Clustering Spectral semi-supervisé avec propagation des contraintes par paires

N Voiron, A Benoit, A Filip, P Lambert, B Ionescu

Clustering Spectral semi-supervisé avec propagation des contraintes par paires

N. Voiron[1], A. Benoit[1], A. Filip[2], P. Lambert[1] et B. Ionescu[2]

[1]*LISTIC, Universite de Savoie, 74940, Annecy le Vieux, France*
{nicolas.voiron, alexandre.benoit, patrick.lambert}@univ-savoie.fr
[2]*LAPI, University Politehnica of Bucharest, 061071, Bucharest, Romania*
{afilip, bionescu}@alpha.imag.pub.ro

RÉSUMÉ. Dans un monde guidé par les données, la classification est un outil essentiel pour aider les utilisateurs à appréhender la structure de ces données. Les techniques d'apprentissage supervisé permettent d'obtenir de très bonnes performances lorsque l'on dispose d'une base annotée, mais un risque de sur-apprentissage existe toujours. Il existe de nombreuses techniques de classification non supervisée qui cherchent à construire la structure des données sans disposer de données d'entraînement. Mais dans des contextes difficiles les résultats sont moins bons que ceux de l'apprentissage supervisé. Pour améliorer les performances, un bon compromis est d'apporter de la connaissance seulement sur les éléments (classes et objets) ambigües. Dans ce contexte, cet article s'intéresse au Clustering Spectral et à l'ajout de contrainte par paires. Nous introduisons une nouvelle généralisation de la propagation des contraintes qui maximise la qualité de partitionnement tout en réduisant les coûts d'annotation.

ABSTRACT. In our data driven world, clustering is of major importance to help end-users and decision makers understanding information structures. Supervised learning techniques rely on ground truth to perform the classification and are usually subject to overtraining issues. On the other hand, unsupervised clustering techniques study the structure of the data without disposing of any training data. Given the difficulty of the task, unsupervised learning tends to provide inferior results to supervised learning. To boost their performance, a compromise is to use learning only for some of the ambiguous classes or objects. In this context, this paper studies the impact of pairwise constraints to unsupervised Spectral Clustering. We introduce a new generalization of constraint propagation which maximizes partitioning quality while reducing annotation costs.

MOTS-CLÉS : Clustering Spectral, apprentissage semi-supervisé, classification vidéo.

KEYWORDS: Graph Cut, Spectral Clustering, semisupervised learning, video clustering.

1. Introduction

Grâce à internet, la quantité de données multimédia disponible a augmenté de façon considérable. Afin de pouvoir fournir aux utilisateurs des outils de recherche et de navigation sur ces énormes quantité de données, des outils d'indexation et de classement automatique efficaces sont nécessaires. Dans la recherche d'informations pertinentes, les outils de sélection de préférences et de suggestions aux utilisateurs permettent d'améliorer les résultats en les adaptant aux intérêts personnels de chacun. Dans le cas des bases d'images et de vidéos, ces outils reposent sur des mesures de similarité. De nombreuses approches ont été explorées pour mesurer ces similarités (Datta *et al.*, 2008) (Voiron *et al.*, 2012). Dans ce papier, nous poursuivons ces explorations dans le contexte des techniques de classification.

Dans ce domaine, de nombreuses techniques de classification existent (Witten *et al.*, 2011). On peut distinguer des méthodes classiques comme les kmeans qui fonctionnent avec des clusters convexes et des méthodes plus élaborées comme les réseaux de neurones artificiels qui sont capables d'identifier des clusters plus complexes. Les techniques de coupe de graphe du Clustering Spectral (von Luxburg, 2007), de la famille des méthodes de Manifold Learning, forment une catégorie particulière de méthodes. Elles sont particulièrement reconnues pour leur capacité d'identification de clusters non convexes. Cependant, le Clustering Spectral standard est non supervisé et ne peut pas intégrer de connaissance externe. Des travaux récents (Rangapuram et Hein, 2012), (Xiong *et al.*, 2014) ont montré l'intérêt d'introduire cette connaissance externe à travers des contraintes par paires pour guider le clustering. De telles approches sont sensiblement similaires à l'apprentissage supervisé classique et aux méthodes "Support Vector Machines" (SVM) mais l'utilisation des connaissances reste différente.

Dans un contexte complexe comme celui de la classification des données vidéos, l'introduction d'une supervision peut résoudre des ambigüités et fortement améliorer les résultats du clustering. Cette supervision est couramment introduite par l'ajout de contraintes communément nommées "Must Link" et "Cannot Link". Ces contraintes indiquent simplement si deux objets sont de la même classe ou non. Elles sont considérées comme étant les plus générales car elles peuvent être extraites d'autres connaissances, comme par exemple d'un étiquetage des objets ou de recommandations d'experts. De plus, ces contraintes correspondent à des annotations par similarité qui sont beaucoup plus faciles à réaliser que des annotations absolues par classe car il est seulement question de savoir si deux objets appartiennent ou non à la même classe. Dans cette démarche, il est aussi important d'optimiser les contraintes pour maximiser la qualité du clustering tout en minimisant la sollicitation des experts. La stratégie la plus couramment rencontrée dans des papiers comme celui de Vu et Labroche (Vu *et al.*, 2012) consiste en une propagation automatique des contraintes. Cependant, dans la littérature, cette propagation n'est souvent qu'à peine évoquée ou utilisée de façon incomplète.

