Un algorithme de classification topographique non supervisée à deux niveaux simultanés

Guénaël Cabanes, Younès Bennani

LIPN - UMR 7030
Université Paris 13 - CNRS
99, av. J-B Clément - F-93430 Villetaneuse
{cabanes, younes}@lipn.univ-paris13.fr

Résumé. Une des questions les plus importantes pour la plupart des applications réelles de la classification est de déterminer un nombre approprié de groupes (*clusters*). Déterminer le nombre optimal de groupes est un problème difficile, puisqu'il n'y a pas de moyen simple pour connaître ce nombre sans connaissance a priori. Dans cet article, nous proposons un nouvel algorithme de classification non supervisée à deux niveaux, appelé S2L-SOM (Simultaneous Two-level Clustering - Self Organizing Map), qui permet de déterminer automatiquement le nombre optimal de groupes, pendant l'apprentissage d'une carte auto-organisatrice. L'estimation du nombre correct de groupes est en relation avec la stabilité de la segmentation et la validité des groupes générés. Pour mesurer cette stabilité nous utilisons une méthode de sous-échantillonnage. Le principal avantage de l'algorithme proposé, comparé aux méthodes classiques de classification, est qu'il n'est pas limité à la détection de groupes convexes, mais est capable de détecter des groupes de formes arbitraires. La validation expérimentale de cet algorithme sur un ensemble de problèmes fondamentaux pour la classification montre sa supériorité sur les méthodes standards de classification à deux niveaux comme SOM+K-Moyennes et SOM+Hierarchical-Agglomerative-Clustering.

1 Introduction

La classification non supervisée, ou clustering, est un outil très performant pour la détection automatique de sous-groupes pertinents (ou *clusters*) dans un jeu de données, lorsqu'on n'a pas de connaissances a priori sur la structure interne de ces données. Les membres d'un même cluster doivent êtres similaires entre eux, contrairement aux membres de groupes différents (homogénéité interne et séparation externe). La classification non supervisée joue un rôle indispensable pour la compréhension de phénomènes variés décrits par des bases de données. Un problème de regroupement peut être défini comme une tâche de partitionnement d'un ensemble d'items en un ensemble de sous-ensembles mutuellement disjoints. La classification est un problème de regroupement qui peux être considéré comme un des plus compétitifs en apprentissage non-supervisé. De nombreuses approches ont été proposées (Jain et Dubes, 1988). Les approches les plus classiques sont les méthodes hiérarchiques et les méthodes partitives.

Les méthodes de classification hiérarchiques agglomératives (CAH) utilisent un arbre hiérarchique (dendrogramme) construit à partir des ensembles à classifier. Dans ce cas, les nœuds de l'arbre issus d'un même parent forment un groupe homogène (Ward, 1963), alors que les méthodes partitives regroupent des données sans structure hiérarchique.

Une méthode efficace utilisée pour la classification est la carte auto-organisatrice ou Self Organizing Map (SOM : Kohonen, 1984, 2001). Une SOM est un algorithme neuro-inspiré qui, par un processus non-supervisé compétitif, est capable de projeter des données de grandes dimensions dans un espace à deux dimensions. Cet algorithme d'apprentissage non supervisé est une technique non linéaire très populaire pour la réduction de dimensions et la visualisation des données. Dans cette approche, la détection des regroupements est en général obtenue en utilisant d'autres techniques de classification telles que K-Moyennes ou des méthodes hiérarchiques. Dans la première phase du processus, une SOM standard est utilisée pour estimer les référents (moyennes locales). Dans la deuxième phase, les partitions associées à chaque référent sont utilisées pour former la classification finale des données en utilisant une méthode de classification traditionnelle. Une telle approche est appelée méthode à deux niveaux. Dans cet article, nous nous intéressons particulièrement aux algorithmes de classification à deux niveaux.

Une des questions les plus importantes pour la plupart des applications réelles, aussi connue comme le "problème de sélection de modèle", est de déterminer un nombre approprié de groupes. Sans connaissances a priori il n'y a pas de moyen simple pour déterminer ce nombre. L'objectif de ce travail est de fournir une approche de classification à deux niveaux simultanés utilisant une SOM, qui peut être appliquée à de grandes bases de données. La méthode proposée regroupe automatiquement les données, c'est-à-dire que le nombre de groupes est déterminé automatiquement pendant le processus d'apprentissage, i.e. aucune hypothèse a priori sur le nombre de groupes n'est exigée. Cette approche a été évaluée sur un jeu de problèmes fondamentaux pour la classification et montre d'excellents résultats comparés aux approches classiques.

