Une méthode de classification visuelle et interactive

David Da Costa[*,**], Gilles Venturini[*]

[*]Laboratoire d'Informatique
Ecole Polytechnique de l'Université de Tours
64, Avenue Jean Portalis, 37200 Tours, France.
{david.dacosta, venturini}@univ-tours.fr
http://www.antsearch.univ-tours.fr/webrtic
[**]COHESIUM
71-73 rue de Saussure, 75017 Paris, France.
ddacosta@cohesium.com
http://www.cohesium.com/

Résumé. L'objectif de notre travail est de pouvoir représenter visuellement des ensembles de données et de laisser l'expert du domaine procéder interactivement à la définition d'une classification de ces données. Notre approche se base sur l'existence d'une fonction de similarité afin de pouvoir traiter des données de tous types (numériques, symboliques, images, textes, etc). Elle permet à l'expert du domaine, grâce à l'utilisation d'une visualisation à base de points d'intérêt (POIs), de définir des classes dans les données grâce à des opérations de sélections graphiques. Nous comparons notre approche interactive avec la classification ascendante hiérarchique (CAH) sur un ensemble de bases classiques. Nous montrons qu'un expert des données peut arriver à des performances similaires à celle d'un algorithme automatique, tout en bénéficiant d'informations complémentaires sur les classes.

1 Introduction

La classification est une des tâches importantes de la fouille de données et a pour but d'affecter à chaque donnée d'un ensemble un label de classe (Jain et Dubes, 1988). La grande majorité des méthodes dans ce domaine utilise des approches non interactives dans lesquelles un algorithme produit des résultats de manière entièrement automatisée. Cependant, on peut noter que l'expert final est le seul à pouvoir valider les résultats, et il arrive même que certaines méthodes (telles que la CAH) nécessite l'intervention de l'utilisateur pour définir de manière fiable un "bon" nombre de classes. Par ailleurs, les méthodes automatiques ne donnent pas systématiquement des informations directes sur la densité des classes, leurs formes, les relations de proximité entre classes : il s'agit souvent d'un travail supplémentaire d'interprétation de la part de l'expert.

Dans ce travail, nous proposons une méthode qui se place dans la lignée des méthodes visuelles et interactives qui vont solliciter l'expert du domaine et obtenir ainsi un résultat directement validé par l'utilisateur et des informations sur les classes obtenues. Notre objectif est

de construire interactivement une classification visuelle de données. Pour ce faire nous définissons une nouvelle méthode de visualisation basée sur les points d'intérêt à partir de laquelle l'utilisateur pourra définir des groupes de points.

L'article est organisé comme suit : la section 2 décrit les techniques de classification interactive. Dans la section 3 nous décrivons notre approche en commençant par spécifier l'utilisation des points d'intérêt pour la classification puis en évaluant cette méthode sur des données aux caractéristiques connues. Dans la section 4 nous décrivons l'application de notre méthode sur des données classiques. Nous concluons et dégageons des perspectives dans la section 5.

2 Classification interactive

2.1 Dans le cas supervisé

De nombreuses méthodes automatiques existent pour faire de la classification supervisée. A titre d'exemple, nous pouvons citer les arbres de décision (Breiman et al., 1984) (Quinlan, 1990) qui sont construits en divisant à plusieurs reprises l'ensemble des données en sous-ensembles disjoints et où une classe est affectée à chaque feuille de l'arbre. Des approches de classification visuelle ont été développées dans ce cadre en faisant appel à des principes très différents des méthodes automatiques : plutôt que de fournir un résultat qui devra être ensuite validé, elles combinent l'aspect découverte et validation en faisant intervenir l'expert.

Ainsi, PBC ("Perception Based Classification"), est basé sur les segments de cercle (Ankerst et al., 1999) et fournit une construction interactive d'arbres de décision pour aider l'utilisateur à établir des modèles de classification. StarClass est une méthode visuelle et interactive de classification : elle visualise des données multidimensionnelles en utilisant les "star coordinates" (Kandogan, 2000) et l'utilisateur peut agir sur une des dimensions pour créer un arbre de décision. Elle est cependant limitée à des petits ensembles de données. Plus récemment, (Teoh et Ma, 2003) ont proposé PaintingClass, qui est une nouvelle technique visuelle interactive de classification qui peut classifier des données symboliques et numériques. PaintingClass permet à des utilisateurs de visualiser des données multidimensionnelles, avec la construction, la visualisation et l'exploration interactives des arbres de décision. Chaque noeud dans l'arbre de décision est montré comme une projection visuelle des données.

