# Caractérisation automatique des classes découvertes en classification non supervisée

Nistor Grozavu*, Younès Bennani*
Mustapha Lebbah*

*LIPN UMR CNRS 7030, Université Paris 13,
99, avenue Jean-Baptiste Clément, 93430 Villetaneuse
Prénom.Nom@lipn.univ-paris13.fr

**Résumé.** Dans cet article, nous proposons une nouvelle approche de classification et de pondération des variables durant un processus d'apprentissage non supervisé. Cette approche est basée sur le modèle des cartes auto-organisatrices. L'apprentissage de ces cartes topologiques est combiné à un mécanisme d'estimation de pertinences des différentes variables sous forme de poids d'influence sur la qualité de la classification. Nous proposons deux types de pondérations adaptatives : une pondération des observations et une pondération des distances entre observations. L'apprentissage simultané des pondérations et des prototypes utilisés pour la partition des observations permet d'obtenir une classification optimisée des données. Un test statistique est ensuite utilisé sur ces pondérations pour élaguer les variables non pertinentes. Ce processus de sélection de variables permet enfin, grâce à la localité des pondérations, d'exhiber un sous ensemble de variables propre à chaque groupe (cluster) offrant ainsi sa caractérisation. L'approche proposée a été validé sur plusieurs bases de données et les résultats expérimentaux ont montré des performances très prometteuses.

## 1 Introduction

La classification automatique - clustering - est une étape importante du processus d'extraction de connaissances à partir de données. Elle vise à découvrir la structure intrinsèque d'un ensemble d'objets en formant des regroupements - clusters - qui partagent des caractéristiques similaires (Fisher, 1996; Cheeseman et al., 1988). La complexité de cette tâche s'est fortement accrue ces deux dernières décennies lorsque les masses de données disponibles ont vu leur volume exploser. En effet, le nombre d'objets présents dans les bases de données a fortement augmenté mais également la taille de leur description. L'augmentation de la dimension des données a des conséquences non négligeables sur les traitements classiquement mis en oeuvre : outre l'augmentation naturelle des temps de traitements, les approches classiques s'avèrent parfois inadaptées en présence de bruit ou de redondance.

La taille des données peut être mesurée selon deux dimensions, le nombre de variables et le nombre d'observations. Ces deux dimensions peuvent prendre des valeurs très élevées, ce qui peut poser un problème lors de l'exploration et l'analyse de ces données. Pour cela, il est

fondamental de mettre en place des outils de traitement de données permettant une meilleure compréhension des données. En effet, plus le nombre de dimensions d'une base de données est important, plus les données sont dispersées dans l'espace de représentation et plus la différences entre les deux données les plus similaires et les deux données les moins similaires est réduit. Ainsi, dans un espace de grande dimensions, il est très difficile pour un algorithme de classification de détecter les variations de similarité qui définissent les regroupements de données. C'est ce qu'on appelle le "fléau de la dimension". Pour contourner cette difficulté, on utilise souvent des techniques de réduction des dimensions afin de faciliter le processus de l'ECD[1]. La réduction des dimensions permet d'éliminer les informations non-pertinentes et redondantes selon le critère utilisé. Cette réduction permet donc de rendre l'ensemble des données plus représentatif du phénomène étudié. Il s'agit d'un problème complexe qui permet d'optimiser le volume d'informations à traiter et faciliter le processus de l'apprentissage. En effet, les principaux objectifs de la réduction des dimensions sont :

- faciliter la visualisation et la compréhension des données,
- réduire l'espace de stockage nécessaire,
- réduire le temps d'apprentissage et d'utilisation,
- identifier les facteurs pertinents.

La réduction du nombre d'observations peut se faire par quantification à travers une classification non supervisée ou par sélection d'exemples. Dans le cadre de cette étude, nous procéderons par une classification non supervisée permettant ainsi de calculer des prototypes (référents : moyennes locales) représentant l'ensemble des données.

Les algorithmes d'apprentissage artificiel requièrent typiquement peu de traits (variables/attributs) très significatifs caractérisant le phénomène étudié. Dans la problématique de la classification non supervisée, il pourrait encore être bénéfique d'incorporer un module de réduction du nombre de variables dans le système global avec comme objectif d'enlever toute information inconséquente et redondante. Cela a un effet important sur la qualité de la classification. En effet le nombre de caractéristiques utilisées est directement lié à l'erreur finale. L'importance de chaque caractéristique dépend de la taille de la base d'apprentissage : pour un échantillon de petite taille, l'élimination d'une caractéristique importante peut diminuer l'erreur. Il faut aussi noter que des caractéristiques individuellement peu pertinentes peuvent être très informatives si on les utilise conjointement. Pour la réduction du nombre de variables nous pouvons procéder de plusieurs manières :

- par sélection : qui consiste à choisir un sous ensemble des caractéristiques initiales dans l'espace de mesure,
- par transformation : qui vise à construire de nouvelles caractéristiques dans un espace transformé - un espace de projection.

Dans cette étude, nous nous intéressons à la réduction de dimension de l'espace de description dans le cadre de la classification non supervisée par sélection à travers la pondération locale des variables. Deux approches différentes, par la technique de pondération locale et proche par l'utilisation de la structure de la carte, seront présentées dans ce papier.