Le reste de cet article est organisé comme suit. La Section 2 présente l'algorithme de classification à deux niveaux simultanés. La Section 3 décrit les bases de données utilisées pour la validation ainsi que le protocole expérimental. Dans la section 4 nous présentons les résultats de la validation et leur interprétation. Une conclusion et des perspectives sont données dans la section 5.

2 Algorithme de classification à deux niveaux simultanés basé sur une carte auto-organisatrice

Dans un espace de grande dimension les données peuvent être fortement dispersées, ce qui rend difficile pour un algorithme de classification la recherche de structures dans les données. En réponse à ce problème, un grand nombre d'approches basées sur une réduction de dimensions ont été développées et testées pour différents domaines d'application. L'idée principale de ces approches est de projeter les données dans un espace de faible dimension tout en conservant leur topologie. Les nouvelles coordonnées des données ainsi projetées peuvent alors être efficacement utilisées par un algorithme de classification. C'est ce qu'on appelle la classification à deux niveaux. De nombreuses approches ont été proposées pour résoudre des problèmes

de classification à deux niveaux (Bohez, 1998; Hussin et al., 2004; Ultsch, 2005; Guérif et Bennani, 2006; Korkmaz, 2006). Les approches basées sur l'apprentissage d'une carte auto-organisatrice sont particulièrement efficaces du fait de la vitesse d'apprentissage de SOM et de ses performances en réduction de dimensions non linéaire (Hussin et al., 2004; Ultsch, 2005; Guérif et Bennani, 2006).

Bien que les méthodes à deux niveaux soient plus intéressantes que les méthodes classiques (en particulier en réduisant le temps de calcul et en permettant une interprétation visuelle de l'analyse, Vesanto et Alhoniemi (2000)), la classification obtenue à partir des référents n'est pas optimale, puisqu'une partie de l'information à été perdue lors de la première étape. De plus, cette séparation en deux étapes n'est pas adaptée à une classification dynamique de données qui évoluent dans le temps, malgré des besoins importants d'outils pour l'analyse de ce type de données. Nous proposons donc ici un nouvel algorithme de classification non-supervisée, S2L-SOM (Simultaneous Two Level - SOM), qui apprend simultanément les prototypes (référents) d'une carte auto-organisatrice et sa segmentation.

2.1 Principe

Une SOM est un algorithme d'apprentissage compétitif non-supervisé à partir d'un réseau de neurones artificiels (Kohonen, 1984, 2001). Lorsqu'une observation est reconnue, l'activation d'un neurone du réseau, sélectionné par une compétition entre les neurones, a pour effet le renforcement de ce neurone et l'inhibition des autres (c'est la règle du "Winner Takes All"). Chaque neurone se spécialise donc au cours de l'apprentissage dans la reconnaissance d'un certain type d'observations. La carte auto-organisatrice est composée d'un ensemble de neurones connectés entre eux par des liens topologiques qui forment une grille bi-dimensionnelle. Chaque neurone est connecté à n entrées (correspondant aux n dimensions de l'espace de représentation) selon n pondérations w (qui forment le vecteur prototype du neurone). Les neurones sont aussi connectés à leurs voisins par des liens topologiques. Le jeu de données est utilisé pour organiser la carte selon les contraintes topologiques de l'espace d'entrée. Ainsi, une configuration entre l'espace d'entrée et l'espace du réseau est construite ; deux observations proches dans l'espace d'entrée activent deux unités proches sur la carte. Une organisation spatiale optimale est déterminée par la SOM à partir des données et quand la dimension de l'espace d'entrée est inférieure à trois, aussi bien la position des vecteurs de poids que des relations de voisinage directes entre les neurones peuvent être représentées visuellement. Le neurone gagnant met à jour son vecteur prototype, de façon à devenir plus sensible à une présentation future de ce type de donnée. Cela permet à différents neurones d'être entrainés pour différents types de données. De façon à assurer la conservation de la topologie de la carte, les voisins du neurone gagnant peuvent aussi ajuster leur vecteur prototype vers le vecteur présenté, mais dans un degré moindre, en fonction de leurs distances au prototype gagnant. Ainsi, les prototypes les plus proches d'une donnée correspondent à des neurones voisins sur la carte. En général, on utilise pour cela une fonction de voisinage gaussienne à symétrie radiale K_{ij}.

