Arbres de recouvrement minimaux duaux et application à la segmentation non supervisée

Laurent Galluccio[1], Olivier Michel[2], Pierre Comon[1]

[1]Laboratoire I3S - UNS - CNRS - UMR 6070
2000, route des Lucioles - Les Algorithmes - BP.121 - 06903 Sophia Antipolis - Cedex, France

[2]Gipsa - Lab - CNRS -UMR 5216
961, rue de la Houille Blanche - Domaine universitaire - BP 46 F - 38402 Saint Martin d'Hères Cedex, France
gallucci@i3s.unice.fr, olivier.michel@gipsa-lab.inpg.fr, pcomon@i3s.unice.fr

Résumé − Cet article propose de nouvelles approches de segmentation non supervisée. Nous proposons d'exploiter les propriétés d'une nouvelle mesure de distance reposant sur la construction d'arbres de recouvrement minimaux duaux : le « **D**ual **R**ooted **Pri**m Tree »(DRooPi), pour construire une paire de classes candidates. Ces classes correspondent à une partition de l'ensemble des sommets connectés par un Droopi. La partition est obtenue par coupure simple du plus grand segment de l'arbre de recouvrement minimal partiel qu'est Droopi, dont les deux racines sont définies par la paire de sommets choisie au hasard. Des fonctions de consensus sont ensuite calculées sur l'ensemble des classes associées à chaque paire de sommets. Pour obtenir la partition finale, un algorithme de classification spectrale est appliqué avec comme mesure de distance les fonctions de consensus définies. Des résultats sont obtenus sur divers données synthétiques et réelles.

Abstract − This paper proposes new approaches to unsupervised clustering. We propose to exploit properties of a new distance measure based on the construction of dual rooted minimal spanning trees : the " **D**ual **R**ooted **Pri**m Tree " (DRooPi), to build a pair of candidate classes. These classes correspond to a partition of the set of vertices connected by a Droopi. The partition is obtained by a single cut of the largest edge of Droopi tree, which is a partial minimal spanning tree. The two roots are defined as the pair of vertices randomly taken. Some consensus functions are computed on the cluster ensemble. In order to obtain a final clustering, a spectral clustering algorithm is applied with as distance measure the consensus functions defined. Results are obtained on various synthetic and real data sets.

1 Introduction

Une tâche de segmentation non supervisée consiste à définir une partition d'un champ de données multidimensionnel en un ensemble de classes pour lesquelles aucune inofrmation a priori n'est disponible (nombre, forme, tailles respectives...). Nous ne discuterons pas dans ce papier du choix de la métrique utilisée pour définir une mesure de ressemblance entre deux « vecteurs » du champ de données analysé.

Parmi les multiples approches développées pour résoudre ce problème, nous citerons par exemple les méthodes de classification hiérarchique, les algorithmes de partitionnement ... [5] Les résultats obtenus par différentes méthodes de segmentation, voire par une seule méthode mais avec des paramètres d'initialisation différents, sont en général très divers. Par conséquent, des approches ont été développées afin de combiner un ensemble de partitions « candidates », visant à produire une segmentation finale unique. Certaines de ces approches utilisent des fonctions de consensus basées par exemple sur des matrices de co-association [3], des votes [2], des représentations en hypergraphe [10], des mélanges de modèles [11].

Dans cette communication, nous proposons d'utiliser le concept de combinaison de méthodes et une nouvelle méthode de segmentation non supervisée basée sur des arbres à recouvrement minimal (minimal spanning tree : MST) à racines duales. Une caractéristique attractive de la segmentation basée sur les MSTs est sa capacité à prendre en compte une information géométrique aussi bien locale que globale sur la distribution des données à partitionner. Dans [4], Griskschat et al. ont montré que les arbres de diffusion à racines duales pour la classification permettent de définir une matrice de similarité qui prend en compte une information de voisinage alors que la distance Euclidienne échoue à capturer cette information. En fait, un arbre à racines duales permet d'extraire des classes à l'intérieur de l'ensemble de données initiales sans définir une partition des données, car tous les sommets n'ont pas besoin d'être connectés. De plus, il présente une variabilité plutôt importante par rapport à l'ensemble d'initialisation des paramètres. Dans ce sens, il peut être considéré comme une méthode de partitionnement faible. Dans notre méthode, la partition finale est obtenue en appliquant un algorithme de classification spectrale [8] sur des matrices de co-association qui ont montré être des matrices d'affinité admissibles. Toutes ces approches seront abordées dans la Section 2. Dans la section 3, des résultats expérimentaux seront présentés sur des données synthétiques et réelles.