Dans ce papier, nous faisons un tour d'horizon exhaustif de la propagation automatique des contraintes par paires et nous apportons ensuite une généralisation originale de la propagation du bi-partitionnement vers le multi-partitionnement. Toutes nos expérimentations sont menées sur deux familles d'ensembles de données. La première est un jeu de données synthétiques avec différents niveaux graduels de séparation des classes allant de très séparé à totalement mélangé. La deuxième famille est issue d'un ensemble de données disponibles pour la classification par genre de vidéos (Schmiedeke *et al.*, 2013).

Le papier est organisé comme suit. Dans la section 2, nous présentons un état de l'art du clustering spectal semi-supervisé et les techniques de propagation automatique des contraintes par paires. La section 3 présente notre contribution sur la généralisation des propagations des contraintes par paires du bi-partitionnement vers le multi-partitionnement. La section 4 examine quels sont les effets de la propagation automatique des contraintes sur les techniques de Clustering Spectral semi-supervisé. Dans la section 5, nous validons l'intérêt de cette propagation et nous discutons son efficacité. La section 6 conclut le papier et évoque les travaux futurs.

2. Clustering Spectral Semi-supervisé

Les performances d'un classifieur sont fortement dépendantes des propriétés et de la structure des données. Lorsque l'on a des clusters convexes, les méthodes classiques telles que les kmeans donnent de bons résultats. Cependant, ces méthodes ne sont pas capables d'identifier des manifolds caractérisés par une connectivité complexe des données. Dans de telles situations, d'autres algorithmes sont opérationnels comme Isomap, le positionnement multidimensionnel (MDS) et le Clustering Spectral. Ces méthodes tentent généralement d'identifier un espace de dimension inférieure qui représente et sépare bien les données. Nous explorons dans ce papier le Clustering Spectral qui est capable de fonctionner efficacement sans hypothèse de forme sur les clusters. Cette méthode peut traiter de grands volumes de données en s'appuyant sur des graphes de similarité sparses. Ce cas de figure est une caractéristique intéressante pour nos travaux portant sur de grandes bases de données vidéos.

Dans sa forme classique, le Clustering Spectral est une méthode totalement automatique qui n'utilise que les données fournies en entrée. Or dans des situations complexes, comme l'analyse ou la compréhension de vidéos, le fossé sémantique entre les caractéristiques de bas niveau extraites et la classification haut niveau attendue est très grand. L'introduction de connaissance pour guider le Clustering Spectral présente donc son intérêt. En suivant cette idée, nous nous intéressons au clustering semi-supervisé par l'ajout d'un faible nombre de contraintes par paires. En choisissant bien certaines contraintes dans l'ensemble des données, on peut obtenir de bons résultats de clustering (Davidson *et al.*, 2006). Ceci diffère de l'apprentissage supervisé qui intègre la connaissance lors d'une phase d'entraînement sur un jeu de données défini, avec un risque évident de surentraînement, et un coût important pour acquérir cette connaissance.

2.1. *Clustering Spectral*

Soit $X = (x_i)_{i \in [\![1,n]\!]}$ l'ensemble des n données que l'on veut partitionner en k classes. Les algorithmes du Clustering Spectral se décomposent en 3 étapes : (i) un graphe de similarité est d'abord construit entre les objets ; (ii) une projection est effectuée sur un espace spectral où les clusters sont plus facilement identifiables ; (iii) pour finir, un clustering convexe standard est effectué sur les données dans cet espace spectral. Ces trois étapes sont présentées dans la suite.

2.1.1. *Étape 1 : la construction du graphe de similarité*

La construction du graphe de similarité peut être séparée en 2 temps : *la construction des liens* suivie de *la pondération des liens*. En général, *la construction des liens* est faite suivant l'une des trois approches suivantes :

1) le graphe des ε-voisinages qui relie les objets distants de moins de ε ;

2) le graphe des k-plus proches voisins qui peut être rendu non orienté en utilisant une procédure de k-plus proches voisins symétrique ou mutuelle ;

3) le graphe totalement connecté qui n'est pas sparse.

En ce qui concerne *la pondération des liens*, la procédure suivante est employée. Une pondération $s(x_i, x_j)$ est assignée à chaque lien construit entre les objets x_i et x_j. Cette pondération est généralement normalisée dans l'intervalle $[0, 1]$. Les pondérations peuvent être :

1) une similarité binaire : $s(x_i, x_j) = 1$ s'il existe un lien entre x_i et x_j et 0 sinon. Dans ce cas, on parle aussi de graphe non pondéré ;

2) une similarité gaussienne : $s(x_i, x_j) = exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma^2}\right)$ avec σ le paramètre contrôlant la largeur des voisinages ;

3) toute autre pondération qui définit une mesure de similarité sur l'ensemble des données.

La similarité s définie ici conduit à la définition d'une matrice d'adjacence W, avec $w_{ij} = s(x_i, x_j)$. W peut être sparse si la construction du graphe limite le nombre de liens. Ceci permet des gains de temps de calcul conséquents avec des algorithmes adaptés.

2.1.2. *Étape 2 : la construction de l'espace spectral*

Définissons maintenant la matrice diagonale des degrés D qui pour chaque nœud contient la somme des pondérations des arêtes dont le nœud est une des extrémités. Ceci nous permet de définir la matrice laplacienne L avec 3 variantes couramment utilisées :

1) le laplacien non normalisé : $L = D - W$

2) le laplacien normalisé "marche aléatoire" : $L_{rw} = D^{-1}L = I - D^{-1}W$

3) le laplacien normalisé symétrique : $L_{sym} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$

Ensuite, les k premiers vecteurs propres associés aux k plus petites valeurs propres du Laplacien sont calculés et disposés dans la matrice $V \in \mathbb{R}^{n\times k}$. V correspond à une projection des similitudes dans un espace propre de plus petite dimension où les clusters sont supposés être plus faciles à identifier. D'un point de vue coupe de graphe, les k dimensions de V fournissent les k premières coupes binaires de plus faible connectivité.