En dehors des arbres de décision, d'autres approches interactives existent dans le cas de l'apprentissage supervisé, et nous pouvons citer par exemple le cas des SVM ("Support Vector Machines") (Vapnik, 1995) qui peuvent faire intervenir l'expert comme dans (Do et Poulet, 2005) où l'utilisateur ajuste le meilleur hyperplan de séparation entre des données.

2.2 Dans le cas non supervisé

La classification non supervisée correspond au cas où l'on ne dispose pas d'une base de référence étiquetée pour effectuer la classification, et de plus, cette classification se fait sans connaissance a priori du nombre de classes. Dans ce domaine, il existe là encore de nombreuses méthodes automatiques. Souvent l'expert du domaine est obligé d'intervenir pour valider les classes trouvées. Des méthodes interactives se sont donc développées de manière à permettre à l'expert de construire et valider les classes en une seule opération.

A titre d'exemple, nous pouvons citer "Neighborgram Clustering" de (Berthold et al., 2002). Cette méthode interactive analyse le voisinage de chaque groupe candidat, puis sélectionne le "meilleur groupe" et supprime l'ensemble des données recouvert par le voisinage et ainsi de suite, de manière interactive. De plus, nous pouvons citer aussi la technique "Object-Centered Interactive MultiDimensional Scaling" (OCI-MDS) (Broekens et al., 2006), qui permet à un expert de proposer des positions alternatives pour des données en les déplaçant dans un espace 2D en temps réel. L'expert est ainsi aidé par plusieurs types de rétroactions visuelles.

Plus récemment, "Interactive Visual Clustering" (IVC) (desJardins et al., 2007) est une nouvelle méthode qui permet à un utilisateur d'explorer interactivement des données afin de les grouper. IVC combine les techniques des forces et ressorts, utilisées dans les graphes (Fruchterman et Reingold, 1991), avec les interactions de l'utilisateur pour faire de la classification non supervisée. L'utilisateur peut alors déplacer des groupes de données à différents endroits sur l'écran afin de former des classes. Un algorithme est ensuite exécuté pour créer des groupes automatiquement avec les nouvelles contraintes implicites, liées aux déplacements.

2.3 Motivations

La contribution principale de cet article consiste à proposer une nouvelle méthodologie interactive de classification non supervisée. Par rapport aux méthodes citées dans la section précédente, notre première motivation consiste à traiter tous les types de données possible et avec des volumes plus importants. Pour cela nous considérons que nous disposons seulement d'une matrice de similarités entre objets à regrouper, sans faire d'autres hypothèses sur la représentation. De plus, nous souhaitons évaluer cette approche avec des données plus volumineuses (les approches citées précédemment proposaient généralement de classifier de petits ensembles de données) et en comparant avec une méthode classique les résultats obtenus par des utilisateurs experts ou non experts. Une autre motivation importante vient du fait que nous souhaitons augmenter les possibilités d'extraction d'information pour l'utilisateur : il s'agit d'obtenir, dans une même visualisation, une idée de la forme et du nombre de classes, des points isolés, des classes plutôt denses ou au contraire plus clairsemées. Enfin, une contrainte importante vient du fait que la méthode doit être simple à comprendre pour l'utilisateur. Même si nous ne pouvons pas à ce stade proposer de comparaisons expérimentales, nous argumenterons dans ce sens.

3 Utilisation des points d'intérêt pour la classification

3.1 Méthode de visualisation

La visualisation que nous allons décrire est la base de notre approche pour faire de la classification interactive. Notre méthode considère n données $D_1, ..., D_n$ et une matrice de similarité Sim entre ces données. $Sim(i, j)$ est la similarité entre les données D_i et D_j, cette matrice étant symétrique et avec une diagonale à 1. On note également que si $Sim(i, j) = 1$ alors les données D_i et D_j sont identiques, et que si $Sim(i, j) = 0$ alors elles sont totalement différentes.