2 Accumulation de preuves

Soient X un ensemble de N points $\in \mathbb{R}^L$ et $\{C_{1i}, \ldots, C_{Ki}\} = P_i$ une partition de X en K classes, telle qu'elle est obtenue par A_i, où $i \in [1, \ldots M]$; A_i désigne ici un algorithme ou un jeu de paramètres d'initialisation d'un algorithme de segmentation particulier. Soit $\mathbf{P} = (P_1, \ldots, P_M)$ l'ensemble des segmentations. Soit $G = (V, E)$ un graphe non dirigé où $V = (v_1, \ldots, v_N)$ est l'ensemble de N sommets et $E = (e_1, \ldots, e_{N-1})$ correspond à l'ensemble de segments. Le poids d'un segment mesure la dissimilarité ou la séparation entre deux sommets. Dans cet article, seule la distance Euclidienne sera considérée bien que l'on peut souligner que d'autres métriques peuvent être utilisées [7].
Très souvent en pratique, la méthode utilisée pour construire $\mathbf{P}$ est l'algorithme K-means, peu coûteux et facilement implantable. L'inconvénient est que K-means tend à générer des classes convexes, ce qui dans le cas général peut être une limitation. Nous proposons ici de construire l'ensemble $\mathbf{P}$ en réalisant une coupure unique (« single cut ») sur le MST obtenu par la réunion de deux sous-MSTs (sMSTs), initialisé en deux sommets distincts et dont la construction est stoppée dès lors que les deux sMSTs possèdent un sommet commun.
Plusieurs algorithmes permettent de construire un MST [6]. Dans ce papier, la méthode proposée est basée sur les propriétés de construction de l'algorithme de Prim ($O(N \log N)$) [9]. Le MST est totalement connecté, acyclique (aucune boucle), unique : indépendant du sommet initial (si il n'existe pas d'égalité dans la matrice de distance). L'algorithme de Prim connecte au graphe partiellement connecté à l'itération i le sommet non encore connecté le plus proche du graphe (au sens d'une métrique arbitraire). Les sous-arbres obtenus après $N-1$ itérations de l'algorithme de Prim sont des MSTs définis sur l'ensemble des N sommets connectés. Le poids d'un MST, défini comme la somme des poids des segments connectant les sommets, est minimal.

2.1 Arbre de recouvrement dual

Dans [4], Griskschat et al. ont proposé une nouvelle mesure de distance entre deux points (ou sommets) d'un champ de données multidimensionnel, basée sur le temps exprimé en nombre d'itérations de l'algorithme de construction de Prim, nécessaire pour observer la « collision »des deux sMSTs. A chaque itération, les deux sMSTs sont incrémentés [1]. Les itérations sont réalisées jusqu'à ce que les deux arbres aient un sommet en commun ; le graphe final n'est pas donc forcément totalement connecté. Tous les points de X ne sont donc pas forcément inclus dans l'un ou l'autre des deux sMSTs quand les itérations sont stoppées.
Nous proposons ici une variante de cet algorithme, qui consiste à n'incrémenter à chaque itération qu'un seul des deux sMSTs suivant un critère de coût minimal. Pour une itération donnée, un seul sommet sera connecté à l'un des deux sous-arbres (et non les deux) qui devra être celui de coût minimal. On notera l le coût ou le poids du segment utilisé pour cette connexion. L'intérêt de cette approche réside dans la propriété suivante : permettre une meilleure prise en compte du voisinage de chaque racine dans la définition d'une nouvelle distance. Ce processus de construction s'arrête également quand les deux sous-arbres se rencontrent. N_{iter} désignera le nombre d'itérations effectuées jusqu'à ce que l'algorithme de construction soit stoppé.

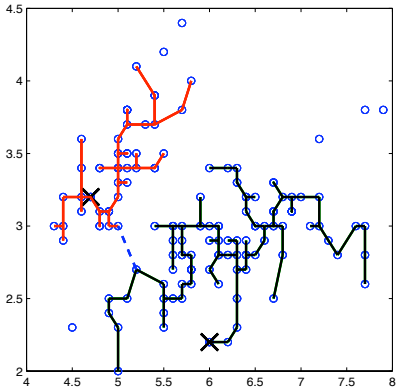

FIG. 1 – Arbres de recouvrement minimaux duaux construit sur un ensemble de données. Les symboles X indiquent les sommets racines. Le segment en pointillé correspond au dernier segment connecté.