2.1.3. *Étape 3 : le partitionnement des données dans l'espace spectral*

Après les deux étapes précédentes, les éventuels manifolds sont censés avoir été dépliés dans un espace propre. Une méthode de clustering convexe classique est maintenant capable d'identifier ces clusters. Les méthodes de l'état de l'art utilisent généralement un simple kmeans (von Luxburg, 2007). Cependant, d'autres méthodes de clustering convexe peuvent également être utilisées, comme par exemple les mixtures de gaussiennes (Xiong *et al.*, 2014).

2.2. *Semi-supervision par paires dans le Clustering Spectral*

Comme indiqué dans (Xiong *et al.*, 2014), le clustering peut introduire des ambigüités sémantiques. Ceci est courant avec la catégorisation d'images car les données en entrée sont généralement des descripteurs de bas niveau, alors que le clustering désiré est fortement sémantique. Dans ce contexte, une solution consiste en l'ajout de contraintes par paires fournies par des connaissances externes. Une approche consiste à introduire entre les objets des contraintes "Must Link" (ML) -les 2 objets sont de la même classe- et "Cannot Link" (CL) -les 2 objets ne sont pas de la même classe-. Cependant, le choix et le nombre de ces contraintes doivent être optimisés pour respecter la qualité du clustering, tout en gardant un faible coût de calcul.

2.2.1. *Prise en compte des contraintes*

Les contraintes peuvent être intégrées à chacune des 2 premières étapes du Clustering Spectral décrites au paragraphe 2.1 : (i) lors de la *construction du graphe de similarité* comme par exemple dans (Xiong *et al.*, 2014), inspiré du Spectral Learning (SL) (Kamvar *et al.*, 2003), où les auteurs proposent d'identifier les objets les plus ambigües, de superviser leurs liens et d'intégrer dans la matrice d'adjacence, W, des 1 pour les ML et des 0 pour les CL. Cependant, il n'y a aucune garantie que les contraintes soient respectées ; ou (ii) lors de la *construction de l'espace spectral*. De nombreuses méthodes ont été explorées comme par exemple "Flexible Constrained Spectral Clustering" (CSP) (Wang et Davidson, 2010) qui intègre les contraintes lors de cette étape et qui se termine par un kmeans ; "Spectral Clustering with Linear Constraints" (SCLC) (Xu *et al.*, 2009) n'utilise pas de kmeans et permet seulement un clustering binaire ; "Constrained Clustering via Spectral Regularization" (CCSR) (Li *et al.*, 2009), prend en compte les contraintes dans une étape intermédiaire qui modifie

l'espace spectral. Une approche différente est proposée dans "Constrained 1-Spectral Clustering" (COSC) (Rangapuram et Hein, 2012). Les contraintes sont intégrées au calcul grâce à la résolution d'un problème spectral d'optimisation convexe. Le résultat est un bi partitionnement sans appel à une méthode de partitionnement de type kmeans. Cette méthode est étendue au cas du multi partitionnement grâce à des appels récursifs. Il est montré que COSC respecte très bien les contraintes en offrant un taux d'erreur plus faible que les autres méthodes décrites précédemment.

À notre connaissance de la littérature, la prise en compte des contraintes est toujours effectuée comme décrite au paragraphe précédent, c'est à dire lors des deux premières étapes du Clustering Spectral. Cependant, nous pourrions envisager une introduction des contraintes plus tardive. Elle interviendrait seulement lors du partitionnement des données dans l'espace spectral. Nous pourrions, par exemple, utiliser l'algorithme MPCK-Means (Bilenko *et al.*, 2004) qui modifie la méthode des kmeans pour lui permettre d'intégrer de la supervision à l'aide de contraintes ML et CL.

2.2.2. *Sélection des contraintes*

Pour comparer plusieurs méthodes, les contraintes sont souvent sélectionnées aléatoirement parmi toutes les paires d'objets possibles. Cependant, Davidson et al. (Davidson *et al.*, 2006) a démontré que dans certains cas, des contraintes mal choisies, dégradent la qualité du clustering et qu'à l'inverse, des contraintes bien choisies peuvent améliorer le résultat de manière significative. (Davidson *et al.*, 2006) introduit deux mesures pour quantifier l'utilité des contraintes : l'informativité qui mesure la quantité d'information apportée et la cohérence qui compare les contraintes aux autres.

Dans (Vu *et al.*, 2012), les auteurs proposent un modèle de sélection active des contraintes. La méthode permet d'identifier et classer des arêtes critiques dans le graphe des plus proches voisins. Ce sont des arêtes passerelles entre des groupes de points fortement connectés qui sont intéressantes à soumettre à l'avis d'un expert pour savoir si l'on doit couper ou non le graphe à cet endroit. Une approche similaire (Xiong *et al.*, 2014) propose d'identifier les objets les plus ambigües et de sélectionner les liens issus de ces objets. Une des originalités de ce travail réside dans le fait que cette sélection d'objet cherche à se focaliser sur les liens qui ont les plus grandes chances d'apporter des changements significatifs dans les résultats du Clustering Spectral.

Ces méthodes utilisent en général un processus itératif avec une sélection à posteriori des contraintes. À chaque itération, les contraintes sont sélectionnées en fonction du clustering obtenu à l'itération précédente. Ces méthodes ont donc un coût de calcul élevé, mais elles s'avèrent efficace d'un point de vue de la qualité de la partition obtenue.