Pour la réduction de dimension de l'espace de description en apprentissage non supervisé les contributions sont plutôt moins conséquentes (Roth et Lange; Liu et al., 2005; Guyon et al., 2006). Dans la littérature nous trouvons généralement des approches basées sur la pondération comme les travaux de Huang et al. (2005), Blansche et al. (2006), Guérif et Bennani (2007),

[1] Extraction des Connaissance à partir des Données

Frigui et Nasraoui (2004) et Grozavu et al. (2008) et des approches de sélection de caractéristiques comme les méthodes proposées par Basak et al. (1998), Bassiouny et al. (2004), Liu et al. (2005), Questier et al. (2005) et Li et al. (2006). Nous trouvons aussi des méthodes permettant la réduction simultanée des dimensions des données (exemples et variables). Ces méthodes sont souvent appelées des techniques de "bi-classification" ou "classification croisée" ou bien encore "Subspace clustering" (Parsons et al., 2004). Ces approches sont très séduisantes en pratique car elles permettent, grâce à une classification simultanée des observations et des variables, de caractériser les groupes identifiés.

Les deux approches que nous proposons dans cet article peuvent être vue comme proches, mais pas identiques à ces dernières techniques de classification croisée. Nos approches sont associées à deux algorithmes d'apprentissage non supervisé et simultané des observations et des variables. La première approche est une technique complètement nouvelle pour pondérer les variables. Cette technique agit plus en amont en pondérant les caracteristiques des observations au cours de l'apprentissage afin de déduire des pondérations locales. La deuxième approche n'est qu'une extension et une reformulation stochastique de l'approche, de pondération locale, proposée dans Grozavu et al. (2008). Dans ce cas la pondération des distances nous permet de déduire les pondérations locales associées à chaque groupe "clusters". Dans le cadre de notre étude, les deux formalismes de pondérations proposés sont associés au modèle des cartes auto-organisatrices. Les pondérations locales estimées sont utilisées pour la caractérisation des groupes de la partition obtenue avec la carte topologique. En effet, contrairement à la pondération globale, qui estime un seul vecteur de pondérations pour tout l'ensemble des référents (c'est à dire toute la carte), la pondération locale associe un vecteur de pondérations à chaque référent de la carte topologique. Ces valeurs sont simultanément estimées au cours de l'apprentissage avec le partitionnement des observations. Par conséquent nous pouvons utiliser ces pertinences, à la fin de l'apprentissage, pour regrouper et caractériser les prototypes.

Le reste de cet article est organisé comme suit. La Section 2 présente brièvement le modèle de base des cartes auto-organisatrices suivi du détail des deux approches proposées de pondération locale adaptative : $lwd$-SOM[2] (pondération de la distance) et $lwo$-SOM[3] (pondération des observations). La technique pour caracteriser automatiquement les groupes en utilisant nos pondérations est présentée dans la section 5. Dans la section 6 nous présentons les résultats des expérimentations. Une conclusion et des perspectives sont données dans la section 7.

## 2 Cartes auto-organisatrices traditionnelles

Les cartes auto-organisatrices (SOM[4]) présentées par Kohonen (2001) ont été largement utilisées pour la classification et la visualisation des bases de données multidimensionnelles. On trouve une grande variété d'algorithmes des cartes topologiques dérivée du premier modèle original proposé par Kohonen Bishop et al. (1998); Cottrell et al. (2004); Lebbah et al. (2007). Ces modèles sont différents les uns des autres, mais partagent la même idée de présenter les données de grande dimension en une simple relation géométrique sur une topologie réduite. Ce modèle consiste en la recherche d'une classification non supervisée d'une base d'apprentissage $A = \{\mathbf{x}_i \in \mathcal{R}^n, i = 1..N\}$ où l'individu $\mathbf{x}_i = (x_{i1}, x_{i2}..., x_{ij}, ..., x_{in})$ est de dimension $n$.

[2] local weight distance using Self Organizing Map
[3] local weight observation using Self Organizing Map
[4] Self-Organizing Map

Ce modèle classique se présente sous forme d'une carte possédant un ordre topologique de $C$ cellules. Les cellules sont réparties sur les nœuds d'un maillage. La prise en compte dans la carte de taille $C$ de la notion de proximité impose de définir une relation de voisinage topologique. Afin de modéliser la notion d'influence d'une cellule $k$ sur une cellule $l$, qui dépend de leur proximité, on utilise une fonction noyau $\mathcal{K}$ ($\mathcal{K} \geq 0$ et $\lim_{|y|\to\infty} \mathcal{K}(y) = 0$). L'influence mutuelle entre deux cellules $k$ et $l$ est donc définie par la fonction $\mathcal{K}_{k,l}(.)$. A chaque cellule $k$ de la grille est associée un vecteur référent $\mathbf{w}_k = (w_{k1}, w_{k2}..., w_{kj}, ..., w_{kn})$ de dimension $n$. On note par la suite par $\mathcal{W} = \{\mathbf{w}_j, \mathbf{w}_j \in \mathcal{R}^n\}_{j=1}^{|W|}$ l'ensemble des référents associés à la carte. Les phases principales de l'algorithme d'apprentissage associé aux cartes auto-organisatrices sont définies dans la littérature et consistent à minimiser la fonction de coût suivante Kohonen (2001) :

