2}$$

$$= \underbrace{0.6016}_{F_3} + \underbrace{0}_{F_2}$$

L'incertitude $\mathcal{I}_t$ étant déterminée à l'itération précédente, X a été sélectionnée car elle entraîne l'incertitude, $\mathcal{I}_{t+1}(F_2, F_3)$, la plus faible et permet, ainsi de maximiser le gain d'incertitude $\mathcal{J}_{t+1}$. Or, on remarque que la feuille F_2 liée à la modalité X_2 possède l'entropie la plus faible. C'est donc cette modalité qui discrimine le plus la variable endogène. Il existe une inégalité de discrimination au sein des modalités d'une même variable exogène. Nous désirons que notre méthode tienne compte de cette remarque. C'est pour cette raison que les nouvelles variables créées seront des conjonctions de modalités de différentes variables. Ces conjonctions devront permettre de discriminer au mieux la variable endogène. Pour obtenir ces conjonctions de modalités, nous nous servirons des règles issues de l'arbre d'induction.

Remarque 2 : Lors d'un processus d'apprentissage, les parties basses des arbres d'induction sont toujours élaguées à cause du trop petit nombre d'individus présents dans leurs différents noeuds ou feuilles. Or, ces parties basses peuvent être pertinentes et, la seule raison qui pousse les arbres d'induction à les éliminer est le gain d'information minimal fixé par l'utilisateur. Aussi, pour prendre en compte ces parties basses et atteindre les différents noeuds et feuilles de ces parties de l'arbre, nous relaxons les contraintes de l'arbre de decision ID3 : la contrainte liée au gain d'incertitude est supprimée de manière à ce que la croissance de l'arbre soit uniquement limitée par : soit le fait que le noeud terminal ne contienne que des individus d'une même classe ; soit le fait que toutes les variables exogènes soient indépendantes de la variable endogène (application du test du χ^2 pour ID3). Les règles issues de cet arbre, nommé arbre non contraint, seront notre base de construction.

4.2. *Déroulement de la méthode*

Notre méthode se décompose en trois étapes :

1. Génération de la base de construction à partir des données de construction : L'utilisation de l'arbre non contraint nous permet d'obtenir notre base de construction. Les règles issues de cette base sont de la forme suivante : $Si\ p_1\ et\ ...\ et\ p_l\ et\ ...\ et\ p_r\ Alors\ C\ =\ C_k$

2. Classement des règles obtenues à l'étape précédente : les règles sont regroupées en fonction de la classe de la variable endogène qui leur est associée. Il y aura donc autant de regroupements de règles que de classes de la variable endogène. Ainsi, la règle décrite en (1) appartient au regroupement de règles k.

3. Construction des nouvelles variables : il y a une variable construite par regroupement de règles. Donc, le nombre de variables construites est égal au nombre de classes de la variable endogène. Chaque règle est une conjonction de modalités de variables et chaque nouvelle variable est la disjonction des règles appartenant à un même regroupement. Les nouvelles variables sont de type booléen. Afin de construire ces variables, la création de variables intermédiaires y_{il} est nécessaire :

$pour\ tout\ k \in [1, ..., K]\ ;\ \forall R_t = (p_l, C_k) \in R_{gk}\,,$

$$\begin{aligned}(R_t, \omega_i) \mapsto y_{il} &= Vrai\ si\ \omega_i\ verifie\ p_l \\ &= Faux\ sinon\end{aligned}$$

Les nouvelles variables Y_{ik} peuvent maintenant être construites. Elles sont au nombre de K et de la forme suivante : $\forall k \in [1, ..., K],\ Y_{ik} = \bigcup_{R_t \in R_{gk}} y_{il}$

5. Expérimentations

La construction de règles s'est effectuée sur 30% des individus tirés aléatoirement. Les 70% restant sont utilisés pour les tests avant et après construction. Nous avons choisi une 10-cross validation et l'algorithme d'apprentissage ID3. Les bases de test utilisées sont issues de la collection de l'UCI Irvine.

Base	Avant Construction (AC) Erreur (Écart type)	Après Construction (ApC) Erreur (Écart type)	Écart $= Err.ApC - Err.AC$	Écart Relatif $= \frac{Ecart}{Err.AC}$
Tic Tac Toe	33,43 (5)	15,97 (4,77)	-17,46	-52,23%
Breast	5,95 (1,95)	5,32 (3,31)	-0,63	-10,59%
Austra	16,6 (4,57)	16,19 (5,03)	-0,41	-2,47%
Cleve	18,53 (8,68)	20,39 (11,29)	1,86	10,04%
CRX	14,73 (5,68)	13,92 (5,86)	-0,81	-5,50%
Diabetes	24,42 (6,41)	24,45 (4,97)	0,03	0,12%
German	31,86 (7,53)	22,14 (2,65)	-9,72	-30,51%
Heart	27,05 (10,29)	17,6 (6,87)	-9,45	-34,94%
Iono	21,37 (8,39)	17,95 (6,89)	-3,42	-16,00%
Pima	26,11 (5,43)	26,11 (3,85)	0	0,00%
Vehicle	34,24 (4,96)	35,76 (4,58)	1,52	4,44%

TAB. 1. *Présentation des résultats.*

Les résultats, présentés dans le tableau 1, nous montrent que dans de nombreux cas, notre méthode améliore la qualité d'apprentissage de la base. En particulier, ceci se vérifie pour les bases Tic Tac Toe, German et Heart qui voient leur taux d'erreur perdre jusqu'à 17 points grâce à la présence des nouvelles variables. Dans de rares cas, l'introduction de nouvelles variables entraîne une légère augmentation du taux d'erreur. Ces résultats sont renforcés par le calcul de l'écart et de l'écart relatif.

6. Conclusion

Notre méthode tient compte de l'algorithme d'apprentissage, de ses caractéristiques et de son influence sur les données étudiées. C'est une méthode qui prend en compte les liens existants entre les variables, et ce par le fait que les variables construites sont des conjonctions de modalités de différentes variables. De plus, le nombre de variables créées est faible, donc l'espace de représentation ne devient pas surchargé. Cependant, les variables construites dépendent du choix des données de construction et, si du bruit est présent dans ces données, les variables construites refléteront ce bruit. Cette méthode nous donne des résultats encourageants pour la suite de notre étude. Aussi, nous envisageons de poursuivre notre recherche dans plusieurs directions : l'utilisation d'autres méthodes pour la génération de la base de construction ; la pondération des règles au sein de chaque regroupement ; et, l'intervention d'un expert pour le choix des règles constituant la base de construction.

7. Bibliographie

[BLO 93] BLOEDORN E., WNEK J., MICHALSKI R., Multistrategy Constructive Induction, 1993.

[BLO 98] BLOEDORN E., Data-driven constructive induction : A Methodology and Its Applications, *IEEE Trans. on Intelligent Systems 13(2)*, 1998, p. 30-37.

[MAT 89] MATHEUS C. J., RENDELL L. A., Constructive Induction On Decision Trees, *IJCAI*, 1989, p. 645-650.

[NGU 98] NGUIFO E. M., NJIWOUA P., Using Lattice-Based Framework as a Tool for Feature Extraction, *European Conference on Machine Learning*, 1998, p. 304-309.

[OLI 94] OLIVEIRA A. L., SANGIOVANNI-VINCENTELLI A., Learning Complex Boolean Functions : Algorithms and Applications, COWAN J. D., TESAURO G., ALSPECTOR J., Eds., *Advances in Neural Information Processing Systems*, vol. 6, Morgan Kaufmann Publishers, Inc., 1994, p. 911–918.

[QUI 86] QUINLAN J., Introduction of Decision Trees, *Machine Learning*, , 1986.

[WNE 94] WNEK J., MICHALSKI R., Hypothesis-Driven Constructive in AQ17HCI : a methode and experiments, 1994.

[ZIG 96] ZIGHED D. A., RAKOTOMALALA R., RABASÉDA S., A discretization method of continous attributes in induction graphs, *13th European Meetings on Cybernetics and System Research*, 1996, p. 997-1002.

Sélection de variables et agrégation d'opinions

Gaëlle Legrand et Nicolas Nicoloyannis

Laboratoire ERIC - Université Lumière Lyon 2
5 av. Pierre Mendès-France
69 676 BRON cedex FRANCE
glegrand@eric.univ-lyon2.fr ; nicolas.nicoloyannis@univ-lyon2.fr

RÉSUMÉ. *La taille des bases de données étant de plus en plus importante, le processus de sélection de variables devient essentiel. Nous proposons une méthode de sélection, pour les variables qualitatives, basée sur l'agrégation d'opinion.*

MOTS-CLÉS : *Sélection de variables, agrégation d'opinion, critères myopes*

1. Introduction

La taille des bases de données étant de plus en plus importante, l'amélioration de la qualité de représentation des données est devenue un problème majeur de l'ECD. L'une des difficultés principales liée à la représentation des données est la dimension des données. La sélection de variables permet de résoudre cette difficulté. C'est un processus choisissant un sous-ensemble optimal de variables selon un critère particulier. Il permet l'élimination de variables inutiles, non pertinentes et redondantes ainsi que l'élimination du bruit généré par certaines variables. Il existe deux types d'approches de sélection :

- Les approches enveloppe, [JOH 94], prennent en compte l'influence du sous-ensemble de variables sélectionnées sur les performances de l'algorithme d'induction. Leur coût calculatoire est bien souvent trop important.
- Les approches filtre sont de 5 types :
  - Les méthodes exhaustives qui possèdent un coût calculatoire trop élevé,
  - Les méthodes heuristiques : la plus connue est RELIEF [KIR 92],
  - Les méthodes probabilistes qui sont représentées par LVF [LIU 96]. Du fait de leur caractéristique probabiliste, le nombre de variables sélectionnées tend vers la moitié du nombre de variables initiales,
  - Les méthodes de sélection en un seul parcours de base qui sont des processus itératifs qui ne nécessitent qu'un seul scan de la base étudiée. Ce type de méthode est représenté par MIFS [BAT 94].
  - Les méthodes pas à pas qui utilisent un critère de sélection myope tel que l'entropie de Shannon.

2. Point de départ

Nous sommes partis du constat suivant : les méthodes pas à pas sont rapides, peu coûteuses et présentent des résultats plutôt encourageants. Il existe 4 catégories de critères permettant de mesurer différentes caractéristiques des variables : les critères d'information, les critères de distance, les critères d'indépendance, et les critères de consistance. Cependant, l'utilisation d'une méthode myope génère trois problèmes : le choix du critère est délicat : quel critère est le plus efficace ? ; la forme du résultat (une liste de variables triées) ne nous permet pas de déterminer le sous-ensemble optimal de variables ; ce type de méthodes ne prend pas en compte l'interaction existante entre les variables exogènes. Notre méthode résoud les deux premiers problèmes soulevés de la manière suivante :

1. Il n'existe pas de critère plus efficace que les autres. Chaque critère met en avant certaines qualités spécifiques à chaque variable. Il semble donc intéressant d'obtenir un résultat tenant compte de l'avis de plusieurs

critères différents. Afin d'obtenir ce type de résultats, nous utilisons une méthode d'agrégation d'opinions.

2. Le fait d'obtenir une liste triée de variables limite l'intérêt de la sélection. En effet, comment peut-on déterminer la taille optimale du sous-ensemble ? Lorsque l'on est en présence d'une liste triée de variables, l'une des méthodes qui semble efficace pour obtenir un sous-ensemble optimal de variables est d'utiliser une approche enveloppe qui ajoute ou ôte itérativement les éléments de la liste triée. A chaque itération, la méthode d'apprentissage est appliquée pour tester si l'ajout ou la suppression d'une variable entraîne une amélioration du taux d'apprentissage. Toutefois, ce processus est bien trop coûteux. Pour cette raison, nous paramétrons la méthode d'agrégation utilisée pour qu'elle nous fournisse non pas un ordre sur les variables mais un préordre total. Ainsi, nous n'ajouterons pas les variables une par une mais par sous-ensemble de variables.

3. Présentation de la méthode

La méthode de sélection proposée se situe à l'intersection des approches filtre et enveloppe. Elle est de type Forward Selection et agrège les classements des variables obtenus à l'aide de plusieurs critères de sélection myopes. Elle traite des variables qualitatives. Cette méthode peut se décomposer en 3 étapes : calcul et discrétisation des différentes valeurs des critères pour chaque variable ; application de la méthode d'agrégation d'opinion sur les résultats obtenus à l'étape précédente ; recherche du sous-ensemble optimal.

3.1. *Calcul et discrétisation des critères*

Nous avons sélectionné un ensemble de 10 critères myopes de sélection : l'entropie de Shannon, le gain d'information, le ratio du gain, le gain normalisé, la distance de Mantaras, le critère de Gini, le chi2, le critère de Tschuprow, le coefficient de Cramer , et le τ de Zhou. Les calculs de chaque critère pour la totalité des variables s'effectuent en parallèle. Le résultat obtenu est un ensemble constitué de 10 listes ordonnées dans l'ordre décroissant l'importance de chaque variable.

Nous introduisons la notion d'équivalence de variables : Deux variables pouvant être aussi pertinentes l'une que l'autre vis à vis de la variable endogène. Afin de définir cette notion, nous considérons un ensemble de variables initial $X = \{x_1, ..., x_i, ..., x_p\}$. Soit $CR = \{cr_1, ..., cr_k, ..., cr_{10}\}$ l'ensemble des 10 critères myopes de sélection choisis avec $cr_k = \{cr_{k1}, ..., cr_{ki}, ..., cr_{kp}\}$, l'ensemble des valeurs du critère k pour les p variables de X.

Les valeurs cr_{ki} de chaque critère sont normalisées. Puis, ces valeurs sont discrétisées en déciles de même largeur. La discrétisation permet d'affecter à chaque variable x_i un rang R_{ki} pour chaque critère cr_k de la manière suivante :
Pour les critères qui doivent être minimisés :
Si $cr_{ki} \in [0; 0.1[$ alors $R_{ki} = 1$; Si $cr_{ki} \in [0.1; 0.2[$ alors $R_{ki} = 2$; ... ; Si $cr_{ki} \in [0.9; 1]$ alors $R_{ki} = 10$
Pour les critères qui doivent être maximisés :
Si $cr_{ki} \in [0; 0.1[$ alors $R_{ki} = 10$; Si $cr_{ki} \in [0.1; 0.2[$ alors $R_{ki} = 9$; ... Si $cr_{ki} \in [0.9; 1]$ alors $R_{ki} = 1$
La variable la plus pertinente est celle possédant le rang le plus faible.
Ainsi, deux variables sont équivalentes du point de vue d'un critère particulier si et seulement si pour ce critère, elles ont le même rang.

3.2. *Agrégation des résultats des critères*

Nous utilisons la méthode d'agrégation d'opinions développée dans [NIC 98], [MAR 81], [MIC 83]. Notre problème consiste donc à construire une opinion OP qui engendre un préordre total sur X et qui maximise le nombre d'accord entre l'opinion OP recherchée et l'opinion de chaque critère. Après l'application de cette technique d'agrégation, nous obtenons une liste ordonnée $L = \{l_1, ..., l_m, ..., l_M\}$ de sous-ensembles disjoints de variables.

3.3. *Découverte du sous-ensemble optimal de variables*

Jusqu'à présent, l'approche de notre méthode était de type filtre. Lors de cette étape, nous sommes dans une approche de type enveloppe. L'avantage d'utiliser une approche enveloppe est lié au fait que l'influence du sous-ensemble de variables sélectionnées sur les performances de l'algorithme d'apprentissage est prise en compte. A la m^{ieme} itération, le sous ensemble de variable $l_m \in L$ est ajouté au sous-ensemble optimal de variables. Le critère d'arrêt est double : il y arrêt du processus soit lorsque le taux d'erreur est constant sur deux itérations soit lorsque l'on assiste à une augmentation du taux d'erreur.

4. Expérimentations

Les variables quantitatives ont été discrétisées avec la méthode Fusinter [ZIG 96]. La sélection de variables s'est effectuée sur 30% des individus tout en gardant la répartition initiale des classes. Les 70% restant sont utilisés pour le calcul du taux d'erreur. Pour cela, nous avons choisi une 10-cross validation et l'algorithme d'apprentissage ID3. Les tests avant sélection ont également été effectués sur 70% de la base. Les bases de test utilisées sont issues de la collection de l'UCI Irvine. Le tableau 1 nous montre les taux d'erreur et les écarts-type associés obtenus avant et après la sélection de variables avec ID3.
Les résultats obtenus sont intéressants. En effet, excepté pour la base Iono, on assiste à une diminution du taux d'erreur et/ou à une stabilisation des résultats (diminution de l'écart-type). Le tableau 2 nous indique le nombre de variables sélectionnées avec ID3. A L'exception de la base Tic Tac Toe, le nombre de variables sélectionnées par notre méthode est inférieur à celui sélectionné par ReliefF et/ou MIFS. Les résultats d'apprentissage obtenus après la sélection sont équivalents pour les 3 méthodes.

Bases	Taux d'erreur avant Sél.	Ecart Type avant Sél.	Taux d'erreur après Sél.	Ecart Type après Sél.
Tic Tac Toe	28,44	7,53	25,16	6,31
Breast	5,9	2,64	4,27	2,8
CRX	14,46	5,44	15,7	3,1
Diabetes	24,3	3,97	23,38	3,32
Pima	25,24	5,76	24,5	5,15
Vehicle	30,59	5,54	28,75	5,44
Austra	17,16	6,21	15,29	3,48
Cleve	27,1	9,18	21,9	8,67
Heart	31,05	6,42	26,32	11,04
Iono	10,92	4,37	11,73	5,59
German	29,57	6,13	26,14	4,87

TAB. 1. *Evaluation de notre méthode de sélection.*

5. Conclusion

Notre méthode est une méthode hybride entre approches filtre et enveloppe qui possède les avantages de chaque approche et qui permet de réduire leurs inconvénients : l'influence des variables sélectionnées sur l'algorithme d'apprentissage utilisé est pris en compte. Ainsi, les variables sélectionnées sont différentes suivant l'algorithme utilisé. Les temps de calcul sont largement inférieurs à ceux des méthodes enveloppe pures grâce à l'utilisation du préordre et à l'obtention d'une liste triée de sous-ensembles de variables. Du point de vue du nombre de variables sélectionnées, les résultats que nous obtenons sont comparables voire supérieurs à ceux obtenus par ReliefF et MIFS. Du point de vue de la qualité d'apprentissage, nous assistons à une diminution des taux d'erreur et à une diminution des écart-types après la sélection. Cependant, nous envisageons d'obtenir comme résultat de la méthode d'agrégation non plus une liste de sous-ensembles de variables, mais le sous-ensemble optimal de variables.

Bases	Sans Sélection	Notre méthode	ReliefF	MIFS
Tic Tac Toe	9	7	5	3
Breast	9	3	6	9
CRX	15	3	2	7
Diabetes	8	2	4	5
Pima	8	2	7	4
Vehicle	18	14	18	6
Austra	14	1	2	13
Cleve	13	7	6	8
Heart	13	2	2	13
Iono	34	2	25	8
German	20	5	14	3

TAB. 2. *Nombre de variables sélectionnées.*

6. Bibliographie

[BAT 94] BATTITI R., Using mutual information for selecting features in supervised neural net learning, *IEEE Trans. on Neural Networks*, vol. 5, 1994, p. 537–550.

[JOH 94] JOHN G. H., KOHAVI R., PFLEGER K., Irrelevant Features and the Subset Selection Problem, *International Conference on Machine Learning*, 1994, p. 121-129, Journal version in AIJ, available at http ://citeseer.nj.nec.com/13663.html.

[KIR 92] KIRA K., RENDELL L., A practical approach to feature selection, *Proceedings of the Tenth International Conference on Machine Learning*, 1992, p. 500-512.

[LIU 96] LIU H., SETIONO R., A Probabilistic Approach to Feature Selection - A Filter Solution, *Int. Conf. on Machine Learning*, 1996, p. 319-327.

[MAR 81] MARCOTORCHINO F., MICHAUD P., Heuristic Approach to the similarity Aggregation Problem, *Methods of Operations Research*, vol. 43, 1981, p. 395-404.

[MIC 83] MICHAUD P., Opinions Agregations, *New Trends in Data Analysis and Applications, J. Janssen, J.F. Marcotorchino and J.M. Proth*, vol. 3, n^o 3, 1983.

[NIC 98] NICOLOYANNIS N., TERRENOIRE M., TOUNISSOUX D., An Optimisation Model for Aggregating Preferences : A Simulated Annealing Approach, *Health and System Science*, vol. 2, n^o 1-2, 1998, p. 33-44.

[ZIG 96] ZIGHED D. A., RAKOTOMALALA R., RABASÉDA S., A discretization method of continous attributes in induction graphs, *13th European Meetings on Cybernetics and System Research*, 1996, p. 997-1002.

Deux modèles de détection des transferts horizontaux de gènes dans une classification des espèces

Vladimir Makarenkov et Alix Boc

Département d'informatique, Université du Québec à Montréal, C.P. 8888, Succ. Centre-Ville, Montréal (Québec), Canada, H3C 3P8

Courriels : boc.alix@courrier.uqam.ca et makarenkov.vladimir@uqam.ca

RÉSUMÉ. Le transfert horizontal de gènes (i.e. transfert latéral de gènes) est un mécanisme évolutif permettant aux différents organismes de s'échanger des gènes au cours de l'évolution. Par exemple, ce phénomène est très fréquent dans l'évolution des bactéries. Dans cet article, nous discutons deux nouvelles approches utilisées pour la détection des transferts horizontaux de gènes dans un arbre phylogénétique (i.e. arbre additif) des espèces considérées. Pour détecter et représenter ces transferts, la première approche exploite un modèle de classification en réseau, alors que la deuxième approche utilise un modèle de classification en arbre. Les deux approches en question procèdent par la réconciliation des topologies de l'arbre du gène considéré et de l'arbre d'espèces. Deux critères d'optimisation ont été utilisés pour réconcilier la classification du gène considéré avec celle d'espèces. Il s'agit d'un côté, d'un critère métrique - les moindres carrés, et de l'autre côté, d'un critère topologique - la distance de Robinson et Foulds [ROB 81]. Cette distance permet de mesurer la similarité topologique entre deux arbres phylogénétiques. Les deux approches discutées dans cette article permettent de générer un scénario d'évolution du gène étudié pour un ensemble d'espèces observées.

MOTS-CLÉS : arbre phylogénétique, arbre additif, transfert horizontal de gènes, critère des moindres carrés, distance topologique de Robinson et Foulds

1. Introduction

La structure d'arbre phylogénétique (i.e. arbre additif) a toujours été utilisée pour représenter la classification des espèces. Dans un tel arbre, chaque espèce n'est reliée qu'avec son plus proche ancêtre et tous autres liens inter-espèces ne sont pas permis. Cependant, pour représenter les phénomènes d'évolution réticulée tels que : l'hybridation, l'homoplasie, la duplication de gènes et le transfert horizontal de gènes, les modèles de classification en réseau doivent être considérés. Dans ce papier nous examinons le cas du transfert horizontal de gènes, qui joue un rôle très important dans l'évolution d'espèces. De nombreuses tentatives d'appliquer des modèles en réseau pour détecter des transferts horizontaux peuvent être trouvées dans la littérature scientifique, voir par exemple [HEI 90] ou [PAG 81]. Un modèle de transfert permettant d'inscrire plusieurs arbres phylogénétiques de gènes dans un arbre phylogénétique d'espèces a été proposé par [HAL 01]. Dans [BOC 03], [BOC 04] et [MAK 04], nous avons proposé deux nouvelles approches pour la détection des transferts horizontaux de gène. Ces approches incorporent plusieurs règles d'évolution pertinentes d'un point de vue biologique. La première approche, [BOC 03] et [MAK 04], considère un modèle d'évolution permettant un transfert partiel du gène du donneur à l'espèce hôte (ceci est possible grâce au mécanisme de la recombinaison), tandis que la deuxième approche, [BOC 04], suppose que le gène du donneur supplante au complet le gène homologue de l'hôte. Le critère des moindres carrés et celui de Robinson et Foulds ont été choisis en tant que critères d'optimisation dans ces deux approches. Dans ce papier, nous comparons les deux approches de réconciliation examinées (i.e. deux modèles d'évolution de base). Nous montrerons comment les différences topologiques entre les arbres phylogénétiques de gène et d'espèces peuvent être exploitées pour déterminer un scénario possible de transferts latéraux du gène considéré.

2. Deux modèles de transferts horizontaux de gènes

Dans cette section nous considérons deux modèles d'évolution génétique supposant le transfert d'une partie du gène (modèle 1) et le transfert du gène au complet (modèle 2). On suppose que les arbres phylogénétiques représentant l'évolution du gène considéré et celle d'espèces sont déjà construits (à l'aide de la méthode NJ [SAI 87], par exemple). L'arbre d'espèces est généralement inféré à partir d'un gène ribosomal, e.g. 16S ARNr ou 23S ARNr, dont l'évolution n'a pas été affectée par des transferts horizontaux.

La Figure 1 ci-dessous illustre les deux modèles considérés. Dans le premier cas, quand une partie du gène transféré du donneur est récupérée par l'espèce hôte, l'arbre phylogénétique d'espèces est transformé en réseau phylogénétique par l'ajout d'une branche orientée représentant le transfert du gène. Sur la Figure 1 (l'arbre du haut), l'arête en pointillé reliant les arêtes (3,4) et (2,C) représente le transfert horizontal de gène entre elles. Dans un arbre phylogénétique, il existe toujours un chemin unique reliant toute paire de nœuds. Si le modèle supposant un transfert partiel est considéré, l'addition d'une branche représentant un transfert horizontal crée un autre chemin entre certains nœuds, en transformant l'arbre phylogénétique en réseau (Figure 1, l'arbre du bas à gauche). Comme le gène qui continue d'évoluer vers l'espèce C comporte maintenant une partie du gène transféré de l'arête (3,4) et une partie originale provenant de l'espèce 2, la distance d'évolution entre l'espèce C et toute autre espèce dans ce graphe doit être calculé comme le chemin de longueur minimum. Ce chemin passera donc soit par l'arête (6,2), soit par l'arête (6,7), voir Figure 1 (l'arbre du bas à gauche). Plusieurs règles d'évolution non considérées ici ont été incorporées dans ce modèle pour permettre une meilleure interprétation biologique. Parmi ces règles, nous avons : la définition du sens de l'évolution – de la racine vers les feuilles, l'interdiction de transferts entre les arêtes situées sur la même lignée, interdiction de plusieurs transferts croisés entre deux lignées données, etc (voir [BOC 04] et [MAK 04], pour plus de détails). Ce modèle est sans doute plus général que le modèle 2, qui suppose que le gène transféré du donneur supplante au complet le gène homologue de l'espèce hôte. Cependant, l'incorporation de nombreuses règles biologiques et la transformation de l'arbre phylogénétique en réseau font en sorte que le calcul des chemins de poids minimum dans le graphe orienté obtenu après l'ajout des arêtes de transferts horizontaux doit se faire par approximation (voir [MAK 04]). L'utilisation des formules d'approximation permet d'effectuer le calcul en temps polynomial. L'algorithme réalisant ce modèle utilise l'optimisation par les moindres carrés pour déterminer les transferts de gène les plus probables sous les contraintes biologiques définies.

Le deuxième modèle d'évolution considéré suppose que suite au transfert du gène au complet l'arbre phylogénétique est transformé en un autre arbre phylogénétique (Figure 1, l'arbre du bas à droite). Les transferts horizontaux sont ajoutés dans l'arbre phylogénétique d'espèces en le transformant en arbre phylogénétique du gène donné. Deux critères d'optimisation, topologique et métrique ont été utilisés dans le calcul pour déterminer les transferts horizontaux les plus probables. Nous avons considéré la distance de Robinson et Foulds [ROB 81 et MAK 00], comme critère topologique, et les moindres carrés, comme critère métrique. À la première itération, le transfert diminuant le plus la distance de Robinson et Foulds entre l'arbre du gène et celui d'espèces est considéré comme le plus probable. L'arête du transfert est par la suite ajoutée à l'arbre phylogénétique d'espèces et ainsi de suite. Comme dans ce modèle nous ne considérons que le transfert du gène au complet, la branche reliant l'espèce affectée par ce transfert et son ancêtre direct est supprimée de l'arbre. Puisque l'ajout d'une arête de transfert est toujours suivi par la suppression d'une arête, nous travaillons toujours avec un graphe connexe et sans cycles, qui est donc un arbre. La méthode réalisant le modèle 2, nécessite $O(kn^4)$ opérations pour ajouter k transferts horizontaux dans l'arbre phylogénétique à n espèces dans le cas des deux critères – topologique et métrique.

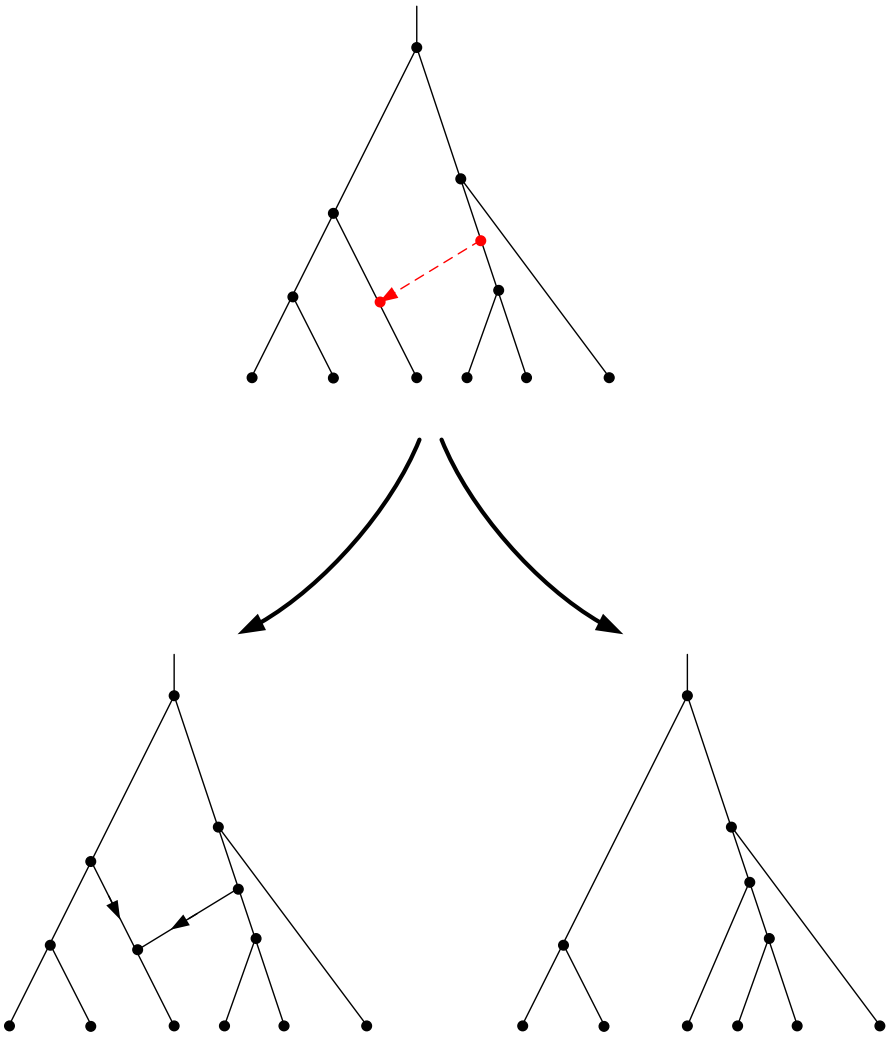


Figure 1. Deux modèles d'évolution génétique supposant que soit le transfert d'une partie du gène (l'arbre du bas à gauche - modèle 1), soit le transfert du gène entier avec la substitution complète du gène homologue chez l'espèces hôte (l'arbre du bas à droite - modèle 2) a eu lieu. Dans le premier cas, l'arbre est transformé en réseau orienté et dans le deuxième en un autre arbre phylogénétique. Les espèces sont associées aux feuilles de l'arbre et représentées par les lettres ; les ancêtres virtuels de ces espèces sont associés aux nœuds internes et représentés par les numéros.

3. Conclusion

Dans cet article nous avons considéré deux modèles de détection des transferts horizontaux de gènes. Les deux modèles sont basés sur un principe de réconciliation d'arbre phylogénétique de gène et d'espèces construits pour le même ensemble d'espèces. Le premier modèle, [BOC 03] et [MAK 04], supposant que n'importe quelle partie d'un gène donné peut être transférée à l'espèce hôte, est basé sur le calcul du chemin de poids minimum entre toute paire de nœuds présents dans l'arbre. Dans ce modèle de transfert, l'arbre phylogénétique d'espèces est transformé en réseau connexe dans lequel plusieurs chemins peuvent relier une paire d'espèces. Dans le deuxième modèle [BOC 04], qui suppose le transfert du gène au complet, l'arbre phylogénétique d'espèces est

transformé en arbre phylogénétique de gène par l'ajout à l'arbre d'espèces d'un transfert horizontal à chaque itération. À chaque étape de transformation de l'arbre d'espèces en arbre de gène, nous ne travaillons qu'avec des structures arborescentes et ne considérons jamais de modèles en réseau. Ce deuxième modèle, bien que moins générique que le premier, permet l'introduction d'un algorithme rapide et exact, sans utiliser le calcul approximatif, pour sa mise en œuvre. De plus, comme c'était montré dans [BOC 04], deux types de critères, topologique et métrique, peuvent être utilisés comme critère d'optimisation de base. Les deux modèles produisent un scénario de transferts horizontaux du gène considéré. Selon [BOC 04], l'utilisation de la distance topologique de Robinson et Foulds permet une meilleure détection des transferts horizontaux par rapport au modèle utilisant seulement le critère métrique (les tests on été fait sur les données génétiques réelles provenant de [DEL 96]). Parmi les pistes pour le futur développement, nous mentionnons la conception d'une technique de validation des résultats obtenus par les algorithmes discutés dans ce papier. Une méthode de validation serait idéalement capable de mesurer un taux de fiabilité à accorder aux transfert retrouvés. D'un autre côté, il est nécessaire de développer des modèles de transferts horizontaux basés sur les critères du maximum de vraisemblance et du maximum de parcimonie, qui sont les deux autres principaux critères utilisés en analyse phylogénétique. Les exécutables Windows pour les deux modèles de transfert exposés ici, de même que le code source en C, sont mis à la disposition des chercheurs à l'adresse URL suivante : <http://www.info.uqam.ca/~boca05/software>. Ces programmes seront bientôt ajouté au logiciel *T-Rex* [MAK 01] disponible pour Windows et Macintosh et bénéficiant d'une interface utilisateur graphique conviviale.

4. Bibliographie

[BOC 04] BOC A., MAKARENKOV V., DIALLO A.B., "Une nouvelle méthode pour la détection de transferts horizontaux de gène : la réconciliation topologique d'arbres de gène et d'espèces", soumis à *JOBIM 2004*, Montréal, Canada.

[BOC 03] BOC A., MAKARENKOV V., "New Efficient Algorithm for Detection of Horizontal Gene Transfer Events", *Algorithms in Bioinformatics*, G. Benson and R. Page (Eds.), 3rd Annual WABI'03, Springer-Verlag, p. 190-201.

[DEL 96] DELWICHE, C.F., PALMER J. D., "Rampant Horizontal Transfer and Duplication of Rubisco Genes in Eubacteria and Plastids", *Mol. Biol. Evol.*, vol. 13, 1996, p. 873-882.

[HAL 01] HALLET, M., LAGERGREN, J., "Efficient algorithms for lateral gene transfer problems", RECOMB 2001, Montréal, ACM, 2001, p. 149-156.

[HEI 90] HEIN, J. "A heuristic method to reconstructing the evolution of sequences subject to recombination using parsimony". *Math. Biosci.*, 1990, p. 185-200.

[MAK 00] MAKARENKOV, V., LECLERC, B., "Comparison of additive trees using circular orders", *Journal of Computational Biology*, vol. 7, 2000, p. 731-744.

[MAK 01] MAKARENKOV,V., "T-Rex: reconstructing and visualizing phylogenetic trees and reticulation networks", *Bioinformatics*, vol. 17, 2001, p. 664-668.

[MAK 04] MAKARENKOV, V. BOC, A., DIALLO, A. B., "Representing lateral gene transfer in species classification. Unique scenario", accepté pour publication à IFCS 2004, Chicago.

[PAG 98] PAGE, R. D. M., CHARLESTON, M. A., "From gene to organismal phylogeny: Reconciled trees", *Bioinformatics*, vol. 14, 1998, p. 819-820.

[ROB 81] ROBINSON D.R., FOULDS L.R., "Comparison of phylogenetic trees", *Mathematical Biosciences*, n° 53, 1981, p. 131-147.

[SAI 87] SAITOU, N., NEI, M., "The neighbour-joining method: a new method for reconstructing phylogenetic trees", *Mol. Biol. Evol*, vol. 4, 1987, p. 406-425.

Etude du comportement d'exposition et de protection solaire chez des adultes français

Emmanuelle Mauger*, Christiane Guinot*, Denis Malvy, Julie Latreille*, Laurence Ambroisine*, Pilar Galan***, Serge Hercberg*** et Erwin Tschachler*,*****

** CE.R.I.E.S., 20 rue Victor Noir, 92521 Neuilly sur Seine, France*
{emmanuelle.mauger, christiane.guinot, julie.latreille, laurence.ambroisine}@ceries-lab.com
*** EA 2323 et CNRS FRE 5036, Université Bordeaux 2, Bordeaux, France*
**** Coordination SU.VI.MAX, INSERM U557, ISTNA, CNAM, Paris, France*
***** Département de Dermatologie, Université de Vienne, Vienne, Autriche*

RÉSUMÉ. Une exposition solaire excessive engendre une accélération du vieillissement et une augmentation du risque de survenue de tumeurs cutanées. Dans le but d'estimer le risque lié à différents types de comportement, une typologie de comportement d'exposition et de protection solaire a été recherchée. Un questionnaire explorant ce sujet auprès d'hommes et de femmes adultes français a été développé dans le cadre de l'étude épidémiologique SU.VI.MAX. Une série d'analyses des correspondances multiples a été effectuée pour résumer l'information. Puis, une classification ascendante hiérarchique (méthode de Ward) a été réalisée à partir des composantes principales retenues permettant d'identifier sept comportements pour les femmes, et six pour les hommes. Finalement, un arbre de décision a été construit afin de pouvoir affecter facilement n'importe quel individu à une classe (algorithme CART).

MOTS-CLÉS : analyse des correspondances multiples, algorithme CART, méthode de Ward, questionnaire auto-administré

1. Introduction

Les rayons ultraviolets sont connus pour jouer un rôle prépondérant dans l'accélération du vieillissement cutané et le développement des tumeurs cutanées. Néanmoins, l'augmentation de la durée des vacances, la facilité des voyages et la mode du bronzage ont entraîné ces cinquante dernières années une plus grande exposition au soleil [ART 95]. Dans le but d'estimer le risque lié à différents types de comportement, une typologie de comportement d'exposition et de protection solaire a été recherchée à partir d'un questionnaire auto-administré développé spécifiquement pour l'étude SU.VI.MAX (SUpplémentation en VItamines et Minéraux Anti-oXydants) [HER 98].

2. Matériel et méthodes

Le questionnaire « soleil SU.VI.MAX » comporte deux parties, la première partie sur les habitudes d'exposition et de protection solaire dans l'année qui vient de s'écouler, et la deuxième partie sur les habitudes d'exposition appréciées globalement au cours de la vie. Soixante dix pour cent des questionnaires ont été récupérés et 63% ont été exploités. Au final les données de 4 825 femmes et 3 259 hommes ont été utilisées [GUI 01].
Les analyses ont été réalisées par genre. Les analyses pour la recherche de typologie de comportement d'exposition et de protection solaire ont été effectuées en réalisant des analyses séparées par groupes d'individus. Une première analyse a été réalisée pour les individus ayant déclaré s'exposer volontairement au soleil et utiliser des produits de protection solaire et une seconde analyse pour les individus ayant déclaré s'exposer volontairement au soleil et ne pas utiliser de produit de protection solaire. Les individus ayant déclaré ne pas s'exposer volontairement au soleil ont été considérés dès le départ comme étant une classe à part entière. La même stratégie d'analyse a été utilisée pour chacun des deux groupes. Une analyse des correspondances multiples a tout d'abord été réalisée dans le but de construire des variables de synthèse résumant au mieux

l'information. Les liens entre les variables décrivant les habitudes d'exposition et de protection solaire ont également été étudiés [JOB 92]. Une classification ascendante hiérarchique (méthode de Ward) des individus a été réalisée à partir des composantes principales retenues à l'étape précédente [EVE 93]. La représentation du dendogramme a été réalisée. Afin de décrire les classes obtenues et de les nommer, des tests de comparaison de moyennes et de pourcentages ont été effectués. Une variable synthétique a ensuite été construite en regroupant la classe C0 des individus ayant déclaré ne pas s'exposer volontairement au soleil, et les classes obtenues sur les deux autres groupes. Afin de décrire et de comparer l'ensemble des types de comportement obtenus, des tests de comparaison de moyennes et de pourcentages ont été effectués. Finalement, dans le but d'assigner facilement n'importe quel individu à un type, un arbre de décision a été construit afin de déterminer des règles de décision basées sur un petit nombre de questions. Pour ce faire, l'algorithme CART a été utilisé avec l'indice de Gini comme mesure d'impureté [BRE 84].

3. Résultats

Pour des raisons de contrainte de place, seuls les résultats sur les femmes sont présentés dans ce document, les résultats sur les hommes étant similaires.

3.1. *Femmes ayant déclaré s'exposer volontairement au soleil et utiliser des produits de protection solaire*

Les quatre premiers axes factoriels restituent 53% de l'information. La figure 1 montre que la première composante oppose les femmes qui ont déclaré utiliser un produit de protection solaire régulièrement (à gauche) à celles qui ont déclaré en utiliser de façon irrégulière (à droite). La seconde composante oppose les femmes qui ont déclaré s'exposer de façon modérée (en haut) à celles qui ont déclaré s'exposer au soleil de façon intense (en bas). La troisième composante oppose les femmes qui ont déclaré utiliser un produit de protection solaire avec des indices moyens et s'exposer de façon modérée à celles qui ont déclaré utiliser une protection autre que moyenne et s'exposer de façon intense. La quatrième dimension oppose les femmes qui ont déclaré utiliser des produits sans filtre solaire à celles qui ont déclaré utiliser un produit de protection avec un bon filtre solaire (le plan factoriel 3-4 n'est pas montré). La typologie obtenue a permis d'identifier 3 classes : les femmes qui utilisent une protection solaire sans filtre solaire et qui s'exposent modérément (C1, n=284), les femmes qui utilisent une protection solaire moyenne et s'exposent de façon intense (C2, n=1364) et les femmes qui utilisent une forte protection solaire et s'exposent modérément (C3, n=466).