2.2.3. *Propagation automatique des contraintes*

Après un ajout de contraintes, des ambigüités voisines peuvent être résolues automatiquement sans appel supplémentaire à l'annotation experte. Ce processus est

illustré dans les figures 1 et 2. Il s'agit de la transitivité de la relation ML et de la combinaison des relations ML et CL. Cette propagation est très intéressante pour réduire le coût de données expertes et atteindre une meilleure qualité de partitionnement avec un coût de calcul inférieur.

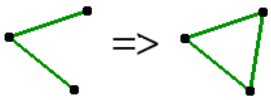

Figure 1. $ML + ML \Rightarrow ML$ *(Règle 1)*

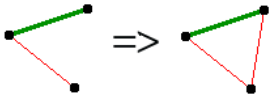

Figure 2. $ML + CL \Rightarrow CL$ *(Règle 2)*

3. Généralisation proposée

Nous proposons une nouvelle façon de propager les contraintes en exploitant la combinaison de deux contraintes CL. Dans le cas général du multi-partitionnement, cette combinaison est indéterminée. Cette configuration est illustrée dans la figure 3 ($CL + CL \Rightarrow ?$). Cependant, tel que présenté dans (Mallapragada *et al.*, 2008), le cas du bi-partitionnement résout cette indétermination avec $CL + CL \Rightarrow ML$ (voir la figure 4). Effectivement dans le cas d'un partitionnement en uniquement 2 classes C_1 et C_2, si un objet X appartient à la première classe C_1 et si X n'est pas dans la même classe que 2 autres objets Y et Z. Alors forcément Y et Z appartiennent à la deuxième et même classe C_2.

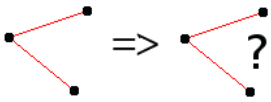

Figure 3. *en multi-partitionnement :* $CL + CL \Rightarrow ?$

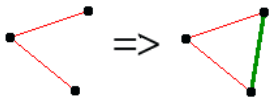

Figure 4. *en bi-partitionnement :* $CL + CL \Rightarrow ML$

Ce cas n'est pas anecdotique. Dans la littérature, il existe beaucoup de bi-partitionnement. Par exemple, dans (Rangapuram et Hein, 2012), la méthode COSC est évaluée dans 3 des 5 cas par des bi-partitionnements. Cependant, dans ces évaluations, aucune propagation n'est considérée.

À notre connaissance, dans le cas d'un partitionnement en 3 classes ou plus, la configuration $CL + CL$ est toujours mentionnée comme étant indéterminée. Cependant, nous pouvons quand même déduire quelque chose d'une configuration ne comportant que des CL. Comme présenté dans la figure 5, dans un tétraèdre en tri-partitionnement, si nous avons 5 arêtes CL alors la sixième et dernière arête est forcément un ML. Effectivement, en prenant 3 classes C_1, C_2 et C_3 et la configuration de la figure 5 comme W et X sont liés par un CL alors forcément W et X appartiennent à 2 classes différentes C_1 et C_2. Comme Y et Z sont eux mêmes reliés à W et X par

des CL alors ils ne font pas partie des classes C_1 et C_2 et forcément ils appartiennent à la même classe restante C_3. Et donc ils sont reliés par un ML.

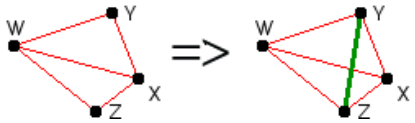

Figure 5. *En tri-partitionnement, dans le tétaèdre :* $5 \times CL \Rightarrow ML$.

Le cas du tétraèdre en tri-partitionnement se généralise pour tout entier n au n-simplexe dans le cas d'un n-partitionnement. Dans un n-simplexe, si l'on a $\left(\frac{n(n-1)}{2} - 1\right)$ arêtes CL alors la dernière et $\frac{n(n-1)}{2}$ème arête est forcément un ML. La démonstration est itérative de manière analogue au cas du tétraèdre en tri-partitionnement. La figure 6 présente le cas du n-simplexe pour n allant de 4 à 7.

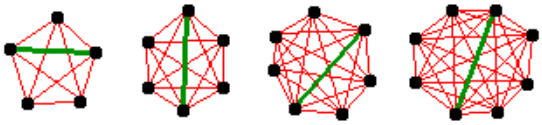

Figure 6. *En n-partitionnement, dans le n-simplexe :* $\left(\frac{n(n-1)}{2} - 1\right) \times CL \Rightarrow ML$.

4. Effet de la propagation automatique sur les méthodes semi-supervisées

La propagation automatique des contraintes agit à 2 niveaux dans les processus de supervision : lors de la sélection des contraintes, elle évite de retenir des contraintes qu'il est inutile de soumettre à l'expert car on peut les obtenir automatiquement, et lors de la prise en compte des contraintes, elle augmente le nombre de contraintes à injecter dans la méthode de classification semi-supervisée. Il est acquis que la première action améliore tous les processus de supervision quels qu'ils soient. Par contre, selon les méthodes et leur prise en compte des contraintes, la deuxième action peut avoir plus ou moins d'effets. Une méthode qui respecte les contraintes sans aucune violation est forcément insensible à cette deuxième action. La figure 7 illustre ce phénomène. A gauche, si le partitionnement respecte les 2 contraintes ML (arêtes vertes continues), alors les 3 sommets sont placés dans la même classe. Et donc forcément, la contrainte ML déduite (arête verte en pointillées) est respectée. Au centre, si le partitionnement respecte les 2 contraintes ML et CL (arêtes verte et rouge continues), alors les 2 premiers sommets sont placés dans une classe différente de celle du troisième sommet. Et forcément, la contrainte CL déduite (arête rouge en pointillées) est respectée. A droite, pour la troisième règle de propagation en bi-partitionnement, si les 2 contraintes CL