$$\mathcal{R}_{\mathcal{SOM}}(\chi, W) = \sum_{i=1}^{N} \sum_{j=1}^{|\mathcal{W}|} \mathcal{K}_{j,\chi(\mathbf{x}_i)} \|\mathbf{x}_i - \mathbf{w}_j\|^2 \tag{1}$$

où $\chi(\mathbf{x}_i) = \arg\min_j \left(\|\mathbf{x}_i - \mathbf{w}_j\|^2\right)$ la fonction d'affectation.
A la fin de l'apprentissage, la carte auto-organisatrice détermine une partition des données en $|\mathcal{W}|$[5] groupes associés à chaque référent/prototype/représentant $\mathbf{w}_k \in \mathcal{R}^n$ de la carte.

# 3 Apprentissage non supervisé et pondération des observations et des variables

Un des inconvénients des cartes auto-organisatrices (SOM) est qu'elles traitent avec égalité toutes les variables. Ceci n'est pas souhaitable dans de nombreuses applications de partitionnement où les observations sont décrites avec un grand nombre de variables. Les groupes fournis par SOM se caractérisent souvent par un sous-ensemble de variables plutôt que l'ensemble des variables définies. Par conséquent certaine variables peuvent occulter la découverte de la structure spécifique d'un groupe - cluster. La pertinence de chaque variable change d'un groupe à un autre. La pondération des variables est une extension de la procédure de sélection des variables où les variables sont associées à un vecteur de poids qui peuvent être considérés comme des degrés de pertinence.

La démarche proposée pour réaliser simultanément le regroupement et la caractérisation des groupes est conçue de telle manière à estimer les meilleurs prototypes, et les ensemble optimaux de poids au cours de la phase d'apprentissage. Chaque prototype $\mathbf{w}_j = (w_{j1}, w_{j2}, ..., w_{jn})$ est associé à un vecteur de poids $\pi_j = (\pi_{j1}, \pi_{j2}, ..., \pi_{jn})$. Nous notons par la suite par $\Pi = \{\pi_j, \pi_j \in \Re^n\}_{j=1}^{|\pi|}$ l'ensemble des vecteurs de poids. Dans ce qui suit, nous présentons deux versions de pondération locale des variables avec les cartes topologiques : une nouvelle approche pour la pondération des observations et une reformulation stochastique de la pondération des distances.

[5] $|\mathcal{W}|$ indique le nombre d'éléments de l'ensemble $\mathcal{W}$

### 3.1 Pondération locale des observations : $lwo$-SOM

Cette technique de pondération estime un vecteur de poids pour pondérer et filtrer les observations en les adaptant dans le processus d'apprentissage. Dans l'architecture proposée, qui est similaire à l'architecture supervisée **?**, nous associons à chaque référent $\mathbf{w}_j$ un vecteur de pondérations $\pi_j$. Ainsi nous proposons de minimiser la nouvelle fonction de coût suivante :

$$R_{lwo}(\chi, \mathcal{W}, \Pi) = \sum_{i=1}^{N} \sum_{j=1}^{|w|} \mathcal{K}_{j,\chi(\mathbf{x}_i)} \|\pi_j \mathbf{x}_i - \mathbf{w}_j\|^2 \tag{2}$$

La minimization de $R_{lwo}(\chi, \mathcal{W}, \Pi)$ se fait itérativement, avec la descente du gradient, en trois étapes jusqu'à stabilisation. Après l'étape d'initialisation de l'ensemble des prototypes $\mathcal{W}$ et l'ensemble des pondérations associés $\Pi$, à chaque étape d'apprentissage $(t+1)$ nous appliquons les étapes suivantes :

– Minimiser $R_{lwo}(\chi, \hat{\mathcal{W}}, \hat{\Pi})$ par rapport à $\chi$ en fixant $\mathcal{W}$ et $\Pi$. Chaque observation pondérée $(\pi_j \mathbf{x}_i)$ est affectée au referent $\mathbf{w}_j$, dont elle est la plus proche au sens de la distance euclidienne :

$$\chi(\mathbf{x}_i) = \arg\min_j \left( \|\pi_j \mathbf{x}_i - \mathbf{w}_j\|^2 \right) \tag{3}$$

– Minimiser $R_{lwo}(\hat{\chi}, W, \hat{\Pi})$ par rapport à $\mathcal{W}$ en fixant $\chi$ et $\Pi$. Les vecteurs référents sont mis-à-jour avec l'expression suivante :

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \epsilon(t) \mathcal{K}_{j,\chi(\mathbf{x}_i)} \left( \mathbf{x}_i - \mathbf{w}_j(t) \right) \tag{4}$$

– Minimiser $R_{lwo}(\hat{\chi}, \hat{W}, \Pi)$ par rapport à $\Pi$ en fixant $\chi$ et $\mathcal{W}$. L'expression de mis-à-jour pour le vecteur des pondérations $\pi_j(t+1)$ est :

$$\pi_j(t+1) = \pi_j(t) + \epsilon(t) \mathcal{K}_{j,\chi(\mathbf{x}_i)} \left( \pi_j \mathbf{x}_i - \mathbf{w}_j(t) \right) \tag{5}$$

Comme l'algorithme stochastique classique de Kohonen, SOM, on note par $\epsilon(t)$ le pas d'apprentissage au temps $t$. L'apprentissage est généralement réalisé en deux phases. Dans la première phase, un grand pas d'apprentissage initial $\epsilon(0)$ et un grand rayon de voisinage $T_{max}$ sont utilisés. Pendant la deuxième phase, $\epsilon$ et $T$ décroient au cours du temps.