Dans l'algorithme S2L-SOM, nous proposons d'associer à chaque connexion de voisinage une valeur réelle qui indique la pertinence des neurones connectés. Étant donné la contrainte d'organisation de la carte, les deux meilleurs représentants de chaque donnée sont reliés par une connexion topologique. Cette connexion sera "récompensée" par une augmentation de sa valeur, alors que toutes les autres connexions issues du meilleur représentant seront "punies" par une diminution de leurs valeurs. Les récompenses et les punitions sont d'autant plus

importantes que l'apprentissage est bien avancé et donc que la structure de la carte est bien représentative de la structure des donnés.

Ainsi, à la fin de l'apprentissage, un ensemble de prototypes interconnectés sera représentatif d'un sous-groupe pertinent de l'ensemble des données : un *cluster*.

2.2 Algorithme S2L-SOM

L'apprentissage connexionniste est souvent présenté comme la minimisation d'une fonction de coût. Dans notre cas, cela correspond à la minimisation de la distance entre les données et les prototypes de la carte, pondérée par une fonction de voisinage K_{ij} (Kohonen, 2001). Pour ce faire, nous utilisons un algorithme de gradient. La fonction de coût à minimiser est définie par :

$$\tilde{R}(w) = \frac{1}{N} \sum_{k=1}^{N} \sum_{j=1}^{M} K_{jN(x^{(k)})} \parallel w_{.j} - x^{(k)} \parallel^2$$

Avec N le nombre de données, M le nombre de neurones de la carte, $w_{.j} = (w_{0j}, w_{1j}, ..., w_{nj})$, $N(x^{(k)})$ est le neurone dont le vecteur prototype est le plus proche de la donnée $x^{(k)}$. K_{ij} est une fonction symétrique positive à noyau : la fonction de voisinage. L'importance relative d'un neurone i comparé à un neurone j est pondérée par la valeur de K_{ij}, qui peut être définie ainsi :

$$K_{ij} = \frac{1}{\lambda(t)} \times e^{-\frac{d_1^2(i,j)}{\lambda^2(t)}}$$

$\lambda(t)$ est une fonction de température qui contrôle l'étendue du voisinage qui diminue avec le temps t de λ_i à λ_f (par exemple $\lambda_i = 2$ à λ_f = 0,5) :

$$\lambda(t) = \lambda_i (\frac{\lambda_f}{\lambda_i})^{\frac{t}{t_{max}}}$$

t_{max} est le nombre maximum d'itérations autorisé pour l'apprentissage. $d_1(i,j)$ est la distance de Manhattan définie entre deux neurones i (de coordonnée (k,m)) et j (de coordonnée (r,s)) sur la grille de la carte :

$$d_1(i,j) = \parallel r - k \parallel + \parallel s - m \parallel$$

Le processus d'apprentissage de S2L-SOM est proche d'un "Competitive Hebbian Learning" (CHL, Martinetz, 1993). La différence essentielle est qu'un CHL ne change pas les références des prototypes en cours d'apprentissage. De plus, avec S2L-SOM, seuls les neurones voisins sur la carte peuvent être connectés, ce qui conserve la topologie en deux dimensions de la carte et permet une réduction de dimensions et une visualisation simple de la structure des données. Par ailleurs, l'utilisation d'une valeur de récompense associée aux connexions donne une information sur la représentativité locale des deux neurones connectés, ce qui n'est pas le cas avec un CHL. Martinetz (1993) a montré que le graphe généré de cette manière préserve la topologie de façon optimale. En particulier chaque arc de ce graphe suit la triangulation de Delaunay correspondant aux vecteurs de référence.