3.2. *Femmes ayant déclaré s'exposer volontairement au soleil et ne pas utiliser de produit de protection solaire*

Les quatre premiers axes factoriels restituent 88% de l'information. La première composante oppose les femmes qui ont déclaré s'exposer de façon modérée à celles qui ont déclaré s'exposer de façon intense et la seconde composante oppose les femmes qui déclarent ne pas s'exposer progressivement à celles qui déclarent penser que lézarder est important. La troisième composante oppose les femmes qui déclarent s'exposer aux heures chaudes et de façon non progressive à celles qui déclarent ne pas s'exposer aux heures les plus chaudes. La quatrième composante oppose les femmes qui déclarent ne pas utiliser de moyen de protection autre que des produits de protection solaire aux autres femmes (les plans factoriels 1-2 et 3-4 ne sont pas montrés). La typologie obtenue a permis d'identifier 3 classes : les femmes qui n'utilisent pas de produit de protection solaire et s'exposent modérément et prudemment (C4, n=58), les femmes qui n'utilisent pas de produit de protection solaire et qui s'exposent de façon modérée et imprudente (C5, n=136) et les femmes qui n'utilisent pas de produit de protection solaire et s'exposent de façon intense (C6, n=43).

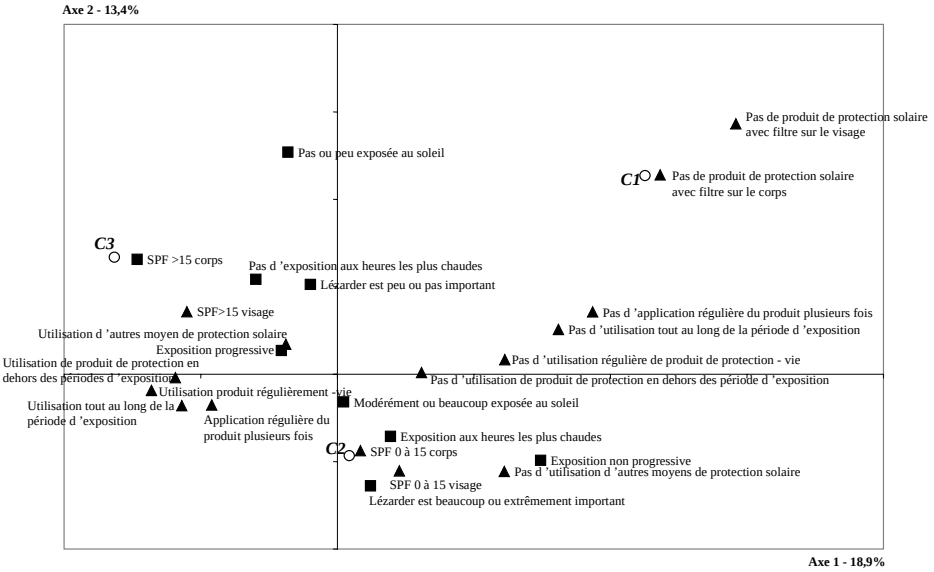


Figure 1. Premier plan factoriel de l'ACM sur les habitudes d'exposition et de protection solaire. ■ Variables décrivant les habitudes d'exposition solaire, ▲ variables décrivant les habitudes de protection solaire, ○ typologie de comportement face au soleil (variable illustrative)

3.3. *Description des classes de la typologie de comportement d'exposition et de protection solaire*

La classe C0 des femmes ayant déclaré ne pas s'exposer volontairement au soleil au cours de la vie comprend 1 558 femmes. La répartition des femmes selon la typologie de comportement d'exposition solaire et de protection solaire et les variables d'exposition et de protection solaire du questionnaire est indiquée dans le tableau 1.

		TYPOLOGIE						
Variables	Modalités	C0	C1	C2	C3	C4	C5	C6
Sur l'année								
Bronzage entre 11h et 16h		10,2	57,9	52,6	36,4	12,1	78,2	85,0
Durée d'exposition >2h par jour		12.7	31,2	32,9	24,8	15,8	19,4	57,5
Exposition progressive		29,5	75,1	83,5	89,1	81,5	66,4	53,9
Intensité d'exposition	Modérément ou beaucoup exposée	29,6	63,4	72,9	62,2	58,6	56,0	88,1
	Pas ou peu exposée	70,5	36,6	27,1	37,9	41,4	44,0	11,9
Importance du fait de lézarder au soleil	Beaucoup ou extrêmement important	4,2	34,4	35,4	26,9	19,0	38,5	54,8
	Pas ou peu important	95,8	65,6	64,7	73,1	81,0	61,5	45,2
Indice de protection utilisé en début de la période d'exposition sur le corps	Pas de filtre solaire	11,7	79,2	0,0	0,0	100,0	100,0	100,0
	SPF 0 à 15	15,7	14,1	97,4	0,0	0,0	0,0	0,0
	SPF >15	12,2	6,7	2,6	100,0	0,0	0,0	0,0
	pas d'exposition volontaire	60,4	0,0	0,0	0,0	0,0	0,0	0,0
Indice de protection utilisé en début de la période d'exposition sur le visage	Pas de filtre solaire	10,5	62,3	0,0	0,0	100,0	100,0	100,0
	SPF 0 à 15	11,3	17,6	72,6	0,0	0,0	0,0	0,0
	SPF >15	17,8	20,1	27,4	100,0	0,0	0,0	0,0
	pas d'exposition volontaire	60,4	0,0	0,0	0,0	0,0	0,0	0,0
Utilisation de produits de protection solaire en dehors des périodes d'exposition		11,1	26,4	33,7	40,6	0,0	0,0	0,0
Utilisation d'un produit de protection solaire tout au long de la période d'exposition		17,4	37,3	58,7	71,7	0,0	0,0	0,0
Application régulière du produit de protection solaire plusieurs fois dans la journée		19,4	46,5	67,7	77,5	0,0	0,0	0,0
Utilisation d'autres moyens de protection contre le soleil		81,2	62,3	73,4	84,0	91,4	62,1	56,1
Au cours de la vie								
Bronzage entre 11h et 16h		0,0	63,4	62,5	52,8	0,0	91,9	93,0
Exposition progressive		0,0	70,4	78,5	83,3	82,8	69,9	53,5
Intensité d'exposition	Modérément ou beaucoup exposée	42,1	82,8	91,7	84,6	87,9	86,0	100,0
	Pas ou peu exposée	57,9	17,3	8,3	15,5	12,1	14,0	0,0
Importance du fait de lézarder au soleil	Beaucoup ou extrêmement important	3,5	40,5	47,9	37,8	17,2	45,6	76,7
	Pas ou peu important	96,6	59,5	52,1	62,2	82,8	54,4	23,3
Utilisation régulière de produits de protection solaire		0,0	29,6	46,6	60,7	5,4	5,3	2,5
Utilisation d'autres moyens de protection contre le soleil		86,4	67,3	76,0	83,3	96,6	61,0	46,5

Tableau 1. Exposition et protection solaire : fréquences (%) selon la typologie de comportement d'exposition et de protection solaire et les variables d'exposition et de protection solaire du questionnaire. Toutes les variables sont significatives (P<0,0001).

3.4. *Arbre de décision*

Un arbre de décision basé sur 6 questions et 9 règles de décision a été obtenu. Les 6 questions sont : « Habitude de pratiquer le bronzage au cours de la vie d'adulte », « Utilisation d'un produit de protection solaire pendant les pratiques de bronzage », « Exposition au cours de la vie », « Pratique du bronzage aux heures les plus chaudes (11h-16h) », « Indice de protection solaire utilisé sur le corps », et « Indice de protection solaire utilisé sur le

visage ». La comparaison entre la classe initiale et la classe d'attribution des femmes selon les règles de décision montre une concordance très satisfaisante (1% de mal classées).

4. Conclusion

Cette analyse a permis de trouver une typologie de comportement face au soleil de femmes adultes françaises en prenant en compte à la fois des habitudes d'exposition et de protection solaire. La même analyse a été conduite pour les hommes permettant l'identification de 6 types de comportement. Cette recherche de typologie a été réalisée dans le but d'estimer le risque lié aux différents types de comportement, afin de pouvoir cibler ultérieurement des groupes d'individus à risque pour des campagnes d'information de santé publique et/ou pour des études d'intervention.

5. Bibliographie

[ART 95] ARTHEY S., CLARKE VA., « Suntanning and sun protection : a review of the psychological litterature ». *Soc Sci Med* 1995, 40:265-274.

[BRE 84] BREIMAN L., FRIEDMAN J.H., OLSHEN R.A. et STONE C.J. *Classification and regression trees*. Chapman & Hall, New-York, 1984.

[EVE 93] EVERITT BS. Eds. Cluster analysis. London : Arnold, 1993.

[GUI 01] GUINOT C., MALVY D., LATREILLE J., PREZIOSI P., GALAN P., VAILLANT L., TENENHAUS M., HERCBERG S., TSCHACHLER E., « Sun exposure behaviour of a general adult population in France ». Dans : Skin and Environment – Perception and Protection (J. Ring, S. Weidinger, U. Darsow, éditeurs), 10e congrès de l'EADV, Munich, 10-14 octobre 2001, Bologne, Monduzzi editore S.p.A., 2001, p.1099-1106.

[HER 98] HERCBERG S., PREZIOSI P., BRIANÇON S., GALAN P., TRIOL I., MALVY D., ROUSSEL AM, FAVIER A., « A primary prevention trial using nutritional doses of antioxidant vitamins and minerals in cardio-vascular diseases and cancers in a general population: « The SU.VI.MAX study » - Design, methods and participants characteristics ». *Control Clin Trials* 1998, 19:336-351.

[JOB 92] JOBSON JD. Applied Multivariate Data Analysis. Volume II : Categorical and Multivariate Methods. New York : Springer Verlag, 1992.

Explication de la corrélation interne aux classes d'une partition

Chérif Mballo[1,2] **— Edwin Diday** [2]

1 : ESIEA Recherche
38, Rue des Docteurs Calmette et Guérin
53000 Laval- France.
mballo@esiea-ouest.fr

2 : LISE-CEREMADE, Université Paris IX Dauphine
Place du Maréchal de Lattre de Tassigny
75775 Paris 16 ième
diday@ceremade.dauphine.fr

RÉSUMÉ. On désire expliquer la corrélation entre deux variables calculées sur chaque classe d'une partition donnée. Pour cela, on transforme d'abord par généralisation chacune des classes de façon à obtenir une description symbolique concise de chaque classe. On obtient ainsi un tableau de données symboliques dont chaque ligne représente la description d'une classe définie par les valeurs prises par les variables symboliques associées aux variables standards initiales. A ce tableau, on ajoute une colonne supplémentaire correspondant à la corrélation entre deux mêmes variables pour chaque classe. C'est la variable à expliquer, non encore partitionnée. On utilise l'algorithme de Fisher pour partitionner la variable à expliquer, puis un arbre de décision utilisant le critère de Kolmogorov-Smirnov basé sur un ordre des valeurs des variables explicatives.

MOTS-CLÉS : Corrélation, algorithme de Fisher, arbre de décision, critère de Kolmogorov-Smirnov.

1. Introduction

On part d'un tableau de données classiques. On s'intéresse à une variable de ce tableau pour en déduire une partition, soit à partir de ses modalités si elle est qualitative, soit en la découpant en classes si elle est quantitative. On peut utiliser pour cela l'algorithme de Fisher ([FIS 58]) qui a pour but de déterminer la partition optimale d'une population (décrite par une variable continue) en un nombre donné de classes d'individus bien agrégés et bien séparés. Ayant obtenu cette partition notée P, on en déduit une description agrégée de chacune de ses classes. Les variables classiques de départ (à valeur numérique ou qualitative) sont ainsi transformées en variables symboliques ([BOC 00]) à valeur intervalle ou diagramme. Par exemple, si la variable *âge* des individus d'une classe varie entre 20 ans et 30 ans, la variable symbolique "*âge*" prendra la valeur [20, 30]. On obtient ainsi un tableau de données symboliques auquel on ajoute une colonne correspondant aux corrélations à expliquer entre deux variables initiales y et y' calculées à l'intérieur de chaque classe de la partition P. On peut alors utiliser l'algorithme de Fisher à nouveau pour découper la variable corrélation *cor(y, y')* afin d'obtenir la variable à expliquer d'un arbre de décision. Différentes méthodes de construction d'arbres de décision sur données symboliques ont été proposées ([PER 96]). Ici nous privilégions la méthode basée sur le critère de découpage binaire de Kolmogorov-Smirnov ([FRI 77] ; [UTG 96]).

2. Présentation du processus pour expliquer une corrélation

Dans l'apprentissage par arbre binaire de décision utilisant le critère de découpage de Kolmogorov-Smirnov (noté KS dans la suite), la variable à expliquer est de type classe et les variables explicatives continues ([FRI 77] ; [UTG 96]) ou qualitatives ([ASS 98]). Nous considérons au départ un tableau de données classiques et nous nous intéressons à la corrélation interne aux classes de deux variables de ce tableau. La variable obtenue des corrélations des classes est une variable quantitative. L'algorithme de Fisher ([FIS 58]) nous permet de partitionner une telle variable. Nous obtenons ainsi une variable classe pour construire un arbre de décision. Comme nous utilisons le critère KS, il faut alors ordonner les variables explicatives qui ne sont plus classiques, mais symboliques. Le critère KS permet de séparer une population en deux sous populations plus homogènes en

se basant sur les fonctions de répartition. Dans le cas où le nombre de classes a priori est k (avec k>2), les fonctions de répartition sont induites par le regroupement de ces k classes a priori en deux groupes ($2^{k-1}-1$ possibilités), mais cette complexité exponentielle a été réduite à une complexité polynomiale par [ASS 98]) appelés super classes par la méthode « twoing splitting process » ([BRE 84]). Cette méthode est utilisée pour générer deux super-classes C_1, C_2 auxquelles sont associées les deux fonctions de répartitions F_1, F_2 d'une variable aléatoire. Selon l'ordre choisi pour ordonner les observations d'une variable explicative symbolique X, la fonction de répartition empirique $\hat{F}_i$ (i=1,2) qui estime F_i en $x \in X$ est donnée par :

$$\hat{\hat{F}}_i(x) = \frac{Cardinal\ (\{y \in X\ /\ y \le x\} \cap C_i)}{Cardinal\ (C_i)}$$

Ainsi, le critère KS est défini par : $KS(x) = \underset{X}{Sup} \sup_x \left| \hat{\hat{F}}_1(x) - \hat{F}_2(x) \right|$

3. Exemple illustratif

Considérons le tableau de données suivant (disponible sur le site du logiciel SODAS) où les individus sont des noms de châteaux et les valeurs pour chaque vin représentent des notes d'experts :

Variables / *Individus*	*Appellation*	*Superficie*	*Cabernet Sauvignon*	*Cabernet Franc*	*Merlot*	*Petit Verdot*	*Nombre Caisses*
Ausone	saint emilion	7	0.001	0.5	0.5	0.001	1800
Cheval Blanc	saint emilion	35	0	0.4	0.6	0	12500
Cos d'Estournel	saint estephe	65	0.6	0	0.4	0.002	20000
Ducru-Beaucaillou	saint julien	50	0.649	0.05	0.25	0.05	20000
Haut-Brion	graves	40	0.5	0.25	0.25	0	12000
Lafite-Rothschild	pauillac	90	0.75	0.04	0.2	0.01	35000
Lafleur	pomerol	4.5	0	0.5	0.5	0	1600
Latour	haut medoc	45	0.4	0.2	0.35	0.05	15000
Léoville Las Cases	saint julien	97	0.65	0.13	0.19	0.03	30000
L'Evangile	pomerol	14	0.29	0	0.71	0	4500
Lynch-Bages	pauillac	80	0.75	0.1	0.15	0	35000
Margaux	saint emilion	66	0.75	0.025	0.2	0.025	25000
Mission Haut-Brion	graves	20	0.65	0.1	0.25	0.001	8000
Montrose	saint estephe	68	0.65	0.1	0.25	0	28000
Mouton-Rothschild	pauillac	75	0.85	0.08	0.07	0	20000
Petit Village	pomerol	11	0.1	0	0.9	0	4500
Petrus	pomerol	11.5	0	0.05	0.95	0	3750
PichonC. de Lalande	pauillac	60	0.57	0	0.35	0.08	28000
Pichon Longueville	pauillac	50	0.75	0	0.25	0	15000
Sociando Mallet	haut medoc	30	0.6	0.1	0.25	0.05	15000
Trotanoy	pomerol	7.5	0	0.1	0.9	0	3000
Vieux Château Certan	pomerol	13.5	0.1	0.3	0.6	0.002	5500

Tableau 1. ***Données initiales***

Avec les modalités de la variable « *Appellation* », nous obtenons une partition dont les classes sont les noms d'appellation. Par exemple la classe *saint emilion = {Ausone, Cheval Blanc, Margaux}*. Nous nous intéressons à l'étude de la corrélation entre les variables *Cabernet Sauvignon* et *Cabernet Franc* à l'intérieur de chaque classe d'appellation. Par exemple pour la classe *saint emilion*, on détermine la corrélation entre les valeurs 0.001 ; 0 ; 0.75 et 0.5 ; 0.4 ; 0.025. Nous obtenons une nouvelle colonne (tableau 2) appelée « *Cor_CS_CF* » contenant les coefficients de corrélation entre ces deux variables pour chaque classe d'appellation. Chaque variable du tableau 1 est alors transformée en une variable intervalle car les nouveaux individus de ce tableau sont les appellations.

Nous partitionnons la variable corrélation en deux classes par l'algorithme de Fisher ([FIS 58]). Nous obtenons les classes suivantes:
classe 1 = {*graves ; haut medoc ; saint emilion ; pomerol*} de description [-1 ; -0.418] ;
classe 2 = { *pauillac ; saint estephe ; saint julien*} de description [0.622 ; 1].
Pour chaque variable symbolique Y, la description d'une classe est un intervalle [α; β] où α est la description de l'individu de plus petite valeur dans la classe et β celle de plus grande valeur prises par les individus de cette même classe pour la variable initiale y associée à Y.

Variables / Individus	***Superficie***	***Cabernet Sauvignon***	***Cabernet Franc***	***Merlot***	***Petit Verdot***	***Nombre Caisses***	***Cor_CS_CF***
saint emilion	[7;66]	[0;0.75]	[0.025;0.5]	[0.2;0.6]	[0.001;0.025]	[1800;25000]	-0.979
saint estephe	[65;68]	[0.6;0.65]	[0;0.1]	[0.25;0.4]	[0;0.002]	[20000;28000]	1
saint julien	[50;97]	[0.649;0.65]	[0.05;0.13]	[0.19;0.25]	[0.03;0.05]	[20000;30000]	1
Graves	[20;40]	[0.5;0.65]	[0.1;0.25]	[0.25;0.25]	[0;0.001]	[8000;12000]	-1
pauillac	[50;90]	[0.57;0.85]	[0;0.08]	[0.07;0.35]	[0;0.08]	[15000;28000]	0.622
pomerol	[4.5;14]	[0;0.29]	[0;0.5]	[0.5;0.95]	[0;0.002]	[1600;5500]	-0.418
haut medoc	[30;45]	[0.4;0.6]	[0.1;0.2]	[0.25;0.35]	[0.05;0.05]	[15000;15000]	-1

Tableau 2

La variable corrélation « *Cor_CS_CF* » du tableau 2 est transformée en une variable classe (classes a priori pour l'arbre de décision) par l'algorithme de classification optimale de Fisher (tableau 3).

Variables / Individus	***Superficie***	***Cabernet Sauvignon***	***Cabernet Franc***	***Merlot***	***Petit Verdot***	***Nombre Caisses***	***Cor_CS_CF***
saint emilion	[7;66]	[0;0.75]	[0.025;0.5]	[0.2;0.6]	[0.001;0.025]	[1800;25000]	1
saint estephe	[65;68]	[0.6;0.65]	[0;0.1]	[0.25;0.4]	[0;0.002]	[20000;28000]	2
saint julien	[50;97]	[0.649;0.65]	[0.05;0.13]	[0.19;0.25]	[0.03;0.05]	[20000;30000]	2
Graves	[20;40]	[0.5;0.65]	[0.1;0.25]	[0.25;0.25]	[0;0.001]	[8000;12000]	1
pauillac	[50;90]	[0.57;0.85]	[0;0.08]	[0.07;0.35]	[0;0.08]	[15000;28000]	2
pomerol	[4.5;14]	[0;0.29]	[0;0.5]	[0.5;0.95]	[0;0.002]	[1600;5500]	1
haut medoc	[30;45]	[0.4;0.6]	[0.1;0.2]	[0.25;0.35]	[0.05;0.05]	[15000;15000]	1

Tableau 3

Nous obtenons ainsi un tableau de données symboliques avec des variables de type intervalle et une variable classe. A partir de ce tableau 3, on fait les arbres de décision à l'aide du critère de découpage binaire KS. Toutes les variables de type intervalle du tableau 3 sont les variables explicatives et la variable corrélation *Cor_CS_CF* est la variable à expliquer. Le critère KS nécessite un ordre des valeurs prises par chaque variable explicative. On peut ordonner des variables à valeur intervalle de différentes façons ([DID 03]). Pour simplifier, nous ordonnons ces variables explicatives par la moyenne. On obtient le tableau 4 donnant l'ordre des intervalles pour chaque variable explicative (la classe a priori correspondante est entre parenthèses).

Superficie	***Cabernet Sauvignon***	***Cabernet Franc***	***Merlot***	***Petit Verdot***	***Nombre Caisses***
[4.5;14] (1)	[0;0.29] (1)	[0;0.08] (2)	[0.07;0.35] (2)	[0;0.001] (1)	[1600;5500] (1)
[20;40] (1)	[0;0.75] (1)	[0;0.1] (2)	[0.19;0.25] (2)	[0;0.002] (2)	[8000;12000] (1)
[7;66] (1)	[0.4;0.6] (1)	[0.05;0.13] (2)	[0.25;0.25] (1)	[0;0.002] (1)	[1800;25000] (1)
[30;45] (1)	[0.5;0.65] (1)	[0.1;0.2] (1)	[0.25;0.35] (1)	[0;0.08] (2)	[15000;15000] (1)
[65;68] (2)	[0.6;0.65] (2)	[0.1;0.25] (1)	[0.25;0.4] (2)	[0.001;0.025] (1)	[15000;28000] (2)
[50;90] (2)	[0.649;0.65] (2)	[0;0.5] (1)	[0.2;0.6] (1)	[0.03;0.05] (2)	[20000;28000] (2)
[50;97] (2)	[0.57;0.85] (2)	[0.025;0.5] (1)	[0.5;0.95] (1)	[0.05;0.05] (1)	[20000;30000] (2)

Tableau 4. *Ordre des intervalles par la moyenne pour chaque variable explicative*

L'algorithme de Kolmogorov-Smirnov calcule la valeur du KS pour chaque variable explicative et pour chaque description au niveau de chaque nœud de l'arbre et récupère la valeur maximale (elle correspond à une variable et à une description). Par exemple pour la variable explicative *Superficie*, KS([20 ;40])=$\left|\frac{2}{4}-\frac{0}{3}\right|=\frac{1}{2}$ car il y a deux intervalles de la classe 1 qui sont avant lui (lui-même compté) sur quatre intervalles de la classe 1 et zéro intervalle de la classe 2 avant lui sur trois intervalles de la classe 2. Avec le tableau 4, on voit que la variable *Superficie* sépare bien la population à l'intervalle [30 ; 45] (affectation pure). C'est une extension naturelle du critère KS, seulement l'argument sélectionné pour le seuil de coupure est un intervalle et non un réel comme dans le cas classique. On peut donc utiliser toutes les autres étapes qui sont communes à tout type de variable pour construire l'arbre de décision (Figure 1). Avec ce petit exemple, nous voyons que la corrélation interne aux classes de la partition produite par la variable « *Appellation* » (deuxième colonne du tableau 1) est expliquée par la variable *Superficie*. On obtient ainsi deux règles de décision correspondant aux deux feuilles (*w* désigne un individu et « ≤ » un ordre d'intervalles ([DID 30])) :
règle 1 : si *Superficie(w)* $\leq$ *[30; 45]*, alors *corrélation(w)* $\in$ *[-1 ; -0.418]*
règle 2 : si *Superficie(w)* > *[30; 45]*, alors *corrélation(w)* $\in$ *[0.622 ; 1]*

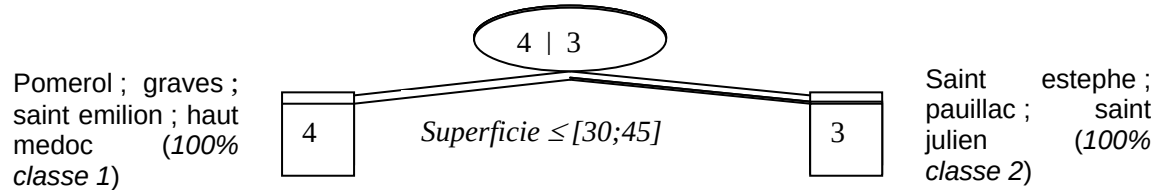


Figure 1. ***Arbre de décision***

Chaque feuille correspond à un concept. En revenant aux individus de départ, nous obtenons :
concept 1 = { *Ausone ; Cheval Blanc ; Haut-Brion ; Lafleur ; Latour ; L'Evangile ; Margaux ; Mission Haut-Brion ; Petit Village ; Petrus ; Sociando Mallet ; Trotanoy ; Vieux Château Certan*} de description [-1 ; -0.418] (*classe 1* : 100 % et *classe 2* : 0 %)
concept 2 = { *Cos d'Estournel ; Ducru-Beaucaillou ; Lafite-Rothschild ; Léoville Las Cases ; Lynch-Bages ; Montrose ; Mouton-Rothschild ; Pichon C. de Lalande ; Pichon Longueville*} de description [0.622 ; 1] (*classe 1* : 0 % et *classe 2* : 100 %).

4. Bibliographie

[ASS 98] ASSERAF, M., Extension et optimisation pour la segmentation de la distance de Kolmogorov-Smirnon , *Thèse de Doctorat, Mathématiques Appliquées, Université Paris IX* , 1998.

[BOC 00] BOCK, H. H. & DIDAY, E., *Analysis of symbolic data : Exploratory methods for extracting statistical information from complex data*, Springer-Verlag, Berlin-Heidelberg, 2000.

[BRE 84] BREIMAN L., FRIEDMAN J., OLSHEN R., STONE C., *Classification and regression trees*, Chapman & Hall, 1984.

[DID 03] DIDAY E., GIOIA F., MBALLO C., Codage qualitatif d'une variable intervalle, *XXXV^{ième} Journées de Statistique*, pages 415-418, Lyon, France, Juin 2003.

[FIS 58] FISHER, W. D., On grouping for maximum homogeneity; *Journal of the American Statistical Association*; Volume 53; pages 789-798; Decembre 1958.

[FRI 77] Friedman, J. H., A recursive partitioning decision rule for non parametric classification; IEEE Transactions on Computers, C-26, pages 404-408.

[PER 96] PERINEL, E., Segmentation et Analyse des données symboliques : Application à des données probabilistes imprécises, *Thèse de Doctorat, Mathématiques Appliquées, Université Paris IX Dauphine*, 1996.

[UTG 96] UTGOFF, P.E., CLOUSE, J.A., A Kolmogorov-Smirnov metric for decision tree induction, *University of Massachusetts, Amherst*, Number 96-3, 1996.

Détection de faibles homologies de protéines par machines à vecteurs de support

Jérôme Mikolajczak[*]**, Gérard Ramstein**[**] **et Yannick Jacques**[*]

[*] *Département de Cancérologie, Institut de Biologie 9 Quai Moncousu, F-44035 Nantes cedex*
jmikolaj@nantes.inserm.fr, yjacques@nantes.inserm.fr
[**]*LINA, équipe LEC, Ecole polytechnique de l'Université de Nantes*
Rue Christian Pauc, BP 50609 44306 Nantes cedex 3
gerard.ramstein@polytech.univ-nantes.fr

RÉSUMÉ. *Cet article décrit une approche discriminative pour la recherche de nouveaux membres dans des familles de protéines à faibles homologies de séquences. L'originalité de la méthode repose sur une modélisation de ces familles par un ensemble M de motifs intégrant les propriétés physicochimiques des résidus. Nous proposons un algorithme de découverte de motifs suivant le paradigme de la classification hiérarchique ascendante. L'ensemble M définit un espace de représentation des séquences : chaque séquence est transformée en un vecteur indiquant la présence ou l'absence de chaque motif appartenant à M. Nous utilisons la technique d'apprentissage par machine à vecteurs de support (SVM) pour discriminer la famille d'intérêt vis à vis des séquences non apparentées. Nous montrons que l'ensemble des motifs hiérarchiques modélise spécifiquement les interleukines par rapport aux autres familles structurales de la base de données SCOP.*

MOTS-CLÉS : *Classification, machines à vecteurs de support, discrimination de protéines, bio-informatique*

1. Introduction

Certaines familles de protéines sont trop hétérogènes pour qu'on puisse retrouver des régions conservées au niveau de leur structure primaire. Des méthodes d'apprentissage ont été proposées [JAA 00]. Une approche particulièrement prometteuse dans le domaine de la classification supervisée repose sur les machines à vecteurs de support [VAP 95] (ou *support vector machines*, nommées SVMs par la suite). Dans cette technique, le jeu d'apprentissage subit une transformation en un ensemble de vecteurs de taille fixe. Plusieurs espaces vectoriels ont été proposés avec des performances remarquables. Une méthode particulièrement efficace et rapide utilise des spectres de chaîne [LES 02]. Un spectre de chaîne regroupe toutes les combinaisons possibles de séquences de n caractères (ou n-gramme) à partir d'un alphabet Ω. Le spectre de chaîne d'une séquence est donc un vecteur représenté par les occurrences de ses k-sous-séquences. Il est à noter que l'espace de représentation est de haute dimension ($|\Omega|^n$ combinaisons possibles de n-grammes). La technique du spectre de chaîne est très simple à mettre en oeuvre et peu coûteuse en temps d'exécution. Les auteurs montrent que la performance de leur algorithme est comparable avec celle faisant intervenir des méthodes complexes, comme les HMMs [KAR 98]. Nos propres expérimentations sur la famille des cytokines démontrent l'efficacité de cette méthode en terme de classification. Nous proposons dans cet article d'utiliser un espace de représentation de faible dimension qui cible des propriétés spécifiques de notre famille d'intérêt. Nous allons dans un premier temps décrire le concept de motif hiérarchique, puis nous donnerons un algorithme d'extraction de ces motifs. Nous rappelerons ensuite les principes des SVMs avant de discuter des résultats obtenus sur la famille des cytokines.

2. Motifs hiérarchiques

La structure primaire d'une protéine est représentée par une séquence $s = \langle s_1 s_2 \dots s_n \rangle$ où chaque s_i appartient à Ω, l'ensemble des acides aminés : $\Omega = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$. Soit $P(\Omega)$ l'ensemble des parties de Ω. Certaines de ces parties possèdent des résidus partageant des propriétés physico-chimiques particulières. Plusieurs variantes de systèmes de classes ont été proposées ; nous avons opté pour celui de Taylor [TAY 86] présenté en table 1. La pertinence de cette classification se vérifie par l'étude des régions conservées : on observe que les mutations s'opèrent généralement au sein d'une même classe (par exemple, les acides aminés I, L et V appartenant à la classe aliphatique sont très fréquemment interchangés). Les classes physico-chimiques définissent un sous-ensemble de $P(\Omega)$, auquel nous ajoutons l'ensemble des singletons de Ω ainsi que l'ensemble Ω lui-même. Nous noterons $C(\Omega)$ l'alphabet suivant : $C(\Omega) = \{\{A\}, \{C\}, \dots, \{Y\}\} \cup \{\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \eta, \theta\} \cup \Omega$. On considérera l'ensemble ordonné $(C(\Omega), \subseteq)$ qui forme un sup-demi-treillis : toute paire (x, y) de $C(\Omega) \times C(\Omega)$ possède une borne supérieure, que l'on notera $sup(x, y)$.

Symbole	Classe	Membres
α	aliphatique	ILV
β	aromatique	$FHWY$
γ	non polaire	$ACFGHIKLMVWY$
δ	chargé	$DEHKR$
ε	polaire	$CDEHKNQRSTWY$
ζ	charge positive	HKR
η	chaîne latérale courte	$ACDGNPSTV$
θ	chaîne latérale très courte	$ACGST$

TAB. 1. *Classes d'acides aminés basées sur des propriétés physico-chimiques*

Un motif $m = \langle m_1 m_2 \dots m_k \rangle$ est une k-séquence formée d'ensembles $m_i \in C(\Omega)$. Pour la simplicité de la notation, on notera le singleton $\{R\}$ par R directement ; le motif $K\alpha$ désignera ainsi le motif composé de la classe $\{K\}$ suivie de la classe $\{I, L, V\}$. Nous appelerons *occurrence* d'un motif m une sous-séquence $\langle s_{i+1} s_{i+2} \dots s_{i+k} \rangle$ de s telle que $s_{i+j} \in m_j \ \forall j,\ 1 \leq j \leq k$. On dira que la séquence s ***vérifie*** le motif m. Le ***support*** d'un motif m dans un jeu de séquences $\mathcal{S}$ est le nombre de séquences de $\mathcal{S}$ qui vérifie m. La séquence MH vérifie ainsi 21 motifs de taille 2, dont les motifs MH, $M\beta$, $\gamma\delta$, et $\Omega\Omega$. Le motif MH ne peut être vérifié que par une seule sous-séquence, tandis que le motif $\Omega\Omega$ est vérifié pour n'importe quelle séquence de taille supérieure ou égale à 2. Il importe donc de qualifier la spécificité d'un motif en prenant en compte la probabilité de le voir apparaître dans une séquence. Comme l'estimation précise de cette probabilité est complexe et inutile pour notre classifieur, nous avons opté pour la fonction de coût suivante : $c(m) = \prod_{i=1}^{k} f(m_i)$ où $f(m_i)$ est la fréquence de la classe m_i dans une base d'apprentissage comprenant de nombreuses familles de protéines différentes. La spécificité d'un motif m sera définie par $\phi(m) = -\log(c(m))$.

Les mesures que nous avons effectuées montrent une bonne corrélation entre l'estimation $\phi(m)$ et le support effectif de m dans la base de contre-exemples de séquences issues de SCOP [MUR 95].

Les motifs peuvent être hiérarchisés selon une relation de généralisation. Soit deux k-motifs m^1 et m^2. Nous noterons $\preceq$ la relation d'ordre suivante : $m^1 \preceq m^2$ ssi pour tout $i \in [1, k]$ on a $m_i^1 \subseteq m_i^2$. L'estimateur $\phi(m)$ est construit de sorte que $\phi(m^1) \geq \phi(m^2)$ pour toute paire de motifs vérifiant $m^1 \preceq m^2$. En spécialisant un motif, on augmente sa spécificité. Nous appellerons borne supérieure des motifs m^1 et m^2 (notée $sup(m^1, m^2)$) le motif $m^{1,2}$ vérifiant : $m_i^{1,2} = sup(m_i^1, m_i^2)$ pour tout $i \in [1, k]$. Le motif $m^{1,2}$ représente le motif le plus spécifique qui généralise m^1 et m^2 : toute sous-séquence vérifiant m^1 ou m^2 vérifiera $m^{1,2}$. La section suivante présente comment extraire les motifs de spécificité minimale.

3. Découverte de motifs hiérarchiques

La recherche de motifs hiérarchiques procède en deux étapes, à savoir l'extraction de motifs germes et la génération des motifs hiérarchiques. La première étape consiste à extraire des motifs germes à partir de la famille $\mathcal{S}$. Un motif germe est un motif qui ne possède pas de minorants. Plus pratiquement, les motifs germes sont les motifs formés uniquement de classes singletons (résidus). Une façon triviale d'obtenir la liste des motifs germes est de relever l'ensemble des k-sous-séquences présentes dans le jeu d'apprentissage. La seconde étape opère un appariement des motifs pour déterminer leurs bornes supérieures. L'algorithme de découverte de motifs s'inspire de la technique de la classification hiérarchique ascendante pour former des clusters de motifs généraux à partir de motifs germes composés uniquement de singletons. L'algorithme décrit ci-dessous permet d'extraire les n motifs les plus intéressants pour caractériser une famille de protéines.

algorithme découverteMotifs
entrées
M, l'ensemble de motifs germes obtenus lors de l'étape 1
$supMin$, le seuil de support minimal recherché
$speMin$, le seuil de spécificité minimale recherchée
sortie
E, l'ensemble de motifs hiérarchiques

$E = \{m \in M \mid support(m) \geq supMin$ et $\phi(m) \geq speMin\}$;
Répéter
Soit m^1 et m^2 la paire de motifs de M telle que :
1. $m^{1,2} = sup(m^1, m^2)$
2. $\phi(m^{1,2}) \geq \phi(m^{i,j})$ pour tout m^i et m^j dans M
$M \leftarrow M - \{m^1, m^2\}$;
$M \leftarrow M \cup \{m^{1,2}\}$;
si $support(m^{1,2}) \geq supMin$ et $\phi(m^{1,2}) \geq speMin$
alors $E \leftarrow E \cup \{m^{1,2}\}$;
jusqu'à $cardinal(M) = 1$ ou $\phi(m^{1,2}) < speMin$;

4. Application à la superfamille des cytokines

Les SVMs ont démontré leur efficacité dans la détection d'homologies éloignées. Les différents méthodes utilisées mettent en évidence l'importance de l'étape de vectorisation des exemples dans la performance de ce type de classifieur. La classification des protéines passe par une première étape de vectorisation suivie de la prédiction proprement dite par SVM. Nous allons transformer une séquence quelconque en un vecteur booléen de dimension n. L'élément de rang i est à vrai ssi le motif de rang i est présent dans la séquence. Il est à noter que cette vectorisation s'effectue en $O(N)$, où N désigne la taille de la séquence. Si l'apprentissage est relativement complexe, la classification est peu coûteuse en temps d'exécution.

Nous avons retenu 45 séquences primaires relatives à la famille des cytokines chez l'homme. La base d'apprentissage qui nous a servi à l'estimation de la spécificité est issue de la base de données SCOP (Structural Classification Of Proteins, [MUR 95]). Les séquences de SCOP forment un échantillon qui recouvre un large spectre de protéines. Après suppression des interleukines, notre base de test comporte 6615 séquences. La table 2 présente les résultats obtenus avec différents classifieurs. Les valeurs présentées sont des moyennes de performances obtenues par la technique de *leave-one-out* (pour laquelle une séquence sert de test et les autres pour l'apprentissage). La ligne KNN présente les résultats obtenus avec la méthode des k plus proches voisins ($k = 3$). La notion de voisinage se réfère à la proximité entre spectres de chaîne : la mesure de similarité utilisée est le produit cartésien des vecteurs normalisés. Les SVMs à base de spectre de chaînes donnent de meilleurs résultats (ligne SCSVM) que

les KNNs, ce qui confirme l'intérêt des machines à vecteurs de support. Les résultats sont identiques pour les fonctions noyaux linéaires et à bases radiales ; la seule différence consiste en une légère amélioration des performances sur la base SCOP pour la fonction à base radiale. Notre méthode MotifsSVM surpasse largement la technique à base de spectre de chaîne (100% de bonne classification) à condition de bien optimiser le type des motifs. Un seuil de spécificité de 14 apparaît comme le meilleur compromis ; au-delà de cette valeur, on ne découvre pas assez de motifs sur certaines cytokines, en deçà, les motifs ne sont assez sélectifs.

classifieur	taux d'erreurs	VP	FN	VN	FP
KNN	18.9	88.9	11.1	73.3	26.7
SCSVM linéaire	13.3	84.4	15.6	88.9	11.1
SCSVM RBF	13.3	84.4	15.6	88.9	11.1
MotifsSVM 13	2.2	95.6	4.4	100	0
MotifsSVM 14	0	100	0	100	0
MotifsSVM 15	5.5	88.9	11.1	100	0

TAB. 2. *Résultats de classification. VP, FN, VN, FP sont les pourcentages respectivement des vrais positifs, faux négatifs, vrais négatifs, faux positifs.*

Les tests opérés sur l'ensemble de séquences négatives de SCOP mettent en évidence le pouvoir discriminant de notre classifieur. Sur les 6615 séquences de la base SCOP, le classifieur MotifsSVM en a en effet mal classé 8. Ce taux d'erreurs de 0.12% est supérieur au meilleur taux obtenu par SCSVM, qui est de 4.08%.

5. Conclusion et perspectives

Les excellents résultats obtenus en classification dans la famille des cytokines démontrent la pertinence d'une description hiérarchique des motifs. Ces derniers assurent un rôle de signature, au sens où ils sont spécifiques à la famille étudiée. Nous avons proposé un paramétrage simple du degré de spécificité souhaité et avons observé qu'une valeur moyennement haute donne les meilleurs performances ($\phi(m) \sim 14$). La capacité des SVMs à gérer des espaces de grande dimension nous permet d'obtenir une classification sans erreurs sur les exemples positifs et un très faible taux d'erreurs sur les exemples négatifs issus de SCOP (0.12%). Ce faible pourcentage de faux-positifs autorise l'emploi de MotifsSVM pour rechercher de nouveaux membres de familles protéiques dans le génome. Certaines améliorations de l'algorithme d'extraction restent à mettre en oeuvre, afin notamment de réduire le nombre de motifs retenus.

6. Bibliographie

[JAA 00] JAAKOLA T., DIEKHANS M., HAUSSLER D., A discriminative framework for detecting remote protein homologies, *Journal of Computationnal Biology*, vol. 7, n^o 1-2, 2000, p. 95-114.

[KAR 98] KARPLUS K., BARRET C., HUGLEY R., Hidden Markov Models for detecting remote protein homologies, *Bioinformatics*, vol. 14, 1998, p. 846-856.

[LES 02] LESLIE C., ESKIN E., NOBLE W. S., The spectrum kernel : a string kernel for SVM protein classification, *Proceedings of the Pacific Biocomputing Symposium*, 2002, p. 564-575.

[MUR 95] MURZIN A., S.E.BRENNER, HUBBARD T., CHOTHIA C., SCOP : a structural classification of proteins database for the investigation of sequences and structures, *Journal of Molecular Biology*, vol. 247, 1995, p. 536-540.

[TAY 86] TAYLOR J., Classification of amino acid conservation, *Theorical Biology*, vol. 119, 1986, p. 205-218.

[VAP 95] VAPNIK V. N., *The nature of statistical learning theory*, Springer-Verlag, 1995.