L'arbre final obtenu en considérant l'ensemble des sommets et segments appartenant à l'un ou l'autre des sMSTs associés à chacune des deux racines sera désigné dans la suite par le nom de « **D**ual **Roo**ted **Pri**m (Droopi) Tree ». Le graphe Droopi enraciné en $x_i \in X$ et $x_j \in X$ sera noté $\mathrm{DR}(\mathrm{x_i}, \mathrm{x_j})$. Différentes mesures $d(x_i, x_j)$ de proximité peuvent être proposées à partir du $\mathrm{DR}(\mathrm{x_i}, \mathrm{x_j})$: $d_{iter} = N_{iter}$, $d_{leng} = \sum_{iter=1}^{N_{iter}} l_{iter}$, $d_{max} = \max_{iter \in [1, N_{iter}]} l_{iter}$.

Le graphe Droopi satisfait quelques propriétés intéressantes, l'une d'elle sera beaucoup utilisée dans le reste du papier : pour un couple de sommets $\{v_1, v_2\}$ servant comme racines des deux sous-arbres, le dernier segment construit, de poids noté l_{max}
- connecte les deux sous-arbres ensemble,
- est toujours le plus long (avec un poids maximal) parmi l'ensemble des segments des deux sous-arbres,
- définit une distance d_{max}

Par conséquent, chaque couple de sommets racines mène à la construction d'un graphe Droopi qui connecte un sous-ensemble de sommets, qui peut être facilement partitionné en deux classes en appliquant une coupure simple pour enlever le dernier segment. Pour chaque couple de noeuds racines considéré, les membres d'une classe reçoivent le même label et les labels sont enregistrés.

Pour chaque réalisation de l'algorithme de partitionnement de graphe avec différentes initialisations, $P_i = \{C_{1i}, C_{2i}\}$, $i \in [1, M]$ représente la partition de l'ensemble de sommets connectés par un couple de sommets $\{v_{1i}, v_{2i}\}$ (où i indique ce choix particulier de racines).

[1] L'incrémentation est ici à comprendre au sens où un nouveau sommet est connecté à l'arbre qui se construit.

Il est important de souligner que $C_{1i} \cup C_{2i} \neq V$ et par conséquent, l'algorithme produit deux classes labelisées et un ensemble de sommets non labelisés. Ceci mènera à quelques affinements dans la définition de mesure de co-association, comme décrite dans la sous-section suivante.

2.2 Mesures de co-association

Suivant les travaux de Fred et Jain [3], nous proposons de combiner les résultats de classification (l'ensemble des P_i) afin de calculer une matrice de similarité par une approche *d'accumulation de preuves*. L'idée directrice en est que des points appartenant à la même classe (inconnue) seront plus souvent affectés à une classe commune C_{ki} par les différentes méthodes de classification ou par les différentes réalisations considérées d'une segmentation partielle (tous les sommets de X ne sont pas pris en compte dans $DR(x_i, x_j)$). Par conséquent, prendre en compte la fréquence (mesure de co-association) avec laquelle les points sont labelisés dans une classe commune permet de mettre en évidence certains groupes dominants. Cette mesure a été souvent utilisée dans la littérature [10, 3]. Pour l'ensemble des M partitions réalisées,

$$co\_assoc(x_i, x_j) = \frac{n(x_i, x_j)}{M}, \qquad (1)$$

où $n(x_i, x_j)$ correspond au nombre de fois qu'une paire de points x_i et x_j a été classée dans le même groupe parmi les M partitions réalisées.
Tous les points ne sont pas labelisés par la réalisation de $\mathrm{DR}(\mathrm{x_i}, \mathrm{x_j})$ (Fig. 1) ; afin de prendre en compte cette information dans la mesure de similarité, nous proposons de définir la mesure de co-association définie *co_assoc*2,

$$co\_assoc2(x_i, x_j) = \frac{n(x_i, x_j)}{m(x_i, x_j)}, \qquad (2)$$

où $m(x_i, x_j)$ est le nombre de partitions dans $\mathbf{P}$ où x_i ET x_j ont été labelisés ($0 \leq m(x_i, x_j) \leq M$).