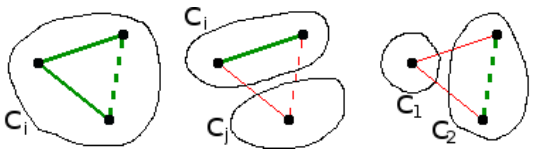

Figure 7. *Un partitionnement qui respecte les contraintes connues (arêtes continues) respecte forcément les contraintes déduites (en pointillés).*

sont respectées, alors le premier sommet appartient à la première classe, tandis que les 2 autres appartiennent à la deuxième classe. Et donc la contrainte ML déduite est respectée. Le même phénomène de respect des contraintes déduites existe avec la généralisation de la troisième règle au cas du n-partitionnement.

L'"Active Clustering" évoqué au paragraphe 2.2.1 n'est pas une méthode qui cible le respect des contraintes. Elles sont injectées dans le graphe de similitude afin de contraindre un peu la coupe effectuée par le Clustering Spectral. Beaucoup de contraintes ne sont pas respectées. En augmentant le nombre d'arêtes concernées par ce pré-conditionnement, la propagation automatique des contraintes améliore forcément la qualité de cette méthode.

Quant à la méthode COSC (Rangapuram et Hein, 2012), elle cible le respect des contraintes. Dans le cas des bi-partitionnements et la plupart des autres cas, cette méthode maximise le respect des contraintes. Avec COSC, le principal effet obtenu par la propagation des contraintes est la non sollicitation inutile de l'expert. Cependant, quelle que soit la méthode, celle-ci doit faire face au facteur humain et au fait qu'un expert peut apporter des contraintes ambigües. La propagation de contraintes majoritairement cohérentes peut permettre d'estomper, dans une certaine mesure, ces contraintes ambigües. De plus, la supervision est nécessaire pour aider à franchir le fossé sémantique données/attentes.

5. Résultats expérimentaux

Les validations expérimentales sont effectuées sur deux types d'ensembles de données : des données synthétiques[1] avec différents niveaux graduels de séparation des classes allant de très séparé à totalement mélangé et des données réelles Blip10000 (Schmiedeke *et al.*, 2013) issues de la classification par genre de vidéos. Les deux ensembles de données sont utilisés dans des configurations de partitionnement bi, tri comme multi-classes. Nous avons expérimenté les deux méthodes de Clustering Spectral mentionnées à la section 4 : la méthode Active Clustering (AC) (Xiong *et al.*, 2014), qui ajoute les contraintes à la matrice du graphe d'adjacence et qui ne garantit pas le respect des contraintes ; Constraint One Spectral Clustering (COSC) (Rangapuram et Hein, 2012), qui intègre les contraintes dans la phase de construction de l'espace propre et cible le respect des contraintes.

Pour évaluer les performances, nous utilisons l'indice de Rand normalisé (Hubert et Arabie, 1985) qui consiste en une normalisation du ratio du nombre de paires classées de la même façon dans les deux partitions sur le nombre de paires totales. Ses principaux avantages sont de prendre ses valeurs dans l'intervalle $[-1, 1]$ où la valeur 1 signifie que les deux partitions sont identiques et où surtout la valeur 0 signifie que les deux partitions sont indépendantes. Nous calculons l'indice de Rand entre le partitionnement de la vérité terrain et le partitionnement obtenu.

Nous avons adapté la procédure de validation suivante (Xiong *et al.*, 2014) : la méthode de clustering est appliquée une première fois sans contrainte. Puis à chaque itération, de nouvelles contraintes sont intégrées au processus, sont propagées automatiquement et la méthode de clustering est appliquée de nouveau. Pour être indépendant et ne pas privilégier un algorithme de clustering sur un autre, les contraintes sont choisies aléatoirement parmi toutes les paires possibles. Les contraintes sont propagées de manière récursive pour garantir que toutes les propagations possibles ont été exécutées à chaque itération. La performance des résultats est ensuite comparée à la vérité terrain grâce à l'indice Rand normalisé. Dans nos expérimentations, les contraintes sont extraites automatiquement de la vérité terrain qui consiste en un étiquetage de tous les objets. Dans le cadre d'une utilisation réelle, les contraintes proviendraient directement de l'appel à un expert qui devrait dire si 2 objets sont liés ou non.

5.1. *Le cas bi-classes synthétique*

Dans ces expérimentations, aucune normalisation n'est appliquée. Le graphe des similarités est construit à l'aide d'un 5 plus proches voisins symétrique. La pondération gaussienne est utilisée.

Nous avons généré 5 datasets synthétiques de 100 points répartis en 2 classes (voir la partie droite de la figure 8) avec les propriétés suivantes :

1) *1er dataset* : a ses points placés aléatoirement ;

2) *2ème dataset* : est partiellement mélangé avec deux classes circulaires qui se chevauchent partiellement ;

3) *3ème dataset* : a deux classes disjointes mais contigües ;

4) *4ème dataset* : similaire au troisième mais avec une zone de séparation entre les 2 classes ;

5) *5ème dataset* : a deux classes fortement séparées.