L'algorithme S2L-SOM procède essentiellement en trois phases :

1. **Phase d'initialisation** :
   - Définir la topologie de la carte.
   - Initialiser aléatoirement tous les prototypes $w_{.j} = (w_{0j}, ..., w_{nj})$ pour chaque neurone j.
   - Initialiser les connexions ν entre chaque couple de neurones i et j :

$$\forall i, j \;\; \nu_{ij} = 0$$

2. **Phase de compétition** :
   - Présenter une donnée $x^{(k)}$ choisie aléatoirement.
   - Parmi les M neurones, choisir les deux meilleurs $N_1(x^{(k)})$ et $N_2(x^{(k)})$ pour représenter cette donnée :

$$N_1(x^{(k)}) = \underset{1 \leq i \leq M}{Argmin} \parallel x^{(k)} - w_{.i} \parallel^2$$
$$N_2(x^{(k)}) = \underset{1 \leq i \leq M, \;\; i \neq N_1}{Argmin} \parallel x^{(k)} - w_{.i} \parallel^2$$

   - Augmenter la valeur de la connexion entre $N_1(x^{(k)})$ et $N_2(x^{(k)})$ et diminuer les valeurs des autres connexions issues de $N_1(x^{(k)})$:

$$\nu_{N_1 N_2}(t) = \nu_{N_1 N_2}(t\text{-}1) + R(t)$$
$$\nu_{N_1 N_i}(t) = \nu_{N_1 N_i}(t\text{-}1) - \delta.R(t) \;\; \forall i \neq 2 \text{ et } N_i \text{ voisin de } N_1$$

   Avec :

$$R(t) = \frac{P}{1 + e^{-\left(\frac{t}{t_{max}}\right)}}$$
P : nombre de connexions punies
δ est une constante (par exemple $\delta = 0,01$)

3. **Phase d'adaptation** :
   - Mettre à jour les prototypes $w_{.j}$ de chaque neurone j selon la règle d'adaptation suivante :

$$w_{.j}(t) = w_{.j}(t-1) - \varepsilon(t) K_{j N_1(x^{(k)})}(w_{.j}(t-1) - x^{(k)})$$

   où $\varepsilon(t)$ est le pas du gradient

4. **Répéter les phases 2 et 3** jusqu'à ce que $t = t_{max}$.

À la fin de l'apprentissage, chaque ensemble de neurones connectés entre eux par des connexions de valeurs positives est representatif d'un groupe homogène de données. L'algorithme attribue un numero à chacun de ces ensemble. Le nombre de groupe est ainsi obtenu automatiquement.

3 Validation

3.1 Description des bases de données utilisées

De façon à démontrer les performances de l'algorithme de classification proposé, huit bases de données présentant différentes difficultés de classification ont été utilisées.

Les bases de données "Hepta", "Chainlink", "Atom" et "TwoDiamonds" proviennent du Fundamental Clustering Problem Suite (FCPS Ultsch, 2005). Nous avons généré aussi quatre autres bases de données intéressantes ("Rings", "Spirals", "HighDim" et "Random"). "Rings" est composée de 3 groupes en 2 dimensions non linéairement séparables et de densité et variance différentes : un anneau de rayon 1 pour 700 points (forte densité), un anneau de rayon 3 pour 300 points (faible densité) et un anneau de rayon 5 pour 1500 point (densité moyenne). "HighDim" est constitué de 9 groupes de 100 points chacun bien séparés dans un espace à 15 dimensions. "Random" est un tirage aléatoire de 1000 points dans un espace à 8 dimensions. Enfin "Spirals" est constitué de deux spirales parallèles de 1000 points chacune dans un anneaux de 3000 points. La densité des points dans les spirales diminue avec le rayon.

Bases	**N**	**D**	**G**	**Difficultés**
Hepta	212	3	7	Densité différentes
Chainlink	1000	3	2	Non linéairement séparable
Atom	800	3	2	Non linéairement séparable différentes densités et variances
TwoDiamonds	800	2	2	Groupes en contact
Rings	2500	2	3	Non linéairement séparable différentes densités et variances
Spirals	5000	2	3	Non linéairement séparable différentes densités et variances
HighDim	900	15	9	Grandes dimensions
Random	1000	8	1	Pas de structure

TAB. 1 – *Description des bases de données pour la validation (N : nombre de données, D : nombre de dimensions, G : nombre de groupes).*