Bien que ces mesures de co-association reposent sur des heuristiques facilement compréhensibles, elles peuvent être sensiblement modifiées pour mieux correspondre aux caractéristiques du problème considéré. Afin d'aborder le problème de sommets non labelisés existant dans l'approche Droopi, on peut choisir de considérer ces sommets comme étant des éléments d'une classe de rejet. Si $nn(x_i, x_j)$ correspond au nombre de fois où aussi bien x_i que x_j appartiennent à cette classe de rejet, une modification naturelle de la définition (1) pourrait être par conséquent $co\_assoc_{mod} = \frac{n+nn}{M}$. Comme cette mesure accorde une importance identique aux classes détectées et à la classe de rejet, nous proposons ici de considérer :

$$co\_assoc3(x_i, x_j) = \frac{n(x_i, x_j)}{m(x_i, x_j)} + \frac{nn(x_i, x_j)}{M}, \qquad (3)$$

Comme $m \leq M$, cela revient à appliquer un plus fort coefficient aux classes détectées et donc un plus faible pour la classe de rejet. Bien que nous ne pouvons pas établir ceci à partir de considérations théoriques, nous observons expérimentalement qu'utiliser cette dernière mesure tend à classer les observations aberrantes ou « outliers »ensemble.

Dans cette étude, nous proposons d'exploiter ces mesures dans le cas particulier d'utilisation d'algorithmes de classification spectrale. Dans ce contexte, la matrice d'affinités A considérée est définie par

$$A(i,j) =\propto \beta \exp\left\{+\frac{co\_assoc(x_i, x_j)}{\sigma}\right\}$$

$(i,j) \in [1,N]^2$, où β et σ sont des constantes à déterminer.

Les résultats présentés sont obtenus par utilisation de l'algorithme de classification spectrale de Ng et al. [8], qui exploite les propriétes des valeurs propres normalisées du Laplacien normalisé d'un graphe totalement connecté décrit par la matrice $A(i,j)$[2]. Cette méthode sera nommée par la suite « Evidence Accumulation Clustering by Droopi tree Cut (EAC-DC) ».

3 Etude expérimentale

Afin de valider et d'évaluer les performances des approches proposées, des tests sur des données synthétiques et des données réelles extraites de la banque de données de « UCI Machine Learning Repository »[1] ont été réalisés. Dans toutes nos expériences, pour chaque algorithme testé, un ensemble de $M = 100$ paramètres d'initialisation tirés aléatoirement a été considéré.

Nous comparons nos approches avec divers algorithmes basés sur des fonctions de consensus sur l'ensemble des classes :

- classification par accumulation de preuves basée sur une mesure de co-association avec un algorithme hiérarchique « average link »(EAC-AL) [3],
- représentation en graphe et en hypergraphe comme décrite dans [10] (CSPA, HGPA et MCLA),
- vote cumulatif, voir [2] pour plus de détails et de définitions (URCV, RCV, ACV),
- partition médiane (QMI) [11].

Ces méthodes utilisent l'algorithme K-means pour former leur ensemble de classes. Il est à noter que, bien que nous considérons le nombre de classes a priori connu, les méthodes citées sont non supervisées.

3.1 Résultats sur des données synthétiques

Le tableau 1 reporte les résultats obtenus avec les différentes fonctions de consensus introduites précédemment (Fig. 2 sont représentés les résultats de nos approches sur ces données). La mesure de performance est définie par le rapport entre le nombre d'objets bien classés sur le nombre d'objets total (la référence est connue). Nos approches arrivent à détecter correctement les classes, malgré leurs formes, tandis que les autres méthodes échouent à trouver ces classes non-convexes.

3.2 Résultats sur des données réelles

Le premier ensemble de données considéré est « Iris »(150 points dans un espace caractéristique à 4 dimensions) [1]. Il contient trois classes de variétés de fleurs

[2]Il faut noter ici qu'en toute généralité, dans le cas où M peut être faible, toutes les distances $d(x_i, x_j)$ peuvent n'être pas définies. Dans ce cas, on choisira arbitrairement de fixer $A(i,j) = 0$.

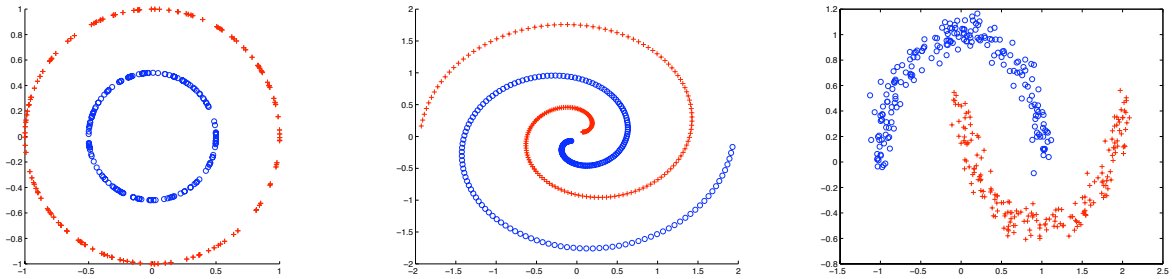

FIG. 2 – Résultats obtenus avec nos approches proposées EAC-DC sur différents ensembles de données synthétiques quelle que soit la mesure de co-association utilisée. Les labels des classes sont indiqués par des symboles.