## 4 La pondération locale de la distance : $lwd$-SOM

A partir de la version $lwd$-SOM analytique Grozavu et al. (2008), nous avons developé la version stochastique de la pondération de la distance. La fonction de coût est décrite par la formule suivante :

$$R_{lwd}(\chi, W, \Pi) = \sum_{i=1}^{N} \sum_{j=1}^{|w|} \mathcal{K}_{j,\chi(\mathbf{x}_i)} (\pi_j)^{\beta} \|\mathbf{x}_i - \mathbf{w}_j\|^2 \tag{6}$$

où $\beta$ est le coefficient de discrimination.
Comme pour l'algorithme $lwo$-SOM, la minimisation de la fonction de coût se fait en trois étapes :

1. Minimiser $R_{lwd}(\chi, \hat{\mathcal{W}}, \hat{\Pi})$ par rapport à $\chi$ en fixant $\mathcal{W}$ et $\Pi$. L'expression d'affectation est la suivante :
$$\chi(\mathbf{x}_i) = \arg\min_j \left( (\pi_j)^\beta \|\mathbf{x}_i - \mathbf{w}_j\|^2 \right) \quad (7)$$

2. Minimiser $R_{lwd}(\hat{\chi}, \mathcal{W}, \hat{\Pi})$ par rapport à $\mathcal{W}$ en fixant $\chi$ et $\Pi$. Les vecteurs des prototypes sont mis-à-jour en utilisant l'expression suivante :
$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \epsilon(t)\mathcal{K}_{j,\chi(\mathbf{x}_i)}\pi^\beta \left(\mathbf{x}_i - \mathbf{w}_j(t)\right) \quad (8)$$

3. Minimiser $R_{lwd}(\hat{\chi}, \hat{\mathcal{W}}, \Pi)$ par rapport à $\Pi$ en fixant $\chi$ et $\mathcal{W}$. La mis-à-jour de ces vecteurs des pondérations $\pi_j(t+1)$ se fait d'après l'expression :
$$\pi_j(t+1) = \pi_j(t) + \epsilon(t)\mathcal{K}_{j,\chi(\mathbf{x}_i)}\beta\pi_j^{\beta-1}\|\mathbf{x}_i - \mathbf{w}_j(t)\|^2 \quad (9)$$

De la même manière que la version $lwo$-SOM, on fait décroître le pas et le rayon d'apprentissage pour constituer deux phases : une phase d'auto-organisation associée aux grandes valeurs des parametres et une phase de quantification associée aux petites valeurs.

# 5 Caracterisation automatique des groupes

Une procédure de sélection de variables comporte trois éléments essentiels : une mesure de pertinence, une procédure de recherche et un critère d'arrêt. Nous distinguons trois types de méthodologie :

– les approches filtres dont la mesure de pertinence est indépendante de l'algorithme qui utilise ensuite les données ;
– les approches symbioses qui évaluent la pertinence des sous-ensembles de variables à l'aide des performances du système que l'on construit ;
– les approches intégrées pour lesquelles la mesure de pertinence est directement incluse dans la fonction de coût optimisée par le système. Les approches intégrées sont de deux types : (1) globale où la mesure de pertinence est calculée globalement sur les individus ; (2) locale pour la quelle chaque référent a son propre vecteur de mesures de pertinence ;

Nos approches $lwo/d$-SOM font partie des catégories des approches integrées avec une mesure localement adaptative de pertinence. Plusieurs critères d'arrêt ont été introduit dans la littérature, mais souvent l'inconvénient de ces critères est la fixation d'un seuil qui dépend de la base des données. Dans notre cas, pour détecter les variables pertinentes nous avons utilisé un test statistique proposé par Cattell en 1960 Cattell (1966) appelé "Scree Test". Ce test va nous permettre une sélection des variables pertinentes d'une manière automatique et sans définir un seuil d'arrêt a priori.