Le principe de notre méthode consiste donc, à l'instar de systèmes de recherche documentaire tels que Sqwid (McCrickard et Kehoe, 1997) ou Radviz (Hoffman et al., 1999), à placer

k données appelées points d'intérêt (POIs) sur un cercle et à représenter les autres données en fonction de leur similarité à ces POIs. Chaque donnée se place sur le barycentre des POIs pondérés par la similarité entre cette donnée et chaque POI (Da Costa et Venturini, 2006). Par exemple, nous avons visualisé la base de données supervisée *Forest CoverType* (Blake et Merz, 1998) (voir figure 1) qui contient un total de 581012 données, 54 attributs et se compose de 7 classes. Lorsque l'on travaille ainsi avec des données supervisées, nous proposons à l'utilisateur de sélectionner, parmi différentes méthodes, comment choisir les POIs. Pour notre exemple, notre méthode place le premier individu de chaque classe comme point d'intérêt initial et positionne les données restantes. L'utilisateur peut ensuite réaliser de nombreuses opérations interactives. Il peut changer les POIs (en ajouter, en enlever, définir des POIs qui ne soient pas des données mais des hypothèses à tester, etc). Il peut également effectuer des zooms de différentes natures, détecter des points isolés et obtenir des informations sur ces points (voir (Da Costa et Venturini, 2006) pour plus de détails). Cette visualisation est très efficace également en ce qui concerne les temps d'affichage : elle peut afficher n données (et donc n^2 similarités) mais en calculant seulement un nombre linéaire de similarités (entre les données et les k POIs). Cela nous permet de traiter des ensembles ayant jusqu'à 1 million de données en moins d'une minute sur un PC standard.

3.2 Visualisation de données non supervisées

Dans cet article nous considérons des données non supervisées. Pour le choix initial des points d'intérêt, notre méthode propose k données (voir la section suivante) et place les données restantes comme précédemment. L'utilisateur peut ensuite modifier ces POIs de manière interactive, en choisissant par exemple les centres des nuages observés. Il peut enlever des POIs ou en rajouter de manière à obtenir un résultat visuellement satisfaisant (par exemple des groupes compacts et bien séparés). Nous rappelons ici que les données sont de tous types (pas seulement numériques) et qu'il n'est par conséquent pas possible d'appliquer des méthodes telles que les KMeans (McQueen, 1967) pour définir ces points. Néanmoins, un choix aléatoire est loin d'être pertinent pour l'utilisateur ce qui nous a mené à développer la méthode décrite dans la section suivante.

3.3 Fonction d'évaluation pour le choix des POIs

Dans cette partie, pour le choix initial des points d'intérêt, notre méthode propose k données choisies en minimisant une fonction d'évaluation. Comme précédemment, l'utilisateur peut donc adapter dynamiquement la visualisation mais il est important de choisir des POIs qui soient le meilleur point de départ possible pour faciliter la classification interactive. La méthode que nous proposons est la suivante : nous allons chercher un triplet de POIs (D_i, D_j, D_k) qui soit plus pertinent que celui choisi aléatoirement et qui donne donc lieu à une meilleure visualisation initiale. Pour cela, nous définissons une fonction d'évaluation $f(D_i, D_j, D_k)$ mesurant le coût d'un triplet et que nous allons chercher à minimiser. Intuitivement, il s'agit de trouver des points de vue différents sur les données : nous souhaitons donc trouver des POIs qui soient les plus dissimilaires possible les uns des autres. Nous définissons donc la fonction de coût f par la formule : $f(D_i, D_j, D_k) = min \sum_{l \in i,j; m \in j,k; l \neq m} Sim(D_l, D_m)$. Autrement dit, nous cherchons un triplet de POIs dans lequel la similarité maximale est minimum.