Incorporation de rotations procrustéennes dans une analyse factorielle multiple

Elisabeth Morand & Jérôme Pagès

Agrocampus Rennes
Laboratoire de mathématiques appliquées
CS 84215
35042 Rennes cedex

RÉSUMÉ. Pour comparer deux nuages de points homologues, la méthode de référence est l'analyse procrustéenne et, dans le cas de plus de deux nuages, l'analyse procrustéenne généralisée (GPA). L'analyse factorielle multiple (AFM) fournit aussi une représentation superposée de nuages de points homologues. Cette dernière représentation bénéficie, par rapport à celle issue de la GPA, d'avantages (elle s'inscrit dans le cadre d'une analyse factorielle riche en aides à l'interprétation) et d'inconvénients (les nuages à comparer subissent des déformations autres que les seules projections et rotations). Il est possible de compléter l'AFM par un ajustement procrustéen de chacun des nuages initiaux sur le nuage moyen de l'AFM. On obtient ainsi une représentation de ces nuages qui à la fois respecte le modèle procrustéen et s'inscrit dans le cadre de l'AFM. D'où le nom d'analyse factorielle multiple procustéenne (AFMP). Nous présentons ici quelques unes de ses propriétés. Cette nouvelle représentation est précieuse lorsque les nuages initiaux sont bidimensionnels. Une application dans ce cas particulier est présentée.

MOTS-CLÉS : Analyse Procrustéenne Généralisée, Analyse Factorielle Multiple, Analyse Factorielle Multiple Procrustéenne.

1. Données et notations

Les données sont constituées d'un ensemble d'individus, $\{i\ ;\ i=1, I\}$, décrits par plusieurs groupes de variables. Ces données peuvent être regroupées sous forme d'un tableau unique structuré en sous-tableaux. On note (figure 1) :

X le tableau complet ;
K l'ensemble des variables ;
J l'ensemble des sous-tableaux ;
K_j l'ensemble des variables du groupe j ;
X_j le tableau associé au groupe j.

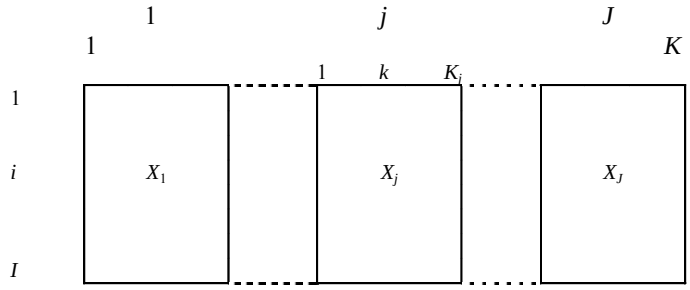


Figure 1. Structure des données.

Au tableau X correspond le nuage des individus, N_I, situé dans l'espace R^K. A chaque groupe de variables, correspond un nuage d'individus, dit partiel et noté N_I^j, situé dans un espace de dimension K_j. Si l'on plonge le nuage N_I^j dans l'espace R^K les coordonnées de chacun des individus de ce nuage se trouvent au sein du tableau, noté $\widetilde{X}_j$, de dimensions (I, K), dans lequel X_j est complété par des 0.

2. Problématique

L'étude simultanée de plusieurs tableaux présente de nombreux aspects. L'un d'entre eux est la représentation superposée des nuages partiels. L'analyse procrustéenne (dans le cas où $J = 2$) ou l'Analyse Procrustéenne Généralisée (GPA) (dans le cas où $J > 2$) est la méthode de référence pour obtenir une telle représentation. L'Analyse Factorielle Multiple (AFM) propose aussi une représentation superposée des nuages partiels. Le cœur de cette analyse est constitué par une ACP effectuée sur le tableau complet X, dont les variables sont pondérées. La pondération utilisée consiste à diviser le poids initial de chaque variable du groupe j par λ_1^j (en notant λ_1^j l'inertie projetée sur le premier axe de l'analyse séparée du groupe j). On obtient ainsi une représentation du nuage N_I , comme dans toute ACP. A cette représentation, on superpose les nuages N_I^j en introduisant les tableaux $\widetilde{X}_j$ en supplémentaires dans l'ACP du tableau complet X. Cette représentation présente quelques propriétés intéressantes, en particulier :

- elle s'inscrit dans une méthode générale qui fournit de nombreux points de vue sur l'analyse simultanée de plusieurs tableaux en particulier de nombreuses aides à l'interprétation ;
- il existe pour cette représentation des relations de transition dites partielles (détaillées ci-après).

La coordonnée, sur l'axe principal de rang s, de l'individu i vu par le groupe j, notée $F_s(i^j)$, s'exprime comme combinaison linéaire des coordonnées des seules variables du groupe j sur ce même axe. Ceci se traduit par la formule de transition suivante, dans laquelle on reconnaît la restriction au groupe j de la relation de transition classique :

$$F_s(i^j) = F_s^j(i) = \frac{1}{\sqrt{\lambda_s}} \frac{1}{\sqrt{\lambda_1^j}} \sum_{k \in K_j} x_{ik} G_s(k)$$

en notant :

$F_s(i^j)$projection de l'individu i^j sur l'axe principal de rang s du nuage des individus N_I ;
$G_s(k)$ projection sur l'axe de rang s de la variable k ;
$\lambda_s$est l'inertie projetée du nuage N_I .

En contrepartie de cette précieuse propriété, cette représentation présente des déformations autres que celles induites par les projections. En effet, chaque nuage partiel est projeté sur deux axes n'appartenant pas à son sous-espace initial (figure 2).

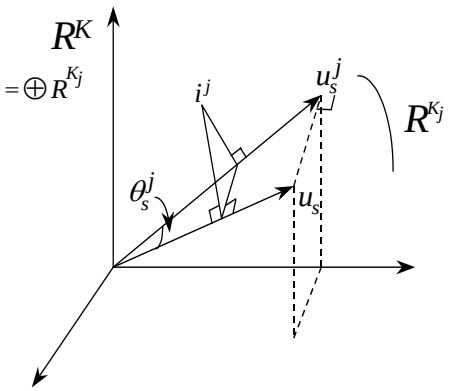


Figure 2. *Projection de l'individu i du nuage j dans N_I*

u_s : axe principal de rang s du nuage des individus
u_s^j: composante de u_s dans l'espace du nuage partiel j.

i^j, $i^{ème}$ ligne de $\widetilde{X}_j$ et appartenant à R^{K_j} , est dans un premier temps projeté sur u_s^j puis, en multipliant les coordonnées par cos(θ_s^i), sur u_s

Nous proposons ci-après une représentation procrustéenne à la fois intégrée à l'AFM, d'où la dénomination d'analyse factorielle multiple procrustéenne (AFMP), et telle que les nuages initiaux ne subissent aucune autre déformation que celles résultant des projections.
Toutefois, il est à noter que dans le cas où les nuages partiels sont dans un espace à plus de deux dimensions, la représentation procrustéenne, bien que non déformée dans l'espace total, le sera lors de sa représentation finale en deux dimensions, par projection sur un espace de dimension inférieure.

3. Méthode

3.1. *Principe*

Chaque nuage partiel N_I^j est ajusté, à l'aide d'une rotation procrustéenne, sur le nuage moyen N_I de l'AFM.
Du point de vue de l'algorithme, on utilise les *S* premières composantes de la représentation moyenne de l'AFM. Le choix de la dimension, *S*, est à discuter suivant les cas. Le tableau ainsi obtenu, de dimension (*I*, *S*), est noté $\widetilde{F}$. On considère ensuite, pour chaque groupe *j*, les tableaux X_j pondérés comme en AFM, soit les tableaux $\frac{1}{\sqrt{\lambda_1^j}} X_j$. Nous cherchons à les rapprocher de $\widetilde{F}$ par une rotation procrustéenne de $\frac{1}{\sqrt{\lambda_1^j}} X_j$ sur $\widetilde{F}$. Cela revient donc à chercher une transformation géométrique T_j telle que $Trace\left[\left(\widetilde{F} - \frac{1}{\sqrt{\lambda_1^j}} X_j T_j\right)\left(\widetilde{F} - \frac{1}{\sqrt{\lambda_1^j}} X_j T_j\right)'\right]$ soit minimum et ce sous la contrainte $T_j T_j' = I$.

On obtient alors les tableaux « procrustéanisés », $\hat{X}_j$ par les relations suivantes.

$$\hat{X}_j = \frac{1}{\sqrt{\lambda_1^j}} X_j T_j$$

Où $T_j = V_j U_j'$ avec :

V_j la matrice orthogonale des vecteurs propres normés de la matrice : $\frac{1}{\lambda_1^j} X_j' \widetilde{F} \widetilde{F}' X_j$

U_j la matrice orthogonale des vecteurs propres normés de la matrice : $\frac{1}{\lambda_1^j} \widetilde{F}' X_j X_j' \widetilde{F}$

Ce calcul revient à effectuer la dernière boucle de l'algorithme de Gower (1975) en prenant comme consensus le nuage moyen issu de l'AFM.

Remarque : Nous avons pris ici l'hypothèse que tous les sous-tableaux étaient de même dimension (K_j constant). C'est ce cas particulier, avec $K_j = S = 2$, qui a suscité l'AFMP et dont on a évalué l'impact pratique.
Toutefois, pour les cas où K_j ne serait pas constant d'autres stratégies sont envisageables et méritent d'être examinées comme :

- prendre K'=max K_j et compléter les tableaux, de dimension inférieure à K', par K'-Kj colonnes de 0 ;
- prende K_j composantes principales du consensus ; on ajuste alors chaque configuration individuelle sur les S=K_j premières composantes de la représentation moyenne de l'AFM, la dimension S étant différente d'une configuration individuelle à l'autre ;
- se limiter à un nombre fixe de composantes principales par tableaux .

3.2. *Propriétés*

Pour la représentation des nuages partiels ainsi obtenue, il n'y a plus de relations de transition partielles. En contrepartie, les nuages partiels n'ont subi aucune déformation autre que les rotations orthogonales.

Remarque. Cette représentation est particulièrement intéressante dans le cas bidimensionnel, puisque la représentation des nuages partiels n'est absolument pas déformée.

4. Exemple

On présente ici un exemple dans le cadre bidimensionnel pour illustrer l'intérêt de la nouvelle méthode par rapport à la représentation superposée usuelle de l'AFM.

4.1. *Données*

On a demandé à 11 dégustateurs de fournir chacun une représentation euclidienne de 10 vins blancs de Val de Loire (5 Chenins, numérotés de 1 à 5, et 5 Sauvignons, numérotés de 6 à 10) en les positionnant sur une nappe. On dispose alors de J=11 représentations euclidiennes. Ainsi, des vins proches sur une nappe sont des vins qui paraissent similaires au juge. Les données analysées sont les coordonnées $X_j(i)$ et $Y_j(i)$ de chaque vin i, mesurées sur la nappe du juge j.

		Juge1			Juge j			Juge 11	
		X_1	Y_1		X_j	Y_j		X_{11}	Y_{11}
	1								
vins	i	$X_1(i)$	$Y_1(i)$		$X_j(i)$	$Y_j(i)$		$X_{11}(i)$	$Y_{11}(i)$
	I=10								

4.2. *Résultats*

A titre d'exemple, nous représentons ici la nappe fournie par le juge 9. Remarquons en particulier sur cette nappe les vins 7 et 10 relativement excentrés.

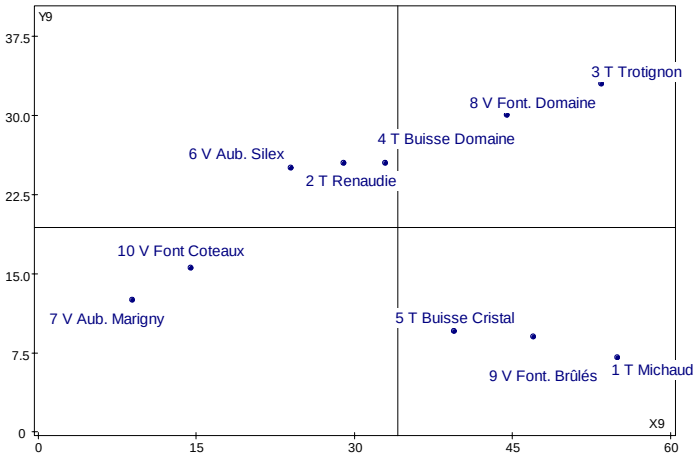


Figure 3 : *Représentation des 10 vins par le juge 9*

Ces données sont traitées par une AFM dans laquelle chaque nappe constitue un groupe de deux variables non réduites.

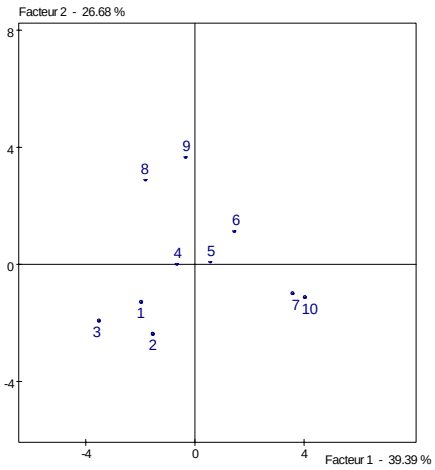


Figure 4a. *Représentation du nuage moyen de l'AFM*

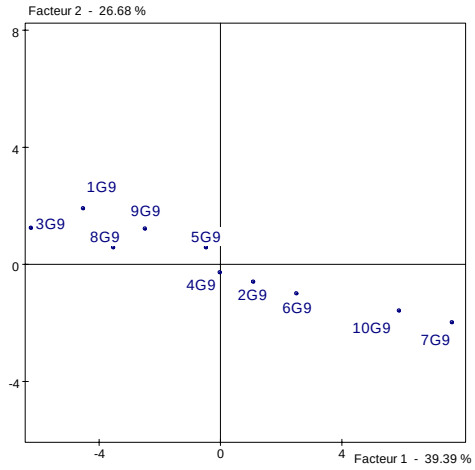


Figure 4b. *Représentation du nuage partiel du juge 9 pour l'AFM*

Sur la représentation moyenne des individus ainsi obtenue (figure 4a) il est commode d'orienter les commentaires selon les deux bissectrices. La première bissectrice sépare les Sauvignons (vins 1 à 5) des Chenins (vins 6 à 10). La seconde bissectrice sépare les 5 Chenins entre eux en mettant en évidence la particularité des vins 7 et 10.
La représentation du nuage partiel du juge 9 (figure 4b) est étirée essentiellement le long de la seconde bissectrice. Cette représentation ne distingue pas les Sauvignons des Chenins. On retrouve comme dans le nuage moyen la variabilité des Chenins où les vins 8 et 9 s'opposent aux vins 7 et 10. Toutefois cette représentation partielle est relativement différente de la nappe du juge 9.

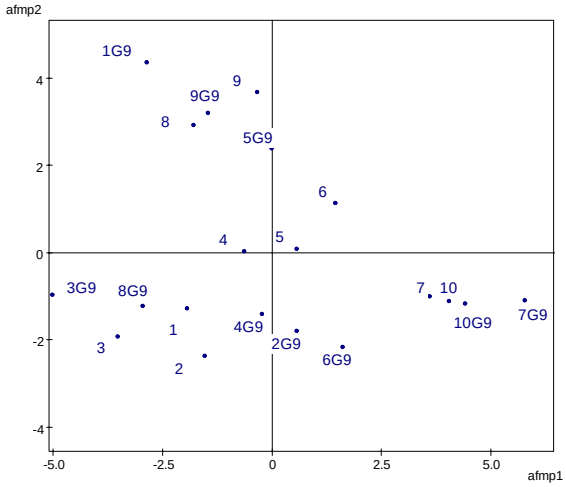


Figure 5. *Représentation superposée du nuage moyen et du nuage partiel du juge 9 en AFMP*

Sur ces mêmes données, une AFMP a été appliquée en utilisant les deux premières dimensions du nuage moyen. Ici encore, nous limitons le commentaire au seul juge 9 (figure 3). Par rapport à la représentation précédente, celle-ci est exactement la configuration « fournie » par le juge 9. Cette représentation (voir figure 5) met en évidence, parallèlement à la représentation moyenne, le caractère particulier des vins 7 et 10 et leur opposition au sein des Chenins aux vins 8 et 9. En revanche, le juge 9 a clairement séparé le vin 8 et le vin 9 alors que dans la configuration moyenne ces vins sont regroupés.
Les distinctions faites par le juge 9, par exemple l'opposition {1,9,5}↔ {3,8,4}, apparaissent par construction dans l'AFMP mais n'apparaissent pas dans l'AFM.

5. Conclusion

La représentation superposée de nuages de points homologues décrite ici est un compromis entre les représentations usuelles de la GPA et de l'AFM. Elle est particulièrement précieuse dans le cas bidimensionnel. En effet dans ce cas précis les nuages partiels ne sont absolument pas déformés. Ceci permet donc un enrichissement de la représentation moyenne de l'AFM par une représentation des nuages partiels contenant toute la spécificité de la configuration partielle initiale.

6. Bibliographie

[ESC 98] ESCOFIER B., PAGES J., *Analyses factorielles simples et multiples ; objectifs méthodes et interprétation*, 284p, Dunod, Paris, 1998.

[GOW 75] GOWER, J.C "Generalized Procrustes Analysis", *Psychometrika*, vol.40, n°1 ,1975, p. 33-51,1975.

Mesure de structuration d'un système de classes

Christophe Osswald

Laboratoire E^3I^2, ENSIETA
2 rue François Verny 29806 Brest Cedex 9
Christophe.Osswald@ensieta.fr

RÉSUMÉ. *L'objectif de ce travail est de cr´eer une mesure du niveau de structure d'un système de classes ou d'une dissimilarit´e, sans approximation sur les donn´ees. Pour cela, nous utilisons la notion de graphe de rigidit´e minimum d'un système de classes, lequel a au moins $n-1$ arêtes, ce minimum ´etant atteint lorsque ce système de classes est une hi´erarchie, une pseudo-hi´erarchie ou plus g´en´eralement un système de classes arbor´e.*
La d´etermination du nombre d'arêtes d'un graphe de rigidit´e minimum est NP-difficile dans le cas g´en´eral, et polynomial si le système de classes est ferm´e ou binaire. Nous utilisons la r´ealisation d'une dissimilarit´e pour obtenir un système de classes binaire, et calculer le nombre d'arêtes associ´e. Enfin, ce nombre est ´evalu´e pour des donn´ees al´eatoires, et nous le comparons à des jeux de donn´ees usuels.

MOTS-CLÉS : *classification, graphes de rigidit´e, r´ealisations binaires, donn´ees al´eatoires.*

Introduction

La recherche d'une structure sous-jacente à un système de classes, ou à un ensemble d'objets décrits par une dissimilarité est l'un des objectifs poursuivis par Flament *et al.* dans le cadre de l'analyse de la similitude [FLA 76] [FLA 79], dans le but de résumer un système de classes. Cette structure sous-jacente se définit par la recherche d'un graphe (Osswald [OSS 03]) ou d'un hypergraphe de rigidité, sur lequel viennent s'accrocher les classes. Les systèmes de classes usuels – partitions, hiérarchies, pseudo-hiérarchies, systèmes de classes arborés – admettent tous des graphes de rigidité qui sont des arbres.

Hansen *et al.* [HAN 94] et Guénoche et Garreta [GUÉ 02] proposent plusieurs mesures d'homogénéité et de séparation sur les classes. Il est possible d'étendre ces indices sur l'ensemble des classes pour obtenir des indices globaux d'adéquation du système de classes avec l'objectif du Vicomte de Buffon : "mettre ensemble ce qui se ressemble, séparer ce qui diffère."

Dans le cadre de l'analyse de la similitude, ou lorsque l'on ne désire pas faire d'approximation sur la dissimilarité d décrivant les données, il est classique d'engendrer un système de classes $\mathcal{K}_d$ formé des cliques maximales des graphes-seuils de la dissimilarité (Jardine et Sibson [JAR 71]), que l'on appelle *classes* de la dissimilarité. D'autres méthodes engendrent un sytème de classes : boules, 2-boules ou réalisations d'une dissimilarité (Brucker [BRU 03a]).

C'est dans ce cadre que nous proposons d'utiliser la notion de graphe de rigidité sur les systèmes de classes induits par les données pour obtenir une mesure de la *structuration* de ces données.

1. Recherche d'une structure sous-jacente

Un graphe $G = (X, E)$ est un *graphe de rigidité* du système de classes $\mathcal{K}$ sur X si pour toute classe C de $\mathcal{K}$, la restriction de G à C est un graphe connexe. Il est clair que le graphe complet K_n sur X, à n éléments est un

graphe de rigidité pour tout système de classes $\mathcal{K}$. Admettre un graphe de rigidité qui soit un arbre caractérise les systèmes de classes qui sont des *hyperarbres*, et admettre un graphe de rigidité qui soit une chaîne caractérise les *prépseudo-hiérarchies* (Durand et Fichet [DUR 88]). Un graphe $G' = (X, E')$ est un *graphe de rigidité minimum* de $\mathcal{K}$ si $\mathcal{K}$ n'admet pas de graphe de rigidité ayant moins d'arêtes que G'. On dit alors que la structuration $s(\mathcal{K})$ de $\mathcal{K}$ est $|E'|$.

Déterminer $s(\mathcal{K})$ est NP-difficile dans le cas général (Brucker *et al.*, [BRU 03b]), et l'est également si $\mathcal{K}$ est construit comme ensemble des boules, des 2-boules ou des classes d'une dissimilarité (Osswald, [OSS 03]). Notons que si $\mathcal{K}$ est composé des classes d'une dissimilarité, il n'est pas garanti que le système de classes obtenu soit borné polynomialement : un graphe peut voir $\mathcal{O}(3^{\frac{n}{3}})$ cliques maximales et il n'est alors pas "possible" de lire la donnée du problème, fut-ce pour l'utiliser au sein d'une heuristique de calcul de $s(\mathcal{K})$.

La *réalisation* de deux éléments x et y d'une dissimilarité d est l'intersection de toutes les classes de d contenant à la fois x et y. C'est aussi l'intersection de toutes les boules de rayon au moins $d(x, y)$ qui contiennent à la fois x et y. La *réalisation* $\mathcal{R}_d$ d'une dissimilarité, ensemble des réalisations des paires d'éléments, se calcule donc en $\mathcal{O}(n^4)$ opérations (Brucker [BRU 03a]). Le système de classes obtenu n'est pas fermé, mais simplement *binaire* (Barthélemy [BAR 03]) : l'intersection de toutes les classes contenant deux éléments est une classe. Il est ainsi possible de calculer $s(\mathcal{R}_d)$ en $\mathcal{O}(n^4)$ opérations.

Si d est une *quasi-ultramétrique* (ses 2-boules sont ses classes), on a $\mathcal{R}_d = \mathcal{K}_d$. Si d est une dissimilarité de Robinson [ROB 51], $\mathcal{K}_d$ et $\overline{\mathcal{K}_d} = \mathcal{R}_d$ admettent les mêmes graphes de rigidité, à $n-1$ arêtes. Enfin, si K_d est un hyperarbre, $\mathcal{R}_d$ l'est également. Plus généralement, comme X est une classe de tout système de classes, le minimum de $s(\mathcal{R}_d)$ est $n-1$, et ce minimum est atteint notamment pour les ultramétriques, les dissimilarités de Robinson et les dissimilarités arborées.

2. Données aléatoires

Afin d'interpréter le nombre $s(\mathcal{R}_d)$, nous avons évalué son comportement sur des données aléatoires, et calculé $p(n)$ la proportion d'arêtes des graphes de rigidité minimum de $\mathcal{R}_d$ pour diverses dissimilarités aléatoires, et étudié certains cas pathologiques qui semblent constituer des maxima.

Nous définissons $p(n)$ pour une forme de dissimilarité aléatoire sur un ensemble à n éléments comme :

$$p(n) = \frac{2}{n(n-1)} E(\mathcal{R}_d(n))$$

La figure 1 représente les évalutions de $p(n)$ pour quatre modèles de distributions aléatoires, et deux cas extrêmes :

- D_2 est une dissimilarité qui peut prendre deux valeurs, 1 et 2, chacune avec une probabilité $\frac{1}{2}$. D_2 est donc une dissimilarité graphique.
- D_3 est une dissimilarité qui peut prendre trois valeurs, de façon équiprobable.
- D_{20} est une dissimilarité qui peut prendre vingt valeurs, de façon équiprobable.
- D_∞ est une dissimilarité dont toutes les valeurs sont différentes.
- d_a est une dissimilarité dont la réalisation est un hyperarbre ; d_a peut être une ultramétrique.
- d_m est une dissimilarité qui vaut 2 pour (x_{2i-1}, x_{2i}) et 1 partout ailleurs. Pour n pair, $s(\mathcal{R}_{d_m}) = \frac{n(n-2)}{2}$.

La dissimilarité d_m maximise la grandeur $s(\mathcal{R})$ à n fixé. Elle est semblable à la dissimilarité d_c qui maximise le nombre de classes pour une dissimilarité graphique (Capobianco et Molluzzo, [CAP 78]) : $d_c(x_{3i-2}, x_{3i-1}) = 2$, $d_c(x_{3i-2}, x_{3i}) = 2$, $d_c(x_{3i-1}, x_{3i}) = 2$ et d_c vaut 1 partout ailleurs.

Pour les dissimilarités D_k, k valant 2 ou 3, $s(D_k)$ converge rapidement vers $1 - \frac{1}{k}$ lorsque n grandit, ce qui correspond à la proportion d'arêtes dans le graphe-seuil G_{k-1} : presque toutes les arêtes qui correspondent à des valeurs de la dissimilarité qui ne sont pas maximales apparaissent dans les graphes de rigidité minimum de $\mathcal{R}_{D_k}$. Pour les valeurs de k plus importantes, la convergence est moins rapide.

Il ne semble pas que $s(D_\infty)$ tende vers 1 aussi vite que $s(d_m)$.

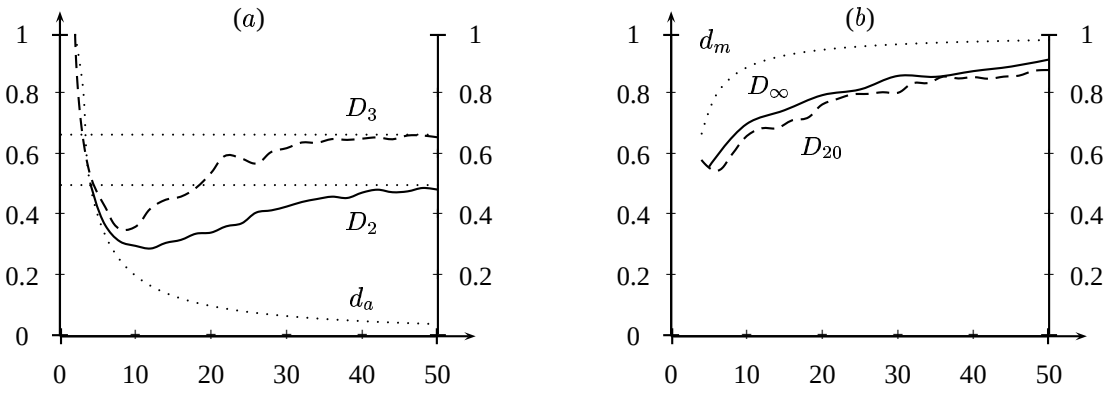


FIG.1. *Une évaluation de $p(n)$ pour cinq dissimilarités*

Nous évaluons $p(n)$ pour les dissimilarités graphiques dont les valeurs sont engendrées selon une loi de Bernoulli de paramètre b : D_2^b vaut 1 avec une probabilité b et vaut 2 avec une probabilité $1-b$:

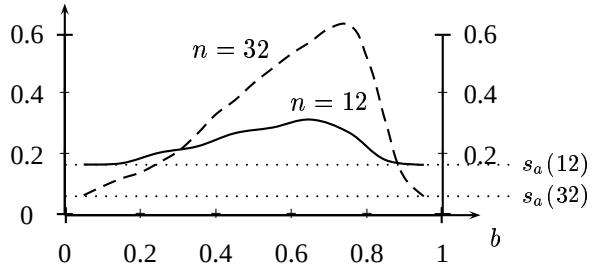


FIG.2. *Une évaluation de $p(12)$ et $p(32)$ pour les dissimilarités D_2^b*

3. Mesure de structuration

Les données les plus structurées sont donc les dissimilarités arboricoles, qui contiennent les dissimilarités de Robinson et les distances d'arbre, pour lesquelles nous avons $s_a(n) = n-1$. Nous disposons d'une mesure $s_\infty(n)$ qui correspond au nombre d'arêtes d'un graphe de rigidité de la réalisation d'une dissimilarité aléatoire ayant toutes ses valeurs deux à deux différentes, ou $s_k(n)$ pour les dissimilarités à k valeurs.

Nous mesurons la structuration S d'une dissimilarité d par interpolation linéaire entre $s_a(n)$ et $s_\infty(n)$ lorsque le nombre de valeurs différentes de d est grand devant n :

$$S(d) = \frac{s(\mathcal{R}_d) - s_a(n)}{s_\infty(n) - s_a(n)}$$

Ainsi $S(d)$ vaut 0 pour des données très structurées, et s'approche de 1 pour des données presque aléatoires.

4. Données de Henley et données textuelles

Les données de Henley [HEN 69] sont issues d'une expérience de rappel libre. Il a été aux sujets d'écrire tous les animaux auxquels ils pouvaient penser en un temps donné. La dissimilarité globale entre deux animaux est la distance moyenne qui les sépare dans ces listes. Ces données ont été abondamment étudiées, par exemple par Barthélemy et Guénoche [BAR 88] et sont réputées résistantes aux méthodes usuelles de classification.

Ours	0	47,2	27,7	40,1	49,6	19,1	29,0	22,6	29,5	21,4	20,3	16,1
Chat		0	30,9	56,1	02,0	29,0	25,3	24,1	24,8	43,0	41,5	47,1
Vache			0	43,6	30,2	11,0	07,7	24,5	34,1	17,0	27,9	08,2
Cerf				0	50,9	44,5	43,0	44,7	39,9	41,1	19,9	53,1
Chien					0	17,0	24,0	26,9	27,5	45,0	39,4	46,8
Ch`evre						0	07,2	23,1	39,6	19,5	21,8	01,8
Cheval							0	28,6	32,6	25,7	30,1	15,2
Lion								0	33,2	29,3	33,3	35,0
Souris									0	34,9	22,6	51,9
Cochon										0	25,9	19,6
Lapin											0	32,5
Mouton												0

FIG.3. *Données de Henley : dissimilarité d_H*

Les graphes de rigidité minimum de la réalisation de d_H ont 34 arêtes, correspondant exactement aux classes à deux éléments de cette réalisation : le graphe de rigidité minimum G_H de la réalisation de d_H est unique. Un arbre sur cet ensemble d'objets a 11 arêtes, et les graphes de rigidité minimum des réalisations d'une dissimilarité aléatoire ont en moyenne 46.7 arêtes – il n'y a que cinq valeurs de la dissimilarité d_H qui sont doublées. Ainsi $s(d_H) = 0,64$. Notons que le Cerf est une feuille de G_H, et est donc porteur d'une part importante de la structuration du jeu de données. La dissimilarité d'_H correspondant à d_H privée du Cerf a un graphe de rigidité minimum à 32 arêtes, ce qui correspond à une structuration de $0,77$. La dissimilarité d_m, qui maximise $s(\mathcal{R}_d)$, a une structuration de $1,37$.

L'indice de connexion lexicale d_G entre 23 œuvres de Jean Giraudoux, obtenu par Brunet [BRU 88] utilise les fréquences d'apparition des mots dans les textes. Les graphes de rigidité minimum des réalisations de d_G ont 32 arêtes, les graphes de rigidité d'une dissimilarité aléatoire sur 23 objets en ont en moyenne 204. Ainsi, la structuration de d_G est $s(d_G) = 0,055$.

5. Bibliographie

[ROB 51] W. S. Robinson, A method for chronologically ordering archeological deposits, vol. 16, 1951, , p. 295-301.

[HEN 69] N. M. Henley, A Psychological Study of the Semantics of Animal Terms, vol. 8, 1969, , p. 176-184.

[JAR 71] N. Jardine and R. Sibson, *Mathematical Taxonomy*, Wiley, London, 1971, part II.

[FLA 76] C. Flament, Hypergraphes et analyse de donn´ees, S´eminaire INRIA, 1976.

[CAP 78] M. Capobianco and J. C. Molluzzo, *Examples and counterexamples in graph theory*, North-Holland, 1978.

[FLA 79] C. Flament and A. Degenne and P. Verg`es, Analyse de similtude ordinale, vol. 40-41, 1979, , p. 223-231.

[DUR 88] C. Durand and B. Fichet, One to one correspondances in pyramidal representation : an unifi ed approach, , p. 85-90, North-Holland, Amsterdam, 1988.

[BAR 88] J.-P. Barthel´emy and A. Gu´enoche, *Les arbres et les repr´esentations de proximit´e*, Masson, Paris, 1988.

[BRU 88] E. Brunet, Une mesure de la distance intertextuelle : la connexion lexicale, vol. 24, 1988, , p. 81-116.

[HAN 94] P. Hansen and B. Jaumard and E. Sanlaville, Partitionning problems in cluster analysis : a review of mathematical approaches, , p. 228-240, Springer-Verlag, 1994.

[GUÉ 02] A. Gu´enoche and H. Garreta, Representation and evaluation of partitions, *Proceedings of IFCS'2002*, Springer, p. , 131-138.

[OSS 03] Christophe Osswald, Classifi cation, analyse de la similitude et hypergraphes, PhD thesis, Ecole des Hautes Etudes en Sciences Sociales, 2003.

[BAR 03] J.-P. Barth´elemy, Classifi cations binaires, *Rencontres de la Soci´et´e Francophone de Classification*, p. , 67-69.

[BRU 03a] F. Brucker, R´ealisations de dissimilarit´es, *Rencontres de la Soci´et´e Francophone de Classification*, p. , 7-10.

[BRU 03b] F. Brucker and C. Osswald and J.-P. Barth´elemy, Rigid hypergraphs : combinatorial optimization problem in clustering and similarity analysis, *Actes de INOC 2003*, 2003.

Méthode factorielle pour l'analyse d'un ensemble de variables quantitatives et qualitatives

Jérôme Pagès

Laboratoire de mathématiques appliquées
Agrocampus, 65, rue de Saint-Brieuc - 35042 Rennes cedex
email : jerome.pages@agrocampus-rennes.fr

RÉSUMÉ. Une méthodologie factorielle permettant d'inclure à la fois des variables quantitatives et qualitatives en tant qu'éléments actifs d'une même analyse a été proposée par B. Escofier (1979a) dans le cadre de l'analyse des correspondances multiples. De son côté, Saporta (1990) a esquissé, dans le cadre de l'analyse en composantes principales, une méthodologie ayant le même objectif. Enfin, la pratique de l'analyse factorielle multiple (AFM) suggère la possibilité de mettre en œuvre une AFM sur des données mixtes en considérant chaque variable, quantitative ou qualitative, comme un groupe d'une seule variable. On montre que ces trois approches conduisent aux même résultats. L'ensemble de ces trois points de vue confère à la méthode proposée initialement par B. Escofier le statut d'une méthode à part entière : l'Analyse Factorielle de Données Mixtes (AFDM).
Cette communication présente le principe de l'AFDM et les principaux graphiques auxquels elle conduit.

MOTS-CLÉS : Analyse ou composantes principales, analyse des correspondances multiples, analyse factorielle multiple, données mixtes.

1. Introduction

L'introduction simultanée de variables quantitatives et qualitatives (données dites mixtes) en tant qu'éléments actifs d'une même analyse factorielle est une problématique fréquente. L'intérêt de conserver telles quelles les variables quantitatives (i.e. sans les coder en qualitatives) vaut essentiellement dans deux cas :

- lorsque le nombre de variables qualitatives est très petit comparé à celui des variables quantitatives
- lorsque le nombre d'individus est faible.

Plusieurs propositions d'analyse factorielle de données mixtes ont déjà été faites. On peut citer, sans prétendre à l'exhaustivité, les travaux suivants de l'Ecole française d'Analyse des données : Tenenhaus (1977), Escofier (1979a et 2003) et Saporta (1990).
Adoptant le point de vue de l'ACM, Escofier (1979a) a proposé d'introduire des variables quantitatives (moyennant un codage approprié) dans une ACM : elle décrit plusieurs propriétés de cette méthodologie ainsi qu'une application.
Il est possible, moyennant une métrique judicieusement choisie, de réaliser une ACP sur un tableau juxtaposant des variables quantitatives réduites et des variables qualitatives codées sous forme disjonctive complète. Cette possibilité est esquissée dans Saporta (1990) sous le nom d'extension de l'ACP et de l'ACM.
Enfin, lorsque les variables constituent des groupes homogènes (i.e. les variables d'un même groupe sont de même type), une analyse factorielle multiple (AFM) peut être réalisée (Escofier & Pagès, 1998 p 173 ; Pagès, 2002).
Si l'on transpose les idées de B. Escofier (1979a) dans le cadre de l'ACP, on retrouve l'extension de Saporta (1990). En outre, cette méthode est équivalente à une AFM dans laquelle chaque groupe est réduit à une seule variable, quantitative ou qualitative.
La convergence entre ces trois points de vue (ACP, ACM et AFM) apporte une justification solide à cette méthodologie qui du coup mérite une dénomination à part entière ; nous proposons : Analyse Factorielle de Données Mixtes (AFDM). Les propriétés de l'AFDM sont étudiées en détail dans Pagès (2004), présentation qui

comporte une application sur des données réelles. Nous nous limitons ici à rappeler le principe de l'AFDM ; dans la communication orale, nous décrirons en outre une application sur des données construites pour illustrer la façon dont l'AFDM équilibre l'influence des différents types de variables.

2. Données, notations

Soient I individus notés i et munis du même poids $p_i = 1/I \ \forall i$. Ces individus sont décrits par :

- K_1 variables quantitatives $\{k = 1, K_1\}$; ces variables seront toujours supposées centrées réduites ; ceci n'est pas une commodité mais une nécessité due à la présence des deux types de variables ;
- Q variables qualitatives $\{q = 1, Q\}$; la $q^{ième}$ variable présente K_q modalités $\{k_q = 1, K_q\}$; l'ensemble des modalités a pour cardinal $\Sigma_q K_q = K_2$.

Soit $K = K_1 + K_2$ le nombre total de variables quantitatives et de variables indicatrices.
Ces notations sont rassemblées dans le tableau de la figure 1 dans lequel les variables qualitatives apparaissent à la fois sous leur forme condensée et sous leur forme disjonctive complète.

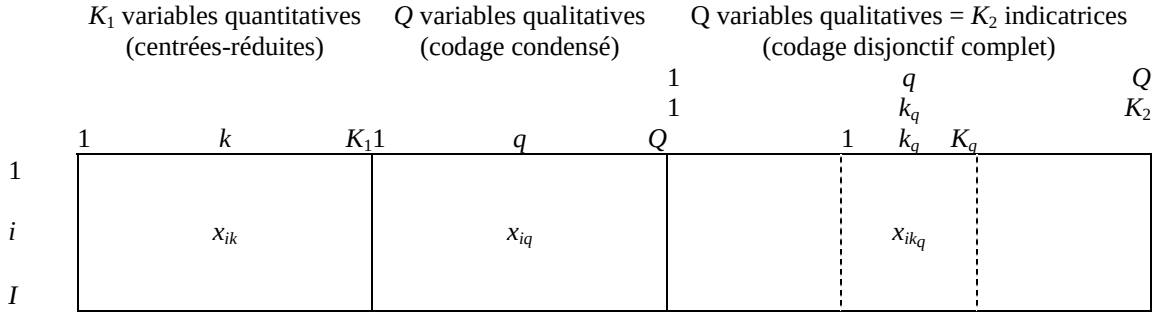


Figure 1. *Structure des données et principales notations.*
x_{ik} : valeur de i pour la variable (centrée-réduite) k ; x_{iq}: modalité de i pour la variable q ;
x_{ik_q} : = 1 si i possède la modalité k de la variable q et 0 sinon

3. Représentation des variables dans RI

Soit R^I l'espace des fonctions sur I. Cet espace est muni de la métrique diagonale des poids des individus notée D : $D(i, j) = 0$ si $j \neq i$

$= p_i$ si $j = i$

Généralement les individus ont le même poids : $D = (1/I)\ \boldsymbol{I_d}$ (en notant $\boldsymbol{I_d}$ la matrice identité de dimension I).
Comme en ACP normée, les variables quantitatives sont représentées par des vecteurs unitaires.
Comme en ACM, la variable q est représentée par le nuage N_q de ses K_q indicatrices centrées. Ce nuage engendre le sous-espace E_q de dimension $K_q - 1$, ensemble des fonctions sur I centrées et constantes sur les classes de la partition définie par q. Pour que N_q ait, dans une ACP non normée, les mêmes propriétés inertielles que dans une ACM, il faut affecter à l'indicatrice k_q le poids $1/p_{k_q}$ (en notant p_{k_q} la proportion des individus possédant la modalité k_q). Comme les programmes d'ACP usuels ne permettent pas l'introduction directe de poids de colonnes, on préfèrera diviser les valeurs de l'indicatrice k_q par $\sqrt{p_{k_q}}$, ce que nous appelons le codage-ACP de la variable qualitative.
En procédant ainsi, on obtient en particulier la propriété fondamentale suivante de l'ACM : l'inertie projetée de N_q sur une variable centrée y est égale au carré du rapport de corrélation $\eta^2(q, y)$ entre q et y.

4. Principe de l'AFDM

En recherchant la direction v de R^I qui rend maximum l'inertie projetée du nuage N_K (comportant à la fois les variables quantitatives et les indicatrices), on rend maximum le critère (en notant r le coefficient de corrélation) :

$$\sum_{k \in K_1} r^2(k, v) + \sum_{q \in Q} \eta^2(q, v)$$

point de départ de la proposition de Saporta (1990 p 66).
Géométriquement (cf. figure 2), les variables k étant réduites, $r(k, v) = \cos\theta_{kv}$, en notant θ_{kv} l'angle entre les vecteurs k et v. De même, v étant centrée, $\eta^2(q, v) = \cos^2\theta_{qv}$ en notant θ_{qv} l'angle entre v et sa projection sur E_q.
Le critère s'écrit alors

$$\sum_{k \in K_1} \cos^2 \theta_{kv} + \sum_{q \in Q} \cos^2 \theta_{qv}$$

point de départ de la présentation de l'AFDM par Escofier (1979a).