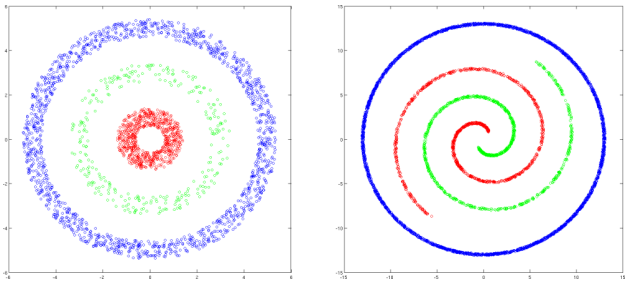


FIG. 1 – *Données "Rings" et "Spirals".*

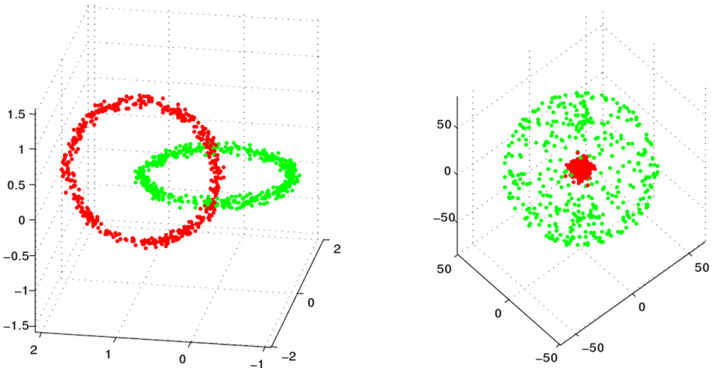


FIG. 2 – *Données "Chainlink" et "Atom".*

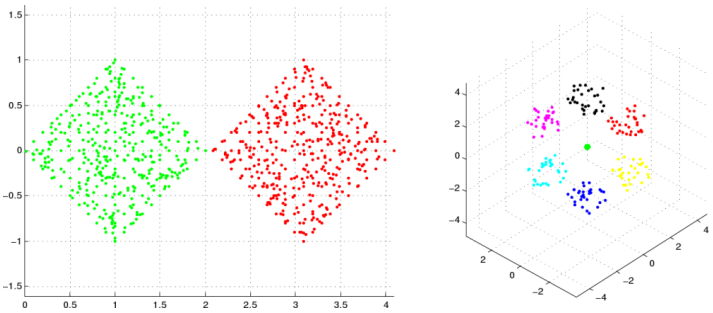


FIG. 3 – *Données "TwoDiamonds" et "Hepta".*

3.2 Protocole expérimental

Nous avons comparé les performances de S2L-SOM en terme de qualité de la segmentation et de stabilité par rapport aux performances des méthodes classiques à un ou deux niveaux. Les algorithmes de comparaison choisis sont K-Moyennes, SingleLinkage et Ward appliqués sur les données ou sur les prototypes de la carte après apprentissage. L'indice de Davies-Bouldin (Davies et Bouldin, 1979) est utilisé pour déterminer le meilleur découpage des arbres (SingleLink et Ward) ou le nombre optimal K de centroïdes pour K-Moyennes. S2L-SOM détermine automatiquement le nombre de classes et n'a pas besoin d'utiliser cet indice.

Dans cet article la qualité de la segmentation à été évaluée à partir d'indices externes (indices de Rand et Jaccard) fréquemment utilisés (Halkidi et al., 2001, 2002). En effet, si des catégories indépendantes des données sont connues, on peut demander si la classification obtenue correspond à ces catégories.

$$Rand = \frac{a_{00} + a_{11}}{a_{00} + a_{01} + a_{10} + a_{11}} \qquad Jaccard = \frac{a_{11}}{a_{01} + a_{10} + a_{11}}$$

Ici a_{00} est le nombre de paires d'objets dont les deux éléments sont dans la même catégorie et le même cluster. a_{01} est le nombre de paires d'objets dont les deux éléments sont dans la même catégorie mais pas le même cluster, alors que a_{10} est le nombre de paires d'objets dont les deux éléments sont dans le même cluster mais pas la même catégorie. Pour finir, a_{11} est le nombre de paires d'objets dont les deux éléments ne sont ni dans le même cluster, ni dans la même catégorie.