Les 5 graphiques de la figure 8 correspondent aux résultats sur les 5 datasets décrits précédemment. Ils présentent les évolutions de la qualité en fonction du nombre de paires sélectionnées aléatoirement. Les courbes en noir correspondent à une sélection/supervision de paires aléatoires, sans aucune propagation. En bleu, le processus est complété par une propagation automatique suivant les 2 premières règles. En rouge, la troisième règle de propagation est ajoutée. Chaque courbe correspond aux valeurs moyennes de 20 exécutions différentes. Les courbes en pointillés correspondent

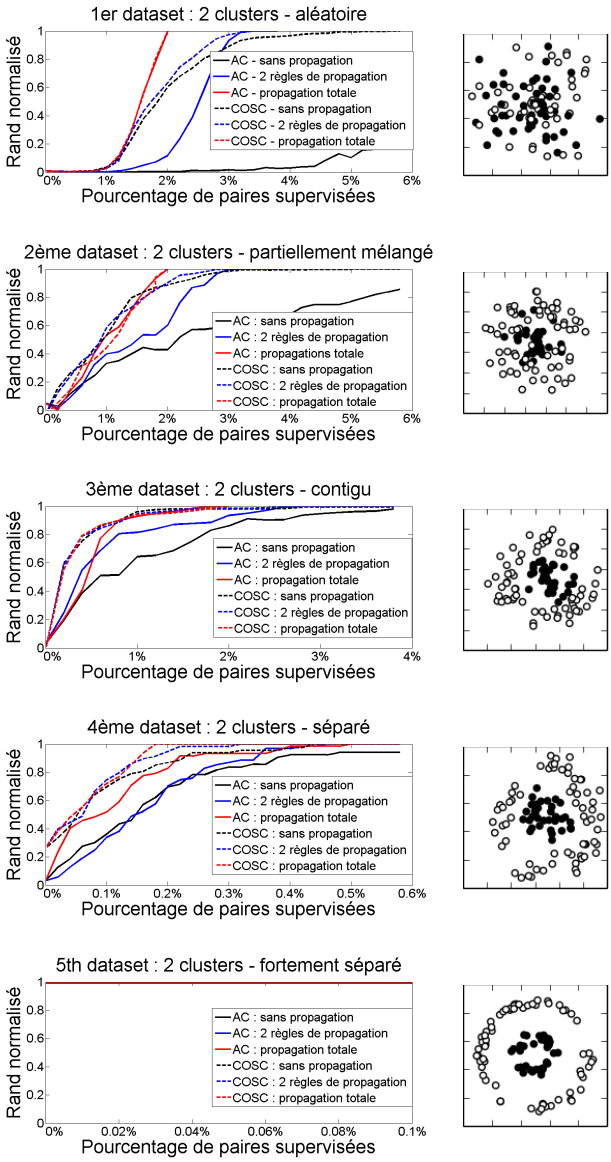

Figure 8. *Qualité du partitionnement en fonction du nombre de paires supervisées avec l'Active Clustering (traits continus) et COSC (pointillés) en utilisant aucune propagation (en noir), les 2 premières (en bleu) ou les 3 règles de propagation (en rouge).*

à la méthode COSC, les courbes en trait continu, à l'Active Clustering. On constate tout d'abord que l'Active Clustering est fortement amélioré par la propagation automatique. L'amélioration apportée à la méthode COSC est sensible lorsque l'on traite des configurations complexes (les deux premiers datasets), mais imperceptible dans les autres cas.

Dans tous les cas, avec l'utilisation des 3 règles de propagation, nous avons la garantie d'atteindre le partitionnement réel en moins de 100 paires supervisées, c'est à dire en supervisant moins de 2% de toutes les paires possibles. Dans les configurations complexes (les deux premiers datasets), la propagation permet à COSC de converger 2 fois plus rapidement que sans propagation. L'Active Clustering converge moins vite que COSC mais le facteur d'amélioration apporté par la propagation automatique est plus grand. Dans un cas simple comme le cinquième dataset, les 2 méthodes obtiennent le partitionnement réel sans avoir besoin de supervision.

Avec les 2 premiers datasets, on remarque que la qualité de la partition commence à augmenter seulement lorsque l'on a déjà ajouté une quantité importante de contraintes. Cet effet de seuil est dû à la progression non linéaire du nombre de contraintes apportées par la propagation automatique. La figure 9 présente le nombre de paires propagées en fonction du nombre de paires choisies aléatoirement et supervisées. Ce graphique montre qu'en dessous d'un certain seuil le nombre de paires propagées est peu important. L'explication de ce phénomène provient du fait que les paires sont choisies aléatoirement et que donc, au début, les objets ne sont que peu connectés. Ensuite, lorsque le graphe des contraintes commencent à être plus connexe, les propagations deviennent importantes. On constate aussi que la contribution de la règle 3 est non négligeable dans les cas des bi et tri partitionnements. Son usage conjoint avec les 2 autres règles permet un gain significatif de contraintes.

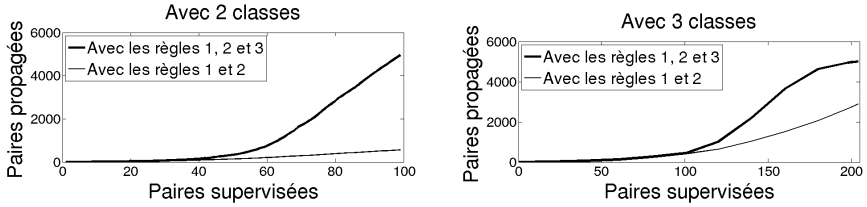

Figure 9. *Nombre de paires propagées selon l'usage ou non de la 3ème règle.*

5.2. *Le cas bi-classes réel*

Nous avons reproduit la même expérimentation qu'au paragraphe 5.1 avec deux datasets réels de 100 séquences vidéos réelles du dataset Blip1000 (Schmiedeke *et al.*, 2013). Les données exploitées sont des vecteurs *descripteurs audio standards* de dimension 196 proposés dans (Mironica *et al.*, 2013). Pour les deux datasets, nous

avons retenus 50 vidéos de deux genres : “Santé” et “Littérature” pour le premier ; “Santé” et “Documentaire” pour le deuxième.