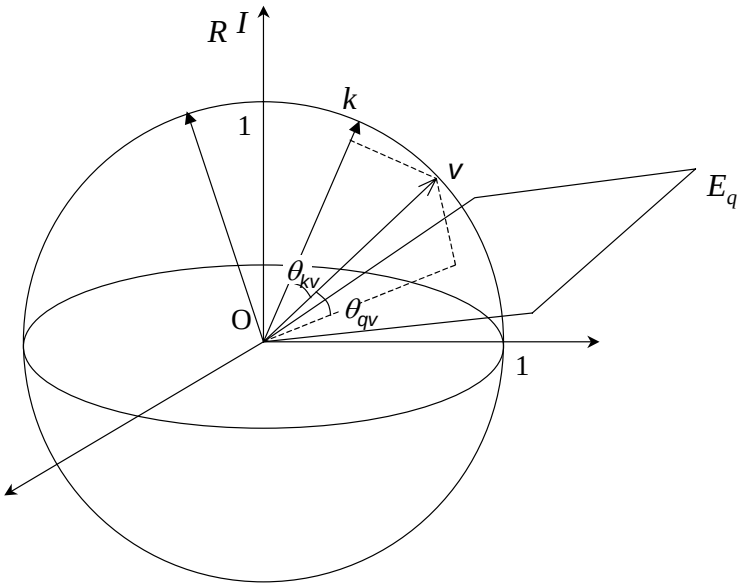


Figure 2. *Sous-espaces engendrés par les deux types de variables (k pour une variable quantitative et E_q pour une variable qualitative) et mesure de liaison avec une composante principale* (v).

Dans sa présentation de l'AFDM, Escofier (1979a) adopte un point de vue technique symétrique de celui choisi ici : elle se place dans le cadre de l'ACM et code la variable quantitative de façon à obtenir un tableau traitable dans ce cadre. Il s'agit donc bien de la même méthode, dont les résultats peuvent être obtenus via une ACM ou une ACP.

En AFM, les groupes de variables sont pondérés de façon rendre égale à 1 leur inertie axiale maximum. En introduisant un tableau de données mixtes dans lequel chaque variable, quantitative ou qualitative, constitue un groupe, on obtient donc les résultats de l'AFDM :

-les variables quantitatives sont centrées-réduites ;

-les variables qualitatives sont codées-ACP (cf. §3) et donc interviennent comme en ACM.

Ceci est la façon la plus simple de mettre en œuvre l'AFDM, par exemple via le logiciel SPAD (2003).

L'idée d'appliquer l'AFM à des groupes constitués chacun d'une seule variable quantitative ou qualitative a déjà été proposée (Abascal-Fernandez et al 2003).

5. Graphiques de l'AFDM

Comme dans toute analyse factorielle on représente :

- le nuage des individus par sa projection sur ses axes d'inertie (on note F_s le facteur sur I de rang s) ;
- les variables quantitatives par leur coefficient de corrélation avec les facteurs F_s ;
- les modalités de variables qualitatives par les centres de gravité des individus correspondant.

En outre, s'inspirant de la représentation des groupes de variables en AFM dite « carré des Lg » on fait figurer sur un même graphique les deux types de variables dans le droit fil des représentations des variables qualitatives en ACM proposés par Escofier (1979b) ou Cazes (1982). La coordonnée de la variables x le long de l'axe s vaut (cf. figure 3):

- $r^2(x, F_S)$ si x est une variable quantitative ;
- $\eta^2(x, F_S)$ si x est une variable qualitative.

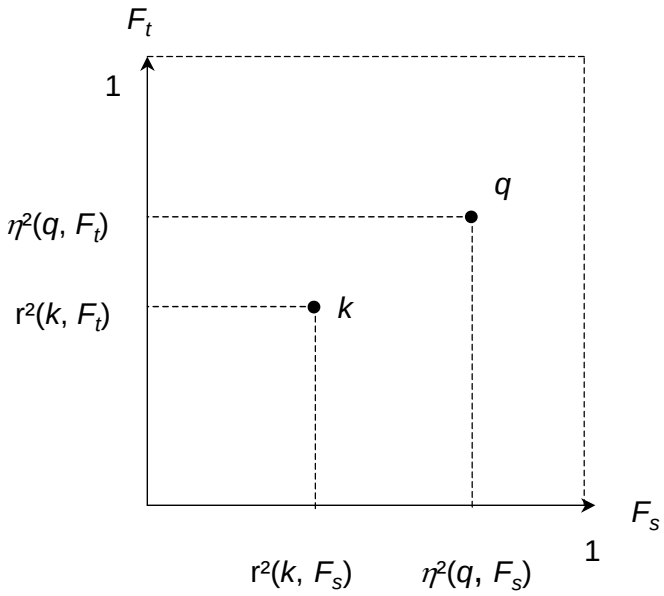


Figure 3. *Représentation des variables dites «carré des Lg »*
Les facteurs F_s et F_t sont en abscisse et ordonnée. Chaque variable a pour coordonnées ses mesures de liaison avec les deux facteurs.

Remarque : cette représentation n'annule pas l'intérêt du classique cercle des corrélations puisqu'elle ne dépend pas du signe de r(x, F_s).

6. Bibliographie

[ABA 03] ABASCAL-FERNANDEZ E., LANDALUCE-CLUO M.I., GARCIA-LAUBE I., Multiple factor analysis of mixed tables : a proposal for analysing problematic metric variables. *Proceeding cf CARME 2003 meeting*. Barcelona, June 2003.

[CAZ 80] CAZES P., Note sur les éléments supplémentaires en analyse des correspondances, *Les cahiers de l'analyse des données*, 1980, vol. 7, n°1, p. 9-23 et vol. 7, n°2, p.133-154.

[ESC 79] ESCOFIER B., Traitement simultané de variables quantitatives et qualitatives en analyse factorielle, *Les cahiers de l'analyse des données*, 1979a, vol. 4, n°2, p.137-146.

[ESC 79] ESCOFIER B., Une représentation des variables dans l'analyse des correspondances multiples. *Revue Statistique Appliquée* XXVII , 1979b, n°4, p.37-47.

[ESC 03] ESCOFIER B., *Analyse des correspondances*, Presses Universitaires de Rennes, 2003.

[ESC 98] ESCOFIER B., PAGÈS J., *Analyses factorielles simples et multiples*, 1998, 3e ed. Dunod.

[LE 02] LE DIEN S., PAGÈS J., Analyse factorielle multiple hiérarchique, *Revue de statistique appliquée* (2002). LI n° 2, p. 47-73.

[PAG 02] PAGÈS J., Analyse factorielle multiple appliquée aux variables qualitatives et aux données mixtes, *Revue de statistique appliquée*, 2002, L n° 4, p. 5-37.

[PAG 04] PAGÈS J., Analyse factorielle de données mixtes. *Revue de statistique appliquée* à paraître 2004.

[SAP 90] SAPORTA G., Simultaneous analysis of qualitative and quantitative data. *Atti della XXXV riunione scientifica ; società italiana di statistica*, 1990, p.63-72.

[TEN 77] TENENHAUS M., Analyse en composantes principales d'un ensemble de variables nominales ou numériques, *Revue de Statistique Appliquée* XXV, 1977, vol.2, p. 39-56.

[SPAD 03] SPAD, Diffusé par DECISIA 2003. – 30 rue Victor Hugo – 92532 Levallois-Perret cedex.

Construction de Hiérarchies Spatiales

Kutluhan Kemal Pak, Edwin Diday

Université Paris 9 Dauphine, Place du Maréchal de Lattre de Tassigny
{pak,diday}@ceremade.dauphine.fr

RÉSUMÉ. *On étend les hiérarchies indicées classiques à des hiérarchies dont les classes sont compatibles avec un maillage. On caractérise les convexes d'un maillage et une dissimilarité induite. Ensuite on présente un algorithme de classification ascendante hiérarchique spatiale dans le cas où le maillage est une grille. La méthode qu'on décrit est accompagnée d'un programme C++, dont la partie graphique est réalisée à l'aide de la bibliothèque OpenGL. On donne un exemple pour illustrer les résultats obtenus.*

MOTS-CLÉS : *Classification, Hiérarchies, 3D*

1. Introduction

Les méthodes classiques de classification ascendante telles les hiérarchies [JOH 67] [BEN 73] ou les pyramides [DID 84] [BER 86] permettent d'obtenir des classes suivant un ordre linéaire. On propose une nouvelle approche de classification ascendante hiérarchique dite spatiale pour obtenir des classes compatibles avec un maillage [DID 04]. On est dans le cadre de l'analyse de données symboliques, c'est pourquoi on utilise des données symboliques stockées dans des fichiers à extensions ".sds" ou ".xml" compatibles avec le logiciel SODAS, respectant les formats prédéfinis pour chaque individu et les variables qui le caractérisent. Chaque individu d'un ensemble $\Omega = \{\omega_1, \omega_2, ..., \omega_N\}$ est décrit par un certain nombre de variables de différents types : numériques qualitatifs classiques ou symboliques telles que les variables à valeurs intervalle ou distribution. On présente un algorithme de construction de hiérarchies spatiales indicées basé sur des mesures de dissimilarités telle que Gowda-Diday, Haussdorf [BOC 00] ou bien sur une mesure de généralité comme défini par Brito [BRI 91].

2. Maillage

2.1. *Définition d'un* (m, k)*-maillage*

Un (m, k)-maillage est un graphe dont chaque sommet est au point de rencontre de m arêtes au maximum formant m angles consécutifs égaux strictement positifs et dont les plus petits cycles (i.e. ceux qui contiennent le minimum de sommets) contiennent k arêtes de même longueur et forment des "cellules" de surface non nulle qui partitionnent et couvrent l'espace dans lequel il est plongé. Plutôt que maillage, on pourrait dire aussi : "tessellation". Une grille, le maillage qu'on utilisera est le cas particulier où $m = k = 4$.

2.2. *Parties convexes d'un maillage*

Soit Ω un ensemble d'individus, une partie C de Ω est dite convexe dans un maillage si et seulement si les chemins de plus courte longueur qui relient deux quelconques de ses sommets sont dans C.

2.3. *Grille Régulière*

Soit $n = 4^p$, le nombre de sommets d'un convexe, avec $p \geq 0$. Ce convexe définit une grille régulière s'il a la forme d'un carré dont chaque coté contient 2^p sommets.

3. Définition des Hiérarchies Spatiales Convexes

Une hiérarchie spatiale convexe est un ensemble H de parties non vides (appelées paliers) d'un ensemble Ω satisfaisant aux propriétés suivantes :

- $\Omega \in H$.
- $\forall \omega \in \Omega, \{\omega\} \in H$.
- $\forall P_1, P_2 \in H$ si $P_1 \cap P_2 \neq \emptyset \Rightarrow P_1 \subseteq P_2$ ou $P_2 \subseteq P_1$.
- Il existe un maillage, dont les sommets sont les éléments de Ω, pour lequel tout élément de H est convexe.

4. Algorithme

On présente les étapes générales d'un algorithme de classification ascendante hiérarchique spatiale. On se restreint, dans ce papier, aux grilles régulières et on définit les fusions autorisées pour former de telles grilles. La mesure utilisée peut aussi bien être une dissimilarité qu'une généralité et l'indice d'agrégation sera déterminé en fonction de cette mesure. Chaque palier est construit selon un ordre défini par cet indice. On introduit quelques notions avant de décrire l'algorithme.

4.1. *Types de paliers*

Soit H, une hiérarchie spatiale en construction. Soient Ω_{tot} l'ensemble de tous les paliers appartenant à H à un instant t et Ω' l'ensemble des paliers de H qui ne sont pas réduits à des singletons au même instant t. Autrement dit $\Omega' = H - \{w_i\} \forall w_i \in \Omega$. Un palier est soit un individu de la population initiale, soit le resultat d'une fusion de plusieurs autres paliers. Dans le deuxième cas, tous les paliers ayant participé à la fusion sont appelés les fils du palier résultant. Il existe deux types de paliers : palier définitif P_{def} et palier en construction P_{ctr}. Un palier P_{def} est soit un individu de la population initiale soit un palier ayant 4 paliers fils. Un palier en construction P_{ctr}, est un palier formé par l'agrégation de 2 paliers fils P_{def}, il faut donc encore une fusion avec un autre P_{ctr} pour former un palier définitif. Il n'existe donc pas qu'une seule forme d'agrégation mais deux. La première est une fusion 1 à 1 qui réunit deux paliers P_{def} et la deuxième est une fusion 2 à 2 réunissant deux paliers P_{ctr} étant donné qu'un palier P_{ctr} est temporaire et représente le regroupement de deux paliers P_{def}. Sachant que Ω_{ctr} représente l'ensemble de tous les paliers temporaires, on peut affirmer que $\forall P_{ctr} \in \Omega_{ctr} \Rightarrow P_{ctr} \notin H$. Donc Ω_{ctr} et Ω_{tot} sont des ensembles entièrement disjoints.

4.2. *Niveau d'un palier*

Le niveau d'un palier est son étage. Soit niv une application de $\Omega_{tot} \cup \Omega_{ctr}$ dans N. En numérotant de 1 à 4 les fils de chaque palier P_{def} et de 1 à 2 ceux de P_{ctr}, on définit niv de façon récurrente à l'aide de l'application $fils_i$ définie de la manière suivante : $fils_i : \Omega_{tot} \cup \Omega_{ctr} \rightarrow \Omega_{tot}$. Sachant que $fils_i$ retourne le fils numéro i du palier définitif ou en construction, niv est définie de la manière suivante :

- $\forall \omega \in \Omega, niv(\omega) = 0$.
- $\forall h \in \Omega', niv(h) = Max(niv(fils_1(h)), niv(fils_2(h)), niv(fils_3(h)), niv(fils_4(h))) + 1$.
- $\forall h \in \Omega_{ctr}, niv(h) = Max(niv(fils_1(h)), niv(fils_2(h))) + 1$.

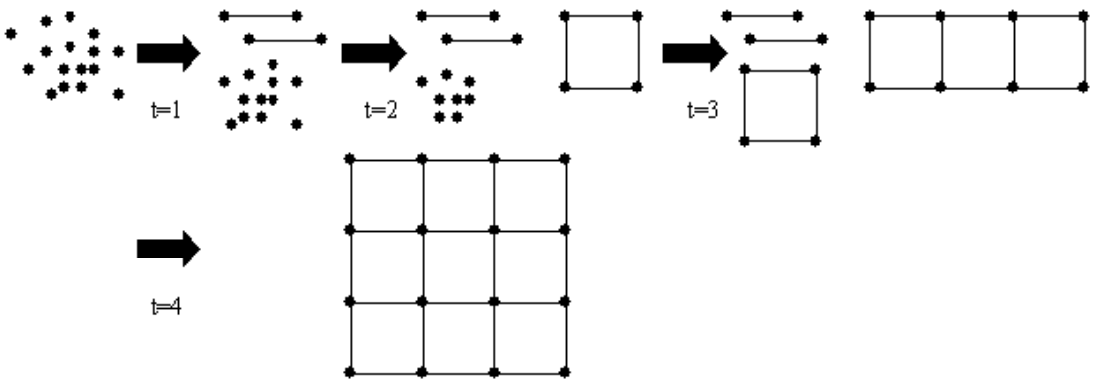


FIG. 1. *Evolution des grilles selon les étapes*

4.3. *Algorithme de Fusions*

Dans cette section, on expose les fusions qui sont autorisés pour former les nouveaux paliers de la hiérarchie spatiale en construction.

Paliers semblables et équivalents : On dit que deux paliers $P1$ et $P2$ sont "semblables" s'ils sont de même type (par exemple tous les deux des paliers en construction) et $niv(P1) \neq niv(P2)$. Et on dit que $P1$ et $P2$ sont "équivalents" s'ils sont de même type et $niv(P1) = niv(P2)$.

On présente une généralisation des étapes successives d'un algorithme de fusion de paliers dans laquelle on se limite à la réunion de paliers "équivalents" pour garantir l'obtention d'une grille régulière. Soit E_h un ensemble permettant de stocker tous les paliers quelque soient leurs types. Initialement, la hiérarchie spatiale en construction H et l'ensemble E_h ne contiennent que les paliers initiaux réduits à des singletons. A l'étape ($t = 1$) suivante, les paliers obtenus sont soit des paliers initiaux de type P_{def} de l'étape précédente, soit la fusion de ces paliers, construisant des P_{ctr} qui induisent des grilles réduites à des segments de droite de même taille. En règle générale, on ajoute à E_h tous les paliers de type P_{ctr} ou P_{def} résultants des fusions de paliers de l'étape précédente et on supprime de E_h tous ceux qui ont participé a leurs constructions. On ajoute à H seulement les paliers de type P_{def} sans jamais en supprimer. On continue ainsi jusqu'à obtenir un palier induisant une grille différente des autres. Si ce palier existe, on passe alors à l'étape suivante et on recommence le processus de construction avec les paliers contenus dans E_h. Ainsi le passage à $t = 2$, est dû à l'agrégation des paliers de type P_{ctr} de l'étape précédente formant un palier P_{def} induisant une grille carrée convexe differente des segments. Au fur et à mesure qu'on avance, on fusionne tous les paliers de E_h jusqu'à ce qu'il en reste qu'un seul dont l'extension sera définie dans une grille (convexe) finale $(2^p) * (2^p)$ contenant tous les individus de l'ensemble initial. H final contiendra tous les paliers appartenant à la hiérarchie spatiale finale.

La figure 1 montre l'évolution de E_h sur 16 individus initiaux. On représente les grilles induites par les paliers de E_h à la fin de chaque étape. Initialement, tous les paliers de l'ensemble initial sont représentés par des points. La fusion entre deux paliers points construit un palier dont le maillage est un segment. On passe donc à $t = 1$. On construit tous les paliers "segments" jusqu'à agréger deux paliers de ce type pour passer à l'étape suivante. On obtient un maillage complet à la fin de la quatrième étape.

Proposition :

1. Chaque palier d'une hiérarchie en construction forme une partie convexe du maillage (grille régulière) contenant 4^p sommets, où p représente le niveau de ce palier.

2. L'algorithme de fusion produit une hiérarchie spatiale convexe si la taille de la population initiale vaut $4^p, p \geq 0$.

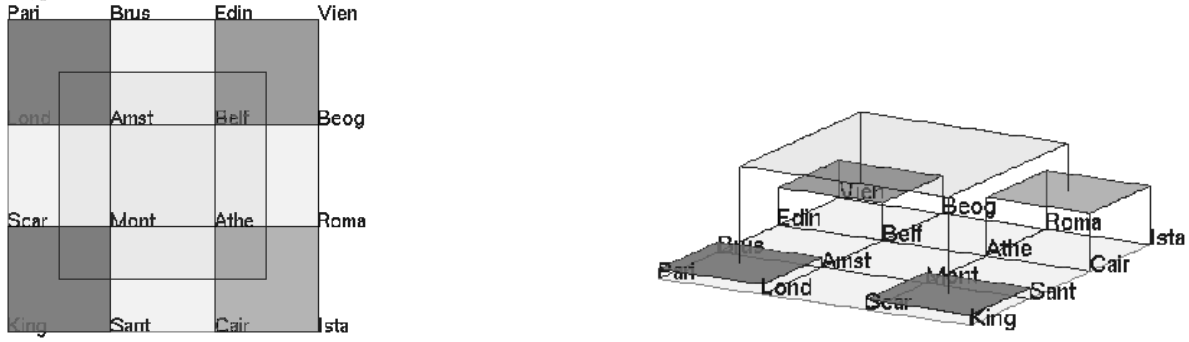


FIG. 2. *Répartition de 16 villes sur grille régulière*

5. Résultats

Quelques propriétés du programme qui accompagne la méthode :

- On utilise des couleurs dégradées pour différencier la hauteur des paliers lorsque la hiérarchie spatiale est vue de haut. Plus la hauteur du palier est élevée, plus sa couleur est claire.
- Chaque palier est interactif. Lorsqu'on "clique" sur un palier, ce palier devient le palier le plus haut, et sa grille devient une partie convexe du maillage contenant les éléments de son extension.
- On peut réaliser un zoom général.
- On peut également effectuer des rotations (sur les trois axes x, y et z).

Exemple La figure 2 représente une hierarchie construite sur un ensemble provenant d'un fichier villes.sds contenant des données sur les températures mensuelles des villes mondiales. Ce fichier contient 16 individus (les villes) et chaque individu est caractérisé par 12 variables intervalles selon la température de tous les mois de l'année. Ces variables correspondent aux températures minimales et maximales enregistrées durant le mois concerné. Les données proviennent du site internet du The World Meteorological Organization, à savoir http : //www.worldweather.org. La mesure de dissimilarité utilisée est par exemple celle definie par Gowda et Diday et l'indice d'agrégation, le maximum.

6. Conclusion

On a présenté une méthode permettant d'obtenir une carte des individus de la population initiale à la Kohonen, tout en gardant les spécificités des méthodes de classification ascendante. On peut espérer que par l'utilisation d'une méthode spatiale s'inspirant des pyramides on obtiendra une grille plus proche de la réalité étant donné que le nombre de paliers serait plus élevé et la grille plus précise.

7. Bibliographie

[BEN 73] BENZECRI J., *L'analyse de données. - Tome 1 et 2.,* Dunod Edition, Paris, 1973.

[BER 86] BERTRAND P., Etude de la représentation pyramidale, Thèse de doctorat, Université Paris 9 Dauphine, 1986.

[BOC 00] BOCK H-H. ET DIDAY E., *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data,* Springer-Verlag, Heidelberg, 2000.

[BRI 91] BRITO P., Analyse de Données Symboliques et Pyramides d'héritage, Thèse de doctorat, Université Paris 9 Dauphine, 1991.

[DID 84] DIDAY E., Une représentation visuelle des classes empiétantes : les pyramides, Rapport de Recherche INRIA N0 291, Rocquencourt, France, 1984.

[DID 04] DIDAY E., *Pyramidal Clustering Based on a Tessellation*, Springer-Verlag, Proceedings IFCS'2004 Chicago, 2004.

[JOH 67] JOHNSON S., Hierarchical clustering schemes, *Psychometrika*, vol. 32, 1967, p. 241-254.

Construction interactive d'arbre de décision avec des données intervalles et taxonomiques

François Poulet

ESIEA – Pôle ECD
38, rue des Docteurs Calmette et Guérin
Parc Universitaire de Laval-Changé
53000 Laval
poulet@esiea-ouest.fr

RÉSUMÉ. Nous présentons une méthode graphique interactive de construction d'arbres de décisions sur des données de types intervalles et taxonomiques. Cette méthode est une extension d'un algorithme existant de construction d'arbre de décision sur des variables "standard". Ce type d'approche (interactive) récente de la fouille de données est appelé fouille visuelle de données. L'utilisateur est le spécialiste des données et non plus un spécialiste de fouille ou analyse de données. Bien entendu, cet utilisateur dispose de mécanismes d'aide pour la création interactive de l'arbre. Ces mécanismes utilisent des méthodes dérivées des SVM (Séparateurs à Vaste Marge). Nous présentons les résultats obtenus sur des ensembles de données intervalles et taxonomiques créés à partir des ensembles de données de l'UCI "Machine Learning Repository".

MOTS-CLÉS : Fouille visuelle de données, classification supervisée, arbre de décision, données intervalles et taxonomiques

1. Introduction

La fouille visuelle de données est une approche récente de la fouille de données (même si l'on peut rapprocher ces travaux de ce qui existait auparavant sous le terme d'analyse exploratoire de données). Les premiers travaux sur le sujet datent de 1999-2000. Les différences principales par rapport aux approches plus classiques sont que l'utilisateur construit "manuellement" (i.e. graphiquement de manière interactive) son modèle et que l'utilisateur du système est le spécialiste des données. Ce dernier point présente au moins les avantages suivants : on peut bénéficier des connaissances du domaine des données tout au long du processus de fouille, la confiance et la compréhensibilité du modèle sont augmentées puisque l'utilisateur a participé à sa création et enfin on peut bénéficier des capacités humaines en reconnaissance de formes. Dans le cas de la classification supervisée, plusieurs algorithmes de création interactive d'arbres de décision ont été présentés, dont [POU 01]. Nous présentons ici une extension de ces travaux : l'adaptation de la méthode aux données de type intervalles et taxonomiques. Nous rappelons d'abord brièvement le principe de construction graphique interactive d'arbre de décision dans le cas de données "standard" puis nous présentons l'extension de la méthode aux cas des données intervalles et taxonomiques et les résultats obtenus sur des ensembles de données intervalles et taxonomiques.

2. Création interactive d'arbre de décision

CIAD est un algorithme de construction interactive d'arbre de décision. Le point de départ est une représentation graphique des données sous la forme d'un ensemble de matrices 2D (les projections des données selon toutes les paires possibles d'attributs ou colonnes de la base de données). Chaque point représente un individu, la couleur correspond à la classe. Une de ces matrices est sélectionnée et représentée à une échelle plus importante dans le coin inférieur droit de la visualisation comme montré sur l'exemple de la figure 1 avec l'ensemble de données Segment de l'UCI.

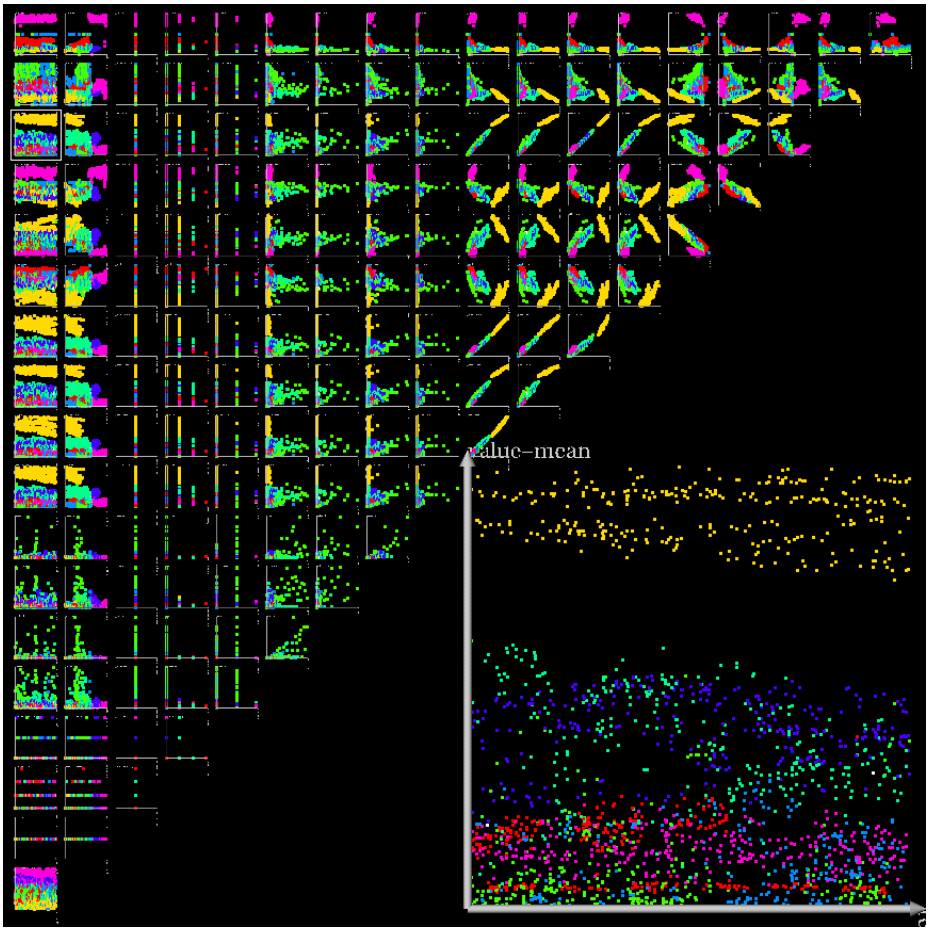


Figure 1 - Représentation de l'ensemble de données Segment

A partir de cette représentation graphique, la stratégie de l'utilisateur est la suivante : il cherche à repérer les zones pures (c'est-à-dire les zones ne comportant que des individus de la même classe ou couleur). Il trace alors une droite de séparation entre ces individus et le reste de l'ensemble de données. Cette zone pure constitue alors une feuille de l'arbre de décision et les individus correspondants sont éliminés de l'ensemble des projections (faisant potentiellement apparaître de nouvelles zones pures). Le processus est alors réitéré sur les individus restants. La figure 2 présente les quatre premières coupes opérées dans quatre projections différentes sur le même ensemble de données (Segment). Ces quatre coupes permettent d'éliminer 57% des individus de l'ensemble de données en classant à 100% quatre des sept classes de l'ensemble de données.

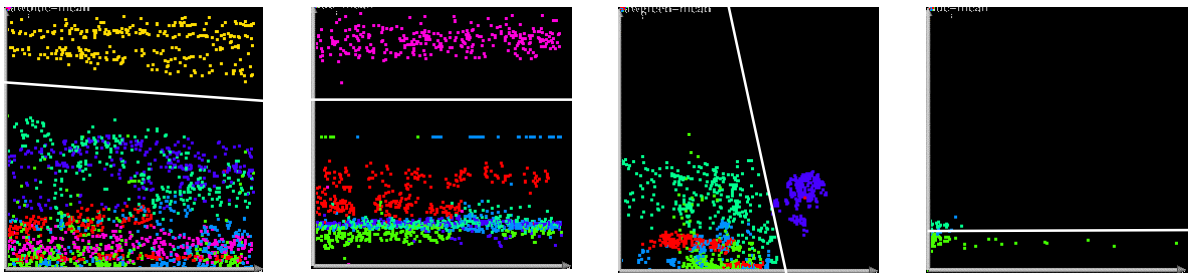
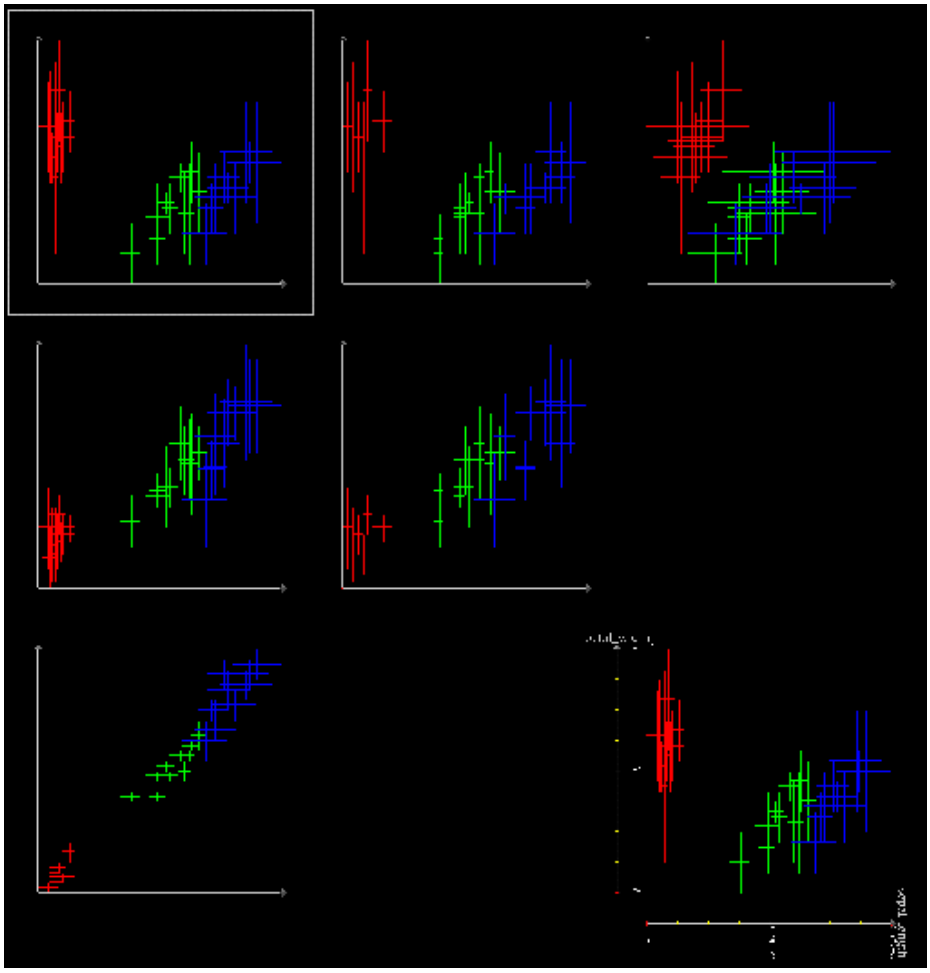


Figure 2 - Les 4 premières coupes sur l'ensemble de données Segment

Bien entendu une telle séparation manuelle n'est pas toujours aisée à effectuer sur les ensembles de données réelles. Pour aider l'utilisateur des mécanismes d'aide ont été ajoutés au système. Ils sont basés sur des algorithmes de SVM (Séparateur à Vaste Marge). La première solution consiste à optimiser la position de la droite tracée pour la transformer en la meilleure droite de séparation (c'est-à-dire la plus éloignée de la zone pure et du reste des données). La seconde solution consiste à rechercher automatiquement la meilleure séparatrice possible dans l'ensemble des projections 2D courantes. Pour ce faire, nous avons utilisé une version simplifiée d'algorithme de SVM : elle cherche la meilleure séparatrice seulement en 2D (et non pas en dimension *n*). L'utilisateur a donc le choix entre une méthode 100% manuelle, une méthode mixte (mélangeant interactions et méthode automatique) et une version 100% automatique ou chaque coupe est calculée par l'algorithme de SVM. Nous avons comparé les résultats obtenus par cette méthode interactive avec les algorithmes classiques d'arbres de décision tels C4.5 [QUI 93], CART [BRE 84] et OC1 [MUR 93] dans [POU 02]. Les taux de bonne classification sont comparables avec en général une taille d'arbre plus petite pour les algorithmes interactifs (et donc une meilleure compréhensibilité des résultats).

Figure 3 - Version intervalle de l'ensemble de données Iris



3. Données intervalles et taxonomiques

Nous avons ensuite étendu la méthode décrite au paragraphe précédent aux données de type intervalle et taxonomie. Le choix a donc été fait de représenter les variables intervalles par des croix (dans le cas de croisement de variable intervalle x variable intervalle) ou des segments (dans le cas de variable intervalle x variable continue) dans les matrices 2D (comme sur l'exemple de la figure 3 avec la version intervalle de l'ensemble de données *Iris*). Pour obtenir cette version intervalle, nous avons créé une nouvelle variable : la surface des pétales (petal_length*petal_width). En triant les données suivant cette nouvelle variable, on classe parfaitement l'ensemble de données. Nous avons ensuite regroupé les individus consécutifs par groupe de 5 pour créer les intervalles (valeurs min et max de chaque attribut original pour chaque groupe de 5). Ensuite le processus de construction de l'arbre de décision est exactement le même que dans le cas de variables continues. L'utilisateur cherche les zones pures et trace la droite de séparation entre ces données appartenant à la même classe et les autres individus de l'ensemble de données.

Les données de type taxonomiques sont traitées de manière analogue aux données intervalles, en effet, on peut se ramener au cas des intervalles en considérant l'ensemble des définitions du niveau inférieur de la taxonomie comme un intervalle. A l'heure actuelle, les mécanismes d'aide disponibles dans la version intervalle-taxonomie sont équivalents à ceux de la version initiale (en raisonnant sur les centres des croix) et utilisent toujours des algorithmes dérivés des Séparateurs à Vaste Marge (SVM) pour optimiser le placement de la droite ou pour trouver de manière automatique la meilleure séparatrice dans l'ensemble des projections 2D disponibles. Les résultats obtenus sont un peu moins bon comparés aux versions originales des ensembles de données utilisés pour l'ensemble de données *Segment*. Ceci peut s'expliquer aisément par la différence entre le nombre d'individus dans la version originale et la version intervalle : l'ensemble original comporte à peu près 40000 individus et la version intervalle quant à elle n'en comporte qu'une centaine. Un individu mal classifié a un coût de 1/40000 (=0,0025%) dans le premier cas et 1/100 dans le second cas.

4. Conclusion – Perspectives

Nous avons présenté une extension de l'algorithme de construction interactive d'arbre de décision aux cas des données de type intervalle et taxonomie. Nous avons comparé nos résultats avec la version automatique développée par [MBA 04] sur l'ensemble de données intervalles *Wave* du logiciel SODAS, les résultats sont similaires en ce qui concerne le taux de bonne classification et la taille de l'arbre obtenu. L'avantage de notre approche (par rapport à la méthode automatique) est qu'elle s'adapte presque immédiatement (sans modification importante) aux variables de type intervalle et taxonomie et qu'elle permet d'avoir simultanément des variables continues, qualitatives, intervalles et taxonomiques. D'un autre côté, ceci semble être un cas particulier puisqu'il nous semble beaucoup plus difficile de traiter d'autres types de variables symboliques (telles que des histogrammes par exemple). Néanmoins l'approche visuelle conserve toujours ses avantages en ce qui concerne l'utilisation des capacités humaines en reconnaissance de formes et l'amélioration de la confiance et de la compréhensibilité dans le modèle obtenu. Ceci n'est qu'un aspect de ce que peut apporter la visualisation dans le processus de fouille de données.

5. Bibliographie

[BLA 98] C.Blake, C.Merz, *UCI Repository of machine learning databases*, [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science, 1998.

[BRE 84] L.Breiman, J.Friedman, R.Olshen, C.Stone, *Classification And Regression Trees*, Chapman & Hall, 1984.

[MBA 04] C.Mballo, E.Diday, *Kolmogorov-Smirnov for decision tree on interval and histogram variables*, to appear in proc. of IFCS'2004, Meeting of the International Federation of Classification Societies, Chicago, Jul.2004.

[MUR 94] S.Murthy, S.Kasif, S.Salzberg, A system for induction of oblique trees, *Journal of Artificial Intelligence Research* 2, 1-32, 1994.

[POU 01] F.Poulet, *CIAD : Construction interactive d'arbres de décision*, actes de SFC'2001, 8e Rencontre de la Société Francophone de Classification, Pointe-à-Pitre, 275-282, 2001.

[POU 02] F.Poulet, FullView: A Visual Data-Mining Environment, in *International Journal of Image and Graphics*, 2(1), 127-144, 2002.

[QUI 93] J.Quinlan, *C4.5: Programs for Machine Learning*, Morgan-Kaufman Publishers, 1993.

Extraction incrémentale de la topologie des données

Yann Prudent, Abdel Ennaji

Laboratoire PSI
Université et INSA de Rouen
F-76821 Mont-Saint-Aignan, France
{yann.prudent, abdel.ennaji}@univ-rouen.fr

RÉSUMÉ. *Cet article introduit un modèle de réseau de neurones incrémental capable d'apprendre les relations toplogiques d'un ensemble de données d'apprentissage. Contrairement aux approches classiques (Kohonen, GCS, NG/CHL, GNG), ce réseau permet de faire un apprentissage par parties si la base d'apprentissage est trop importante pour être apprise en une seule fois. C'est-à-dire qu'il permet de reprendre un apprentissage même si les données précédemment apprises ne sont plus accessibles.*

MOTS-CLÉS : *Apprentissage non supervisé, Réseaux de neurones, Carte topologique, Apprentissage incrémental, Clustering.*

1. Introduction

Un des objectifs possibles de l'apprentissage non supervisé est d'effectuer un apprentissage qui peut être qualifié d'« apprentissage topologique ». Un apprentissage topologique peut se définir de la manière suivante :
Étant donnée une distribution de données P(**x**), il faut trouver une structure qui représente au mieux la topologie de cette distribution.
Dans ce but, les cartes de Kohonen [KOH 82] et le *Growing Cell Structure* [FRI 94] effectuent une projection non linéaire des données d'apprentissage dans un sous-espace discret de dimension choisie *a priori*. Pour la carte de Kohonen la taille du réseau doit elle aussi être fixée au préalable. A l'inverse le réseau GCS ajoute des neurones et des connexions au fur et à mesure de son apprentissage. Dans [MAR 94] les auteurs proposent d'allier deux méthodes, l'algorithme *Neural Gas*(NG) [MAR 91] et le *Competitive Hebbian Learning*(CHL) [MAR 93]. L'algorithme NG permet de déplacer les centres des classes en suivant la distribution des données. Le CHL permet d'établir des relations topologiques en induisant un sous-graphe de la triangulation de Delaunay (Triangulation de Delaunay induite). Dans ce type de réseau (NG/CHL) la dimension n'est pas fixe (Rectangle pour Kohonen, Hypertétrahèdre pour GCS) mais varie d'une région de l'espace à l'autre. De plus celle-ci n'est pas fixée *a priori*. Malgré ces avantages, la taille de ce réseau reste constante, et fixée *a priori*. Dans [FRI 95], Fritzke propose le réseau *Growing Neural Gas*. Ce réseau combine les avantages des réseaux GCS et NG/CHL, il n'a ni taille ni dimension à fixer *a priori*. De plus, cette approche est supposée être incrémentale en données.

L'approche est constructive et permet de prendre en compte de nouvelles données, mais qu'advient-il des connaissances déjà acquises ?

Nous introduirons et discuterons le principe de l'algorithme GNG. Puis nous proposerons un nouvel algorithme s'inspirant fortement de celui proposé par Fritzke mais qui comporte quelques avantages par rapport à celui-ci.

2. Growing Neural Gas

L'idée principale de cette méthode est d'ajouter successivement de nouveaux neurones à un réseau composé initialement de deux neurones connectés, grâce à une analyse locale de l'erreur quadratique engendrée par les données précédentes. Ce réseau se caractérise par une gestion dynamique du nombre de neurones basée sur un principe compétitif pour ajouter ou supprimer des neurones et des connexions.
Chaque neurone c_i du réseau GNG ne possède qu'un vecteur de référence w_i qui correspond à ses coordonnées dans l'espace de description. A chaque étape, le réseau tente de minimiser l'erreur quadratique engendrée par le nouvel exemple en déplaçant les neurones les plus proches de cette nouvelle donnée. Les connexions qui relient les neurones n'ont pas de poids, elles n'ont pour but que de définir la structure topologique des données (voisinage). Ces connexions ont un âge, et chaque connexion ayant un âge trop élevé est supprimée. Ainsi, chaque donnée présentée au réseau active la connexion qui relie les deux neurones situés le plus près d'elle dans l'espace de description. Si la connexion existe, son âge est réinitialisé à 0, sinon, elle est créée et son âge initial vaut 0. Des neurones sont rajoutés au réseau après un nombre constant d'étapes. L'erreur quadratique accumulée par les différents neurones permet de définir le vecteur de référence et les connexions du nouveau neurone.
Fritzke décrit ce réseau comme un réseau incrémental car il traite les données une par une et qu'il a une architecture évolutive. Toutefois, aucun résultat sur un apprentissage incrémental n'est présenté. Un réseau GNG peut-il apprendre de nouvelles données sans utiliser celles précédemment apprises ni détériorer le réseau déjà entraîné ?
Le problème illustré par la figure 1 a pour but de montrer les limites du réseau GNG pour la visualisation de la topologie dans le contexte d'un apprentissage incrémental. Ce problème est un problème synthétique composé de deux classes en deux dimensions avec une distribution sphérique pour la première classe et cubique pour la deuxième. L'apprentissage est effectué en deux étapes, une première où le réseau apprend la topologie de la première classe, et une deuxième où les données de la deuxième classe sont présentées au réseau. La figure 1(a) nous montre que pour pouvoir apprendre la topologie de la deuxième classe (distribution cubique), le réseau GNG a détérioré la connaissance acquise sur la topologie de la première classe (distribution sphérique). Ce problème est connu sous le nom de dilemme « plasticité/stabilité ».