## 5.1 Critère de sélection : "*Scree Acceleration Test*"

L'utilisation initiale du test "Scree Test", Cattell (1966) était la détermination visuelle du nombre de valeurs propres à prendre en compte lors d'une analyse en composantes principales. L'idée de base est de représenter graphiquement les valeurs propres et de trouver à partir de quelle valeur le graphique semble présenter un changement brutal. Selon Cattell, nous devons

trouver ce qui représente la ligne du changement brutal ”Scree”. Le nombre de composantes à garder correspond au nombre de valeurs propres précédant ce ’Scree’. Fréquement, ce ’Scree’ apparait là où la pente du graphe change radicalement. Ainsi, il s’agit de trouver la décélération maximale dans ce graphique.
Par analogie, l’utilisation de ce test avec nos modèles de pondérations, consiste à détecter, par exemple, le changement brutal dans le graphique des pertinences $\pi_k = (\pi_{k1}, \pi_{k2}..., \pi_{kj}, ..., \pi_{kn})$. Il faudrait donc détecter la plus forte décélération. La procédure de sélection est ainsi composée des étapes suivantes :

1. Ordonner le vecteur des pondérations $\pi_k = (\pi_{k1}, \pi_{k2}..., \pi_{kj}, ..., \pi_{kn})$ en suivant un ordre décroissant. Le nouveau vecteur ordonné est noté $\pi_k = (\pi^1_{..}, \pi^2_{..}..., \pi^i_{..}, ..., \pi^n_{..})$ ; où l’exposant $i$ de la pondération $\pi^i$ indique l’ordre.
2. Calculer les premières différences $df_i = \pi^i_{..} - \pi^{i+1}_{..}$ ;
3. Calculer les deuxièmes différences (l’accélération) $acc_i = df_i - df_{i+1}$
4. Chercher le changement brutal ’scree’ à l’aide de la fonction suivante : $\max_i (abs(acc_i) + abs(acc_{i+1}))$

Ce processus permet de sélectionner toutes les variables se trouvant avant le changement brutal.

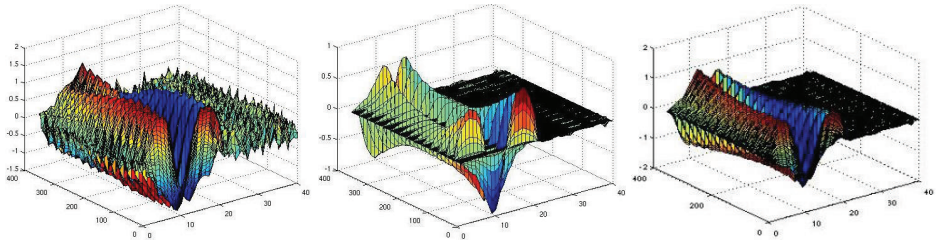
# 6 Validation des approches proposées

Nous avons utilisé différents jeux de données disponibles sur UCI Asuncion et Newman (2007) de taille et de complexité variable pour évaluer nos approches de pondération locale adaptative et de sélection. En particulier dans la partie validation, nous allons donner plus de détails sur la base des vagues de Breiman. Les bases utilisées sont :

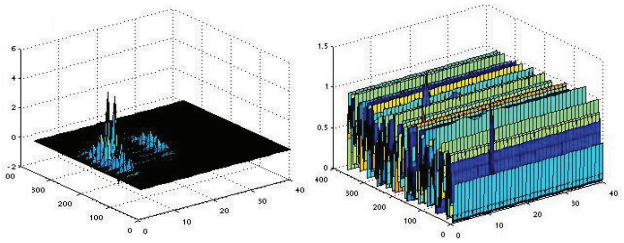
- Vagues de Breiman bruitées : La base est composée de 5000 exemples divisés en 3 classes. La base originale comportait 21 variables, mais 19 variables additionnelles distribuées selon une loi normale ont été rajoutées sous forme de bruit. Chaque observation a été généré comme une combinaison de 2 sur 3 vagues.
- Base de cancers : "*Wisconsin Diagnostic Breast Cancer (WDBC)* : Ce jeu de données contient 569 individus qui sont décrit par 32 variables. 357 individus sont atteint de cancer bénigne et les 212 autres ont des cancers malignes. Les variables décrivent les caractéristiques des noyaux de cellules présentes dans l’image numériques.
- Jeu de données Isolet : Ces données ont été généré comme suit : 150 sujets prononcent chaque lettre de l’alphabet à deux reprises. Ainsi, nous avons 52 exemples de formation de chaque locuteur. Les données sont constituées de 1559 individus et 617 variables. Toutes les variables sont continues.
- La base Madelon : Ces données posent un problème à 2 classes proposé à l’origine pendant la compétition sur la sélection de variables organisée lors de la conférence NIPS’2003, Guyon et al. (2006). Les exemples sont situés sur les sommets d’un hypercube en dimension 5, mais 15 variables redondantes et 480 dimensions bruitées ont été ajoutés. Le jeu de données original était séparé en trois parties (apprentissage, validation et test) mais nous n’avons utilisé que les 2600 observations de l’ensemble d’apprentissage et de validation pour lesquels les classes étaient connues.
- La base "SpamBase" est un jeu de données composé de 4601 observations décrites par 57 variables, chacune decrivant un mail et sa categorie : spam ou non-spam. Les attributs

descriptifs de ces mails sont les frequences d'apparition de certains mots ou caractères ainsi que des informations sur la quantité de caractères mis en capitale.