Nous avons aussi utilisé pour ces travaux les indices internes de Davies-Bouldin (1979) et Calinski-Harabasz (1974). Ici la principale question est d'estimer à quel point une segmentation des données correspond à sa structure interne. En l'absence de connaissances a priori cette segmentation peut toujours être évaluée par des critères internes comme l'homogénéité intra-groupes et la séparation entre groupes. Les valeurs de ces indices ont été normalisés entre 0 et 1 pour chaque test pour une meilleure lisibilité.

L'indice suggéré par Davies et Bouldin (1979) pour différentes valeurs de K (nombre de cluster) est typiquement introduit comme suit pour évaluer les concepts de séparation entre groupes (le dénominateur) et l'homogénéité intra-groupes (le numérateur). Soit s_i la racine carrée de l'erreur standard (variance intra-groupes) du groupe i de centroïde c_i :

$$DB(K) = \frac{1}{K} \sum_{i=1}^{K} \frac{s_i + s_j}{\parallel c_i - c_j \parallel^2}$$

L'indice de Calinski et Harabasz ((1974) est le plus largement utilisé dans les méthodes de classification classiques. Il peut être défini comme suit :

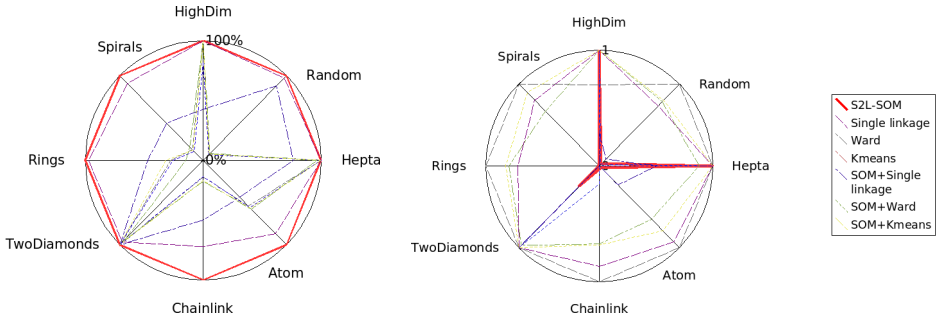
$$CH(K) = \frac{trace(B)/(K-1)}{trace(W)/(N-K)}$$

Avec N le nombre de points de données et K le nombre de groupes. $trace(W)$ est la somme des distances carrées inter-groupes alors que $trace(B)$ est la somme des distances carrées intra-groupes. Cet indice présente une valeur élevée lorsque le nombre de groupes est optimum.

Le concept de stabilité est aussi utilisé pour estimer la validité de la segmentation. Pour évaluer la stabilité des différents algorithmes, nous utilisons une méthode de sous-échantillonnage (Ben-Hur et al., 2002). Pour chaque base de donnée, chaque sous-échantillon est segmenté par un algorithme de classification, et nous comparons deux à deux avec l'indice de Jaccard les différences entre les segmentations obtenues. Ce processus est répété un grand nombre de fois et la moyenne de l'indice est considérée comme une estimation fiable de la stabilité de la classification.

4 Résultats

Les résultats pour les indices externes montrent que pour ces données S2L-SOM est capable de retrouver sans aucune erreur la segmentation attendue. Ce n'est pas le cas des autres algorithmes, en particulier lorsque les groupes sont de formes arbitraires ou lorsqu'il n'y a pas de structure dans les données (voir figure 4(a)). Les segmentations obtenues par S2L-SOM sont d'excellentes qualités selon les indices internes lorsque les données sont regroupées en amas compacts et plus ou moins hypersphériques ("Hepta" ou "HighDim" par exemple). Par contre ces indices ne sont pas du tout adaptés à des groupes de formes arbitraires ("Rings", "Spirals", "Chainlink" ou "Atom", figure 4(b)), ce qui explique les mauvaises performances de S2L-SOM pour ces données. Il faut noter aussi que nous ne pouvons pas évaluer de cette façon une segmentation en un seul groupe, comme c'est le cas pour la segmentation des données "Random" par S2L-SOM.



(a) Indice externe de Jaccard. (b) Indice interne de Calinski-Harabasz.