3. Notre contribution : Incremental Growing Neural Gas (IGNG)

Le modèle que nous proposons dans ce papier est un modèle constructif et incrémental de cartes auto-organisatrices. Comme le modèle *Growing Neural Gas*, notre approche n'impose aucune contrainte sur la structure du réseau. Celui-ci est mis à jour de façon continue par un apprentissage Hebbien compétitif. Contrairement aux réseaux décrits précédemment, celui-ci possède deux types de neurones :

- des neurones matures,
- des neurones embryons.

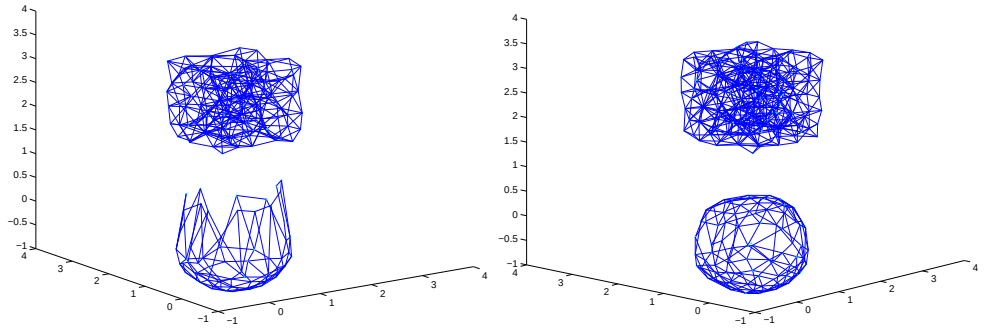
Chaque neurone possède un âge, un vecteur de référence et un type (mature ou embryon). Chaque connexion possède un âge. Afin de construire notre réseau, nous effectuons avant l'apprentissage de chaque donnée **x** un test défini par l'équation 1 où σ est un seuil fixé *a priori*. Ce test sert à vérifier si l'insertion d'un neurone est nécessaire. S'il n'existe pas au moins deux neurones qui satisfont cette équation, alors l'ajout d'un nouveau neurone est effectué.

$$dist(\mathbf{x}, w_n) \leq \sigma \quad [1]$$

Quand un neurone est inséré, c'est un neurone de type embryon avec un âge nul.

Initialement le réseau est vide. A chaque itération, nous cherchons le neurone gagnant s_1 (le plus proche de la nouvelle donnée). Si celui-ci n'existe pas (réseau vide) ou s'il ne satisfait pas la condition de l'équation 1, un nouveau neurone est ajouté avec $w_{new} = \mathbf{x}$.

Un deuxième cas peut se produire quand un neurone gagnant s_1 satisfaisant l'équation 1 existe. Nous recherchons alors le deuxième neurone s_2 le plus proche de la nouvelle donnée. Si s_2 n'existe pas ou s'il ne satisfait pas le test, un nouveau neurone est ajouté avec $w_{new} = \mathbf{x}$ et une connexion est créée entre ce neurone et s_1.



(a) GNG (b) IGNG

FIG. 1. *Comportement du GNG (a) et du IGNG (b) dans le cas d'un apprentissage incrémental*

Enfin, le dernier cas de figure correspond à l'existence d'au moins deux neurones qui satisfont le test de l'équation 1. Aucun neurone n'est ajouté dans cette situation et l'adaptation des vecteurs de référence se fait comme pour le GNG. L'âge des connexions émanant de s_1 est incrémenté. Comme dans l'apprentissage Hebbien compétitif, nous ajoutons une connexion entre les deux plus proches neurones de la donnée. Si celle-ci existe déjà son âge est réinitialisé à zéro. Toutes les connexions ayant un âge supérieur à a_{max} sont supprimées, et si ceci a pour conséquence l'isolement d'un neurone, celui-ci est supprimé. L'âge de tous les neurones connectés à s_1 est incrémenté, et tous les neurones embryons ayant un âge supérieur à a_{mature} deviennent des neurones matures. L'apprentissage de ce réseau est synthétisé dans l'algorithme 1.

si *le neurone* s_1 le plus près de l'entrée **x** *n'est pas activé* **alors**
  crée un nouveau neurone *embryon* de coordonnée **x** ;
  retourner ;
si *le neurone* s_2 le deuxième plus près de l'entrée *n'est pas activé* **alors**
  créer un nouveau neurone *embryon* de coordonnée **x** ;
incrémenter les connexions émanant de s_1 ;
$w_{s_1} += \epsilon_b(\mathbf{x} - w_{s_1})$;
$w_n += \epsilon_n(\mathbf{x} - w_n)$; //(n étant les voisins directs de s_1)
si s_1 *et* s_2 *sont connectés* **alors**
  $age_{s_1 \to s_2} = 0$;
sinon
  créer une connexion entre s_1 et s_2 ;
Supprimer les connexions avec un âge supérieur à a_{max} ;
si *ceci à pour conséquence l'isolement d'un neurone* **alors**
  supprimer ce neurone ;
Incrémenter l'âge des neurones émanant de s_1 ;
pour *Chaque neurone embryon s* **faire**
  si *age(s)*$\geq a_{mature}$ **alors**
    s devient un neurone *mature*

Algorithme 1: Algorithme d'apprentissage du réseau IGNG

Dans le cas du problème illustré par la figure 1, contrairement au réseau GNG, notre réseau se comporte parfaitement et préserve bien les connaissances déjà modélisées. Il répond donc bien, dans ce cas, au dilemme « plasticité/stabilité ».

Dans le but de comparer les performances de notre réseau à celui de Fritzke, nous avons supervisé les deux réseaux en étiquetant chaque neurone par un vote majoritaire des étiquettes des données qui l'ont activé. Le problème que nous avons utilisé pour ce test est un problème de reconnaissance de chiffres manuscrits de la base NIST. Le vecteur de caractéristiques considéré est constitué par les 85 caractéristiques (niveaux de gris) issues d'une pyramide de résolution à 4 niveaux (1+4+16+64) [BAL 82]. Dans le mode passif, la base est apprise en passant plusieurs fois toute la base, ce nombre de fois étant appelé nombre de cycles. Dans le mode incrémental, nous avons séparé la base d'apprentissage en quatre parties. L'apprentissage des deux modèles s'est fait par présentations successives des quatre parties de la base. Ces résultats nous montrent que notre réseau se prête bien à un apprentissage

	Cycles	λ	ϵ_b	ϵ_m	σ	passif (neurones)	incremental (neurones)
GNG	50	400	0.1	0.006	-	91.44% (330+0)	81.29% (328+0)
IGNG	10	-	0.01	0.002	2.7	91.71% (313+9)	90.18 (244+16)%

TAB. 1. *Résultats obtenus sur la base de reconnaissance de chiffres manuscrits*

incrémental puisque son comportement reste stable dans ces conditions, contrairement au GNG dont les performances se dégradent de manière importante.

4. conclusion

Le réseau IGNG est capable de rendre compte des relations topologiques d'une distribution de données P(**x**). Il apparaît que notre réseau peut reprendre un apprentissage sans tenir compte des données précédemment apprises. Ceci est très utile dans le cas, par exemple, où la base d'apprentissage est trop grande pour être apprise en une seule fois, ou pour poursuivre l'entraînement d'un réseau quand les données déjà apprises ne sont plus disponibles. De plus, le réseau IGNG possède tous les avantages du GNG. En d'autres termes, il n'est pas nécessaire de fixer *a priori* des paramètres tels que le nombre de clusters ou la dimension de la carte. De plus, les paramètres utilisés sont constants dans le temps contrairement aux réseaux NG et aux cartes de Kohonen. Enfin, le réseau IGNG converge plus vite que le GNG vers une bonne représentation de la topologie des données. Ceci a été confirmé sur le problème de clustering à densité de points très variable proposé dans [RIB 00]. Tout ceci fait du réseau IGNG une alternative aux approches classiques. La combinaison de ce modèle avec un apprentissage supervisé nous parait être une bonne voie pour l'élaboration un système d'apprentissage incrémental.

5. Bibliographie

[BAL 82] BALLARD D., BROWN C. M., *Computer Vision*, Prentice Hall, 1982.

[FRI 94] FRITZKE B., Growing Cell Structures — A Self-Organizing Network for Unsupervised and Supervised Learning, *Neural Networks*, vol. 7, n^o 9, 1994, p. 1441–1460.

[FRI 95] FRITZKE B., A growing neural gas network learns topologies, TESAURO G., TOURETZKY D. S., LEEN T. K., Eds., *Advances in Neural Information Processing Systems 7*, p. 625–632, MIT Press, Cambridge MA, 1995.

[KOH 82] KOHONEN T., Self-organized formation of topologically correct feature maps, *Biological Cybernetics*, vol. 43, 1982, p. 59–69.

[MAR 91] MARTINETZ T., SCHULTEN K., A "Neural-Gas" Network Learns Topologies, KOHONEN T., MÄKISARA K., SIMULA O., KANGAS J., Eds., *Proc. International Conference on Artificial Neural Networks* (Espoo, Finland), vol. I, Amsterdam, Netherlands, 1991, North-Holland, p. 397–402.

[MAR 93] MARTINETZ T., Competitive Hebbian Learning Rule Forms Perfectly Topology Preserving Maps, GIELEN S., KAPPEN B., Eds., *Proc. ICANN'93, International Conference on Artificial Neural Networks*, London, UK, 1993, Springer, p. 427–434.

[MAR 94] MARTINETZ T., SCHULTEN K., Topology Representing Networks, *Neural Networks*, vol. 7, n^o 2, 1994.

[RIB 00] RIBERT A., ENNAJI A., LECOURTIER Y., Clustering Data : Dealing with Hight Density Variation, SOCIETY I. C., Ed., *ICPR'2000*, vol. 2, 2000, p. 740-744.

La topologie des données pour une distribution fiable des tâches de classification

Yann Prudent, Abdel Ennaji

Laboratoire PSI
Université et INSA de Rouen
F-76821 Mont-Saint-Aignan, France
{yann.prudent, abdel.ennaji}@univ-rouen.fr

RÉSUMÉ. *Ce papier présente un système d'apprentissage multi-classifieurs dont la conception est pilotée par la topologie des données d'apprentissage. La démarche adoptée consiste à mettre en place un système de classification multiple où un ensemble de classifieurs de base sont controlés par une carte neuronale auto-organisatrice. Celle-ci permet de rendre compte de la topologie des données d'apprentissage et d'activer le ou les classifieurs concernés. De plus, le système proposé permet d'établir un critère de confiance s'affranchissant totalement du type de classifieurs utilisés. Ce coefficient permet de régler le compromis* Erreur/Rejet. *Le modèle de carte que nous utilisons est le* Incremental Growing Neural Gas, *réseau que nous avons proposé dans [PRU 04]. Ce réseau est une extension incrémentale du réseau GNG.*
Des résultats comparatifs de cette architecture sont donnés sur la base de segmentation d'images de l'UCI et sur des données de reconnaissance de chiffres manuscrits.

MOTS-CLÉS : *Classification, Apprentissage supervisé et non supervisé, Réseaux de neurones, Coopération, Reconnaissance de formes.*

1. Introduction

En reconnaissance de formes, les phases d'apprentissage et de classification constituent des étapes fondamentales qui conditionnent en grande partie les performances du système. Les techniques d'apprentissage artificiel ont connu ces dernières décennies des avancées fondamentales à travers des modèles tels que les réseaux de neurones et les *Séparateurs à Vaste Marge* (SVM)[VAP 95], qui montrent globalement de bonnes performances. Dans un premier temps, ces méthodes, basées sur différentes théories et méthodologies, ont été considérées comme autant de solutions possibles à un même problème. Cependant, leur développement n'a pas permis de mettre en évidence la supériorité incontestable d'une méthode sur une autre pour répondre aux contraintes des applications pratiques. De plus, peu de modèles permettent de fournir, en plus d'une décision, une estimation de la fiabilité de la décision ou un coefficient de confiance qui ne soit pas simplement une probabilité *a posteriori* comme c'est souvent le cas. Une solution admise dans la communauté pour résoudre ce type de problèmes consiste à adopter une conception modulaire et distribuée du système de classification [DUI 02] [KUN 02][GOR 98].
Dans ce contexte, ce papier présente un système d'apprentissage multi-classifieurs dont la conception est pilotée par la topologie des données d'apprentissage. Plusieurs contributions sont à noter dans le domaine de la décomposition de tâches de classification. Nous pouvons distinguer les approches purement supervisées des approches hybrides combinant à la fois des algorithmes supervisés et non supervisés. Jacobs et Jordan proposent dans [JAC 91] une approche qui décompose l'ensemble d'apprentissage en plusieurs sous-ensembles puis entraîne plusieurs réseaux sur ces sous-ensembles. Cette approche est basée sur une étude complètement supervisée. L'approche présentée dans [GOR 98] procède par une phase d'apprentissage non supervisé (clustering) par cartes auto-organisatrices. Celle-ci permet de spécialiser des groupes de neurones d'un Perceptron Multi-Couches (PMC) en fonction des clusters détectés. Ces deux méthodes fondent la distribution du problème de classification sur des informations

strictement supervisées ou des informations *a priori*. Elles ne prennent pas en compte de la distribution réelle des données dans l'espace de représentation.
Pour effectuer la distribution, certaines méthodes hybrides réalisent une première étape qui consiste à capturer la topologie des données dans l'espace de description. Ceci revient à un problème de classification automatique et implique donc de déterminer le nombre et la constitution des groupes (clusters) dans l'ensemble d'apprentissage. Les techniques les plus généralement utilisées dans ce domaine sont certainement les cartes Auto-Organisatrices de Kohonen [KOH 82] et les méthodes de regroupement par partitionnement [MAC 67]. Dans la pratique, le principal inconvénient de ces méthodes est la nécessité de fournir le nombre de groupes à l'avance pour obtenir une bonne représentation des données. [RIB 98] se sert d'une étude non supervisée du problème pour le distribuer sur plusieurs PMCs. Il utilise une classification ascendante hiérarchique pour déterminer les clusters sur les données d'apprentissage et entraîne un PMC sur chaque cluster. Il obtient des résultats équivalents à un classifieur global (K Plus Proches Voisins) pour les taux d'erreur maximaux, mais de bien meilleures performances quand l'erreur doit être très faible. Toutefois la classification hiérarchique ascendante, bien que très efficace sur les problèmes de *clustering* et ne nécessitant pas de connaître le nombre de clusters *a priori*, reste très coûteuse en temps et en espace. Dans [HéB 99] l'auteur tente de palier ces inconvénients en se servant d'un algorithme d'apprentissage non supervisé qui n'a besoin d'aucune connaissance préalable du problème : le réseau *Growing Neural Gas* [FRI 95]. Cette approche lui permet de reprendre les travaux de [GOR 98] sans avoir à connaître le nombre de clusters *a priori*. Nous nous servons d'une extension du réseau GNG, le réseau IGNG [PRU 04], pour distribuer le problème initial sur plusieurs classifieurs différents. La section suivante est consacrée à la conception modulaire du classifieur et à la présentation du facteur de confiance qui est associé à chaque module. Des résultats préliminaires sont présentés dans la section 3 puis discutés dans la conclusion.

2. Conception du classifieur distribué

La structure topologique représentée par le réseau *Incremental Growing Neural Gas* permet de distribuer le problème sur un ensemble de classifieurs. En effet, grâce à ce réseau notre système génère plusieurs sous-ensembles à partir de l'ensemble d'apprentissage. Il utilise chacun d'eux pour entraîner un ou plusieurs classifieurs. Chacune des régions définies par le IGNG est ainsi traitée, de manière indépendante, par un ou plusieurs classifieurs supervisés. Pour chaque sous-ensemble, notre système a la possibilité d'entraîner plusieurs classifieurs de différents types (PMC, SVM, KPPV, $\cdots$), ou aux paramètres d'apprentissage différents (noyaux, nombre de neurones, $\cdots$). Notre système sélectionne ensuite le ou les classifieurs adéquats au problème par les méthodes de sélection de classifieurs proposées dans la littérature [GIA 00] [KUN 02].
Il reste donc à trouver une méthode qui permette, à partir du IGNG, de générer les sous-ensembles de données d'apprentissage. Nous proposons une méthode qui génère plusieurs sous-ensembles s_i de l'ensemble d'apprentissage S. L'union de tous ces sous-ensembles n'est pas forcément égale à l'ensemble d'apprentissage. Elle associe à chaque neurone c_i du IGNG les données d'apprentissage telles que :

$$s_i = \{x \in S / d(x, w_i) \leq \sigma_i\},$$

σ_i est fixé *a priori* ou peut être calculé pendant la phase d'apprentissage du réseau IGNG. Chaque neurone est donc associé à une zone d'influence hypersphérique de rayon σ_i. Cette méthode a pour avantage de fermer les frontières de décision engendrées par les classifieurs. En effet, la génération de frontières de décisions ouvertes est un des principaux défauts de beaucoup de classifieurs tels que les PMCs ou les SVMs [GOR 98]. Ainsi, bien que reconnus comme performants, ils ne peuvent rejeter de manière efficace les données n'appartenant à aucune classe.

Une fois la base d'apprentissage décomposée en plusieurs sous-ensembles, un ou plusieurs classifieurs supervisés sont associés à chaque sous-ensemble d'apprentissage défini pour chaque neurone du IGNG. Nous appellerons par la suite cette association (neurone, classifieurs, sous-ensemble) un GNeurone. Les classifieurs d'un GNeurone vont se prononcer en phase de test pour tous les points qui l'activent. La décision d'un GNeurone, décision prise localement, résulte de la fonction de combinaison des classifieurs associés à celui-ci.
Lorsqu'une nouvelle donnée x à classifier est présentée au système, tous les Gneurones répondant à la condition $d(\mathbf{x} - w_i) \leq \sigma_i$ sont sélectionnés pour la décision. Nous choisissons alors les K plus proches GNeurones de la

nouvelle donnée dans l'espace de description (K est fixé *a priori*). Si le nombre de GNeurones sélectionnés est inférieur à K, on les prend tous. A chaque GNeurone sélectionné on associe un coefficient de confiance v donné par :

$$v = \frac{|s_i|}{d(x, w_i)^\beta}$$

Ce coefficient tient compte de deux facteurs importants pour la fiabilité d'une décision :

- le nombre de données d'apprentissage du GNeurone ($|s_i|$),
- la distance de celui-ci avec la donnée à classifier.

β est une constante permettant de définir un compromis ***distance/taille de la base d'apprentissage*** . Ce coefficient de confiance permet de pondérer le vote de chaque GNeurone, et par la même occasion de rejeter des données lorsque la décision prise par l'ensemble des K GNeurones est jugée peu fiable. Il est à noter que le coefficient de confiance de chaque GNeurone s'affranchit totalement du ou des classifieurs associés à celui-ci.

3. Résultats

La validation de notre approche a été effectuée à travers plusieurs expérimentations. Celles-ci ont pour but de mettre en évidence la performance des classifieurs, indépendamment de la pertinence des caractéristiques. Nous avons utilisé comme classifieurs pour les GNeurones uniquement des SVMs. Notre système, appelé KPPG (K Plus Proches GNeurones), a été comparé à des classifieurs qui ont déjà fait leurs preuves : le KPPV et le SVM. Le noyau utilisé pour le SVM a été déterminé par un procédé *d'essai/erreur*. La première expérimentation a été effectuée avec la base de segmentation d'images de l'UCI [MUR 92] appelée *Image*. Ce problème est composé d'une base d'apprentissage de 210 exemples et d'une base de test de 2100 exemples. L'espace de description est de dimension 20, et le problème composé de 7 classes (BRICKFACE, SKY, FOLIAGE, CEMENT, WINDOW, PATH, GRASS).

Nous avons ensuite validé notre approche sur le problème de la reconnaissance de chiffres manuscrits sur la base NIST. Le vecteur de caractéristiques considéré était constitué par les 85 caractéristiques (niveaux de gris) issues d'une pyramide de résolution à 4 niveaux (1+4+16+64) [BAL 82]. La base d'apprentissage est composée de 2626 exemples et la base de test de 2621.

Les résultats des classifieurs sur ces bases sont donnés par le tableau 1. Les deux courbes de la figure 1 ont été obtenues pour chaque classifieur considéré. La courbe du KPPV a été obtenue en diminuant k, tout en vérifiant que tous les voisins soient de la même classe (le taux maximum d'identification est obtenu quand k=1). La courbe du KPPG a été obtenue de la même manière. Comme les SVMs ne sont pas, à notre connaissance, capables de rejeter efficacement des données, il est impossible d'obtenir une courbe pour ce classifieur.

4. Conclusion

Les résultats préliminaires obtenus avec notre approche sont prometteurs. Les performances sont équivalentes aux autres classifieurs pour un taux d'erreur maximal, mais se révèlent supérieures quand des taux d'erreurs faibles sont recherchés. En effet, les résultats obtenus jusqu'à présent montrent que dans le cas où un taux de reconnaissance maximal est nécessaire, le système proposé est au moins aussi performant qu'un SVM. De plus, dans le cas où le taux d'erreur doit être faible, le KPPG obtient de bien meilleurs résultats que le KPPV. Quand aux SVMs, ils ne sont pas, à notre connaissance, capables de rejeter efficacement des données. Notre méthode permet donc d'allier les performances en reconnaissance de certains classifieurs comme les SVMs et un rejet efficace. Ce rejet est obtenu grâce au coefficient de confiance de chaque GNeurone qui s'affranchit totalement du ou des classifieurs associés à celui-ci. Le choix du réseau IGNG, et la liberté totale du choix de classifieurs, nous permettent de voir ce système comme un nouvel outil pour la conception d'un système d'apprentissage incrémental. Enfin, seul un classifieur de type SVM a été associé à chaque neurone du réseau. Le fait de multiplier les classifieurs à ce niveau permettra probablement d'améliorer les résultats obtenus.

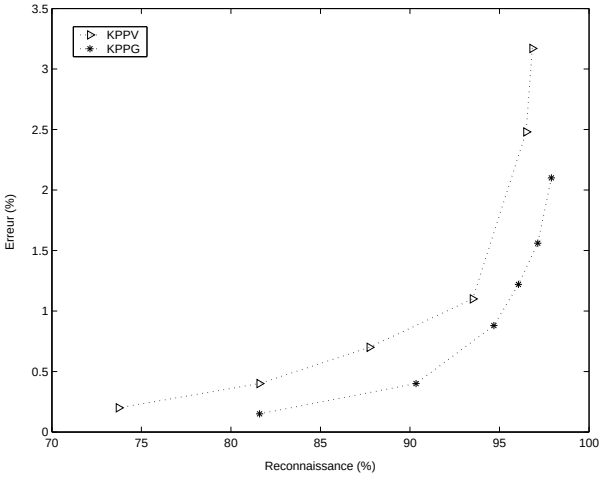


FIG. 1. *Erreur/Reconnaissance pour le KPPV et le KPPG sur la base NIST(85)*

	KPPV	SVM	KPPG
image UCI	12.43%	6.05%	**5.66%**
NIST(85)	3.17%	**2.10%**	**2.10%**

TAB. 1. *taux d'erreur des trois classifieurs*

5. Bibliographie

[BAL 82] BALLARD D., BROWN C. M., *Computer Vision*, Prentice Hall, 1982.

[DUI 02] DUIN R., The Combining Classifier : to Train or Not to Train, SOCIETY I. C., Ed., *ICPR*, Quebec, 2002, p. 765-771.

[FRI 95] FRITZKE B., A growing neural gas network learns topologies, TESAURO G., TOURETZKY D. S., LEEN T. K., Eds., *Advances in Neural Information Processing Systems 7*, p. 625–632, MIT Press, Cambridge MA, 1995.

[GIA 00] GIACINTO G., ROLI F., Dynamic Classifier Selection, SPRINGER, Ed., *Proc. of the First Int. Workshop on Multiple Classifier Systems*, jun 2000, p. 177-189.

[GOR 98] GORI M., SCARSELLI F., A Multilayer Perceptron adequate for pattern recogition and verification, *IEE Transactions on pattern analysis and machine intelligence*, vol. 20, 1998, p. 1121-1132.

[HéB 99] HÉBERT J.-F., PARIZEAU M., GHAZZALI N., An Hybrid Architecture for Active and Incremental Learning : The Self-Oganizing Perceptron (SOP) Network, *IJCNN99*, Washington, DC, USA, July 1999.

[JAC 91] JACOBS R., JORDAN M., NOWLAN S., HINTON G., Adaptive mixtures of local experts, *Neural Computation*, vol. 3, n^o 1, 1991, p. 79-87.

[KOH 82] KOHONEN T., Self-organized formation of topologically correct feature maps, *Biological Cybernetics*, vol. 43, 1982, p. 59–69.

[KUN 02] KUNCHEVA L. I., A theorical study on six classifier fusion strategies, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, Feb 2002.

[MAC 67] MACQUEEN J., Some methods for classification and analysis of multivariate observations, *Proceedings of the 5th Symposium on Mathematics Statistics and Probability*, 1967, p. 281-297.

[MUR 92] MURPHY P. M., AHA D. W., UCI repository of machine learning databases, Machine-readable data repository, 1992, University of California, Department of Information and Computer Science, Irvine, CA.

[PRU 04] PRUDENT Y., ENNAJI A., Extraction incrémentale de la topologie des données, *11èmes Rencontres de la Société Francophone de Classification*, bordeaux, septembre 2004.

[RIB 98] RIBERT A., Structuration Evolutive de données : Application à la construction de classifieurs distribués, PhD thesis, Université de Rouen, 1998.

[VAP 95] VAPNIK V., *The nature of statistical theory*, Springer, 1995.

Evaluation de la pertinence de paramètres biochimiques et classification pour la caractérisation des états physiologiques dans un bioprocédé par la théorie de l'évidence.

S. Régis, A. Doncescu, J-P. Asselin de Beauville, et J. Desachy

GRIMAAG Université Antilles-Guyane, Campus de Fouillole 97159 Pointe-à-Pitre / Laboratoire d'Informatique de l'Université de Tours, 64 Av. J. Portalis 37200 Tours / LAAS CNRS, 7 Av. du Col. Roche 31077 Toulouse Cedex 04
sregis@univ-ag.fr, adoncesc@laas.fr, jean-pierre.asselin@auf.org, jdesachy@univ-ag.fr

RÉSUMÉ. *L'objectif de ce travail est de proposer une nouvelle méthode basée sur la théorie de l'évidence capable de déterminer la pertinence d'une source pour la classification. L'évaluation de la pertinence repose sur la notion de conflit. Une mesure du conflit basée sur une distance est proposée comme alternative à la mesure de conflit classique. Les résultats de l'application en bioprocédé fermentaire sont présentés.*

MOTS-CLÉS : *Paramètres biochimiques, Pertinence, Théorie de l'évidence, Classification*

1. Introduction

A l'instar de la physique au siècle dernier, la biologie fournit aux mathématiques, à l'informatique et aux technologies de nombreux champs d'études et d'applications. Les données que nous analysons proviennent de procédés biotechnologiques expérimentaux ou industriels réalisés dans un bioréacteur utilisant des micro-organismes. Ces biotechnologies sont développées au Laboratoire Biotechnologies-Bioprocédés de Toulouse (LBB)[1]. Jusqu'à présent les travaux réalisés concernaient les analyses des classifications de ces données biotechnologiques [Y.N 02][S.R 03]. Lors de ces expériences, de nombreux paramètres biochimiques sont mesurés en temps réel et sont des sources potentielles d'information pour la classification. Jusqu'ici tous ces paramètres n'avaient pas été utilisés pour effectuer la classification : seuls ceux jugés pertinents par les experts en microbiologie étaient utilisés. Dans ce papier nous proposons une méthode basée sur la théorie de l'évidence ou théorie de Dempster - Shafer (DS) pour évaluer la pertinence des sources biochimiques de façon automatique et réaliser la classification en tenant compte de cette pertinence. Dans le premier paragraphe nous présentons plus en détails la notion de pertinence de paramètres biochimiques ainsi que les motivations de l'automatisation de l'évaluation de cette pertinence. Dans le second, nous parlons de la classification préliminaire effectuée par la méthode LAMDA (Learning Algorithm for Multivariate Data Analysis) [J.A 80][WAI 00] qui fournit des masses d'évidence utilisées par la théorie de l'évidence. Dans la troisième section la théorie de l'évidence est présentée ainsi que le calcul de la pertinence basée sur la notion de conflit entre paramètres. Une nouvelle mesure de conflit basée sur une distance est proposée. Enfin dans le dernier paragraphe nous présentons les premiers résultats et les premières conclusions.

1. Nous remercions les équipes du LBB pour leur aide et leur collaboration

2. Pertinence des paramètres biochimiques

On rappelle que les données biotechnologiques que nous cherchons à classer sont des signaux numériques issus d'un procédé de fermentation alcoolique utilisant des micro-organismes (levures appelées *Saccharomyces Cerevisiae*). Il s'agit de mesures de paramètres biochimiques effectuées à des intervalles de temps réguliers. Ces données sont représentées sous forme de vecteur où chaque composante d'un vecteur correspond à la valeur d'un paramètre biochimique à un instant donné t. Le nombre de vecteurs qui dépend du temps total du bioprocédé et de la fréquence de mesure des paramètres, est ici égal à 1012. Les données n'ont subi aucun traitement ni filtrage et aucune hypothèse n'est faite sur la nature du bruit éventuellement présent au niveau de ces données. Les experts en microbiologie cherchent à trouver une classification de ces données qui corresponde aux différents états physiologiques des micro-organismes (un état doit correspondre à une ou plusieurs classes). Ces experts cherchent à identifier au moins 3 états physiologiques principaux (on notera qu'il peut exister des sous-états à l'intérieur d'un état principal) des levures qui sont : la fermentation (état 1), la diauxie (état 2) et l'oxydation (état3). Le nombre de paramètres biochimiques mesurés dépend de la nature du bioprocédé et peut varier entre 6 et 20 (voire plus). Pour la classification des données et la recherche des états physiologiques, les experts utilisent seulement les paramètres biochimiques qu'ils jugent pertinents. Cependant la notion de pertinence des paramètres est basée sur des connaissances *subjectives* et *empiriques*. De plus les connaissances de ces experts concernent tout au plus 5 ou 6 paramètres alors qu'il peut y en avoir beaucoup plus. Par conséquent, une partie des informations fournies par l'ensemble des paramètres peut être soit inexploitée, soit redondante, voire erronée. Ainsi, en plus d' avoir une classification automatique des données qui leur fournisse les états physiologiques, ces experts désirent une évaluation de la pertinence des paramètres biochimiques ayant une base *objective* ou du moins suffisement *théorique*. C'est ici que la théorie de l'évidence peut fournir une aide pour l'évaluation de la pertinence de ces paramètres. Mais l'utilisation de la théorie de l'évidence implique que pour chaque paramètre des masses d'évidence affectées à chaque classe soient fournies au préalable. Ces masses d'évidence sont calculées à partir de la classification réalisée par la méthode LAMDA.

3. Classification préliminaire par LAMDA pour les masses d'évidence

La classification LAMDA est une méthode de classification non supervisée (mais qui peut aussi être utilisée en supervisée) developpée au LAAS de Toulouse qui tente de concilier les propriétés de la loi bayésienne et celles des méthodes neuronales simplifiées, tout en utilisant des opérateurs d'aggrégation flous issus de l'intelligence artificielle. Nous ne nous attarderons pas sur les détails de cette méthode car elle a déjà été présentée à plusieurs reprises [J.W 98][Y.N 02]. Rappelons cependant que pour chaque paramètre biochimique, LAMDA calcule un degré d'appartenance associée à chacune des classes existantes (ces classes sont créées au fur et à mesure de la classification [Y.N 02]) grâce la généralisation d'une loi binomiale appelée Degré d'Adéquation Marginal (DAM) :

$$\rho_{ji}^{1-\alpha(x_i,c_{j,i})}(1-\rho_{ji})^{\alpha(x_i,c_{j,i})} \quad [1]$$

où $c_{j,i}$ représente la composante i du centre c_j de la classe j, x_i est la composante i de l'élément x à classer, $\rho_{i,j}$ est la probabilité qu'un élément appartienne à la classe c_j et $\alpha(x_i, c_{i,j})$ représente la distance entre x_i et $c_{i,j}$. Une fois que les DAM ont été calculés pour chaque paramère biochimique et pour chacune des classes existantes, LAMDA effectue une fusion des informations issues de tous les paramètres biochimiques en utilisant un opérateur d'aggrégation. Il existe divers opérateurs d'aggrégation (T-norme et T-conorme, moyenne, etc) mais LAMDA utilise le triple Π développé par Yager et Rybalov [R.Y 98] pour sa propriété de renforcement total. Le triple Π est utilisé pour calculer le Degré d'Adéquation Global (DAG) pour chaque classe j :

$$DAG_j(x) = \frac{1}{1+\Pi_{i=1}^{n}\left[\frac{1-DAM_{j,i}(x_i)}{DAM_{j,i}(x_i)}\right]} \quad [2]$$

On pourrait se demander à ce stade pourquoi l'on tient compte de l'information issue de tous les paramètres biochimiques alors que l'on cherche à garder uniquement ceux qui sont pertinents, mais cette première fusion

présente un avantage certain. En effet, à partir de cette information globale on peut déterminer s'il faut créer une nouvelle classe ou non. Ainsi si le DAG est inférieur à 0.5 quelle que soit la classe, alors on crée une nouvelle classe (dont le centre sera l'élément que l'on cherchait à classer) ; sinon l'élément sera classé dans la classe dont le DAG est le plus important. De ce fait on tient compte de toutes les classes possibles : on passe ainsi d'un *monde ouvert* à un *monde fermé*. On entend par monde fermé le fait que toutes les classes aient été prises en compte pour la classification ; sinon on parle de monde ouvert. Cette notion de monde ouvert ou fermé est très importante pour la théorie de DS c ! ar elle peut grandement influencer la classification finale. Une fois que tous les DAM ont été calculés pour chaque paramètre, une normalisation est effectuée sur ceux-ci pour respecter l'équation 3 présentée dans le paragraphe suivant.

4. La théorie de l'évidence et son application

4.1. *La théorie de l'évidence*

La théorie de l'évidence est une généralisation de la théorie bayésienne qui tient compte des notions d'incertitude et d'imprécision de l'information. Elle a été introduite par Dempster [DEM 68] et complétée par Shafer [SHA 76]. Considérons l'ensemble de tous les évènements possibles (on parle d'ensemble de toutes les hypothèses) ; cet ensemble est appelé *ensemble de discernement* et est noté Θ. Toutes ces hypothèses sont mutuellement exclusives et sont nommées *singletons*. La théorie de Dempster-Shafer porte sur l'ensemble des sous-ensembles A de Θ. Cet ensemble de sous-ensemble de Θ est noté 2^{Θ}. A peut être composée d'un singleton ou d'une union de plusieurs singletons. Une masse d'évidence est alors définie sur l'ensemble des sous-ensembles A avec les propriétés suivantes :

$$\sum_{A \subset \Theta} m(A) = 1 \qquad [3]$$

$$m(\oslash) = 0$$

Les notions de *plausibilité* ($Pl(A)$) et de *croyance* ($Bel(A)$) sont aussi introduites mais ne servent qu'à la fin, au moment de la prise de décision :

$$Pl(A) = \sum_{B \cap A \neq \oslash} m(B) \qquad [4]$$

$$Bel(A) = \sum_{B \subset A} m(B)$$

Pour obtenir une information de deux sources différentes 1 et 2, il existe une combinaison de leur masses d'évidence appelée règle de Dempster-Shafer :

$$(m_1 \oplus m_2)(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B).m_2(C) \qquad [5]$$

$$A, B, C \subset 2^{\Theta}$$

où K est défini comme suit :

$$K = \sum_{B \cap C = \oslash} m_1(B).m_2(C) \qquad [6]$$

Le dénominateur $1 - K$ est un facteur de normalisation. Plus précisément K représente la mesure du conflit entre les sources 1 et 2. Plus K est important, plus les sources sont en conflit et moins la fusion a de sens. Si $K = 1$ alors le conflit est total et la fusion n'a pas de sens. On peut généraliser la règle de DS à n sources :

$$(\oplus m_i)_{i=1,\ldots,n}(A) = \frac{1}{1-K} \sum_{X_1 \cap \ldots \cap X_n = A} \left(\Pi_{i=1}^{n} m_i(X_i) \right) \qquad [7]$$

$$A, X_i \subset 2^{\Theta}$$

$$K = \sum_{X_1 \cap \ldots \cap X_n = \oslash} \left(\Pi_{i=1}^{n} m_i(X_i) \right)$$

Si les sources sont en conflit fort (K est grand) alors la règle de DS peut conduire à des résultats erronés, en particulier si l'on travaille dans un *monde ouvert*. Dans ce cas, la bonne hypothèse a dû être omise [ZAD 84][SME 90]. Cependant comme nous travaillons dans un monde fermé, s'il y a conflit entre les classes, cela provient du fait qu'au moins une des sources est erronée ou non pertinente.
Après la fusion de l'information, la prise de décision se fait par le choix du maximum soit de la plausibilité, soit de la croyance. Le choix de la fonction de plausibilité correspond à un choix optimiste tandis que l'utilisation de la fonction croyance correspond à une décision pessimiste.

4.2. *Théorie de l'évidence et pertinence des paramètres biochimiques*

Comme nous l'avons vu au paragraphe 2, l'évaluation de la pertinence des paramètres n'est pas une tâche facile. Nous proposons donc d'utiliser la notion de conflit pour évaluer cette pertinence. Ainsi en calculant le conflit deux à deux entre les paramètres, il est possible de se faire une idée de la pertinence des sources. Si un paramètre est en conflit avec plus de la moitié des autres paramètres alors il est considéré comme non pertinent et est écarté de la classification ; sinon il est pertinent et utilisé pour la classification. On notera que la pertinence est évaluée pour chaque élément : l'évaluation de la pertinence d'un paramètre est donc locale et non globale. Cette méthode se rapproche de la méthode de fusion par vote de Dubois et Prade [D.D 92] mais dans cette dernière, il n'y a pas de caractérisation des paramètres. Par ailleurs les méthodes statistiques traditionnelles comme l'ACP ne semblent pas adé quates pour la détection de conflit car d'une part la notion de corrélation ou de décorrélation est plus liée à la redondance d'information qu'à la notion de conflit, et d'autre part il n'est pas possible de les utiliser de façon ponctuelle c'est-à-dire pour un seul point.

4.3. *Vers une autre mesure du conflit*

Comme on l'a vu, la valeur K permet de mesurer le conflit entre sources. Cependant cette mesure K peut parfois conduire à des résultats erronés. Considérons le paramètre température (T°) qui fournit des masses d'évidence sur 9 classes différentes (voir le tableau suivant). Si l'on calcule le conflit entre T° et un paramètre P2 qui fournit

T°	0.111612	0.102348	0.101244	0.112802	0.151594	0.107448	0.107448	0.107264	0.098240

les mêmes masses d'évidence que T°, on trouve $K = 0.886865$! Ceci est dû au fait que K est adapté à la mesure de conflit de sources travaillant sur des unions de classes différentes. Or ici, tous les paramètres travaillent sur exactement les mêmes classes. C'est pourquoi nous proposons comme alternative au conflit K une mesure alternative D basée sur la norme 1 :

$$D = \frac{1}{2} \sum_{i} |m1(C_i) - m2(C_i)| \qquad [8]$$

Le facteur $\frac{1}{2}$ est un facteur de normalisation. En utilisant cette mesure D, on trouve un conflit entre T° et P2 égal à 0 ce qui est plus logique puisque T° et P2 fournissent les mêmes masses d'évidence.

5. Résultats et conclusion

Les premiers résultats sont encourageants. En fixant comme seuil de conflit entre deux paramètres une valeur de 0.3, et en éliminant les paramètres qui sont en conflit avec plus de la moitié des autres paramètres, on obtient les résultats suivants :

- 8 des 22 paramètres mesurés pendant le bioprocédé, sont considérés comme non pertinents vers la fin de l'expérience. Cette analyse confirment les connaissances des experts qui considèrent la fin d'un bioprocédé comme chaotique en raison de la mort d'un grand nombre de micro-organismes, cette mort provoquant une incohérence au niveau des mesures de plusieurs paramètres.

- certains paramètres sont éliminés de la classification au milieu de l'expérience. Quand on analyse les signaux, on constate que ces éliminations correspondent exactement à l'apparition de pics ou de singularités souvent inexplicables par les experts. Autrement dit, la méthode tend à éliminer les artefacts.

- l'élimination de certains paramètres de la classification permet d'une part, l'apparition de nouvelles classes auparavant absentes à certains moments de l'expérience, et entraîne d'autre part la disparition d'autres classes autrefois présentes à d'autres moments de l'expérience.

A titre comparatif, les résultats d'une ACP sur les données n'ont permis de tirer aucune conclusion quant à la notion de pertinence car les paramètres formaient plusieurs groupes plus ou moins décorrélés entre eux ; et ces groupes ne correspondaient pas vraiment aux groupes de paramètres non pertinents observés par les microbiologistes. Les résultats de la méthode présentée dans ce papier sont donc encourageants car ils montrent que la méthode agit comme un système expert en éliminant les artefacts et en confirmant de manière totalement indépendante les connaissances des experts. Les prochains travaux concerneront l'analyse plus détaillée de la classification et éventuellement l'utilisation de la méthode pour d'autres applications utilisant plusieurs sources d'informations.

6. Bibliographie

[D.D 92] D. DUBOIS H. P., On the relevance of non-standard theories of uncertainty in modeling, pooling expert opinions, *Reliability Engineering and System Safety*, vol. 36, n^o 2, 1992.

[DEM 68] DEMPSTER A., A generalisation of Bayesian Inference, *Journal of the Royal Statistical Society*, vol. 30, 1968, p. 205-247.

[J.A 80] J. AGUILAR MARTIN M. BALSSA R. D. M., Estimation récursive d'une partition. Exemple d'apprentissage et auto apprentissage dans **R**, rapport n^{o}880139, 1980, LAAS-CNRS.

[J.W 98] J. WAISSMAN-VILANOVA J. AGUILAR B. D. G. R., Généralisation de degré d'adéquation marginale dans la méthode de classification LAMDA, *6èmes Rencontres de la Société Francophone de Classification*, 1998.

[R.Y 98] R. YAGER A. R., Full Reinforcement Operators in Aggregation Techniques, *IEEE Transactions on Systems, Man, Cybernetics-Part B : Cybernetics*, vol. 28, n^o 6, 1998.