Dans ces expérimentations, la comparaison des différents résultats est mesurée à l'aide de deux critères externes. On peut utiliser ces indices lorsque la segmentation souhaitée est connue, en particulier sur nos jeux de données. Il s'agit de la comparaison entre la segmentation proposée et une segmentation souhaitée. Ainsi nous avons utilisé, le taux de la pureté et l'indice de Rand, qui calcule le pourcentage du nombre de couples d'observations ayant la même classe et se retrouvant dans le même sous ensemble après segmentation de la carte Saporta (2006).
Nous avons lancé l'apprentissage avec nos algorithmes $lwo/d$-SOM et le test de sélection de variables sur les cinq bases décrites ci-dessus. Nous calculons par la suite les valeurs des indices de qualité pour les deux cartes ($lwo/d - SOM$).



(a) ($W$ de SOM) (b) ($W$ de $lwo$-SOM) (c) $W$ de $lwd$-SOM



(d) $\Pi$ de $lwo$-SOM (e) $\Pi$ de $lwd$-SOM

FIG. 1 – *Carte topologique de taille* $26 \times 14$ *(364 neurones). Les graphiques* $a$*,* $b$ *et* $c$ *indiquent l'ensemble des référents* $\mathcal{W}$ *issus respectivement de SOM classique,* $lwo$*-SOM et* $lwd$*-SOM. Les graphiques* $d$ *et* $e$ *représentent les pondérations locales estimées respectivement par* $lwo$*-SOM et* $lwd$*-SOM.*

## 6.1 Déroulement de l'approche sur un exemple : Vagues de Breiman

Nous utilisons ce jeu de données pour montrer l'intégralité du processus permettant la caractérisation des groupes, à partir de l'apprentissage avec les deux approches ($lwo/d$-SOM)

en passant par la détection et la sélection des variables pertinentes. L'apprentissage d'une carte de dimension $26 \times 14$ pour toutes les observations permet de fournir pour chaque cellule un vecteur référent $\mathbf{w}_k = (w_{k1}, w_{k2}..., w_{kj}, ..., w_{k40})$ et un vecteur de pondérations $\pi_k = (\pi_{k1}, \pi_{k2}..., \pi_{kj}, ..., \pi_{k40})$ de dimension $n = 40$.

La figure 1 ($a$, $b$, $c$) montre, avec une visualisation 3D, l'ensemble des référents $W$ issus de l'apprentissage des trois algorithmes : SOM classique, $lwo$-SOM et $lwd$-SOM. Les pondérations locales $\Pi$ sont preséntées dans la figure 1 ($d$, $e$) issues de l'apprentissage avec $lwo$-SOM et $lwd$-SOM respectivement. Les axes $X$ et $Y$ indiquent respectivement les variables et les indices des référents. L'amplitude indique la valeur estimée. Nous rappelons ici, que dans le cas de l'algorithme $lwo$-SOM, les référents calculés $W$ représentent des observations pondérées ($\pi\mathbf{x}$). En observant les trois graphiques ($a$, $b$, $c$) nous constatons que le bruit qui represente les variables de 19 à 40 est bien visible avec de faibles amplitudes. Cette analyse visuelle des résultats est plus claire avec la version nouvelle $lwo$-SOM que nous avons proposée. Les deux graphiques representant les référents $W$ et les pondérations $\Pi$ montrent que les variables du bruit ne sont pas pertinantes. Après l'analyse des deux types de pondérations (figure 1 ($d$, $e$)), nous constatons que les pondérations $\Pi$ issues de $lwo$-SOM corespondent plus à la structure des données (pertinence des variables) que les pondérations de $lwd$-SOM. Ce phénomène a lieu car les pondérations des observations réprésentent un filtre pour les données et l'estimation des référents tient compte de ce filtrage. Afin de vérifier qu'il est possible de sélectionner les variables d'une manière automatique avec nos algorithmes $lwo$-SOM, nous avons appliqué la phase de sélection sur l'ensemble des référents $\mathcal{W}$ pour la version $lwo/d - SOM$ après segmentation de la carte. Cette phase consiste à détecter les variations brutales pour chacun des vecteurs en entrée. Pour la phase de segmentation nous avons utilisé la classification hiérarchique Vesanto et Alhoniemi (2000).

En utilisant la version $lwo$-SOM la segmentation de la carte avec les référents $\mathcal{W}$, qui sont

TAB. 1 – *Les résultats de la séléction par groupe de la base vagues de Breiman (l'intervalle $[i - j]$ indique les variables de $i$ à $j$)*

| Db | # groupes réels. | lwd-SOM : $\Pi\mathcal{W}$ | lwo-SOM : $\mathcal{W}$ | Classification croisée |
|---|---|---|---|---|
| wave-form | 3 | $cl_1$ : [6-15]<br>$cl_2$ : [4-10]<br>$cl_3$ : [7-19] | $cl_1$ : [3-8 ; 11-16]<br>$cl_2$ : [8-11 ; 14-19]<br>$cl_3$ : [3-20] | $cl_1$ : [3-12]<br>$cl_2$ : [7-15]<br>$cl_3$ : [10-18]<br>$cl_4$ : [5-17] |
| Pureté<br>Rand | | 0,5374<br>0,6068 | 0,5416<br>0,6164 | |

déjà pondérés, permet d'obtenir du premier coup 3 groupes. Par contre avec la version $lwd$-SOM, la segmentation de la carte utilisant les référent $W$ fournie 6 groupes "clusters", mais la segmentation à partir du produit $\Pi\mathcal{W}$ permet d'oboutir à 3 groupes ce qui est significatif pour notre exemple (vagues de Breiman).