FIG. 4 – *Valeur des indices de qualité de la segmentation pour chaque algorithme sur chaque base de données.*

En ce qui concerne la stabilité (figure 5), S2L-SOM montre des résultats excellents pour les données regroupées en hypersphères, quelle que soit la dimension ("Hepta" et "HighDim"), mais aussi dans les cas où les groupes sont de formes arbitraires en deux dimensions ("Rings" et "Spirals") et lorsque les données ne sont pas structurées ("Random"). On remarque que dans ce dernier cas la segmentation obtenue par les méthodes classiques est extrêmement instable. Lorsque les données ne sont pas linéairement séparables dans des dimensions supérieures à deux ("Atom" et "Chainlink"), l'algorithme est limité par la contrainte topologique en deux dimensions de la carte auto-organisatrice et la stabilité de la segmentation n'est pas maximale. On peut cependant noter que même dans ce cas S2L-SOM reste plus stable que la quasi totalité des méthodes classiques. Par contre, tout en présentant une stabilité relativement élevée, S2L-SOM est moins stable que la plupart des méthodes classiques lorsque les groupes pressentent un point de contact ("Diamonds"). En effet, ce point de contact favorise la création et l'augmentation par l'algorithme de la valeur des connexions entre les deux groupes.

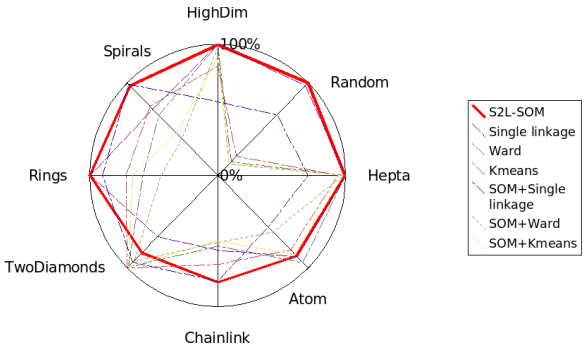
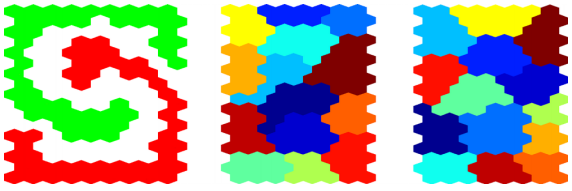
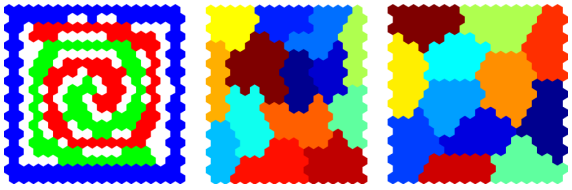


FIG. 5 – *Valeur de l'indice de stabilité de la segmentation pour chaque algorithme sur chaque base de données.*

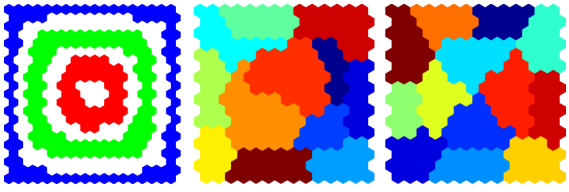
La visualisation des groupes obtenus confirme ces résultats. En effet, l'algorithme S2L-SOM est un puissant outil pour la visualisation en deux dimensions de la segmentation obtenue. Les groupes sont aisément et clairement identifiables, ainsi que les zones sans données. Tel qu'on peut le voir sur la figure 6, les résultats obtenus avec l'algorithme S2L-SOM sont plus proches de la réalité que ceux obtenus par des méthodes classiques.



(a) Données "Chainlink".



(b) Données "Spirals".



(c) Données "Rings".

FIG. 6 – *Visualisation des groupes obtenus avec, de gauche à droite, S2L-SOM, SOM+WARD et SOM+K-Moyennes.*

5 Conclusions et perspectives

Dans cet article, nous proposons une méthode de classification à deux niveaux simultanés. On utilise une SOM comme technique de réduction de dimensions et effectue en parallèle une classification optimisée. Les performances de cette méthode ont été évaluées à partir de tests sur une série de problèmes fondamentaux pour la classification, et comparées aux méthodes à deux niveaux classiques s'appuyant sur CAH ou K-Moyennes. Les résultats expérimentaux démontrent que l'algorithme proposé produit une classification de meilleure qualité que les approches classiques. Ils montrent aussi que le grand avantage de l'algorithme S2L-SOM est qu'il n'est pas limité aux groupes de formes convexes, mais est capable d'identifier des groupes de formes arbitraires. Pour finir, le nombre de groupes est déterminé automatiquement dans notre approche pendant l'apprentissage, c'est-à-dire qu'aucun a priori sur ce nombre n'est requis.