[SHA 76] SHAFER G., *A Mathematical Theory of Evidence*, Princeton University Press, New Jersey, 1976.

[SME 90] SMETS P., The combination of evidence in the transferable belief model, *IEEE Trans. on Pattern Analysis, Machine Intelligence*, , n^o 12, 1990, p. 447-458.

[S.R 03] S. REGIS J. DESACHY A. D. J. A.-M., Comparaison de classification non supervisées de données biotechnologiques, *Xe Rencontre de la Société Francophone de Classification*, Neuchâtel, Suisse, Septembre 2003.

[WAI 00] WAISSMAN-VILANOVA J., Construction d'un modèle comportemental pour la supervision de procédés : application à une station de traitement des eaux, PhD thesis, LASS - CNRS, Toulouse, Novembre 2000.

[Y.N 02] Y. NAKKABI S. REGIS J. D. A. D.-G. R., Apport de la Transformée en Ondelettes pour affiner les résultats de classifications, *IXe Rencontre de la Société Francophone de Classification*, Toulouse, France, Septembre 2002, p. 287-291.

[ZAD 84] ZADEH L. A., Book Review : A Mathematical Theory of Evidence, *AI Magazine*, vol. 5, n^o 3, 1984, p. 81-83.

Discriminer les inversions de courts segments dans une séquence biologique

David Robelin, Marie-Pierre Etienne

Laboratoire Statistique et Génome, UMR CNRS 8071, Tour Evry 2, 523 place des Terrasses, 91000 EVRY
robelin@genopole.cnrs.fr

RÉSUMÉ. *Deux méthodes permettant de détecter des segments retournés dans une séquence d'ADN sont présentées. La séquence est modélisée par une chaîne de Markov. La matrice de transition est estimée sur la séquence ; la matrice de transition de la chaîne inversée est ensuite calculée. Cela permet de comparer les vraisemblances d'un segment obtenu selon ces deux modèles de Markov. Chacune des méthodes produit un score. La distribution asymptotique qui lui est associée est présentée dans ce papier. La première des méthodes considère que la taille du retournement est connue a priori, alors que la deuxième explore toutes les tailles possibles.*

MOTS-CLÉS : *Chaîne de Markov, séquence d'ADN, segment inversé, valeurs extrêmes*

1. Introduction

Pendant l'évolution, de courts segments d'ADN peuvent s'inverser comme illustré dans la figure 1. A cause de la polarité de la molécule, l'opération d'inversion est accompagné d'un passage au brin complémentaire ($Adenine \leftrightarrow Thymine$ et $Cytosine \leftrightarrow Guanine$). Goldstein et coll. ont étudié leurs conséquences sur les protéines répertoriées dans les bases de données [GOL 00, GOL 03]. Ils ont trouvé un excès significatif de mots (de taille 3 à 7 acides aminés) qui sont inverse-complémentaires d'eux mêmes. Ils ont conclu à l'existence de courts segments retournés dans l'ADN codant. Goldstein et coll. suspectent que l'inversion de courts segments d'ADN pourrait être un mécanisme majeur de l'évolution des génomes. A la manière de Goldstein, on nommera un segment inversé *dincom* (pour **D**NA **in**verse **com**plementary).

On présente ici deux méthodes permettant de détecter les *dincoms* présents dans une séquence. Elles sont fondées sur une modélisation Markovienne de la séquence. Une approximation de la distribution des statistiques utilisées par chacune des méthodes est présentée dans le cas où il n'y a aucun retournement. La première méthode considère que l'on connaît a priori la taille du *dincom* . Elle est fondée sur une fenêtre glissante. Dans la deuxième méthode, cette taille n'est pas connue, et on utilise une statistique de type CUSUM [PAG 54, BAS 93]. Dans une première partie, la méthodologie de chacune des méthodes est présentée. Des résultats probabilistes sont donnés à cette occasion. Dans une deuxième partie, on présente succintement une application des deux méthodes au génome du HIV.

TAB. 1. *Un exemple de dincom . W pour le brin Watson. C pour le brin Crick.*

$$\begin{array}{cccccccc} 5' & W_1 & W_2 & W_3 & W_4 & W_5 & W_6 & 3' \\ & | & | & | & | & | & | & \\ 3' & C_1 & C_2 & C_3 & C_4 & C_5 & C_6 & 5' \end{array} \quad \rightarrow \quad \begin{array}{cccccccc} 5' & W_1 & \mathbf{C_5} & \mathbf{C_4} & \mathbf{C_3} & \mathbf{C_2} & W_6 & 3' \\ & | & | & | & | & | & | & \\ 3' & C_1 & \mathbf{W_5} & \mathbf{W_4} & \mathbf{W_3} & \mathbf{W_2} & C_6 & 5' \end{array}$$

2. Description des méthodes

La séquence est modélisée par une chaîne de Markov à l'état stationnaire $X = (X_1, ..., X_n)$. On note $\mathrm{P}(v|u)$ avec $u, v \in \{a, c, g, t\}$ la probabilité que u soit suivi par v. La probabilité stationnaire de chaque lettre est notée $\mu(u)$.

En retournant cette chaîne, on définit $X^- = (\bar{X}_n, \bar{X}_{n-1}, ..., \bar{X}_1) = (X_1^-, ..., X_n^-)$ où $\bar{X}_i$ désigne le nucléotide complémentaire de X_i. On peut montrer que X^- est également une chaîne de Markov si quelques hypothèses sont vérifiées, ce qui est généralement le cas pour des chaînes de Markov estimées sur des séquences réelles. La distribution stationnaire de X^- est facilement obtenue : $\forall u \in \{a, c, g, t\},\ \mu^-(u) = \mu(\bar{u})$.
La probabilité $\mathrm{P}^-(v|u)$ que u soit suivi par v pour tout $u, v \in \{a, c, g, t\}$ dans ce modèle se calcule en utilisant P, μ and μ^- :

$$\mathrm{P}^-(v|u) = \mathrm{P}(\bar{u}|\bar{v})\frac{\mu(\bar{v})}{\mu^-(u)}$$

2.1. *Approche par fenêtre glissante*

La séquence observée est notée $s_1, ..., s_n$. Pour chaque segment de taille $l < n$, on peut calculer

$$T_i = \log\left(\frac{\mathbb{P}^-(s_i, .., s_{i+l-1})}{\mathbb{P}^+(s_i, .., s_{i+l-1})}\right),\ i = 1, ..., n-l+1$$

où $\mathbb{P}^+(s_1, .., s_l)$ (resp. $\mathbb{P}^-(s_1, .., s_l)$) est la probabilité d'observer $(s_1, .., s_l)$ selon le modèle de Markov X^+ (resp. X^-).

Pour distinguer les pics pouvant refléter la présence de *dincom* des fluctuations aléatoires sur le graphique des T_i, nous avons besoin de connaître la distribution de T_i lorsqu'il n'y a pas de retournement dans la séquence. Si l est suffisamment petit, on peut calculer les probabilités d'apparition ainsi que les valeurs associées des 4^l différents segments sous le modèle X^+ et en déduire la distribution exacte de T_i. Le nombre de segments différents devient rapidement trop grand quand l grandit pour appliquer cette méthode. Un résultat asymptotique est utilisé dans ce cas. T_i peut s'écrire comme une combinaison linéaire des nombres d'occurrences des mots de deux lettres. On note $N_{i,i+l-1}(u, v)$ le nombre de fois où apparaît le mot "uv" dans la séquence $s_i, .., s_{i+l-1}$:

$$T_i = \log\left(\frac{\mu^-(s_i)}{\mu^+(s_i)}\right) + \sum_{u,v\in\{a,c,g,t\}} \log\left(\frac{\mathrm{P}^-(v|u)}{\mathrm{P}^+(v|u)}\right) \times N_{i,i+l-1}(u, v)$$

Le vecteur des comptages $\{N_{i,i+l-1}(u, v), u, v \in \{a, c, g, t\}\}$ est asymptotiquement gaussien quand n tend vers l'infini [WAT 95a]. Donc, T_i l'est également. Le calcul de son espérance est direct et le calcul de sa variance qui met en jeu les covariances entre les comptages des mots de deux lettres n'est pas détaillé ici.

Cette distribution nous permet de quantifier l'exceptionnalité d'une valeur de T_i donnée. Mais, nous ne pouvons l'utiliser pour évaluer l'ensemble des segments de taille l de la séquence sans nous confronter à un problème de test multiple. On s'intéresse alors à la distribution de la plus grande valeur observée : $S_l^n = \max_{i=1,...,n-l+1} T_i$ quand il n'y pas de *dincom*. Afin d'obtenir une approximation de cette distribution, on réécrit T_i :

$$T_i = \log\left(\tfrac{\mu^-(s_i)}{\mu^+(s_i)}\right) + \textstyle\sum_{j=1}^{j+l-1} Y_j \quad \text{où } Y_j = \log\left(\tfrac{\mathbb{P}(X_{j+1}|X_j)}{\mathbb{P}(X_{j+1}^-|X_j^-)}\right)$$

Glaz et Balakrishnan proposent l'approximation suivante de la distribution du maximum d'une somme glissante calculée sur une série indépendante d'entiers positifs aléatoires.

$$\mathbb{P}(S_l^n \leq s) \approx \mathbb{P}(S_l^{3l} < s)\left(\frac{\mathbb{P}(S_l^{3l} < s)}{\mathbb{P}(S_l^{2l} < s)}\right)^{n/l-3}$$

$3l$ étant relativement petit, $\mathbb{P}(S_l^{3l} < s)$ et $\mathbb{P}(S_l^{2l} < s)$ peuvent être évaluées par une approche de Monte-Carlo dans un temps relativement court, pour en déduire ensuite la distribution de S_l^n pour tout $n > 3l$.

Dans notre cas, les variables ne sont ni entières, ni positives, ni indépendantes. Une étude de simulation intensive a néanmoins montré un très bon comportement de cette approximation (résultats non présentés ici).

2.2. *Approche par score local*

Dans cette approche la longueur du *dincom* n'est pas fixée a priori. En utilisant les notations précédentes, $(Y_i)_{1 \leq i \leq n-1}$ peut être considéré comme le score de "retournement" du couple (X_i, X_{i+1}). On cherche ici le segment de score maximal parmi tous les segments possibles de la séquence. Son score appelé score local H_n est défini ci-dessous :

$$H_n = \max_{1 \leq i \leq j \leq n-1} Y_i + \ldots + Y_j$$

Plusieurs auteurs se sont intéressés à la distribution de H_n sous l'hypothèse nulle d'indépendance des Y_i ou lorsque les Y_i sont issues d'une chaîne de Markov ([DEM 91a, DEM 91b, KAR 92, DAU 99, DAU 03]). On utilise dans notre cas, le résultat de Karlin et Dembo :

$$\lim_{n \to +\infty} \mathrm{P}\left(H_n \leq \frac{\ln n}{\lambda} + x \right) = \exp\left(-K^* \exp\left(-\lambda x\right)\right).$$

Les constantes K^* et λ dépendent de la distribution de Y_i, mais leur détermination analytique est difficile. Nous avons donc utilisé la méthode préconisée par Waterman [WAT 95b] : on détermine par Monte Carlo la distribution $F(y)$ de H_{n_0} pour n_0 fixé : $F(y) = \mathbb{P}\left(H_n \leq y\right)$. Ensuite, une simple régression linéaire nous permet de déterminer λ et K car : $\ln\left(-\ln\left(F(y)\right)\right) \approx \ln K^* + \lambda\, y - \ln n$.
Dans ce cas encore, des simulations intensives non présentées ici ont montré une bon comportement de la méthode.

3. Application au génome du HIV

Chacune de ces méthodes a été appliquée à titre d'exemple à la séquence du virus HIV1. La figure 1 représente les valeurs de la statistique T_i obtenus en chaque point avec une fenêtre de longueur $l = 200$ ainsi que les scores locaux. Dans les deux cas, deux pics significatifs au risque 5% se distinguent en début et en fin de séquence.

4. Bibliographie

[BAS 93] BASSEVILLE M., V. N. I., *Detection of abrupt changes. Theory and application*, Prentice Hall information and systems sciences series, Prentice Hall, Englewood Cliffs, NJ, USA, 1993, 528 pp.

[DAU 99] DAUDIN J., MERCIER S., Distribution exacte du score local d'une suite de variables indépendantes et identiquement distribuées., *C. R. Acad. Sciences*, vol. 9, 1999, p. 815-820, Série I, Math.

[DAU 03] DAUDIN J., ETIENNE M., VALLOIS P., Asymptotic behaviour of the local score of independant and identically distributed random sequence, *Stochastic Processes and their Applications*, vol. 107, 2003, p. 1-28.

[DEM 91a] DEMBO A., KARLIN S., Strong limit theorems of empirical distributions for large segmental exceedances of partial sums of Markov variables., *Ann. Probab.*, vol. 19, n^o 4, 1991, p. 1756-1767.

[DEM 91b] DEMBO A., KARLIN S., Strong limit theorems of empirical functionals for large exceedances of partial sums of i.i.d. variables., *Ann. Probab.*, vol. 19, n^o 4, 1991, p. 1737-1755.

[GOL 00] GOLDSTEIN D., MURI F., SARAGUETA P., PRUM B., Inverse complementary homologues of short cysteine signatures., *C R Acad Sci III*, vol. 323, 2000, p. 167–172.

[GOL 03] GOLDSTEIN D. J., FONDRAT C., MURI F., NUEL G., SARAGUETA P., TOCQUET A.-S., PRUM B., Short inverse complementary amino acid sequences generate protein complexity, *Comptes Rendus Biologies*, vol. 326, n^o 3, 2003, p. 339–348.

[KAR 92] KARLIN S., DEMBO A., Limit distributions of maximal segmental score among Markov-dependent partial sums., *Adv. Appl. Probab.*, vol. 24, n^o 1, 1992, p. 113-140.

[PAG 54] PAGE E. S., Continuous Inspection Scheme, *Biometrika*, vol. 41, 1954, p. 100–115.

[WAT 95a] WATERMAN M., *Introduction to Computational Biology : Maps, sequences and genomes*, Chapitre Probability and Statistics for Sequence Patterns, p. 305–326, Chapman & Hall, 1995.

[WAT 95b] WATERMAN M., *Introduction to Computational Biology : Maps, sequences and genomes*, Chapman & Hall, 1995.

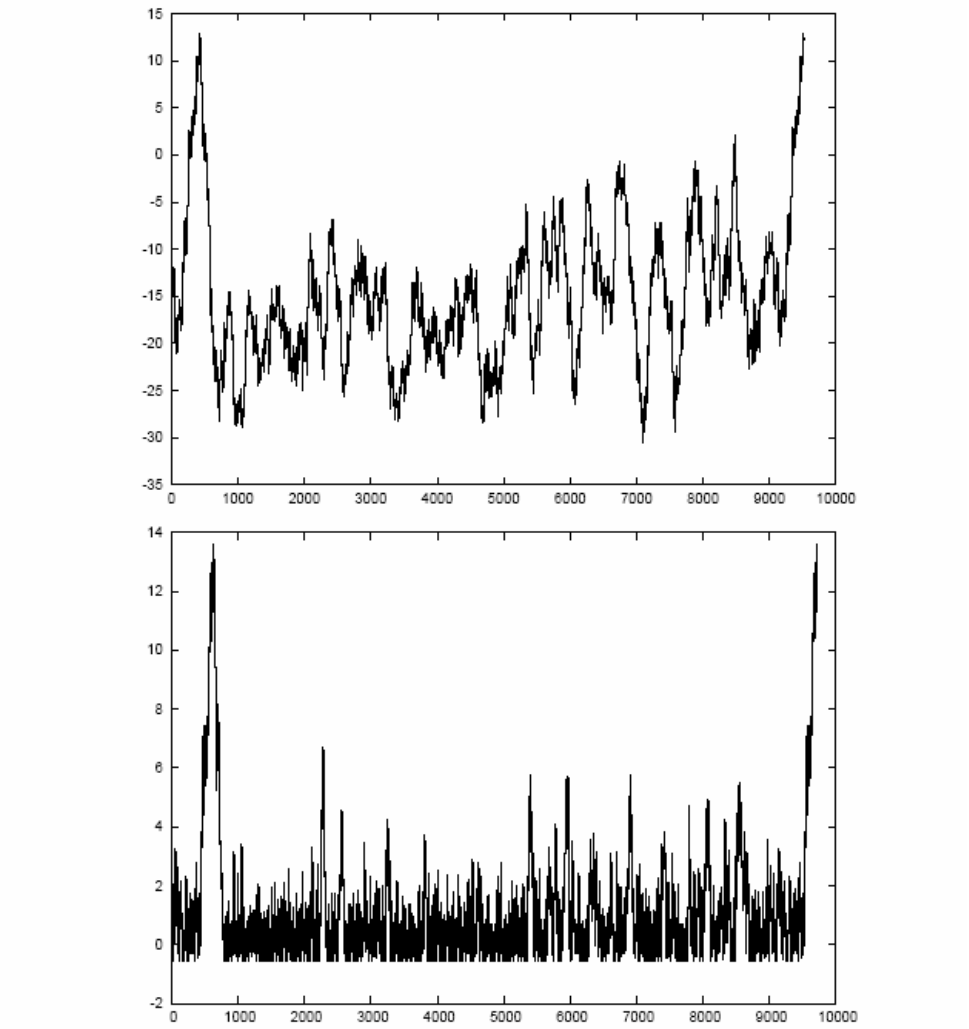


FIG. 1. *Valeurs prises par la statistique T_i le long du génome hiv. Haut : avec une fenêtre glissante de taille 200 - Bas : avec la méthode du score local*

Utilisation de coefficients statistiques d'association ordinaux pour comparer des hiérarchies

F. Sousa

CEC, Faculdade de Engenharia da Universidade do Porto, Departamento de Engenharia Civil
Rua Dr. Roberto Frias,
4200-465 Porto, Portugal
E-mail: fcsousa@fe.up.pt

RÉSUMÉ. Le résultat le plus fréquent d'une méthode de classification ascendante hiérarchique est un arbre de classification ou dendrogramme, qui définit une structure ultramétrique sur l'ensemble des éléments à classifier. On a souvent besoin de comparer des dendrogrammes ou bien leurs matrices ultramétriques associées. Dans ce travail on utilise des coefficients d'association ordinaux pour comparer des dendrogrammes obtenus à partir d'un même tableau de données. On obtient la distribution empirique de chacun de ces coefficients d'association en recourant à la simulation. Trois méthodes de génération aléatoire de dendrogrammes sont discutées et utilisées.

MOTS-CLÉS: Classification, Hiérarchie, Génération Aléatoire d'Arbres de Classification, Coefficients d'Association Ordinaux, Simulation

1. Introduction

La définition d'une méthode de classification ascendante hiérarchique (C.A.H.) présuppose le choix de deux fonctions de comparaison, l'une entre paires d'éléments et l'autre entre paires de parties de l'ensemble à classifier. Diverses méthodes de C.A.H. ont été proposées dans la littérature. Si on applique sur un tableau de données plusieurs de ces méthodes on est pratiquement sûr d'obtenir des hiérarchies différentes. Comment savoir quel est le meilleur algorithme? Cette question et d'autres se posent, sans qu'on y ait jusqu'à présent donné des réponses claires et générales.

Dans les études de validation ou de sensibilité il faut fréquemment comparer des dendrogrammes obtenus par l'application de différents algorithmes de classification à un même tableau de données. Divers travaux ont déjà été menés sur ce thème [LAP 92, SOK 63, NIC 84, GOR 99, SOU 01] qui est l'objectif central de cet article. Plusieurs coefficients statistiques peuvent être utilisés pour mesurer le degré de similarité d'une paire de dendrogrammes. Dans ce travail, divers coefficients d'association sont utilisés, tous de caractère ordinal.

L'utilisation de méthodologies inférentielles dans ce contexte est très délicate, car quelques hypothèses de base ne sont pas vérifiées. Par exemple si on considère la situation dans laquelle on souhaite comparer deux arbres de classification ou leurs matrices ultramétriques associées, en choisissant un coefficient d'association, on obtient une valeur de similitude entre les deux dendrogrammes. Mais comment mesurer la signification statistique de cette valeur, en tenant compte du fait que les deux matrices à comparer sont dépendantes et que dans chaque matrice les valeurs ne sont pas indépendantes car elles vérifient l'inégalité ultramétrique? Dans ce contexte les lois de distributions de probabilité des coefficients ne sont pas vérifiées et il faut obtenir des lois statistiques adéquates pour cette situation spécifique. Ceci n'est possible que si on peut construire tous les dendrogrammes non isomorphes et en comparer toutes les paires. Le nombre de dendrogrammes non isomorphes augmente très rapidement en fonction du nombre de nœuds terminaux. Donc l'énumération exhaustive doit être remplacée par des tirages aléatoires.

Dans ce travail on présente une méthodologie pour générer aléatoirement des arbres de classification. Cette méthodologie est comparée avec d'autres proposées par Lapointe & Legendre [LAP 91] et Podani [POD 00].

2. Arbres de classification ou dendrogrammes

Il faut commencer par définir ce qu'on entend par dendrogramme, qui est un type particulier d'arbre (pour des définitions générales des différents types d'arbres voir par exemple [BAR 88]). Un dendrogramme vérifie les propriétés suivantes:

- C'est un arbre avec racine, ce qui veut dire qu'on choisit un de ses nœuds pour être la racine, induisant ainsi une direction aux branches de l'arbre.
- C'est un arbre sans groupements empiétants.
- C'est un arbre valué, ce qui veut dire que des valeurs sont associées aux branches de l'arbre.
- Ses nœuds terminaux ont des étiquettes et ils sont tous à la même distance de la racine.
- À chaque nœud interne de l'arbre est associée une valeur d'indice de niveau de fusion, valeur qui dépend des fonctions de comparaison utilisées.

Quelques auteurs considèrent trois classes de dendrogrammes [POD 00 et leurs références]:

- *Les dendrogrammes indicés*, qui vérifient toutes les propriétés énoncées ci-dessus.
- *Les dendrogrammes complètement ordonnés* ou *invariants d'ordre global* (GOI) [SIB 72], qui sont des dendrogrammes indicés dans lesquels on substitue aux valeurs d'indice des niveaux de fusion les rangs correspondants.
- *Les dendrogrammes partiellement ordonnés* ou *invariants d'ordre local* (LOI), dans lesquels l'information sur les nœuds internes est prise en compte seulement localement.

Dans ce travail on considère que la définition complète d'un arbre de classification passe par l'identification de trois aspects: la topologie ou forme, les nœuds terminaux et les valeurs de l'indice de niveaux de fusion. Ceci correspond à considérer des dendrogrammes indicés. Toutefois les limites invoquées par l'utilisation des valeurs numériques de l'indice des niveaux de fusion sont bien connues. Pour comparer des dendrogrammes on utilise des coefficients statistiques ordinaux. On a donc besoin uniquement de l'ordre des valeurs de l'indice des niveaux de fusion, ce qui revient à considérer les dendrogrammes comme étant complètement ordonnés.

Soit m le nombre de nœuds terminaux d'un arbre de classification. Le nombre de dendrogrammes, binaires et sans niveaux de fusion égaux, non isomorphes est donné par l'expression suivant [FRA 81] :

$$d(m) = \frac{m!(m-1)!}{2^{m-1}}$$

Notons que $d(m)$ augmente très vite, par exemple $d(10) = 2571912000$ et $d(50) > 10^{100}$. Ceci justifie ce que nous mentionnions plus haut, il faut remplacer l'énumération exhaustive par des tirages aléatoires.

2.1. *Génération aléatoire de dendrogrammes*

Il existe de nombreux travaux sur la génération aléatoire d'arbres en général. Dans [FUR 84] sont présentées et discutées des méthodes de générations aléatoires de diverses catégories d'arbres binaires du type LOI. Cependant peu d'auteurs ont proposé des méthodes pour la génération d'arbres de classification. On considère ici trois méthodes pour générer des arbres qui tiennent compte de leur topologie, de l'étiquetage de leurs nœuds terminaux et des valeurs de leurs nœuds internes. Ces trois méthodes génèrent des dendrogrammes, pour m fixé, uniformément au sens de Furnas [FUR 84]. Ce qui signifie que chaque élément de l'ensemble des dendrogrammes non isomorphes à m nœuds terminaux a une probabilité d'être généré constante et égale à $1/d(m)$.

- La méthode de *Génération Uniforme*, proposée par Sousa [SOU 00] est récursive. Elle opère sur chaque nœud par un algorithme très simple. Deux vecteurs seulement sont nécessaires comme données de l'algorithme : un vecteur avec tous les nœuds terminaux et un autre avec les niveaux de fusion associés aux nœuds internes. Le processus commence par la racine, qui possède l'information des m éléments terminaux et des $m-1$ nœuds internes. L'algorithme divise aléatoirement les m nœuds terminaux en deux sous-ensembles et, tenant compte des cardinaux de ces sous-ensembles, divise aléatoirement les nœuds internes, ce qui crée deux sous-arbres. L'algorithme est appliqué à chacun de ces sous-arbres, successivement jusqu'à l'obtention de la division triviale associée aux nœuds terminaux.

- La méthodologie de *Permutation Double*, proposée par Lapointe & Legendre [LAP 91], qui permet de générer aléatoirement les matrices ultramétriques ou dendrogrammes indicés pour un ensemble d'indices de niveaux fixé.
- La méthode d'*Agglomération Aléatoire*, due à Podani, [POD 00], a comme point de départ les *m* éléments isolés et en chacun des $m-1$ pas l'algorithme choisit aléatoirement les classes à réunir.

Des études par simulations ont montré que les trois algorithmes de génération aléatoire d'arbres de classification sont équivalents.

2.2. *Comparaison ordinale de dendrogrammes*

En C.A.H. il existe divers types de structures de relations entre les éléments à classifier qu'on a fréquemment besoin de comparer. En particulier dans cet article on s'est intéressé à la comparaison de structures associées à des hiérarchies.

Pour comparer des paires de dendrogrammes ou les matrices ultramétriques correspondantes obtenus par l'application de deux critères classificatoires distincts au même tableau de données, on choisit une approche ordinale. Cette approche bien que moins informative est sans doute plus robuste qu'une approche numérique. Les coefficients statistiques d'association ordinaux utilisés sont les coefficients de corrélation de Spearman, de Kendall et de Goodman-Kruskal.

Quand on obtient une valeur d'association pour une paire de hiérarchies, on peut connaître sa signification statistique en ayant recours à la simulation de la distribution empirique de la loi du coefficient utilisé.

La procédure qui a été développée consiste à générer, pour une valeur de *m* fixée et en utilisant une des méthodes référées, une paire de dendrogrammes et à calculer la valeur du coefficient d'association entre eux. La répétition de ce calcul pour un grand nombre de paires de dendrogrammes, permet de déterminer la loi empirique du coefficient. Pour chaque coefficient d'association, différentes valeurs de *m* sont considérées.

3. Bibliographie

[BAR 88] BARTHÉLEMY, J.-P., GUÉNOCHE, A., *Les Arbres et les Représentations des Proximités, Méthodes + Programmes*, Masson, Paris, 1988.

[FRA 81] FRANK, O., SVENSSON, K., "On Probability Distributions of Single-Linkage Dendograms", *Journal of Statistical Computation and Simulation*, vol. 12, 1981, p. 121-131.

[FUR 84] FURNAS, G. W., "The Generation of Random, Binary Unordered Trees", *Journal of Classification*, vol. 1, 1984, p. 187-233.

[GOR 99] GORDON, A. D., *Classification*, 2nd ed., Chapman & Hall, London, 1999.

[LAP 91] LAPOINTE, F.-J., LEGENDRE, P., "The Generation of Random Ultrametric Matrices Representing Dendrograms", *Journal of Classification*, vol. 8, 1991, p. 177-200.

[LAP 92] LAPOINTE, F.-J., LEGENDRE, P., "Statistical Significance of the Matrix Correlation Coefficient for Comparing Independent Phylogenetic Trees", *Systematic Biology*, vol. 41, nº 3, 1992, p. 378-384.

[NIC 84] Nicolau, F. C., "Problemas de Validade em Classificação Automática", *Actas do III Colóquio de Estatística e Operação Investigação Operacional*, SPEIO, Lagos, 1984.

[POD 00] PODANI, J., "Simulation of Random Dendrograms and Comparison Tests: Some Comments", *Journal of Classification*, vol. 17, 2000, p. 123-142.

[SIB 72] SIBSON, R., "Order Invariants Methods for Data Analysis", *Journal of the Royal Statistical Society, Series B*, nº 34, 1972, p. 311-349.

[SOK 63] SOKAL, R. R., SNEATH, P. H. A., *Principles of Numerical Taxonomy*, Freeman, San Francisco, 1963.

[SOU 00] SOUSA, F., *Novas Metodologias em Classificação Hierárquica Ascendente*, Dissertação de Doutoramento, Universidade Nova de Lisboa, Lisboa, 2000.

[SOU 01] SOUSA, F., NICOLAU, F., "Uma Abordagem ao Problema da Comparação de Estruturas Classificatórias", in *A Estatística em Movimento* (M. M. Neves, J. Cadima, M. J. Martins e F. Rosado, eds.), Sociedade Portuguesa de Estatística, 2001, p. 409-418.

Les treillis de Galois pour l'organisation et la gestion des connaissances

Laszlo Szathmary, Amedeo Napoli

Équipe Orpailleur
LORIA/INRIA-Lorraine, 615 rue du Jardin Botanique, BP 101
F-54600, Vandœuvre-lès-Nancy, France
{szathmar,napoli}@loria.fr

RÉSUMÉ. *Dans cet article, nous étudions l'application des treillis de Galois sur différentes sources d'information ou de données (par exemple des documents du Web ou des notices bibliographiques) afin d'organiser les connaissances qui peuvent en être extraites. Dans notre cas, les connaissances sont organisées en un treillis de Galois. Cette organisation des connaissances peut alors être utilisée pour un certain nombre de buts, comme par exemple la gestion de la connaissance dans une organisation, la recherche documentaire sur le Web, etc. De plus, une base de connaissances, ou ontologie, peut s'appuyer sur la structure de treillis de Galois. Notre objectif global est de mettre en place un processus de classification par treillis pour enrichir une ontologie qui à son tour permet de guider le processus de découverte de connaissances dans les données.*

MOTS-CLÉS : *gestion des connaissances, treillis de Galois, ontologie, découverte des connaissances*

1. Introduction

Dans cet article[1], nous cherchons à analyser le travail global d'une organisation. Dans notre cas, nous avons considéré une équipe de recherche comme une petite entreprise. L'expérience montre que fréquemment les chercheurs, pouvant même appartenir à la même équipe, au même laboratoire, ne savent pas exactement sur quoi travaillent les autres chercheurs. Notre but est de trouver des interconnexions entre les travaux des différents membres pour faire émerger et comprendre les orientations de recherche principales/marginales dans l'équipe, et ainsi fournir des explications sur le travail de recherche.

Dans une équipe de recherche, les publications sont une bonne façon de décrire les centres d'intérêts d'un chercheur. C'est pourquoi nous avons choisi d'analyser les notices bibliographiques de l'équipe. Nous avons travaillé avec les descriptions BibTeX qui nous fournissent les méta-données d'un article, par exemple le titre, les auteurs, les mots-clés, le résumé, l'année de publication, etc. Une entrée BibTeX s'appuie globalement sur le standard du Dublin Core (`http://www.dublincore.org`). La norme Dublin Core comprend un ensemble d'éléments simples et représentatifs décrivant les caractéristiques des ressources sur le réseau, articles scientifiques en particulier. Cette norme est généralement utilisée pour la gestion des méta-données dans les pages HTML. Les descriptions BibTeX ayant un "vocabulaire contrôlé", c'est-à-dire un ensemble limité et consistant de termes bien définis, peuvent être considérées comme étant alignées sur le Dublin Core.

Pour analyser les publications, nous avons utilisé la classification par treillis comme technique d'extraction des connaissances à partir de données. En général, une ontologie fournit un modèle des connaissances d'un domaine et peut être utilisée de façon partagée dans différentes applications. Dans notre cas, nous avons construit une

1. Ce travail de recherche est réalisé en partie dans le cadre d'un programme de recherche franco-hongrois Balaton (Balaton F-23/03).

ontologie exprimant certains éléments de connaissances sur les membres et les publications de l'équipe. En utilisant cette ontologie, nous pouvons essayer de mettre en place un processus de fouille et d'analyse de données sur les publications de l'équipe.

Dans cet article nous avons étudié plus précisément les interconnexions entre les personnes, les publications et les thématiques. L'ontologie construite sert à guider la classification, en tenant compte de certaines erreurs et d'une certaine redondance dans les données. Nous montrons et discutons les premiers résultats obtenus.

L'article est organisé comme suit. Dans la section 2, nous présentons le rôle des ontologies dans le processus d'organisation des données avec une classification par treillis, et nous détaillons l'étape de suppression d'erreurs. La section 3 décrit le processus de classification proprement dit avec un exemple. Dans la section 4, nous discutons les perspectives du travail de recherche présenté dans cet article.

2. Ontologies

Les ontologies facilitent le partage et la réutilisation des connaissances. Plusieurs ontologies simples sont disponibles sur le Web, comme celles de la bibliothèque DAML Ontology[2] ou le projet DMOZ[3] (Directory Mozilla). À notre connaissance, il n'existait pas d'ontologie de thématiques de recherche, et nous avons donc décidé d'en construire une pour les besoins de notre travail de recherche. Cette ontologie va servir de base de connaissances de référence pour le domaine considéré et elle va également nous servir à guider le processus de classification (corrections d'erreurs, gestion de synonymes).

2.1. *Ontologies dans le processus d'ECBD*

L'extraction de connaissances dans les bases de données (ECBD) est un processus qui cherche à extraire des unités de connaissances nouvelles et réutilisables dans de grands volumes de données [FAY 96, GOE 99]. Ce processus peut être guidé à la fois par un analyste, qui est un spécialiste du domaine, et par une ontologie portant sur le domaine des données. En retour, les résultats du processus d'ECBD peuvent venir enrichir l'ontologie considérée.

Notre travail de recherche porte plus spécialement sur l'organisation en un treillis de connaissances relatives à une équipe de recherche. En particulier, la classification par treillis est ici une des techniques d'ECBD qui est considérée. Pour mettre en oeuvre la classification par treillis, nous exploitons une ontologie qui porte sur les thématiques de recherche étudiées dans l'équipe, et qui nous permet de :

- Corriger des erreurs d'orthographe et de traiter des problèmes de synonymie en regroupant les diverses étiquettes attribuées à une même information.
- Guider le processus de classification en permettant de considérer les connaissances d'un domaine à différents niveaux de granularité, selon qu'un élément de connaissance est plus ou moins spécifique par rapport à l'ontologie. La prise en compte du niveau de généralité permet de construire une famille de treillis sur le même ensemble de données.

De plus, la classification par treillis peut être utilisée en parallèle comme un module de base pour la recherche d'informations selon certains critères, comme cela est introduit dans [CAR 00]. Nous avons également exploité cette approche, mais il n'en sera pas question dans cet article.

2.2. *La correction d'erreurs et la mise en facteur d'information avec une ontologie*

Dans cette section, nous détaillons pourquoi et comment une ontologie peut être utilisée pour guider le processus d'extraction et d'organisation de connaissances à partir d'une base de données.

2. http://www.daml.org/ontologies/
3. http://www.dmoz.org

Divers problèmes ont pu être traités grâce aux connaissances intégrées à l'ontologie :

– Les publications sont indexées par des mots-clés qui sont donnés manuellement. Ces données sont très souvent entachées d'erreurs d'orthographe. Si un mot-clé est syntaxiquement incorrect, alors une recherche sur la base du même mot-clé écrit correctement ne peut pas aboutir.
– Synonymes : les mots-clés peuvent avoir divers synonymes. Comme l'association d'un mot-clé à un document ne s'appuie pas sur des règles bien définies ou une grammaire précise, plus d'un mot-clé peut être attaché à un document pour la même thématique. Dans ce cas, une recherche de documents doit pouvoir aboutir en utilisant n'importe lequel des synonymes.
– Langues : comme nous travaillons avec une bibliographie où il y a au moins deux langues, français et anglais, des mots-clés peuvent être employés dans les deux langues. Ceci est pris en considération dans l'ontologie de la même façon que les synonymes : les mots-clés sont donc fournis dans les deux langues étudiées.

Nous avons exploité l'ontologie des thématiques de la façon suivante. Tous les mots-clés qui servent à indexer les publications de l'équipe ont été répertoriés et "filtrés" à l'aide de l'ontologie : le filtrage a permis de regrouper en une même classe les synonymes, les mots-clés de même nature et de langues différentes, les variantes syntaxiques. Cependant, si ce processus fonctionne bien pour un nombre limité de documents, il est difficile de l'étendre à l'échelle du Web car la liste des variantes pour les thématiques de recherche est impossible à construire. Un processus de classification approximative dans ce cas reste à construire.

Pratiquement, 147 publications auxquelles sont attachés 335 mots-clés différents ont été analysées. L'ensemble de 335 mots-clés s'est réduit à 89 mots-clés après filtrage. Par exemple, l'ensemble de mots-clés suivants ('DL', 'DLs', 'case-based problem solving', 'CBR', 'galois connection') se transforme après le filtrage en ('description logics', 'case-based reasoning', 'Galois lattices').

3. L'organisation de documents avec des treillis de Galois

Les treillis de Galois permettent d'organiser en treillis des données se présentant sous la forme d'un tableau binaire *individus* $\times$ *propriétés* associé à une relation. Cette problématique est également appelée et étudiée en tant qu'analyse de concepts formels [GAN 99a, GAN 99b].

Dans ce paragraphe, nous nous intéressons plus particulièrement à la classification par treillis (de Galois) de documents ayant une thématique commune. Le treillis résultant va donner des indications sur la proximité ou l'éloignement des thématiques pour les membres de l'équipe, mais aussi sur les groupes d'auteurs publiant ensemble sur un sujet donné.

Ainsi, pour étudier une interaction entre les documents x qui traitent d'une thématique y, il est possible de construire le treillis de Galois de la relation "le document x traite de la thématique y". Rappelons qu'une telle relation est une donnée de base de notre problème.

Pour traiter cet exemple, nous avons pris un tableau de départ volontairement simplifié (voir tableau 1) où figurent 5 individus et 6 thématiques (ontologies, web sémantique, raisonnement à partir de cas, règles d'association, bioinformatique, adaptation). Le treillis de Galois associé à ce tableau booléen est donné à la figure 1, où les concepts sont donnés sous forme de couples {extension} $\times$ {intension}. Par exemple, il est possible de voir pour un concept quels auteurs travaillent sur un sujet donné, sachant qu'une clé comme *cadot03b* permet d'accéder à la notice bibliographique correspondante et au champ "Auteur" associé.

	ont.	sw	cbr	assoc. rules	bioinfo.	adapt.
cadot03b				x		
cherfi03c				x	x	
daquin02a						x
daquin03a	x	x	x			
lieber02a			x			x

Tableau 1. Tableau d'entrée d'articles $\times$ mots-clés

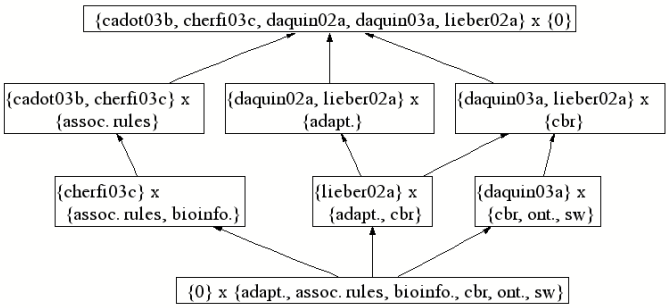


Figure 1. Treillis de Galois regroupant les documents selon leurs thématiques

En outre, il est encore possible d'extraire à partir de treillis des règles d'association, comme par exemple ici : *bioinformatique* $\Rightarrow$ *règles d'association*. Les règles donnent un point de vue alternatif sur les données étudiées. Pour plus d'informations sur les régles d'association, consultez [AGR 96, KLE 94, PAS 99].

En changeant le point de vue d'entrée, c'est à dire la relation et donc le tableau booléen, il est possible d'avoir un point de vue différent sur les données. Ainsi, nous avons travaillé sur trois relations différentes, *personnes* $\times$ *mots-clés*, *mots-clés* $\times$ *documents*, et *documents* $\times$ *personnes*. A chaque point de vue nous avons associé un treillis, qui permet de structurer les données et de répondre à des questions du type "quel x est en relation R avec quel y".

4. Conclusion et perspectives

Dans cet article, nous avons brièvement montré comment organiser les informations sur un domaine sous la forme d'un treillis, pour essayer ensuite de satisfaire des requêtes et rechercher des informations. Ce travail demande à être continué et approfondi, notamment sur l'organisation multi-dimensionnelle des informations en plusieurs treillis, ainsi que sur la façon de pouvoir gérer conjointement ces treillis, et enfin sur la construction de familles de treillis en fonction d'une ontologie du domaine et de la granularité des connaissances considérées.

5. Bibliographie

[AGR 96] AGRAWAL R., MANNILA H., SRIKANT R., TOIVONEN H., VERKAMO A. I., Fast discovery of association rules, *Advances in knowledge discovery and data mining*, p. 307–328, American Association for Artificial Intelligence, 1996.

[CAR 00] CARPINETO C., ROMANO G., Order-Theoretical Ranking, *Journal of the American Society for Information Science*, vol. 51, n^o 7, 2000, p. 587–601, John Wiley & Sons, Inc.

[FAY 96] FAYYAD U., PIATETSKY-SHAPIRO G., SMYTH P., From data mining to knowledge discovery in databases, *AI Magazine*, vol. 17, 1996, p. 37–54.

[GAN 99a] GANTER B., Attribute exploration with background knowledge, *Theoretical Computer Science*, vol. 217, n^o 2, 1999, p. 215–233, Elsevier Science Publishers Ltd.

[GAN 99b] GANTER B., WILLE R., *Formal concept analysis : mathematical foundations*, Springer, Berlin/Heidelberg, 1999.

[GOE 99] GOEBEL M., GRUENWALD L., A survey of data mining and knowledge discovery software tools, *SIGKDD Explorations Newsletter*, vol. 1, n^o 1, 1999, p. 20–33, ACM Press.

[KLE 94] KLEMETTINEN M., MANNILA H., RONKAINEN P., TOIVONEN H., VERKAMO A. I., Finding interesting rules from large sets of discovered association rules, *Proceedings of the third international conference on Information and knowledge management*, ACM Press, 1994, p. 401–407.

[PAS 99] PASQUIER N., BASTIDE Y., TAOUIL R., LAKHAL L., Efficient mining of association rules using closed itemset lattices, *Inf. Syst.*, vol. 24, n^o 1, 1999, p. 25–46, Elsevier Science Ltd.