La caractérisation des groupes avec l'algorithme "ScreeTest" est fourni dans la Table 1. Pour chaque technique nous montrons les variables sélectionnées et associées à chaque groupe. Nous observons que les deux techniques fournissent 3 groupes caractérisés par des variables

différentes, mais qui se recouvrent. Nous constatons que pour les groupes $cl_1$, $cl_2$ et $cl_3$, les variables détectées avec la carte $lwd$-SOM sont inclues dans l'ensemble des variables détectées avec la carte $lwo$-SOM. Nous observons aussi qu'aucune variable du bruit n'est sélectionnée avec la méthode $lwd$-SOM, au contraire de la technique $lwo$-SOM qui détecte une seule variable du bruit 20. Ceci ne réduit pas la qualité de la segmentation puisque le calcul de la pureté des deux partitions, confirme une meilleur segmentation avec la méthode $lwo$-SOM (Table 1). Cette augmentation est faible, mais elle est significatif puisqu'acune connaissance n'est a priori utilisée pour cette tâche. En comparant ces résultats avec des approches de classification croisée, nous constatons qu'on selectionne les mêmes variables.

## 6.2 Résultats sur d'autres bases de données

En ce qui concerne les autres bases de données, nous allons ce contenter dans cette section d'indiquer les résultats obtenus après la phase de sélection des variables. Pour la base WDBC, l'application de nos approches $lwo/d$-SOM nous a permis d'obtenir les variables 4 et 24 comme variables pertinentes de la base avec une forte importance pour le premier groupe et pour le 9ieme groupe (acune variable n'est selectionées pour le rest des groupes). En ce qui concerne la base Isolet nous avons constaté un accord de la sélection des variables non pertinentes. Les algorithmes $lwo/d$-SOM associés au test de sélection fournissent les variables non pertinentes dont les indices appartiennent à l'intervalle 300 à 500. En comparant les indices de qualité de partionement (Indice de pureté), nous constatons une amélioration pour l'approche $lwo$-SOM par rapport à $lwd$-SOM.

TAB. 2 – *Détéction des variables pertinantes par groupes pour les db. : wdbc, madelon, isolet et spambase (l'intervalle $[i-j]$ indique les variables de $i$ à $j$ ; $cl_i$ : groupe $i$)*

| Db. | # gr. réel | $lwd$-SOM | | $lwo$-SOM | | Classification croisée |
|---|---|---|---|---|---|---|
| | | sélection $\Pi\mathcal{W}$ | Pureté | sélection sur $\mathcal{W}$ | Pureté | |
| wdbc | 2 | $cl_1$-$cl_9$ : (4 ;24) | 0,6274 | $cl_1$-$cl_9$ : (4 ;24) | 0,8682 | $cl_1$-$cl_5$ (4 ;24) |
| Madelon | 2 | $cl_1$ :1<br>$cl_2$ : (91, 281, 403-424) | 0,5242 | $cl_1$ :1<br>$cl_2$ : (242, 417-452] | 0,5347 | $cl_1$ :[445-450]<br>$cl_2$ :[450-460] |
| Isolet | 26 | $cl_1$-$cl_{13}$ :<br>[1-330, 450-617] | 0,5242 | $cl_1$-$cl_{13}$ :<br>[5-302, 434-488]<br>[545-551,586-593] | 0,5261 | $cl_1$-$cl_{15}$ :<br>[1-300, 450-620] |
| Spam | 2 | $cl_1$ :56 ; $cl_2$ :57 | 0,6103 | $cl_1$ : 56 ; $cl_2$ : 57 | 0,6413 | $cl_1$ :56 ;$cl_2$ :57 |

Après l'analyse des expérimentation nous pouvons déduire quelques caractéristiques particulières et quelques comparaisons entre $lwo$-SOM et $lwd$-SOM :

- Les deux approches itératives de pondérations $lwo/d$-SOM ne donne aucune importance aux variables non-pertinentes à l'opposé avec le SOM classique où nous avons des valeurs plus élevées pour les variables non pertinentes.
- Les deux méthodes stochastiques sont plus rapides en les comparant particulièrement à la version analytique "batch" de pondération déjà proposé de Anonyme.

- Les deux algorithmes ($lwo/d$-SOM) et la Scree Test fournissent une caractérisation des groupes plus adapté grâce à l'utilisation des pondérations locales.
- Les indices de qualité de la segmentation (Indice de Pureté) sont meilleurs pour les deux approches avec un avantage pou l'algorithme $lwo$-SOM.
- Les pondérations $\Pi$ après l'apprentissage de $lwo$-SOM apporte plus d'information que celles fournis par $lwd$-SOM du fait de l'adaptation des pondérations aux observations et pas aux distances. L'avantage de la méthode $lwo$-SOM est obtenu grâce à la pondération en amont des observations.