Cependant, cette méthode ne peut fonctionner que si les clusters sont suffisamment séparés dans l'espace de données. En effet, des groupes qui se touchent ne sont définis que par une diminution de la densité dans la zone de contact, ce qui ne peut pas être détecté par S2L-SOM. Dans le futur, nous prévoyons donc d'utiliser des informations sur la densité des données pour améliorer les performances de l'algorithme. Nous prévoyons aussi d'incorporer de la plasticité à l'algorithme S2L-SOM, pour rendre le modèle incrémental et évolutif.

Remerciements : Ce travail a été soutenu en partie par le projet "Sillage", financé par l'ANR (Agence Nationale de la Recherche).

Références

Ben-Hur, A., A. Elisseeff, et I. Guyon (2002). A stability based method for discovering structure in clustered data. *Pacific Symposium on Biocomputing 7*, 6–17.

Bohez, E. L. J. (1998). Two level cluster analysis based on fractal dimension and iteratedfunction systems (ifs) for speech signal recognition. *IEEE Asia-Pacific Conference on Circuits and Systems*, 291–294.

Calinski, T. et J. Harabasz (1974). Dendrite method for cluster analysis. *Communications in Statistics 3*(1), 1–27.

Davies, D. L. et D. W. Bouldin (1979). A cluster separation measure. *IEEE Transactions on Pattern Recognition and Machine Intelligence 1*(2), 224–227.

Guérif, S. et Y. Bennani (2006). Selection of clusters number and features subset during a two-levels clustering task. In *Proceeding of the 10th International Conference Artificial intelligence and Soft Computing 2006*, Palma de Mallorca, Spain, pp. 28–33.

Halkidi, M., Y. Batistakis, et M. Vazirgiannis (2001). On Clustering Validation Techniques. *Journal of Intelligent Information Systems 17*(2-3), 107–145.

Halkidi, M., Y. Batistakis, et M. Vazirgiannis (2002). Cluster Validity Methods. *SIGMOD Record 31*(2,3), 40–45, 19–27.

Hussin, M. F., M. S. Kamel, et M. H. Nagi (2004). An efficient two-level SOMART document clustering through dimensionality reduction. In *ICONIP*, pp. 158–165.

Jain, A. K. et R. C. Dubes (1988). *Algorithms for clustering data*. Upper Saddle River, NJ, USA : Prentice-Hall, Inc.

Kohonen, T. (1984). *Self-Organization and Associative Memory*. Berlin : Springer-Verlag.

Kohonen, T. (2001). *Self-Organizing Maps*. Berlin : Springer-Verlag.

Korkmaz, E. E. (2006). A two-level clustering method using linear linkage encoding. *International Conference on Parallel Problem Solving From Nature, Lecture Notes in Computer Science 4193*, 681–690.

Martinetz, T. (1993). Competitive hebbian learning rule forms perfectly topology preserving maps. In S. Gielen et B. Kappen (Eds.), *Proceedings of the International Conference on Artificial Neural Networks (ICANN-93), Amsterdam*, Heidelberg, pp. 427–434. Springer.

Ultsch, A. (2005). Clustering with SOM : U*C. In *Proceedings of the Workshop on Self-Organizing Maps*, pp. 75–82.

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

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association 58*(301), 236–244.

Summary

One of the most crucial questions in many real-world cluster applications is determining a suitable number of clusters. Determining the optimum number of clusters is an ill posed problem for which there is no simple way of knowing that number without a priori knowledge. In this paper we propose a new two-level clustering algorithm based on self organizing map, called S2L-SOM, which allows an automatic determination of the number of clusters during learning. Estimating true numbers of clusters is related to the cluster stability which involved the validity of generated clusters. To measure this stability we use the sub-sampling method. The great advantage of our proposed algorithm, compared to the common partitional clustering methods, is that it is not restricted to convex clusters but can recognize arbitrarily shaped clusters. The validity of this algorithm is superior to standard two-level clustering methods such as SOM+K-means and SOM+Hierarchical-Agglomerative-Clustering. This is demonstrated on a set of critical clustering problems.