Use of PLS Regression and PLS Path Modeling for Multiple Table Analysis

Michel Tenenhaus

HEC-School of Management, 78351-Jouy-en-Josas (France)
tenenhaus@hec.fr

RÉSUMÉ. *A situation where J blocks of variables are observed on the same set of statistical units is considered in this paper. A factor analysis logic is applied to tables instead of individuals. The latent variables of each block should well explain their own block and in the same time the latent variables of same rank should be as positively correlated as possible. In the first part of the paper we describe the hierarchical PLS path model and remind that it allows to recover some usual multiple table analysis methods. In the second part we suppose that the number of latent variables can be different from one block to another and that these latent variables are orthogonal. PLS regression and PLS path modeling may be used for this situation.*

MOTS-CLÉS : *Multiple factor analysis, PLS regression , PLS path modeling, Generalized canonical correlation analysis.*

1. Introduction

We consider in this paper a situation where J blocks of variables $X_1, ..., X_J$ are observed on the same set of n statistical units. All variables are supposed to be standardized. We can follow a factor analysis logic on tables instead of variables. In the first section of this presentation we suppose that each block X_j with dimension $n \times k_j$ is multidimensional and is summarized by m latent variables plus a residual E_j. Each data table is decomposed into two parts : $X_j = \{t_{j1}p_{j1}^{'}+...+t_{jm}p_{jm}^{'}\}+E_j$ where t_{jh} is an n-dimension column vector and p_{jh} a k_j-dimension column vector. The first part of the decomposition is $t_{j1}p_{j1}^{'}+...+t_{jm}p_{jm}^{'}$. The latent variables $t_{j1}, ..., t_{jm}$ should well explain the data table X_j and in the same time the latent variables of same rank h, $t_{1h}, ..., t_{Jh}$, should be as *positively* correlated as possible. The second part of the decomposition is the residual E_j which represents the part of X_j not related to the other block, i.e. the specific part of X_j. We show that the PLS approach allows to recover some usual methods for multiple table analysis. In section two we suppose that the number of latent variables can be different from one block to another and that these latent variables are orthogonal. PLS regression and PLS path modeling may be used for this situation.

2. Multiple Table Analysis : a classical approach

In Multiple Table Analysis it is usual to introduce a super-block X_{J+1} obtained by concatenating all the blocks X_j. This super-block is summarized by m latent variables $t_{J+1,1}, \ldots, t_{J+1,m}$ also called auxiliary variables. The path model describing this situation is given in Figure 1. This model corresponds to the hierarchical model proposed by Wold (1982). The latent variables $t_{j1}, \ldots, t_{jm}$ should well explain their own block X_j. In the same time the latent variables of same rank $(t_{1h}, \ldots, t_{Jh})$ and the auxiliary variable $t_{J+1,h}$ should be as *positively* correlated as possible. In some usual Multiple Table Analysis (= MTA) methods, as Horst's (1961) and Carroll's (1968) Generalized Canonical Correlation Analysis, orthogonality constraints are imposed on the auxiliary variables $t_{J+1,h}$ and the latent variables t_{jh} related to block j have no orthogonality constraints. We define for the super-block X_{J+1} the sequence of blocks $E_{J+1,h}$ obtained by deflation : each block $E_{J+1,h}$ is defined as the residual of the

regression of X_{J+1} on the latent variables $t_{J+1,1}, ..., t_{J+1,h}$. Figure 2 corresponds to step h. For computing the latent variables t_{jh} and the auxiliary variables $t_{J+1,h}$ we use the general PLS algorithm (Wold, 1985) defined as follows for step h of this specific application :

External estimation :

- Each block X_j is summarized by the latent variable $t_{jh} = X_j w_{jh}$

- The super-block $X_{J+1,h}$ is summarized by the latent variable $t_{J+1,h} = E_{J+1,h-1} w_{J+1,h}$

Internal estimation :

- Each block X_j is also summarized by the latent variable $z_{jh} = e_{jh} t_{J+1,h}$, where e_{jh} is the sign of the correlation between t_{jh} and $t_{J+1,h}$. We will however choose $e_{jh} = +1$ and show that the correlation is then positive.

- The super-block $E_{J+1,h-1}$ is summarized by the latent variable $z_{J+1,h} = \sum_{j=1}^{J} e_{J+1,j,h} t_{jh}$, where $e_{J+1,j,h}$ $= +1$ when the centroid scheme is used, or the correlation between t_{jh} and $t_{J+1,h}$ for the factorial scheme, or finally the regression coefficient of t_{jh} in the regression of $t_{J+1,h}$ on $t_{1h}, \ldots, t_{Jh}$ for the path weighting scheme.

We can now describe the PLS algorithm for the J-block case. The weights w_{jh} can be computed according to two modes : mode A or B.

In mode A simple regression is used :

$$w_{jh} \propto X_j^{'} t_{J+1,h},\ j = 1 \text{ to } J, \text{ and } w_{J+1,h} \propto E_{J+1,h-1}^{'} z_{J+1,h} \tag{1}$$

where $\propto$ means that the left term is equal to the right term up to a normalization.

For mode B multiple regression is used :

$$\begin{aligned} w_{jh} &\propto (X_j^{'} X_j)^{-1} X_j^{'} t_{J+1,h},\ j = 1 \text{ to } J, \\ \text{and } w_{J+1,h} &\propto (E_{J+1,h-1}^{'} E_{J+1,h-1})^{-1} E_{J+1,h-1}^{'} z_{J+1,h} \end{aligned} \tag{2}$$

The normalization depends upon the method used. For some method w_{jh} is of norm 1. For other methods the variance of t_{jh} is equal to 1.

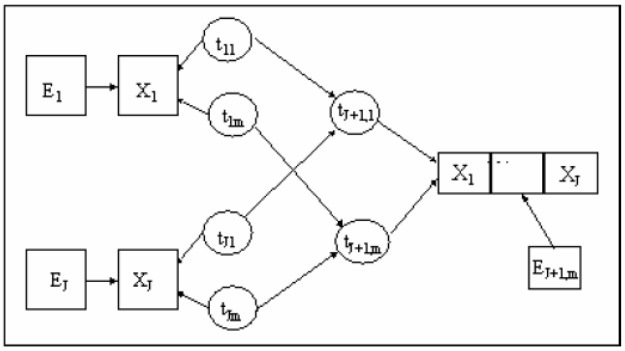


Figure 1 : *Path model for the J-block case*

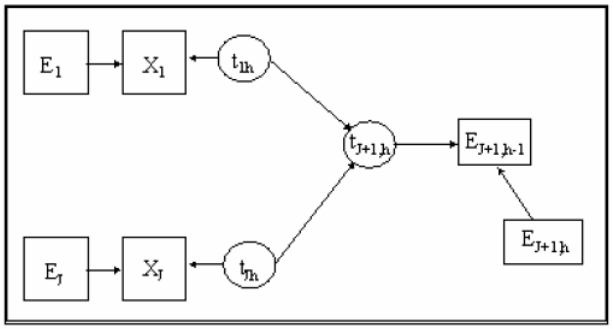


Figure 2 : *Path model for the J-block case : Step h*

It is now easy to check that the correlation between t_{jh} and $t_{J+1,h}$ is always positive : $t'_{J+1,h}t_{jh} = t'_{J+1,h}X_j w_{jh} \propto t'_{J+1,h}X_j X'_j t_{J+1,h} > 0$ when mode A is used. The same result is obtained when mode B is used.

The PLS algorithm can now be described. We begin by an arbitrary choice of weights w_{jh}. We get the external estimations of the latent variables, then the internal ones. Using the equations (1) or (2) we get new weights. This procedure is iterated until convergence always verified in practice, but only mathematically proven for the two-block case. The various options of PLS Path Modeling (mode A or B for external estimation ; centroid, factorial or path weighting schemes for internal estimation) allow to unify within a common framework many methods for Multiple Table Analysis : Generalized Canonical Analysis (the Horst's one (1961) and the Carroll's one (1968)), Multiple Factor Analysis (Escofier & Pagès, 1988), Lohmöller's split principal component analysis (1989), Horst's maximum variance algorithm (1965). The links between PLS and these methods have been demonstrated in Lohmöller (1989) or Tenenhaus (1999) and studied on practical examples in Guinot, Latreille and Tenenhaus (2001) and Pagès and Tenenhaus (2001). These various methods are obtained by using the PLS algorithm according to the options described in Table 1. The super-block only is deflated ; the original blocks are not deflated.

Scheme of calculation for the inner estimation	Mode of calculation for the outer estimation	
	A	*B*
Centroid	PLS Horst's generalized canonical correlation analysis	Horst's generalized canonical correlation analysis (SUMCOR criterion)
Factorial	PLS Carroll's generalized canonical correlation analysis	Carroll's generalized canonical correlation analysis
Path weighting scheme	- Lohmöller's split principal component analysis - Horst's maximum variance algorithm - Escofier & Pagès Multiple Factor Analysis	

No deflation on the original blocks, deflation on the super-block

Table 1 : *Multiple Table Analysis and PLS algorithm*

Discussion on the orthogonality constraints

There is some advantage on imposing orthogonality constraints only on the latent variables related to the super-block : no dimension limitation due to block sizes. If orthogonality constraints were imposed on the block latent variables, then the maximum m of latent variables would be the size of the smallest block. The super-block X_{J+1} is summarized by *m* orthogonal latent variables $t_{J+1,1}, \ldots, t_{J+1,m}$. Each block X_j is summarized by m latent variables $t_{j1}, \ldots, t_{jm}$. But these latent variables can be highly correlated and consequently don't reflect the real dimension of the block. In each block X_j the latent variables $t_{j1}, \ldots, t_{jm}$ represent the part of the block correlated with the other blocks. A principal component analysis of these latent variables will gives the actual dimension of this part of X_j. It can be preferred to impose orthogonality on the latent variables of each block. But we have to remove the dimension limitation due to the smallest block. This situation is going to be discussed in the next section.

3. Multiple Table Analysis : new perspectives

We describe in this section a new approach more focused on the blocks than on the super-block. This approach is called PLS-MTA : a PLS approach to Multiple Table Analysis. We now suppose a variable number of common components in each block :

$$X_j = t_{j1}p_{j1}^{'} + ... + t_{jm_j}p_{jm_j}^{'} + E_j \tag{3}$$

A two steps procedure is proposed to find these components.

Step 1

For each block X_j we define the super-block X_{-j} obtained by concatenating all the other blocks X_i for $i \neq j$. For each j we carry out a PLS regression of X_{-j} on X_j. So we obtain m_j orthogonal and standardized PLS components $\tilde{t}_{j1}, ..., \tilde{t}_{jm_j}$ which represent the part of X_j related with the other blocks. The choice of the number m_j of components is determined by cross-validation.

Step 2

One of the procedures described in Table 1 is used on the blocks $\tilde{T}_j = \left\{\tilde{t}_{j1}, ..., \tilde{t}_{jm_j}\right\}$ for h = 1. We obtain the rank one components $t_{11}, \ldots, t_{J1}$ and $t_{J+1,1}$. Then, to obtain the next components we only consider the blocks with $m_j > 1$. For these blocks we construct the residual $\tilde{T}_{j1}$ of the regression of $\tilde{T}_j$ on t_{j1}. A MTA is then applied on these blocks and we obtain the rank two components $t_{12}, \ldots, t_{J2}$ (for *j* with $m_j > 1$ and $t_{J+1,2}$. The components t_{j1} and t_{j2} are uncorrelated by construction, but the auxiliary variables $t_{J+1,1}$ and $t_{J+1,2}$ can be slightly correlated as we did not impose orthogonality constraint on these components. This research of components is iterated until the various m_j common components are found. These components can finally be expressed in term of the original variables. There is a great advantage on imposing orthogonality constraints on each block components : the new m_j orthogonal and standardized components $t_{j1}, \ldots, t_{jm_j}$ are deduced from the m_j orthogonal and standardized PLS components $\tilde{t}_{j1}, ..., \tilde{t}_{jm_j}$ by a rotation. That means that

$$[t_{j1}, \ldots, t_{jm_j}] = [\tilde{t}_{j1}, ..., \tilde{t}_{jm_j}]A_j \tag{4}$$

where A_j is an orthogonal (rotation) matrix.

4. Bibliographie

[CAR 68] CARROLL J. D., A generalization of canonical correlation analysis to three or more sets of variables, *Proc. 76th Conv. Am. Psych. Assoc.*, 1968, p. 227-228.

[ESC 88] ESCOFIER B., PAGÈS J., *Analyses factorielles simples et multiples*, Dunod, Paris, 1988.

[GUI 01] GUINOT C., LATREILLE J., TENENHAUS M., PLS Path modellind and multiple table analysis. Application to the cosmetic habits of women in Ile-de-France, *Chemometrics and Intelligent Laboratory Systems*, vol. 58, 2001, p. 247-259.

[HOR 61] HORST P., Relations among m sets of variables, *Psychometrika*, vol. 26, 1961, p. 126-149.

[HOR 65] HORST P., *Factor Analysis of Data Matrices*, Holt, Rinehart and Winston, New York., 1965.

[LOH 65] LOHMOLLER J.-B., *Latent Variables Path Modeling with Partial Least Squares*, Physica-Verlag, Heildelberg, 1965.

[PAG 01] PAGÈS J., TENENHAUS M., Multiple factor analysis combined with PLS path modeling. Application to the analysis of relationships between physico-chemical variables, sensory profiles and hedonic judgements, *Chemometrics and Intelligent Laboratory Systems*, vol. 58, 2001, p. 261-273.

[TEN 99] TENENHAUS M., L'approche PLS, *Revue de Statistique Appliquée*, vol. 47, 1999, p. 5-40.

[WOL 82] WOLD H., Soft Modeling : The Basic Design and Some Extensions, *Systems under indirect observation, Part 2*, K.G. Joreskog & H. Wold (Eds),North-Holland,Amsterdam, 1982, p. 1-54.

[WOL 85] WOLD H., Partial Least Squares, *Encyclopedia of Statistical Sciences*, vol. 6, Kotz, S & Johnson, N.L. (Eds), John Wiley & Sons, New York, 1985, p. 581-591.

Classification de processus ARMA basée sur l'analyse structurelle.

Carole Toque

Ecole Nat. Sup. des Télécommunications
46, rue Barrault,
75634 Paris Cedex 13
Email : carole.toque@educagri.fr

RÉSUMÉ. Pour l'identification des processus ARMA, la méthode proposée combine à la fois une approche structurelle par analyse des points de retournements et de critères d'information avec des techniques de classification (classification ascendante hiérarchique (CAH)) et factorielle (analyse des correspondances multiples (ACM)). La méthode est appliquée à des séries AR(1) et MA(1) simulées.

MOTS-CLÉS : Analyse structurelle, processus ARMA, identification de processus, théorie de l'information, entropie, mesure d'incertitude, analyse des correspondances multiples, classification hiérarchique ascendante.

1. Introduction

Pour la prévision des séries chronologiques, la méthodologie historique de Box et Jenkins [BOX 76], basée essentiellement sur l'examen des fonctions d'autocorrélation (FAC) et d'autocorrélation partielle (FAP), reste toujours d'actualité. Cependant, l'étape d'identification de la chronique échantillon à la classe des processus ARMA linéaires et stationnaires, est délicate et un peu trop restrictive. Le recours à d'autres techniques peut donc se justifier lorsque se combinent par exemple des changements structurels dont on veut mesurer la puissance (reprise (ou pic) et essoufflement (ou creux)).
On propose alors une méthode qui utilise à la fois une approche structurelle avec l'analyse des points de retournement et la théorie de l'information, et une approche par des techniques de classification.
Une classification est le résultat d'une succession de choix : le choix de la matrice des données initiales (quelles sont les variables à utiliser ?), le choix de la distance, le choix de la méthode de classification et le choix de la méthode d'agrégation pour une classification hiérarchique.
Précisément, les matrices initiales étudiées sont successivement la matrice temporelle issue de la simulation de processus AR(1) et MA(1), la matrice des points de retournements et la matrice des mesures 'entropiques' des séries simulées. Enfin, la méthode de classification retenue est la méthode hiérarchique ascendante avec le plus souvent la distance euclidienne ou la distance binaire pour la matrice des vecteurs d'état (0) ou (1), et le critère du lien complet d'agrégation, à l'exception de la CAH sur facteurs d'une ACM pour laquelle le critère de Ward est utilisé.

2. «Séparabilité» de processus ARMA en classes

2.1. *Les simulations de trajectoires et la matrice initiale temporelle*

18 processus AR(1) et 18 processus MA(1) centrés et stationnaires, chacun de longueur 500, ont été générés à partir des valeurs de coefficients ϕ_1 et θ_1, comprises entre -0.9 et +0.9 et avec un pas de 0.1.

Par ailleurs, nous avons fixé comme valeurs des paramètres du modèle : la variance du bruit ($\sigma_u^2 = 90$) et une valeur de calage générateur aléatoire, d'un bruit dont la loi est posée gaussienne et centrée.
La matrice temporelle à analyser est donc de dimension (36x500) et les observations sont les processus AR(1) de coefficients -0.9 à -0.1 numérotés de 1 à 9 puis de coefficients +0.1 à +0.9 numérotés de 19 à 27, et les processus MA(1) de coefficients -0.9 à -0.1 numérotés de 10 à 18 puis de coefficients +0.1 à +0.9 numérotés de 28 à 36.

2.2. *Classification hiérarchique sur la matrice temporelle*

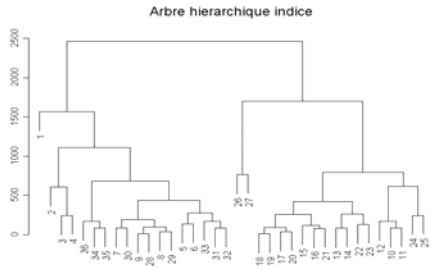


Fig. 1. Dendrogramme

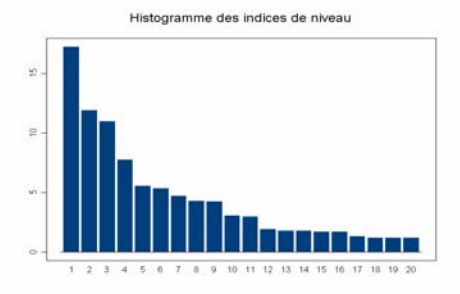


Fig. 2. Histogramme des indices de niveau

La CAH obtenue avec la distance euclidienne et le lien complet d'agrégation, est représentée par le dendrogramme ci-dessus (Fig. 1.) et est complétée par le graphique des 'indices de niveaux par ordre décroissant' (Fig. 2.) pour le choix du nombre de classes.
On peut alors constater que l'arbre hiérarchique ne présente pas de 'forts' effets de chaînage : les individus se répartissent en classes. Cette structure globale de l'arbre est déjà un bon indicateur de la 'séparabilité' des processus AR(1) et MA(1) en classes.
Pour ce qui est du choix du niveau de coupure de l'arbre, on retient une séparation en 4 classes avec, d'une part le seul processus AR(1) à 'fort' coefficient (-0.9) en module, d'autre part les 2 processus AR(1) à 'forts' coefficients positifs (+0.8 et +0.9), puis la classe des AR(1) à coefficients négatifs et des MA(1) à coefficients positifs, et la classe des MA(1) à coefficients négatifs et des AR(1) à coefficients positifs. A l'exception des deux premières classes qui isolent les processus AR(1) à très 'forts' coefficients en module, on retrouve certaines propriétés de symétrie des comportements de la FAC d'un AR(1) et de la FAP d'un MA(1), et de la FAP d'un AR(1) et de la FAC d'un MA(1).
Ces premiers résultats montrent que la classification est un outil d'aide à l'identification de processus ARMA. Cependant, il manque la description des classes pour renforcer la méthode : on propose alors de recourir à l'analyse structurelle.

3. Identification structurelle de processus ARMA

3.1. *L'analyse structurelle et les mesures d'incertitude*

Pour décrire et mesurer les changements structurels d'une série temporelle, on choisit de transformer la série initiale en une série de points de retournements ou série d'états. On construit par différences premières des données, une série de symboles (0) ou (1) correspondant respectivement aux 'pics' ou aux 'creux' de la série initiale (Kendall et Stuart [KEN 76]). Puis, on mesure les fréquences des symboles et séquences de k symboles sur chaque série d'états pour estimer les différentes probabilités.
Cependant, comment qualifier l'information dont on dispose sur ces probabilités ? Ou encore, comment qualifier l'incertitude ? C'est bien sûr la théorie de l'information qui intervient avec les mesures entropiques de divers ordres (Shannon [SHA 48] et, Yaglom et Yaglom [YAG 59]).

Un projet développé en Fortran 90 puis intégré à 'Splus' permet de construire les séries d'états, d'estimer les probabilités $P_k = \left\{ p_{k,1}, p_{k,2} ..., p_{k,m^k} \right\}$ des m^k k-uplets d'états d'une série $\{Z_t\}$, de calculer les entropies simples H_k d'ordre k (notées aussi 'Shk') définies par $H_k(P_k) = -\sum_{i=1}^{m^k} p_{k,i} \log_2 p_{k,i}$, de calculer les entropies conditionnelles h_k d'ordre k (notées aussi 'Condk') définies par $h_1 = H_1(P_1)$ sinon $h_k = H_k(P_k / P_{k-1}) = H_k(P_k) - H_{k-1}(P_{k-1})$, et de calculer les entropies résiduelles d_k d'ordre k (notées aussi 'Resk') définies par $d_k = h_k - h_{k+1}$, pour k allant de 1 à q ou (q-1) pour d_k.

Par exemple, l'entropie d_k s'interprète comme la réduction moyenne d'incertitude sur un symbole selon qu'on connaît le k-gramme précédent plutôt que le $(k-1)$–gramme.
Enfin, le projet est appliqué aux séries AR(1) et MA(1) déjà simulées, avec comme valeur de paramètre q=5 pour les calculs des fréquences et des entropies. Il en résulte pour chaque processus : un vecteur d'états de (0) et de (1) et un vecteur des entropies de dimension (14).

3.2. *Classification sur la matrice des points de retournements*

Fig. 3. Dendrogramme

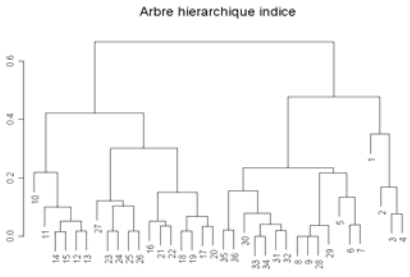
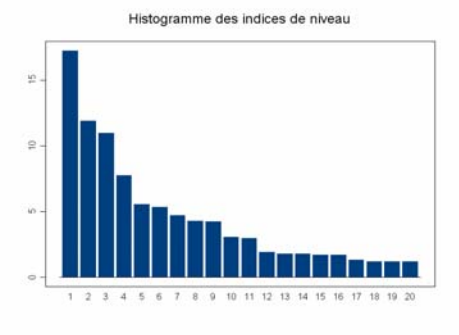


Fig. 4. Histogramme des indices de niveau



Pour cette première CAH ‘structurelle’, la matrice initiale des données est la matrice des 36 vecteurs de (0) et (1), la métrique (pour le calcul de la matrice des distances) est la métrique ‘binaire’ (la distance entre deux vecteurs lignes est le nombre d’occurrences de (01) ou de (10) divisé par le nombre de colonnes où au moins un de ces individus a un (1)) et la méthode d’agrégation est le lien complet.
L’arbre en Fig.3. présente des classes plus compactes avec moins d’effets de chaînage que dans la classification obtenue sur la matrice temporelle. Aussi, des classes de processus à coefficients ‘faibles à semiforts’ en module se différencient de classes de processus à coefficients ‘forts’ en module. En effet, pour une séparation en 6 classes, on distingue par exemple, des classes de processus dits ‘faibles à semi-forts’ (5-6-7-8-9 et 28-29) et (16-17-18 et 19-20-21-22) qui reflètent les résultats de la symétrie des comportements des fonctions d’autocorrélations, et une classe de processus dits ‘forts’ (1-2-3-4).
La méthode par l’analyse structurelle et la classification est encourageante. Pour faciliter l’interprétation des classes et pour faire apparaître des classes encore plus compactes, on choisit de construire une ACM à 3 modalités sur les vecteurs d’entropie.

3.3. *Classification sur composantes principales d’une ACM ‘structurelle’*

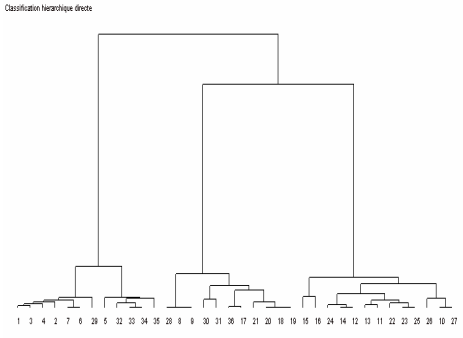


Fig. 5. Dendrogramme

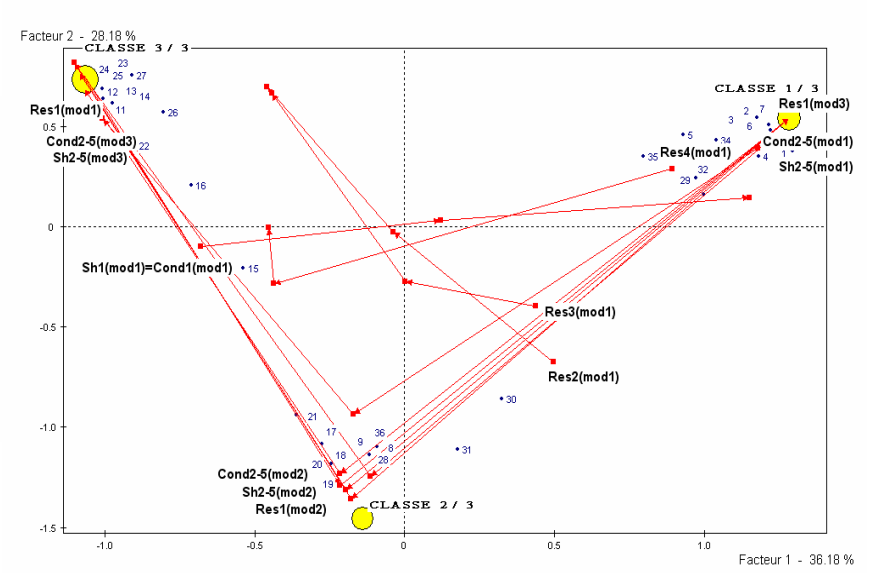


Fig. 6. Représentation des classes et trajectoires

Une classification réalisée après une ACM permet d'illustrer les graphiques de projection des variables (en nombre réduit) et des individus en y ajoutant les classes obtenues à l'issue de la CAH (Fig. 5.).
La lecture graphique (Fig. 6.) est en effet facilitée dans ce plan factoriel (F1,F2) qui explique à lui seul près de 64% de l'inertie totale et dans lequel les individus et caractères sont assez bien représentés ('sommes de carré de cosinus' souvent proches de 0.70).
On peut aussi mettre en évidence une non-linéarité entre les différents critères et traduire une éventuelle progression en reliant leurs modalités respectives.
On constate alors une certaine cohérence des données avec la présence de nombreuses lignes 'polygonales' régulières qui suivent 3 classes d'individus. La méthode et le choix du nombre des modalités semblent pertinents.
Il en résulte les rapprochements des variables et des classes d'individus dont les contributions à la formation des axes sont souvent voisines de 0.75. L'axe F1, avec surtout les entropies conditionnelles d'ordre 2 à 5 (modalités 1 et 3) et l'entropie résiduelle d'ordre 1 (modalités 3 et 1) sépare la classe des AR(1) à coefficients négatifs et des MA(1) à coefficients positifs, de la classe des MA(1) à coefficients négatifs et des AR(1) à coefficients positifs. L'axe F2 oppose du côté positif les processus dits 'forts' à 'semi-forts' aux processus dits 'faibles' du côté négatif, avec surtout des entropies conditionnelles d'ordre 2 à 5 (modalité 2) et une entropie résiduelle d'ordre 1 (modalité 2).
Cette ACM sur les entropies, complétée par une CAH, fait donc apparaître explicitement les modalités de mesures d'incertitude et de réduction d'incertitude pour caractériser aussi bien les 2 classes de processus dits 'forts' à 'semi-forts' que la classe des processus dits 'faibles'.

4. Conclusion

La méthode avec les mesures d'incertitude et la classification hiérarchique est justifiée et peut s'étendre à tous les processus ARMA.
Cependant, tout comme le choix des variables à utiliser est essentiel, le choix de la distance l'est aussi.
Par exemple, la distance 'binaire' utilisée peut être améliorée en tenant compte des k-uplets d'états des séries de points de retournement pour k>2. Ou encore, au-delà des méthodes de classification hiérarchique, il est possible de construire une partition non supervisée, basée sur l'entropie calculée à partir de fréquences de k-uplets d'états (ou points de retournements) d'une série quelconque.
Des résultats bien meilleurs sont alors attendus en estimant les distances entre points, non plus avec la norme L^2 comme il a été fait, mais avec la norme L^1 en vue de la prévision de processus linéaires et non linéaires.

5. Bibliographie

[BOX 76] BOX G.E.P., JENKINS G.M., *Time Series Analysis, Forecasting and Control*, Holden Day, San Francisco, 1976.

[KEN 76] KENDALL M., STUART A., *The advanced theory of statistics (Volume 3 : design and analysis, and time-series)*, Charles Griffin and Co Ltd, London, 1976.

[SHA 48] SHANNON C.E., *A Mathematical Theory of communication*, Bell Syst. Tech. J. 27, 1948, pp379-423.

[YAG 59] YAGLOM A. M., YAGLOM I. M., *Probabilité et Information*, Dunod, Paris, 1959.

De la variation des classifications en fonction des méthodes utilisées et des objets lexicaux étudiés. Une application au discours des acteurs en gouvernance d'entreprises

TREBUCQ Stéphane

CRECCI, Université Montesquieu Bordeaux 4
Avenue Léon Duguit,
33608 Pessac
trebucq@u-bordeaux4.fr

RÉSUMÉ. Au cours de l'année 2003, la mission d'information parlementaire Clément a interrogé différentes classes d'acteurs professionnels afin d'étudier les possibilités de réforme des modes de gouvernance des entreprises. Les données textuelles de ces auditions peuvent être traitées grâce aux méthodes statistiques de classification ascendante hiérarchique. Elles peuvent servir à identifier les acteurs les plus influents, en étudiant la proximité entre les discours des acteurs interrogés et la structure du rapport final rédigé par les députés. L'objet de cet article vise à présenter l'influence des objets lexicaux choisis et des méthodes de classification utilisées sur les résultats empiriques obtenus.

MOTS-CLÉS : classification ascendante hiérarchique, données textuelles, sensibilité des résultats, stabilité des résultats, logiciel Tropes, logiciel SPSS

1. Introduction

Le 16 octobre 2002, une mission d'information, placée sous la présidence de M. le Député Pascal Clément, a été confiée à un groupe de dix-neuf parlementaires de l'Assemblée Nationale, de toutes tendances politiques confondues. Le rapport final, déposé le 2 décembre 2003 à l'Assemblée Nationale, prend acte de la crise de confiance des marchés financiers, et propose de nouveaux principes afin de réorganiser le système de gouvernance des entreprises. Les conclusions auxquelles ont abouti les parlementaires ont été tirées partiellement de la consultation d'un panel d'une cinquantaine de professionnels qualifiés, représentant dix catégories d'acteurs de la vie économique (voir tableau 1). On peut toutefois s'interroger sur l'influence effective qu'ont pu exercer ces différentes catégories d'experts sur la réflexion du groupe de députés matérialisée par la rédaction d'un rapport final.

L'ensemble des auditions ayant été consigné par écrit, il est possible de procéder à diverses analyses lexicales. Celles-ci peuvent donner lieu, dans un second temps, à des traitements statistiques utilisant la classification hiérarchique [LEB 94]. Après constitution d'un tableau de correspondances croisant les différents objets lexicaux observés avec les onze catégories d'acteurs relevées (hommes politiques inclus), la technique de classification ascendante hiérarchique permet de tester la proximité existant entre le discours des différentes catégories d'experts et la présentation définitive du rapport rédigé par les parlementaires.

Tableau 1 - Synthèse statistique des auditions de la commission parlementaire

	Catégories	Abréviations utilisées dans les classifications	Nombre de personnes interrogées	en %	Nbre de mots	en %	Nbre de RN*	en %	Nbre de MR*	en %
01	Avocats	AVOCATS	10	20%	17359	14%	355	38%	354	15%
02	Dirigeants	DIRIGEAN	8	16%	32102	26%	495	53%	602	25%
03	Conseils	CONSEIL	7	14%	14358	11%	285	30%	257	11%
04	Rep. entreprises	ENTREPR	6	12%	20959	17%	326	35%	346	14%
05	Régulateur-Etat	REGULAT	5	10%	10324	8%	246	26%	162	7%
06	Expert-comptables	EXPCOMP	4	8%	9335	7%	229	24%	134	6%
07	Universitaires	UNIV	4	8%	8792	7%	194	21%	132	6%
08	Rep. actionnaires	ACTIONN	3	6%	5733	5%	123	13%	68	3%
09	Agence de notation	AGENCEN	1	2%	1373	1%	25	3%	15	1%
10	Rep. banques	BANQUE	1	2%	4682	4%	113	12%	58	2%
	Sous-total		*49*		*125017*		*937*		*2395*	
11	Hommes Politiques	HOMPOL	_		30433		552		965	

*** RN : référents-noyaux, MR : mises en relations, ou co-occurrences de référents-noyaux (voir la partie méthodologie)**

2. Méthodologie

La mise en œuvre du logiciel Tropes[1] permet de réaliser, dans un premier temps, une analyse propositionnelle [GHI 98] des auditions parlementaires. L'unité d'analyse retenue dans le texte n'est autre que la proposition grammaticale (sujet/verbe/complément). Il est alors possible de repérer non seulement les termes pivots, ou RN (référent-noyau), de chaque proposition mais également les mises en relations, ou MR, pouvant être opérées entre deux RN présents au sein d'une même proposition.

A titre d'exemple, dans des propositions du type « le dirigeant peut avoir un intérêt à accroître ses prélèvements non pécuniaires », ou encore « les dirigeants agissent dans les intérêts des anciens actionnaires », l'analyse automatique du logiciel Tropes sera en mesure d'identifier les termes clés tels que dirigeant et intérêt. Un classement sémantique sera ensuite constitué sous les catégories de « patron » (avec pour classe d'équivalents[2] : dirigeant, employeur, administrateur, gérants) et d' « intérêt » (avec pour classe de termes : intérêt ou taux d'intérêt). Il est également possible d'identifier dans le texte l'usage des verbes. S'agissant de l'étude des mises en relation entre les RN, elle fournit une description beaucoup plus précise des associations d'idées présentes dans le texte. Ainsi en se référant de nouveau aux deux propositions citées précédemment en exemple, on obtiendrait notamment la mise en relation suivante : patron→intérêt. On parlera alors de cooccurrence entre ces deux RN. La flèche qui les relie précise ici le sens de la relation indiquant que le RN « patron » précède le RN « intérêt » dans les propositions observées.

Trois matrices ou tableaux de correspondances, décrivant les discours des dix catégories d'experts, ainsi que la structure du rapport final rédigé par les députés[3], peuvent être de ce fait constituées. La première matrice est fondée sur les occurrences des RN. La seconde matrice retient uniquement les occurrences de verbes. Enfin, la troisième matrice est établie à partir des MR des RN, pris deux à deux au sein des propositions. L'objet de l'expérimentation qui suit consiste à étudier les variations observables dans les résultats empiriques obtenus en fonction de ces trois matrices, ainsi que les différences induites par l'usage de différentes méthodes de classification hiérarchique.

3. Résultats obtenus

La première matrice fondée sur les occurrences de RN est traitée successivement avec le logiciel SPSS, en fonction des trois méthodes suivantes de classification hiérarchique : distance moyenne entre classes (figure 1),

[1] Voir le site internet : http://www.acetic.fr

[2] Les classes d'équivalents regroupent les mots qui apparaissent fréquemment dans le texte et qui possèdent une signification très voisine.

[3] Comparer des discours d'auditions à un rapport rédigé peut représenter une première limite.

agrégation suivant le saut minimum (figure 2), et critère de Ward (figure 3). La mesure d'intervalle choisie est le carré de la distance euclidienne.

Figure 1. Classification en fonction des RN selon la méthode de la distance moyenne entre classes

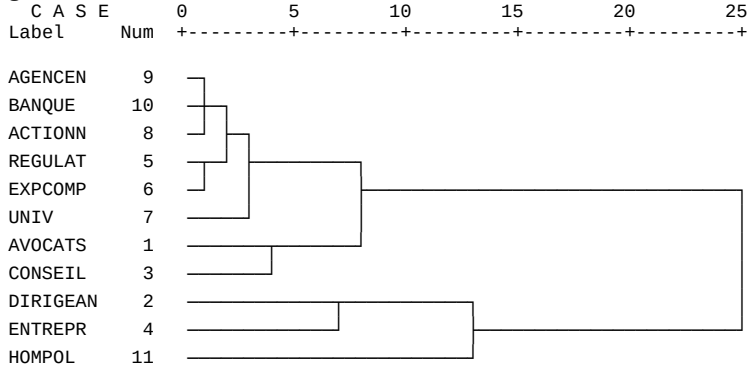


Figure 2. Classification en fonction des RN selon la méthode d'agrégation suivant le saut minimum

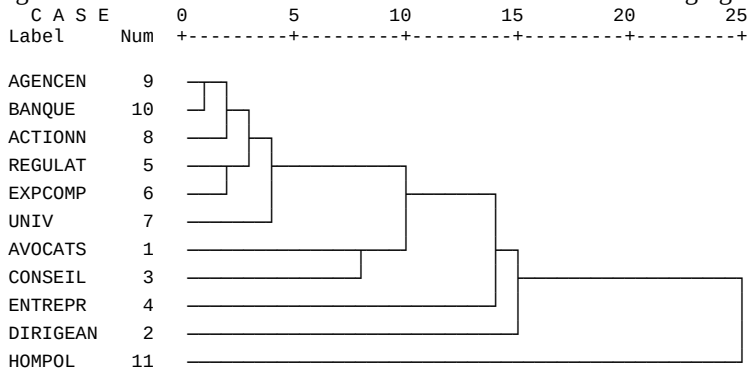
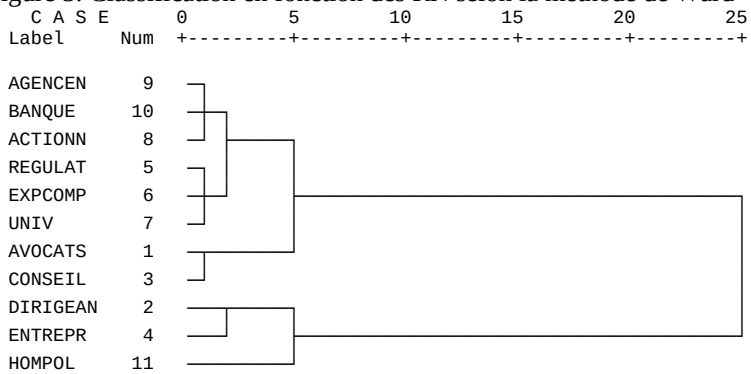


Figure 3. Classification en fonction des RN selon la méthode de Ward



On remarque que l'utilisation de trois méthodes conduit à des résultats sensiblement différents. S'agissant de la méthode du saut minimum, force est de constater qu'elle fournit des résultats relativement atypiques par rapport aux deux autres. On peut logiquement se demander si de telles observations ne sont pas purement contingentes à la nature des objets lexicaux étudiés, à savoir les RN. Or, les classifications obtenues en fonction des verbes (figure 4) conduisent à des résultats quasiment identiques. En revanche, celles que l'on obtient à partir des MR (figures 5 et 6) indiquent une distanciation plus forte entre le discours émanant des représentants des entreprises ou des dirigeants et celui des hommes politiques. Si les hommes politiques et les dirigeants font usages de références communes, les idées qu'ils développent ne sont pas pour autant totalement similaires.

Figure 4. Classification en fonction des verbes selon la méthode de la distance moyenne entre classes

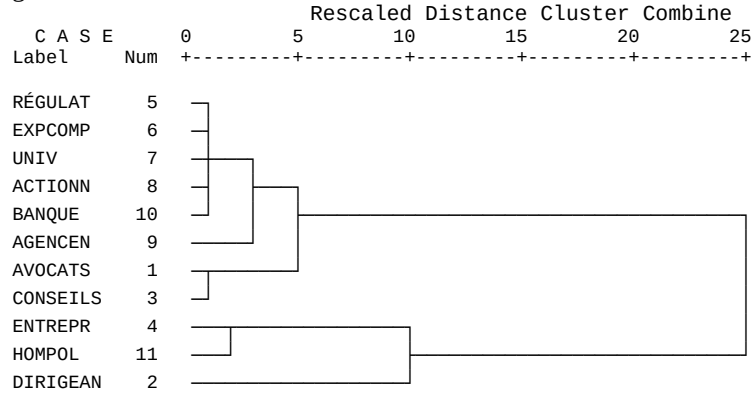


Figure 5. Classification en fonction des MR selon la méthode de la distance moyenne entre classes

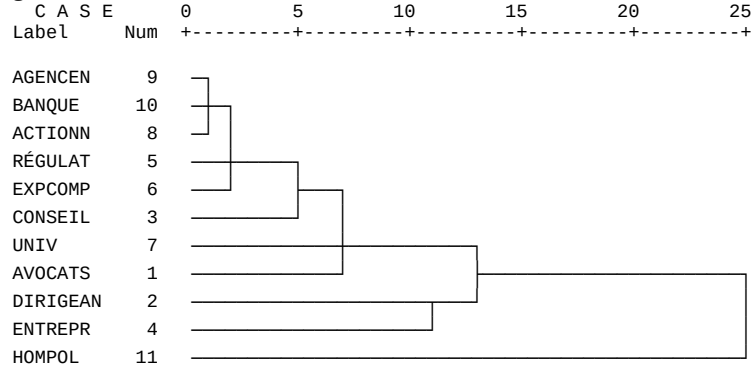
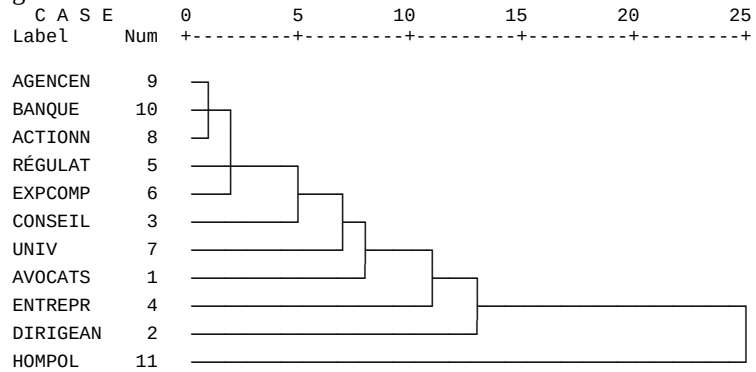


Figure 6. Classification en fonction des MR selon la méthode du saut minimum



4. Conclusion

Les classifications fondées sur les RN, les verbes ou les MR peuvent conduire à des conclusions sensiblement différentes. De plus, l'utilisation de la méthode du saut minimum produit des résultats relativement atypiques par rapport aux deux autres méthodes utilisées (distance moyenne entre classes, et critère de Ward). Si l'on retient les résultats obtenus à partir des RN et des verbes, on peut conclure à l'existence d'une proximité entre le discours des hommes politiques et celui des dirigeants. En revanche, une telle proximité doit être nuancée si l'on se fonde sur l'étude des MR. Le constat de cet écart peut s'expliquer par la volonté des députés de proposer des réformes, alors que les dirigeants souhaitent conserver leurs prérogatives en matière de définition des règles de gouvernance.