## 7 Conclusion

Dans ce papier, nous avons introduit deux approches pour caractériser les groupes "clusters" en utilisant les cartes auto-organisatrices. La première est une nouvelle technique de caractérisation en pondérant les observations ($lwo$-SOM). La deuxième est une reformulation stochastique de la pondération classique des distances ($lwd$-SOM). Nous avons utilisé un test statistique original "Scree Test" qui nous a permis de détecter automatiquement les variables les plus pertinentes. Nous avons montré à travers différents exemples l'intérêt de l'estimation des pertinences des variables pour la visualisation et la sélection des variables. Nous avons montré aussi que les deux approches peuvent être considérées comme une pseudo-classification croisée ou simultanée des observations et des variables. Enfin, contrairement à la classification croisée, la méthode proposée dans cet article permet de caractériser les groupes d'une manière automatique. L'estimation du nombre correct de groupes est en relation avec la stabilité de la segmentation et la validité des groupes générés. En perspective, nous allons mesurer cette stabilité pour nos algorithmes par des techniques de sous-échantillonnage.

## Références

Asuncion, A. et D. Newman (2007). UCI machine learning repository.

Basak, J., R. K. De, et S. K. Pal (1998). Unsupervised feature selection using a neuro-fuzzy approach. *Pattern Recogn. Lett. 19*(11), 997–1006.

Bassiouny, S., M. Nagi, et M. F. Hussein (2004). Feature subset selection in som based text categorization. In *IC-AI*, pp. 860–866.

Bishop, C. M., M. Svensén, et C. K. I.Williams (1998). Gtm : The generative topographic mapping. *Neural Comput 10*(1), 215–234.

Blansche, A., P. Gancarski, et J. Korczak (2006). Maclaw : A modular approach for clustering with local attribute weighting. *Pattern Recognition Letters 27(11)*, 1299–1306.

Cattell, R. (1966). The scree test for the number of factors. *MBR 1*, 245–276.

Cheeseman, P., J. Kelly, M. Self, J. Stutz, W. Taylor, et D. Freeman (1988). Autoclass : A bayesian classification system. In *Fifth International Workshop on Machine learning*.

Cottrell, M., S. Ibbou, et P. Letrémy (2004). Som-based algorithms for qualitative variables. *Neural Netw. 17*(8-9), 1149–1167.

Fisher, D. (1996). Iterative optimization and simplification of hierarchical clusterings. *Journal of Artificial Intelligence Research. (JAIR) 4*, 147–178.

Frigui, H. et O. Nasraoui (2004). Unsupervised learning of prototypes and attribute weights. *Pattern Recognition 37(3)*, 567–581.

Grozavu, N., Y. Bennani, et M. Lebbah (2008). Pondération locale des variables en apprentissage numérique non-supervisé. *Extraction et Gestion des Connaissances (EGC 08)*, 45–54.

Guérif, S. et Y. Bennani (2007). Dimensionality reduction trough unsupervised features selection. *International Conference on Engineering Applications of Neural Networks*.

Guyon, I., S. Gunn, M. Nikravesh, et L. Zadeh (Eds.) (2006). *FE, Found. and Appl.* Springer.

Huang, J. Z., M. K. Ng, H. Rong, et Z. Li (2005). Automated variable weighting in k-means type clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence 27(5)*.

Kohonen, T. (2001). *Self-organizing Maps*. Springer Berlin.

Lebbah, M., N. Rogovschi, et Y. Bennani (2007). Besom : Bernoulli on self organizing map. In *International Joint Conferences on Neural Networks. IJCNN 2007, Orlando, Florida*.

Li, Y., B.-L. Lu, et Z.-F. Wu (2006). A hybrid method of unsupervised feature selection based on ranking. In *ICPR '06*.

Liu, L., J. Kang, J. Yu, et Z. Wang (2005). A comparative study on unsupervised feature selection methods for text clustering. pp. 597–601.

Parsons, L., E. Haque, et H. Liu (2004). Subspace clustering for high dimensional data : a review. *SIGKDD Explor. Newsl. 6*(1), 90–105.

Questier, F., R. Put, D. Coomans, B. Walczak, et Y. V. Heyden (2005). The use of cart and multivariate regression trees for supervised and unsupervised feature selection. pp. 45–54.

Roth, V. et T. Lange. Feature selection in clustering problems. In S. Thrun, L. Saul, et B. Schölkopf (Eds.), *Advances in Neural Information Processing Systems 16*.

Saporta, G. (2006). *Probabilités, analyse des données et statistiques*. Editions Technip.

Vesanto, J. et E. Alhoniemi (2000). Clustering of the self-organizing map. *Neural Networks, IEEE Transactions on 11*(3), 586–600.

## Summary

We introduce a new approach, which provide simultaneously Self-Organizing Map (SOM) and local weight vector for each cluster. The proposed approach is computationally simple, and learns a different feature vector weights for each cluster. Clustering and feature weighting offers two advantages: First, they guide the SOM process to cluster the data set into more meaningful clusters; Second, they can be used to characterize a cluster using a feature selection method. Based on the Self-Organizing Map approach, we present two new simultaneously clustering and weighting algorithms, called lwo-SOM and lwd-SOM respectively. These two algorithms achieve the same goal, however, they minimize different objective functions. lwo-SOM estimates the feature vector weights by weighting observations, while lwd-SOM estimates the feature vector weights by weighting distance between observations and prototypes. We illustrate the performance of the proposed approach using different data sets.