Les résultats obtenus sont présentés dans la figure 10. Ils sont analogues à ceux obtenus avec les datasets de référence et pour la méthode COSC vus dans la section précédente. En terme de vitesse de convergence, “Santé” et “Littérature” nous donnent des résultats intermédiaires entre ceux des 3ème et 4ème datasets. Ce qui semblerait indiquer que les vidéos de genre “Santé” et “Littérature” ont des données audios assez séparées. “Santé” et “Documentaire” nous donnent des résultats analogues à ceux du 2ème dataset. Ce qui semblerait indiquer que les vidéos de genre “Santé” et “Documentaire” ont des données audio partiellement mélangées pour le descripteur utilisé.

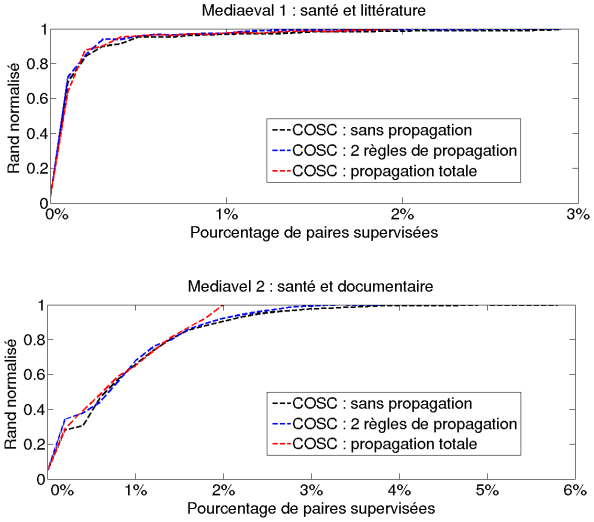

Figure 10. *Qualité du partitionnement en fonction du nombre de paires supervisées avec COSC en utilisant aucune propagation (en noir), les 2 premières (en bleu) ou les 3 règles de propagation (en rouge).*

5.3. *Le cas multi-classes*

Nous avons reproduit la même expérimentation qu’au paragraphe 5.1 sur des datasets multi-classes. Le premier dataset est composé de 100 points placés aléatoirement sur le disque unité bi-dimensionnel et répartis en 3 classes équilibrées. Il est représenté dans la partie droite de la figure 11. Les deuxième et troisième datasets sont composés des données vidéos réelles décrites au paragraphe 5.2. Le deuxième dataset est composé de 100 vidéos des genres “Santé”, “Documentaire” et “Littérature” réparties en 3 classes égales. Le troisième dataset est composé des 5197 vidéos réparties en 26 classes inégales. Il s’agit de l’ensemble des données du challenge MediaEval pour l’année 2012.

Les résultats sont présentés dans la partie gauche de la figure 11. La courbe bleue correspond à l'utilisation des 2 premières règles de propagation. La courbe rouge correspond à l'ajout de la troisième règle que nous avons proposé au paragraphe 3.

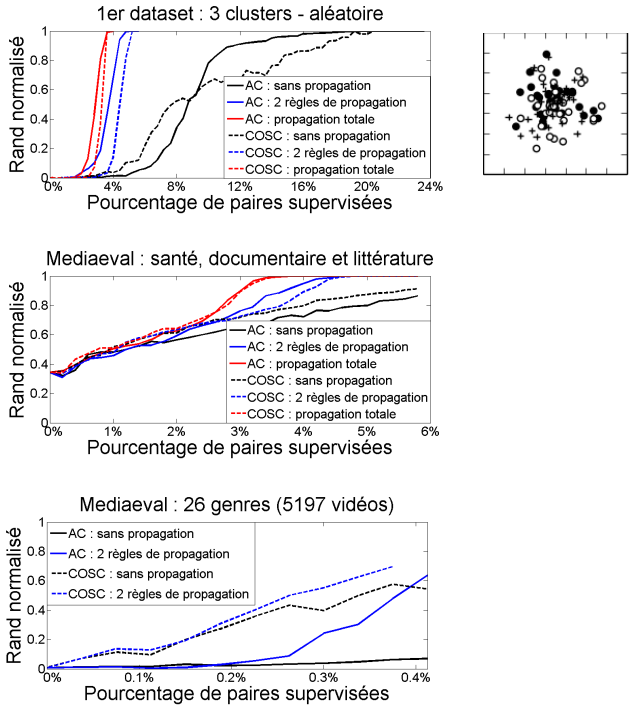

Figure 11. *Qualité du partitionnement en fonction du nombre de paires supervisées avec l'Active Clustering (traits continus) et COSC (pointillés) en utilisant aucune propagation (en noir), les 2 premières (en bleu) ou les 3 règles de propagation (en rouge).*

On peut observer que dans ce nouveau cas d'utilisation, la propagation automatique apporte un gain de vitesse de convergence significatif. Plus précisément, les deux premiers jeux de données montrent l'avantage de la troisième règle de propagation proposée dans la section 3 qui permet d'obtenir le partitionnement parfait avec 20% de contraintes en moins que la propagation limitée aux deux premières règles. Dans cette expérimentation, COSC atteint une performance inférieure à l'Active Clustering. Ceci peut s'expliquer par le fait que COSC effectue des coupes binaires de manière hiérarchique, ce qui n'est pas adapté à des datasets tri-classes.