5. Bibliographie

[GHI 98] GHIGLIONE R., LANDRE A., BROMBERG M., MOLETTE P., *L'analyse automatique des contenus*, Dunod, Paris, 1998.

[LEB 94] LEBART L., SALEM A., *Statistique textuelle*, Paris, Dunod, 1994.

Classification croisée d'un tableau de données symboliques : application à l'analyse du comportement des utilisateurs d'un site Web

Rosanna Verde[1], Yves Lechevallier[2]

[1]*Dip. Strategie Aziendali e Metodologie Quantitative - SUN – Seconda Università di Napoli, Piazza Umberto I, 81043 Capua, Italie - rosanna.verde@unina2.it*

[2]*INRIA - Institut National de Recherche en Informatique et en Automatique Domaine de Voluceau - Rocquencourt B.P. 105 - 78153 Le Chesnay Cedex, France yves.lechevallier@inria.fr*

RÉSUMÉ. Dans ce travail nous présentons une méthode de classification croisée appliquée aux tableaux des données symboliques. Le critère optimisé est un Φ^2. Cette méthode a été utilisée dans le cadre du Web Usage Mining pour découvrir des classes de comportements de navigateurs.

MOTS-CLÉS : Nuées Dynamiques, Données symboliques, Fichiers « log » du WEB.

1. Introduction

Dans ce papier nous proposons une généralisation de la méthode de classification croisée ([GOV77] ; [GOV95]) dans le cadre de l'analyse des données symboliques et tout particulièrement lorsque les descripteurs sont des variables de type intervalle ou modal [BOC00, pages 42-48]. L'objectif est de classer simultanément les lignes et les colonnes d'un tableau de données. La convergence de l'algorithme, démontrée dans [GOV77], est basée sur la cohérence entre la fonction d'affectation des lignes et la fonction d'affectation des colonnes du tableau de données, le prototype étant un tableau de données « réduit » de même type que le tableau de données de départ. Si le tableau de données est un tableau de contingence alors la fonction d'affectation est une distance du χ^2 entre les distributions des lignes et colonnes de ce tableau et les prototypes des classes associées. Cependant dans notre cadre, le critère à optimiser sera la mesure d'association du ϕ^2 calculée sur le tableau de données « réduit » servant de prototype.

Dans le cas des variables modales nous montrerons qu'il suffit de travailler sur un tableau de profils normalisés au lieu d'un tableau de contingence. Dans le cas des variables intervalles nous proposons de découper le domaine de ces variables en un ensemble d'intervalles élémentaires disjoints formant une partition du domaine de la variable étudiée où la description de chaque ligne est formée par une séquence d'intervalles élémentaires. Dans ce cas l'étape d'agrégation de ces intervalles élémentaires se fera par un algorithme de programmation dynamique.

La méthode de classification croisée dans le contexte de l'Analyse de Données Symboliques représente une solution efficace pour la recherche conjointe d'une typologie sur l'ensemble des individus (représentés par les lignes du tableau de données) et une taxonomie sur les modalités des variables (représentés par les colonnes du tableau). Ces modalités représentant des intervalles disjoints dans le cas des variables intervalles.

Une application sur données du *web* provenant du *web server* de l'INRIA permettra de valider la procédure et l'introduire comme une méthodologie de recherche de typologies dans le contexte du *Web Usage Mining* [ARN03].

2. Schéma général de la méthode de classification croisée appliquée à un tableau de données symboliques

Soit E l'ensemble de n objets représentés par p variables multi-valuées $Y_1,\ldots,Y_p$. A chaque objet de E est associée une description symbolique (intervalles, distributions, liste de valeurs…) qui est un vecteur de dimension p, nous appelons X le tableau de ces descriptions. Dans le cas des variables discrètes, multi-valuées et pondérées les modalités sont les valeurs qui ont été observées. Dans le cas des variables intervalles, il est nécessaire de réaliser une transformation de ces variables afin d'homogénéiser l'ensemble des descripteurs et de permettre une stratégie commune de regroupement. Dans ce cas, à partir de l'ensemble des n intervalles observés sur les éléments de E on construit un ensemble $I=\{I_1,\ldots,I_h,\ldots,I_H\}$ de H intervalles disjoints, dits *élémentaires*, tels qu'ils constituent une *base* de l'ensemble des intervalles et que chaque intervalle x_s, d'un objet s, peut être représenté par l'union d'un ensemble d'intervalles élémentaires disjoints. A cet intervalle x_s on associe une distribution donnée par le vecteur $q_s=(q_{s1},\ldots, q_{sh},\ldots,q_{sH})$ où les poids sont définis par:

$$q_{sh} = \begin{cases} |I_h|/|x_s| & \text{si } I_h \subseteq x_s \\ 0 & \text{sinon} \end{cases} \qquad (1)$$

où $|I_h|$ et $|x_s|$ sont respectivement les longueurs des intervalles I_h et x_s. Ainsi les variables intervalles seront assimilées à des variables modales avec une contrainte d'ordre total [BOC00, pages 153-165, CHA03] où l'ensemble des modalités est l'ensemble des intervalles élémentaires.

Rappelons rapidement le schéma général de l'algorithme de type Nuées Dynamiques [DID71 ; VER00] sur un tableau de données multi-valuées qui sera utilisé, ici, pour la classification des lignes et des colonnes du tableau X de données symboliques. Cet algorithme est basé sur la recherche de la meilleure partition P^* de E en c classes non vides, au sens d'un critère Δ qui mesure l'adéquation entre G et P, où P est une partition en c classes non vides de E et G est le vecteur des c prototypes associés aux c classes de P:

$$\Delta(P^*,G^*)=Min\{\Delta(P,G)\}. \qquad (2)$$

Dans le cadre de la classification simultanée des lignes et des colonnes d'un tableau de données, quand le tableau de données est un tableau de contingence, des auteurs ([GOV77)] ; [GOV03]) ont proposé pour Δ le critère χ^2. Comme dans notre contexte les variables sont modales il doit être défini à partir de la partition « ligne » P en k classes non vides de E, de la partition « colonne » Q en m classes non vides de l'ensemble V des modalités associées aux p variables symboliques et du prototype G défini comme résumé du croisement de ces deux partitions. Du fait qu'il doit être aussi indépendant des pondérations de chacune de ces variables modales, nous proposons d'utiliser le critère du Φ^2 sur le tableau de données y_{sb} transformé de la manière suivante :

Soit V_v l'ensemble des modalités de la variable Y_v et soit x_{sb} la valeur de l'objet s pour la modalité $b \in V_v$ alors la hiérarchie sur les modalités induites par les p variables implique une normalisation sur chaque variable de cette valeur :

$$y_{sb} = x_{sb} / \tilde{x}_{sv} \text{ et } \tilde{x}_{sv} = \sum_{e \in V_v} x_{se} \qquad (3)$$

Le critère du Φ^2 appliqué aux vecteurs de profils normalisés y_{sb} est additif par rapport aux variables. Les variables intervalles sont transformées en variables modales où l'ensemble des modalités correspond à l'ensemble de intervalles élémentaires avec une pondération définie par l'équation (1).

Ainsi la valeur résumée g_{ij} associée au croisement de la classe $C_i \subset E$ de la partition P avec la classe $C^j \subset V_v$ de la partition Q est égale à $g_{ij} = \sum_{s \in C_i} \sum_{b \in C^j} y_{sb}$ et le critère d'adéquation $\Delta(P,Q,G)$ est optimisé en façon itérative dans les deux étapes suivants :

On fixe la partition « ligne » P, on recherche la meilleure partition « colonne » Q, parmi toutes les partitions en m classes de l'ensemble de modalités V tel que $\Delta(Q,G|P)=Min\{\Delta(Q',G'|P)\}$ avec comme prototype G l'ensemble des k vecteurs G_i , le vecteur $G_i = \left(g_{ij} / g_{i.}\right)_{j=1,...,m}$ étant associé à la classe C_i de la partition P de l'ensemble E.

Dans le cas des variables intervalles la partition « colonne » Q doit être compatible avec l'ordre total donné par l'ensemble des intervalles élémentaires, cette recherche est réalisée par l'algorithme de programmation dynamique proposé dans [LEC76] sinon on utilise l'algorithme de type Nuées Dynamiques décrit ci-dessus pour trouver cette partition « colonne ».

On fixe la partition « colonne » Q, on recherche la meilleure partition « ligne » P, parmi toutes les partitions en k classes de l'ensemble de E telle que $\Delta(P,G|Q)=Min\{\Delta(P',G'|Q)\}$ avec comme prototype G l'ensemble des m vecteurs G^j, le vecteur $G^j = \left(g_{ij} / g_{.j}\right)_{i=1,...,k}$ étant associé à la classe C^j de la partition Q de l'ensemble V. Ici l'algorithme de classification décrit ci-dessus est toujours utilisé.

3. Application à l'analyse de l'usage à partir de fichiers « logs »

L'objectif d'une classification automatique des visites en groupes homogènes est de révéler des catégories de comportement de navigation d'internautes. Cependant l'objet « visite » [SÄU 01], construit à partir d'un ensemble d'actions sur les sites WEB, est caractérisé par un ensemble de lignes contenues dans les fichiers « log » aussi la description de cet objet peut se faire sous forme de description symbolique où chaque variable symbolique représente un site ou bien une rubrique de ce site (par exemple sur le site de l'INRIA on peut avoir une description en fonction des unités de recherche ou bien en fonction des rubriques du site (information générale, les projets, la documentation, la DRH,…).

4. Perspectives

Une des limites de notre approche est que la structure du site est faite à partir de l'implémentation physique des pages aussi elles sont difficilement interprétables du fait qu'elles décrivent les parcours qu'en termes de noms de documents HTML principalement connus par les concepteurs du site. Un marquage sémantique des pages faciliterait donc la lecture des résultats obtenus, et pourrait intervenir dans la conception même de ces outils de Web Usage Mining. Une autre contrainte est que de plus en plus de sites sont aujourd'hui conçus de manière dynamique, et non plus comme un ensemble de pages HTML reliées les unes aux autres par des liens hypertextes. Ainsi, chaque page est générée automatiquement suivant les précédentes requêtes de l'utilisateur à l'aide d'éléments de construction contenus dans la base de données du serveur sur fond d'une page HTML standard, qui peut elle-même être personnalisée par rapport à l'utilisateur. Dans ce cas, le marquage sémantique des pages visualisé s'avère d'autant plus difficile que nécessaire. On pourrait même aller jusqu'à se demander si la distinction actuelle entre Web-Content-Mining et Web-Usage-Mining ne sera pas amener à disparaître via un rapprochement nécessaire des méthodes.

5. Bibliographie

[ARN03] ARNOUX, A., LECHEVALLIER, Y., TANASA, D., TROUSSE, B., VERDE, R., "Automatic Clustering for Web Usage Mining" SYNASC-2003, *5th International Workshop on Symbolic and Numeric Algorithms for Scientific Computing*, Timisoara, 2003.

[BOC00] BOCK, H. H., DIDAY, E. (eds.), *Analysis of Symbolic Data*. Studies in Classification, Data Analysis and Knowledge Organisation, Springer-Verlag, 2000.

[CHA03] CHAVENT, M., DE CARVALHO, F. A. T., LECHEVALLIER, Y., VERDE, R., “Trois nouvelles méthodes de classification automatique de données symboliques de type intervalle”. *Revue de Statistique Appliquées*, n. 4 , 2003, p. 5-29.

[DID71] DIDAY, E., “Le méthode des Nuées dynamique”, *Revue de Statistique Appliquée*, vol. 19, n. 2, 1971, p. 19-34.

[GOV77] GOVAERT, G., “Algorithme de classification d'un tableau de contingence”. *First international symposium on Data Analysis and Informatics*, INRIA, Versailles, 1977, p. 487-500.

[GOV95] GOVAERT, G., “Simultaneous clustering of rows and columns”. *Control Cybernet*, vol. 24, 1995, p.437-458.

[GOV03] GOVAERT, G., NADIF M., “Clustering with block mixture models”. *Pattern Recognition*, Elservier Science Publishers,, vol. 36, 2003, p. 463-473

[JOL03] JOLLOIS, F.-X., “Contribution à la classification automatique à la fouille de données”. Thèse de l’Université de Metz, 2003.

[LEC76] LECHEVALLIER, Y., “Classification automatique optimale sous contrainte d’ordre total”. Rapport de recherche IRIA, n. 200, 1976.

[SÄU 01] SÄUBERLICH F., HUBER K.-P., ”A Framework for Web Usage Mining on Anonymous Logfile Data “, GfKl’2001, Munich, 2001.

[VER00] VERDE, R., DE CARVALHO, F.A.T., LECHEVALLIER, Y., “A Dynamical Clustering Algorithm for Multi-Nominal Data”. Kiers, H.A.L. *et al.* (eds.): *Data Analysis, Classification, and Related Methods*. Springer-Verlag, Heidelberg, 2000, p. 387-394.

[VER03] VERDE, R., DE CARVALHO, F.A.T., LECHEVALLIER, Y., “General dynamical clustering methods on symbolic data”. *CLADAG2003*, Bologna, 2003.

Modèle de Mélange par Copules et Données Fonctionnelles

Mathieu Vrac

Centre de Recherche de Mathématiques de la Décision,
Université Paris IX Dauphine,
Place de Lattre deTassigny,
75755 Paris cedex 16

RÉSUMÉ. *L'algorithme EM et ses variantes (SEM, MCEM, etc.) sont largement utilis´es pour d´eterminer et ´etudier un m´elange de densit´es de probabilit´e, dans le cas, par exemple, de lois normales et de donn´ees purement num´eriques. Ce papier propose une version de EM et de ses variantes pour obtenir un m´elange de lois pour donn´ees fonctionnelles, ainsi qu'un m´elange de lois - dites "Copules" - utilisable aussi bien sur donn´ees num´eriques que fonctionnelles.*

MOTS-CLÉS : *Modèle de M´elange, EM, Copules, Donn´ees Fonctionnelles*

1. Introduction

La première apparition d'un mélange de densités de probabilité dans la littérature scientifique est due à Pearson (1894) et fût résolu par la méthode dite des moments. L'algorithme EM de Dempster et al. (1977) est depuis l'algorithme d'estimation des mélanges de densités le plus répandu. Il a donné lieu à de nombreuses variantes telles que SEM, SAEM, ou MCEM, tentant de palier à certains défauts de EM, tels que la sensibilté à la solution intiale ou le choix du nombre de composantes. Ces approches considèrent que les données observées ont été générées à partir d'une densité f exprimée comme un mélange de K densités paramétriques f_k de paramètres α_k :

$$f(x) = \sum_{k=1}^{K} p_k f_k(x|\alpha_k). \tag{1}$$

Une procédure est ici proposée afin d'étendre EM et ses variantes au cas de données fonctionnelles. Celle-ci repose sur la notion de "Fonction de Distribution de Distribution", introduite par Vrac (2002) et Diday et Vrac (2004) pour les données fonctions de répartition, et étendue ici au cas des données fonctionnelles sous le nom de "Fonction de Distribution de Fonctions". Un mélange de lois est ensuite développé en utilisant des densités dérivant de fonctions copules. La théorie de ces fonctions de répartition ayant la particularité d'avoir des marginales uniformes sur $[0, 1]$, apparut pour la première fois sous la plume de Sklar (1959). Etant donné un n-uple de variables aléatoires $X = (X_1, ..., X_n)$, les copules joignent la fonction de répartition jointe F_X n-dimensionnelle avec ses marginales $F_{X_1}, ..., F_{X_n}$ unidimensionnelles, modélisant ainsi la dépendance entre les variables aléatoires. Ce modèle, est ainsi utilisé pour écrire un mélange de copules pour données numériques, et est ensuite appliqué pour définir un mélange de densités pour données fonctionnelles, grâce aux fonctions de distribution de fonctions.

2. Modèle de Mélange pour Données Fonctionnelles

Afin d'étendre les approches de décomposition de mélange de densités au cas des données fonctions, la notion de fonction de répartition est introduite pour ces données. Celle-ci étend la définition de "Fonction de Distribution de Distributions" donnée par Vrac (2002) et Diday et Vrac (2004). Soit Ω un ensemble d'individus statistiques

w, chacuns décrits par une fonction f_w appartenant à Ω_f, un ensemble de fonctions unidimensionnelles. Nous supposons sans perte de généralités que ces fonctions f_w sont définies d'un sous-espace de $\mathbb{R}$ dans un sous-espace de $\mathbb{R}$. Soit A, la σ-algèbre engendrée sur Ω_f par les singletons $\{f\}$ de Ω_f. Une variable aléatoire X est alors définie, qui à tout w associe sa fonction $X(w) = f_w \in \Omega_f$:

$$\begin{array}{rccc} X: & (\Omega, M, \mathbb{P}) & \longrightarrow & (\Omega_F, A) \\ & w & \mapsto & f_w \in \Omega_F, \end{array}$$

avec M une σ-algebra sur Ω et $\mathbb{P}$ une mesure de probabilité sur (Ω, M).

Definition 1 *Une "Fonction de Distribution de Fonctions" (FDF, ou "Fonction de Distribution de Données Fonctionnelles") p-dimensionnelle au point $T = (T_1, ..., T_p) \in \mathbb{R}^p$ est la fonction H_T définie par :*

$$\begin{array}{rccc} H_T: & \mathbb{R}^p & \longrightarrow & [0,1] \\ & x = (x_1, ..., x_p) & \mapsto & H_T(x) \end{array}$$

avec

$$H_T(x) = \mathbb{P}(\{f \in \Omega_f \,|f(T_1) \leq x_1, ..., f(T_p) \leq x_p\}) \forall\, x \in \overline{\mathbb{R}^p}. \tag{2}$$

Nous supposons alors que nous disposons d'un ensemble E de N fonctions $\{f_1, ..., f_N\}$ appartenant à Ω_f et nous cherchons à modéliser une fonction H_T de distribution de fonctions à partir de E (pour un $T = (T_1, ..., T_p)$ donné), ce qui équivaut à modéliser h_T, la densité associée à H_T (h_T est la dérivée p-ième de H_T). Diday et Vrac (2004) ont prouvé que H_T est une fonction de répartition et nous pouvons alors supposer que H_T (respectivement h_T) est un mélange de K fonctions de distribution de fonctions (respectivement dérivées de FDF) paramétriques H_T^k (respectivement h_T^k) de paramètre α_k :

$$H_T(x_1, ..., x_p) = \sum_{k=1}^{K} p_k H_T^k(x_1, ..., x_p|\alpha_k), \tag{3}$$

équivalent à $h_T(x_1, ..., x_p) = \sum_{k=1}^{K} p_k h_T^k(x_1, ..., x_p|\alpha_k)$. Cette formulation par h_T correspond à un mélange de densités et nous pouvons alors appliquer un algorithme de type EM pour résoudre ce mélange, algorithme se résumant de la manière suivante : A partir d'une solution initiale (p_k^0, α_k^0), l'itération $n \geq 1$ se compose de deux étapes successives avec $\boldsymbol{x}_i = (f_i(T_1), ..., f_i(T_p))$

– Estimation (E) : Calcul de $t_k^n(\boldsymbol{x}_i)$, la probabilité a posteriori que $\boldsymbol{x}_i$ appartienne à la composante k (que f_i appartienne à la composante k) :

$$t_k^n(\boldsymbol{x}_i) = \frac{p_k^n h_k(\boldsymbol{x}_i|\alpha_k^n)}{\sum_{k=1}^{K} p_k^n h_k(\boldsymbol{x}_i|\alpha_k^n)} \tag{4}$$

– Maximisation (M) : Pour $k = 1, ..., K$, calcul des proportions du mélange

$$p_k^{n+1} = \frac{1}{N} \sum_{i=1}^{N} t_k^n(\boldsymbol{x}_i) \tag{5}$$

et pour $k = 1, ..., K$, résolution des équations de log-vraisemblance

$$\sum_{i=1}^{N} t_k^n(\boldsymbol{x}_i) \frac{\partial \log(h_k(\boldsymbol{x}_i|\alpha_k^{n+1}))}{\partial \alpha_k} = 0. \tag{6}$$

A la convergence de cet algorithme, nous disposons d'un modèle de mélange pour données fonctionnelles (grâce aux paramètres $(p_k, \alpha_k)_{k=1,...,K}$) et par application du principe de Maximum A Posteriori (MAP) nous obtenons une classification en K classes P_1, ..., P_K des fonctions $\{f_1, ..., f_N\}$.

$$P_k = \{\boldsymbol{x}_i | t_k(\boldsymbol{x}_i) \geq t_j(\boldsymbol{x}_i) \text{ for all } j \neq k\} \tag{7}$$

3. Mélange de Fonctions Copules : EM-Copules

Une fonction copule peut être vue comme une fonction de répartition multivariée dont les marginales sont uniformes sur $[0,1]$. Regardons la définition formelle d'une copule avant de présenter le mélange de copules.

Definition 2 (Schweizer and Sklar (1983)) *: Une copule n-dimensionnelle (ou n-copule) C est une fonction de $[0,1]^n$ dans $[0,1]$ telle que :*

1. Pour tout u dans $[0,1]^n$,

$$C(u) = 0 \text{ si au moins une coordonnée de } u \text{ est égale à } 0, \tag{8}$$

$$\text{et si toutes les coordonnées de } u \text{ sont égales à } 1 \text{ sauf } u_k \text{ alors } C(u) = u_k; \tag{9}$$

2. Pour tout a et b dans $[0,1]^n$ tels que $a \leq b$, alors $V_C([a,b]) \geq 0$, avec

$$V_C([a,b]) = \Delta_a^b C(t) = \Delta_{a_n}^{b_n} \Delta_{a_{n-1}}^{b_{n-1}} ... \Delta_{a_1}^{b_1} C(t),$$

$$\text{où } \Delta_{a_k}^{b_k} C(t) = C(t_1, ..., t_{k-1}, b_k, t_{k+1}, ..., t_n) - C(t_1, ..., t_{k-1}, a_k, t_{k+1}, ..., t_n)$$

Le résultat essentiel de la théorie des copules est le théorème de Sklar :

Theorem 1 (Sklar (1959)) *:*
Soit H une fonction de répartition p-dimensionnelle de marginales unidimensionnelles $F_1, ..., F_p$. Alors, il existe une copule C telle que pour tout $(x_1, ..., x_p)$ dans $\overline{\mathbb{R}}^p$,

$$H(x_1, ..., x_p) = C(F_1(x_1), ..., F_p(x_p)). \tag{10}$$

Si $F_1, ..., F_p$ sont continus, alors C est unique.

Les copules permettent ainsi de lier la loi jointe d'un n-uplet de variables aléatoires avec ses lois unidimensionnelles. La notion de mélange de copules se définie en remarquant que certaines fonctions copules sont des fonctions de répartition auxquelles on peut associer des densités. Disposant d'un N-échantillon du p-uple de variables aléatoires $(X_1, ..., X_p)$, nous souhaitons estimer une fonction de répartition H (respectivement une densité de probabilité $h = \frac{\partial^p H}{\partial x_1 ... \partial x_p}$) comme mélange de K fonctions de répartition H_k (respectivement densités h_k) paramétriques de paramètres α_k :

$$H(x_1, ..., x_p) = \sum_{k=1}^{K} p_k H_k(x_1, ..., x_p). \tag{11}$$

D'après le théorème de Sklar, l'équation (11) devient

$$H(x_1, ..., x_p) = \sum_{k=1}^{K} p_k C_k(F_1^k(x_1), ..., F_p^k(x_k)) \tag{12}$$

où C_k est la copule de la composante k et F_i^k est la fonction de répartition margininale de X_i dans la composante k. Notons que si $F_1^k, ..., F_p^k$ sont uniformes sur $[0,1]$, le mélange (12) est $H(x_1, ..., x_p) = \sum_{k=1}^{K} p_k C_k(x_1, ..., x_p)$. Nous posons par ailleurs que chaque copule C_k appartient à une famille donnée de l'ensemble des copules dites Archimédiennes (voir Schweizer et Sklar (1983) et Nelsen (1998)). Les familles de cet ensemble ont la particularité d'être paramétrées par un paramètre qu'on notera β_k de dimension $p-1$. Alors, nous avons le mélange de copules

$$h(x_1, ..., x_p) = \sum_{k=1}^{K} p_k \frac{dF_1^k}{dx_1}(x_1) \times ... \times \frac{dF_p^k}{dx_p}(x_p) \times \frac{\partial^p C_{\beta_k}}{\partial u_1 ... \partial u_p}(F_1^k(x), ..., F_p^k(x_p)), \tag{13}$$

où les marginales F_i^k sont supposées paramétriques de paramètres b_i^k. Nous noterons $b_k = (b_1^k, ..., b_p^k)$ et $\boldsymbol{x}_i$ les valeurs $(x_1, ..., x_p)$ de l'individu i. Afin de résoudre ce mélange de densités original, nous utilisons une version adaptée de EM : l'algorithme EM-copules, qui à partir d'une solution initiale $(p_k^0, \beta_k^0, b_k^0)_{k=1,...,K}$, se résume ainsi à l'itération n :

– Estimation (E) : Calcul de $t_k^n(\boldsymbol{x}_i)$, la probabilité a posteriori que $\boldsymbol{x}_i$ (l'individu i) provienne de la composante k (avec $\Phi_k^n = (\beta_k^n, b_k^n)$, les paramètres courants) :

$$t_k^n(\boldsymbol{x}_i) = \frac{p_k^n h_k(\boldsymbol{x}_i|\beta_k^n, b_k^n)}{\sum_{k=1}^K p_k^n h_k(\boldsymbol{x}_i|\beta_k^n, b_k^n)}. \tag{14}$$

– Maximisation (M) : Pour $k = 1, ..., K$, calcul des proportions du mélange selon (5) et pour $k = 1, ..., K$:

1. pour $j = 1, ..., p$ estimation des paramètres b_j^k des distributions marginales dans la composante k en résolvant les équations de log-vraisemblance pour F_1^k, ..., F_p^k

$$\sum_{i=1}^N t_k^n(\boldsymbol{x}_i) \frac{\partial \log(\frac{\partial F_j^k}{\partial x_j}(x_j|b_j^{k,(n+1)}))}{\partial b_j^k} = 0 \tag{15}$$

2. estimation de β_k, paramètre de la k^{eme} copule, solution de

$$\sum_{i=1}^N t_k^n(\boldsymbol{x}_i) \frac{\partial \log(C''_{\beta_k^{(n+1)}}(F_1^k(x_1|b_1^{k,(n+1)}), ..., F_p^k(x_p|b_p^{k,(n+1)}))}{\partial \beta_k} = 0 \tag{16}$$

avec $C''_{\beta_k} = \frac{\partial^2 C_{\beta_k}}{\partial u \partial v}$, dérivée seconde de C selon u et v.
Alternativement, ces deux étapes peuvent être remplacées par la résolution du système d'équations

$$\sum_{i=1}^N t_k^n(\boldsymbol{x}_i) \nabla_{(\beta_k, b_k)} \log(h_k(\boldsymbol{x}_i|\beta_k, b_k)) = 0. \tag{17}$$

A la convergence de cet algorithme, chaque composante est décrite par une copule qui fournit ainsi une indication de la dépendance entre les deux fonctions de répartition également données. L'algorithme EM-Copules s'adapte bien évidemment au cas des données fonctionnelles, tel que présenté en section 2. Cette approche a été employée par Vrac (2002) pour l'étude de données climatologiques.

4. Conclusion

Cet article propose tout d'abord une version de l'algorithme EM permettant de résoudre un mélange de lois de données fonctionnelles. Pour cela la notion de fonction de distribution de fonctions a été développée de manière générale et peut être représentée (sous quelques contraintes) par une loi statistique quelconque. Puis, les fonctions copules sont utilisées dans un mélange de densités pour définir l'algorithme EM-Copules traitant aussi bien les données numériques que fonctionnelles.

5. Bibliographie

[DEM 77] DEMPSTER A., LAIRD N., RUBIN D., Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society*, vol. 39, 1977, p. 1-38.

[DID 04] DIDAY E., VRAC M., Mixture Decomposition of Distributions by Copulas in the Symbolic Data Analysis Framework, *Discrete Applied Mathematics*, vol. to appear, 2004.

[NEL 98] NELSEN R. B., *An Introduction to Copulas*, Springer Verlag. In Lectures Notes in Statistics, New-York, 1998.

[PEA 94] PEARSON K., Contributions to the theory of mathematical evolution, *Philosophical transactions of the royal society of London*, vol. A 185, 1894, p. 71-110.

[SCH 83] SCHWEIZER B., SKLAR A., *Probabilistic Metric Spaces*, Elsevier North-Holland, New-York, 1983.

[SKL 59] SKLAR A., Fonction de répartition à n dimensions et leurs marges, *Inst. Statist. Univ. Paris Pub.*, vol. 8, 1959, p. 229-231.

[VRA 02] VRAC M., Analyse et Modélisation de Données Probabilistes par Décomposition de Mélange de Copules et Application à une Base de Données Climatologiques, Thèse de doctorat, Université Paris IX Dauphine, 2002.

WF : méthode de sélection de variable combinant une méthode filtre rapide et une approche enveloppe

Erray Walid

ERIC, Université Lumière Lyon 2
5 Avenue Pierre Mendès-France
69676 Bron cedex

RÉSUMÉ. Dans cet article nous allons en premier lieu introduire de manière générale le procédé de sélection de variables. Ensuite nous allons présenter une étude de quatre méthodes de sélection de variables : Abb, Lvf, Relief et une méthode myope. Le but de cette étude est de déterminer, d'une part, l'utilité des méthodes de sélection de variables et d'autre part de voir si une méthode plus complexe est forcément meilleure. Nous introduirons par la suite une variante de la méthode myope qui sera combinée à une approche enveloppe afin d'améliorer la qualité du sous-ensemble obtenu.

MOTS-CLÉS : Sélection de variables, Méthodes filtres, Méthodes enveloppes, Qualité d'apprentissage.

1. Introduction

Aujourd'hui, les bases de données ont des tailles de plus en plus importantes. Pour cela, l'amélioration de la qualité de la représentation des données est devenu un problème majeur de l'ECD. L'une des difficultés principales liée à ce problème est la dimension de cet espace de représentation.
La sélection de variables permet de résoudre ce problème. C'est un processus choisissant un sous-ensemble optimal de variables selon un critère particulier. Il permet l'élimination de variables inutiles, non pertinentes et redondantes ainsi que l'élimination du bruit généré par certaines variables. Le processus d'apprentissage sera ainsi accéléré et la précision prédictive des algorithmes d'apprentissage sera améliorée. Il existe deux familles d'algorithmes de sélection de variables : les méthodes "enveloppes" [JOH 92] et les méthodes "filtres" [KIR 92]. La différence fondamentale entre ces deux familles réside dans le fait que la première est liée à l'algorithme d'apprentissage utilisée, alors que la seconde en est totalement indépendante.
Dans cet article nous allons étudier trois méthodes fréquemment citées : Abb, Lvf et Relief ainsi qu'une méthode de sélection rapide que nous appellerons méthode myope. Nous commencerons par décrire les différents algorithmes. Ensuite nous présenterons une étude expérimentale effectuée sur des jeux de données standards. Enfin nous terminerons par une troisième partie qui proposera une amélioration de la méthode myope afin de trouver le sous-ensemble optimal et ceci à l'aide d'une approche enveloppe.
Ce travail a été effectué dans le cadre d'un projet de collaboration entre le laboratoire ERIC et la société France Telecom.

2. Méthodes filtres

2.1. *Définition*

Le filtrage est un processus de prétraitement des données qui filtre les variables non pertinentes avant que n'intervienne la phase d'induction [LIU]. Il utilise les caractéristiques générales de l'ensemble d'apprentissage pour sélectionner certaines variables et en exclure d'autres.

2.2. *Algorithmes étudiés*

2.2.1 Abb [LIU 98]

L'algorithme Abb commence à partir de l'ensemble total des variables (Backward Elimination) et cherche ensuite la variable à retirer afin de maximiser le taux d'inconsistance. Donc, la racine contiendra toutes les variables, et le nœud fils ne sera exploré que si la valeur du taux d'inconsistance du sous-ensemble de variables est supérieure à un certain seuil.

2.2.2 Lvf [LIU]96)

Initialement, le meilleur sous-ensemble de variables est l'ensemble total des variables. Il génère alors aléatoirement un sous-ensemble *S*. Si son cardinal est inférieur ou égal à celui du meilleur sous-ensemble et son taux d'inconsistance est inférieur au seuil fixé, S sera donc considéré comme étant le meilleur sous-ensemble. L'algorithme s'arrête au bout d'un nombre I fixé d'itérations.

2.2.3 Relief [KON 96]

Relief est un algorithme basé sur l'attribution de poids aux variables. L'algorithme commence par choisir un échantillon d'instances (ou individus) dont le nombre est fourni par l'utilisateur. Il recherche ensuite pour *T* instances (*T* choisi par l'utilisateur), la plus proche instance de réussite (de même classe) et les plus proches instances d'échec (de classes différentes) en se basant sur une mesure de distance. L'algorithme met à jour les poids des différentes variables qui sont initialisés à zéro. Cette démarche est basée sur une idée intuitive qui est : une variable est plus pertinente qu'une autre si elle distingue une instance de son instance d'échec la plus proche, et moins pertinente si elle distingue une instance de son instance de réussite la plus proche.

2.2.4 Myope.

Le fonctionnement de la méthode myope consiste à mesurer l'entropie de Shannon (Shannon, 1948) correspondant à chaque variable. Dans le cas où la variable est continue, on procède tout d'abord à une bipartition avant de mesurer l'entropie de Shannon.

3. Etude expérimentale

3.1. *But*

A l'aide de nos expérimentations nous essayerons de répondre à deux questions. Il s'agit, en premier lieu, de mesurer le pouvoir de ces méthodes à retrouver le meilleur sous ensemble qui pourra améliorer la qualité de l'apprentissage. Il serait également très utile de savoir s'il y'a un apport important des méthodes les plus complexes comme Relief et Abb par rapport à Lvf et la méthode myope.
Pour mesurer la qualité du sous-ensemble nous nous baserons sur le taux d'erreur obtenu en cross validation avec 10 partitions par l'algorithme ID3 [QUI 83]. Pour les méthodes Relief et la méthode myope nous allons déterminer le meilleur sous-ensemble de variables donc qui a le plus petit taux d'erreur en validation croisée. Pour les méthodes Abb et Lvf, nous allons comparer la qualité des cinq meilleurs sous-ensembles obtenus et déterminer le sous-ensemble dont le taux d'erreur est le plus petit. Les bases de données étudiées sont issues de la base UCI Irvine [BLA 98].

3.2. *Résultats et commentaires*

Le tableau ci-dessous donne le taux d'erreur minimum obtenu en validation croisée ainsi que le pourcentage de variables obtenues dans ce cas. (Table1).

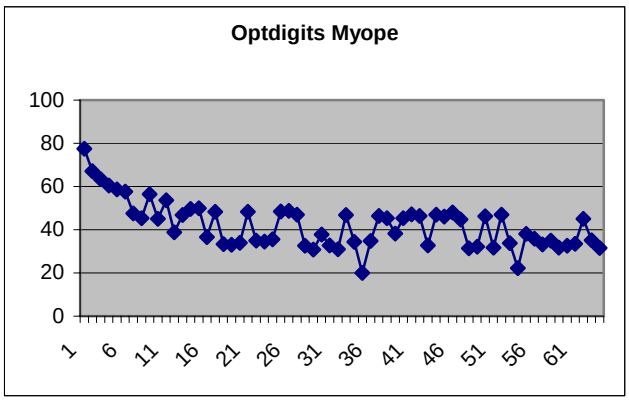
Table 1. Résultats obtenus par Abb, Lvf, Relief et Myope

Base	Abb		Lvf		Relief		Myope	
	Erreur minimale	% Variables	Erreur minimale	% Variables	Erreur minimale	% Variables	Erreur minimale	% Variables
Dermatology	8,46	32,35%	11,74	100,00%	49.39	75	49.39	62
Flag	37,11	32,14%	40,20	21,43%	27.21	100	26.95	50
Iono	8,83	23,53%	12,25	26,47%	26.51	100	27.13	100
Iris	4,00	100,00%	4,00	100,00%	80.69	68	80.75	100
Letter Rec.	42,29	75,00%	54,76	75,00%	7.92	68	7.92	75
OptDigit	49,35	18,75%	56,26	21,88%	20.41	43	16.35	56
Pen Digit	41,95	56,25%	73,95	31,25%	10	42	10.47	42
Pima	25,26	87,50%	32,16	25,00%	28.06	33	26.36	90
Segmentation	25,23	31,58%	20,95	100,00%	43.81	42	43.29	96
Tic Tac Toe	28,81	50,00%	30,68	11,11%	7.65	91	29.23	44
Wave	30,14	57,14%	32,96	33,33%	9.68	97	9.11	57
Yeast	52,76	100,00%	55,72	50,00%	17.42	60	20.06	54
Zoo	5,94	12,50%	23,76	25,00%	49.39	75	49.39	62

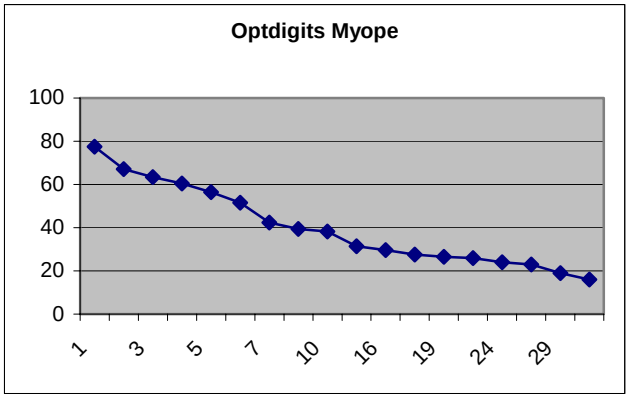
Grâce à ces résultats nous pouvons dire d'une part que le processus de sélection de variable est un processus très utile puisque, dans la majorité des cas, le minimum en taux d'erreur n'est pas forcément obtenu à l'aide de la totalité des variables. D'autre part, les méthodes les plus complexes comme Abb et Relief ne sont pas forcément les plus performantes. Nous pouvons également dire que pour les bases étudiées la méthode myope fournit des résultats très satisfaisants que ce soit par rapport aux taux d'erreurs obtenus mais également par rapport aux temps de calcul très inférieurs à ceux des autres méthodes.

4. WF : méthode combinant une méthode filtre et une approche enveloppe

Un exemple de graphe obtenu par la méthode myope est le suivant :



Il s'agit d'une courbe qui représente les taux de succès en validation croisée de différents sous-ensembles. Nous partons de la meilleure variable et nous ajoutons à chaque fois une variable jusqu'à l'obtention de l'ensemble total des variables. Nous remarquons que lors de l'ajout de certaines variables, le taux d'erreur augmente. Nous proposons donc d'appliquer une approche enveloppe qui commence par la meilleure variable et qui éliminera à chaque itération chaque variable dont l'ajout ne permet pas de diminuer le taux d'erreur. Sur le même exemple que ci-dessus, nous obtenons le graphe suivant :



A l'aide de cette méthode nous avons réussi à avoir un sous-ensemble de variables bien défini et non pas une liste ordonnée. Aussi ce sous-ensemble a une meilleure qualité que tous les sous-ensembles obtenus par les méthodes étudiées, que ce soit par sa taille plus petite (28% des variables sélectionnées) mais également par son taux d'erreur plus faible (16%)

5. Conclusion et perspectives

A la suite de cette étude nous pouvons conclure d'une part que la procédure de sélection de variables et certainement très utile. Aussi son utilité devrait être beaucoup plus importante pour des bases ayant un nombre plus élevé de variables. D'autre part, l'apport en qualité des méthodes les plus complexes comme Abb, Relief et même Lvf par rapport à la méthode myope, qui a une complexité très faible, n'est pas évidente. Sachant que dans le cadre d'étude de bases de données réelles, où nous avons à traiter un grand volume de données, une complexité faible, donc un temps de calcul minimum, devient primordial.
Les premiers résultats obtenus par la méthode WF sont très encourageants. Il serait très intéressant d'effectuer des tests sur des bases plus volumineuses avec un nombre de variables plus grand pour étudier la stabilité de cette méthode.

6. Bibliographie

[BLA 98] Blake, C.L. & Merz, C.J. (1998). UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science. ftp://ftp.ics.uci.edu/pub/machine-learning-databases/.

[JOH 92] John G. & Kohavi R. (1992). Wrappers for feature subset selection. AIJ issue on relevance. Methodology, Systems, Applications, 31--40. IOS Press.

[KIR 92] Kira K. & Rendell L. A. (1992). The feature selection problem: Traditional methods and a new algorithm. In Tenth National Conference on Artificial Intelligence, 129--134. MIT Press.

[KON 96] Kononenko I. & Robnik-Sikonia M. (1996). Relief for estimation and discretization of attributes in classification, regression and ILP problems. In Ramsay, A., ed., Artificial Intelligence.

[LIU 96] Liu H. & Setiono R. (1996). A probabilistic approach to feature selection – A filter approach. Proc. of the 13th International Conference on Machine Learning pp. 319-327, Morgan Kaufmann.

[LIU] Liu H. & Setiono S.. Some issues on scalable feature selection. In 4th World Congress of Expert Systems: Application of Advanced Info. Technologies.

[LIU 98] Liu H. & Motoda H., & Dash M. (1998). A monotonic measure for optimal feature selection. In Proceedings of European Conference on Machine Learning, pages 101--106.

[QUI 86] Quinlan J. (1986). Introduction of Decision Trees, Machine Learning, vol. 1, 1986, p. 81-106.

[SHA 48] Shannon C.E. (1948). A mathematical theory of communication. Bell System Technical Journal, 27:379--423,623--656, 1948.