TAB. 1 – Résultats obtenus sur les données synthétiques

Méthodes	Demi-lunes	Deux spirales	Deux cercles
EAC-DC (1, 2, 3)	**1**	**1**	**1**
EAC-AL	0.7675	0.5550	0.5200
QMI	0.7550	0.5050	0.5150
URCV	0.7600	0.5000	0.5200
RCV	0.7650	0.5300	0.5200
ACV	0.5000	0.5000	0.5000
CSPA	0.7300	0.5025	0.5550
HGPA	0.5000	0.5000	0.5000
MCLA	0.7275	0.5050	0.5400

avec une classe bien séparée des autres et deux classes proches. Le deuxième ensemble de données « breast cancer Wisconsin »a été analysé. Il est constitué de 683 vecteurs de 9 dimensions caractéristiques séparés en deux classes. Le troisième ensemble de données testé est « Wine », composé de 178 vecteurs de 13 dimensions incluant trois classes de cépages différentes. Les résultats obtenus par les différents algorithmes sont présentés dans le tableau 2.

On constate que la méthode développée donne de meilleures résultats sur les données Iris et breast cancer. En ce qui concerne les données Wine, les performances de nos approches sont plutôt convaincantes.

TAB. 2 – Résultats obtenus sur les données réelles

Méthode	Iris	Breast Cancer Wisconsin	Wine
EAC-DC 1	0.9067	**0.9678**	0.7022
EAC-DC 2	**0.9267**	0.9605	0.7022
EAC-DC 3	0.9200	0.9606	0.7022
EAC-AL	0.8733	0.9429	0.6910
CSPA	0.9200	0.8448	0.7135
HGPA	0.9200	0.6501	0.7022
MCLA	0.8933	0.9575	**0.7247**
URCV	0.8800	0.9590	0.6742
RCV	0.9000	0.9663	0.6854
ACV	0.9067	0.9649	0.6966
QMI	0.8800	0.9356	0.6011

4 Conclusion

Dans ce papier, nous avons présenté une nouvelle méthode de segmentation non supervisée, basée sur le concept d'accumulation de preuves et sur la combinaison de résultats de multiples segmentations. Nous avons proposé d'exploiter les propriétés d'une nouvelle construction de graphe nommée Droopi pour établir l'ensemble de classes. Cette construction permet de mieux capturer la géométrie intrinsèque globale et locale des données. De multiples réalisations de la segmentation par une coupure sur le MST sont effectués à partir d'un ensemble de paramètres d'initialisation choisis aléatoirement et permettent de construire un ensemble de classes grâce à de nouvelles mesures de co-association proposées. L'extension de la méthode proposée avec des arbres à K racines est en cours d'étude. Les performances de cette nouvelle méthode ont été évaluées sur un ensemble de données synthétiques et réelles, soulignant des caractéristiques des algorithmes proposés très intéressantes et prometteuses.

Références

[1] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.

[2] H. G. Ayad and M. S. Kamel. Cumulative voting consensus method for partitions with a variable number of clusters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(1) :160–173, January 2008.

[3] A. L. N. Fred and A. K. Jain. Combining multiple clusterings using evidence accumulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6) :835–850, June 2005.

[4] S. Griskschat, J. A. Costa, A. O. Hero, and O. Michel. Dual rooted-diffusions for clustering and classification on manifolds. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Toulouse, France, 2006.

[5] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering : a review. *ACM Computing Surveys*, 31(3) :264–323, 1999.

[6] D. J. Marchette. *Random graphs for statistical pattern recognition*. Wiley, 2004.

[7] O. J. J. Michel, P. Bendjoya, and P. RojoGuer. Unsupervised clustering with mst : Application to asteroïd data. In *Physics in Signal and Images Processing*, 2005.

[8] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering : Analysis and an algorithm. In *Advances on Neural Information Processing Systems*, volume 14, 2001.

[9] R. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36 :1389–1401, 1957.

[10] A. Strehl and J. Ghosh. Cluster ensemble - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3 :583–617, 2002.

[11] A. Topchy, A. K. Jain, and W. Punch. Clustering ensembles : Models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12) :1866–1881, December 2005.