Avec le troisième dataset qui contient un plus grand nombre de données et clusters, COSC surpasse l'Active Clustering. Une fois encore, les méthodes sont bien toutes deux améliorées par la propagation automatique des contraintes. Les résultats montrent que les deux premières règles de propagation appliquées à 50 000 liens, ce

qui représente seulement $0,37\%$ de tous les liens possibles, permettent d'améliorer la qualité de 21% pour COSC et de 650% pour l'Active Clustering.

Cependant, dans un tel cas, il faut discuter des coûts de calcul de la technique de propagation. En effet, les deux premières règles peuvent être appliquées efficacement grâce à la vectorisation de produit de matrices sparses. Les algorithmes et codes sources sont disponibles sur la même url que les données[1] . Mais la troisième règle consiste en un examen de tous les 26-simplexes du graphe. Pour l'instant, une telle analyse n'est pas optimisée et devient rapidement trop coûteuse en temps de calcul et en consommation mémoire. En conséquence, seules les deux premières règles de propagation ont été appliquées. D'autres travaux pourraient être conduits pour améliorer ces aspects et supporter le passage à l'échelle.

6. Conclusions

Ce papier présente une généralisation de la propagation automatique des contraintes utilisée dans le cas d'optimisations du clustering sur des graphes de similarités. Nous avons mené nos expérimentations sur un ensemble de données synthétiques et sur un ensemble de données réelles issues d'une classification par genre de vidéos. Nous avons montré les bénéfices de la généralisation de la propagation automatique sur le clustering de telles données. Ces gains ont été mis en évidence avec deux techniques différentes du Clustering Spectral semi-supervisé.

La principale contribution de ce papier réside dans la généralisation de la propagation automatique des contraintes. A l'issue de ce papier, nous pouvons donc recommander son usage car il réduit le coût d'ajout des contraintes en améliorant la qualité du clustering obtenu. Et dans le pire des cas, les performances ne sont pas inférieures à celles des méthodes d'origine.

Des travaux futurs pourront concerner l'optimisation de la propagation de contraintes en terme de coûts de calculs. En outre, la sélection des contraintes est intéressante dans le but d'améliorer l'efficacité des méthodes. Une autre perspective consiste à comparer cette approche à l'état de l'art des techniques de clustering supervisé dans le contexte de défis tels que le challenge MediaEval. La mesure de comparaison pourrait être simplement le taux d'erreur de classification.

Remerciements

Cette soumission du travail de Nicolas Voiron est soutenue financièrement par la région Rhône-Alpes (ARC6).

1. `http://www.polytech.univ-savoie.fr/index.php?id=listic-nicolas-voiron`.

7. Bibliographie

Bilenko M., Basu S., Mooney R. J., « Integrating Constraints and Metric Learning in Semi-supervised Clustering », *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, ACM, New York, NY, USA, p. 11-, 2004.

Datta R., Joshi D., Li J., Wang J. Z., « Image retrieval : Ideas, influences, and trends of the new age », *ACM Comput. Surv.*, vol. 40, n° 2, p. 5 :1-5 :60, May, 2008.

Davidson I., Wagstaff K. L., Basu S., « Measuring constraint-set utility for partitional clustering algorithms », *In : Proceedings of the Tenth European Conference on Principles and Practice of Knowledge Discovery in Databases*, Springer, p. 115-126, 2006.

Hubert L., Arabie P., « Comparing partitions », *Journal of classification*, vol. 2, n° 1, p. 193-218, 1985.

Kamvar S. D., Klein D., Manning C. D., « Spectral learning », *In IJCAI*, p. 561-566, 2003.

Li Z., Liu J., Tang X., « Constrained clustering via spectral regularization », *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, p. 421-428, 2009.

Mallapragada P. K., Jin R., Jain A. K., « Active query selection for semi-supervised clustering », *19th International Conference on Pattern Recognition (ICPR 2008), December 8-11, 2008, Tampa, Florida, USA*, p. 1-4, 2008.

Mironica I., Ionescu B., Knees P., Lambert P., « An In-Depth Evaluation of Multimodal Video Genre Categorization », *IEEE International Workshop on Content-Based Multimedia Indexing*, 2013.

Rangapuram S. S., Hein M., « Constrained 1-Spectral Clustering », *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2012, La Palma, Canary Islands, April 21-23, 2012*, p. 1143-1151, 2012.

Schmiedeke S., Xu P., Ferrané I., Eskevich M., Kofler C., Larson M., Estève Y., Lamel L., Jones G., Sikora T., « Blip10000 : A Social Video Dataset Containing SPUG Content for Tagging and Retrieval », *ACM Multimedia Systems Conference*, 2013.

Voiron N., Benoit A., Lambert P., « Automatic difference measure between movies using dissimilarity measure fusion and rank correlation coefficients », *Content-Based Multimedia Indexing (CBMI), 10th International Workshop*, 2012.

von Luxburg U., « A Tutorial on Spectral Clustering », *Statistics and Computing*, vol. 17, n° 4, p. 395-416, December, 2007.

Vu V., Labroche N., Bouchon-Meunier B., « Improving constrained clustering with active query selection », *Pattern Recognition*, vol. 45, n° 4, p. 1749-1758, 2012.

Wang X., Davidson I., « Flexible constrained spectral clustering », *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010*, p. 563-572, 2010.

Witten I., Frank E., Hall M., « Data Mining : Practical Machine Learning Tools and Techniques », *Morgan Kaufmann Publishers*, 2011.

Xiong C., Johnson D. M., Corso J. J., « Active Clustering with Model-Based Uncertainty Reduction », *CoRR*, 2014.

Xu L., Li W., Schuurmans D., « Fast normalized cut with linear constraints », *2013 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 0, p. 2866-2873, 2009.