La recherche s'effectue ensuite de la manière suivante :

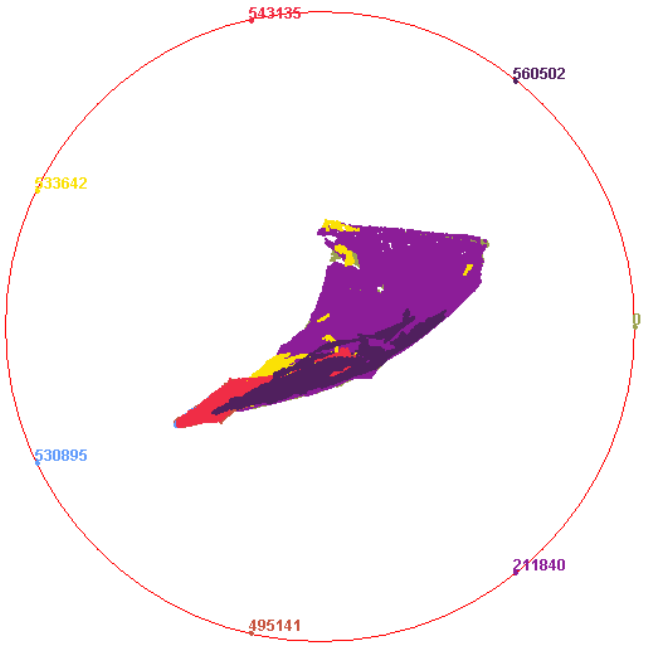
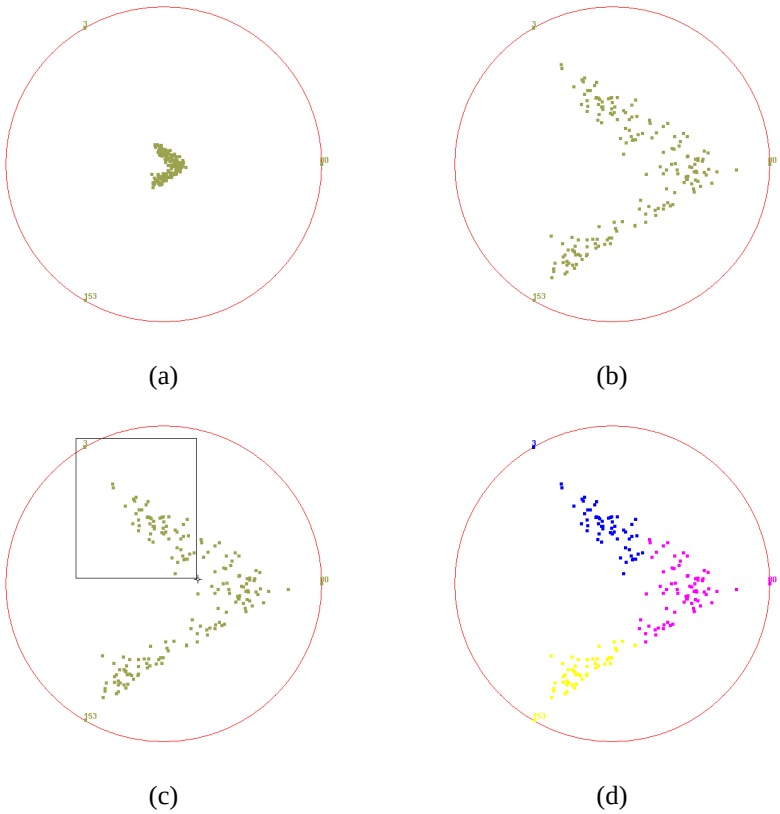


FIG. 1 – *Visualisation de la base de données Forest CoverType (le temps d'affichage est de 91 secondes sur un PC standard pour 581012 données).*

1. Générer aléatoirement et évaluer un triplet de POIs $T = (D_i, D_j, D_k)$
2. Pendant 1000 itérations :
   (a) Générer un triplet T' voisin de T en remplaçant l'un des POIs de T par un autre choisi aléatoirement, puis évaluer T',
   (b) Si $f(T') < f(T)$ alors $T \leftarrow T'$
3. Donner T en sortie

Cet algorithme effectue donc une ascension locale stochastique ("random hill climbing"). Le nombre d'itérations est limité par une constante car nous travaillons éventuellement avec de grands ensembles de données : l'affichage des données a lieu en temps linéaire et nous ne souhaitons accorder que quelques secondes de temps de calcul à l'opération de choix des POIs (sachant que l'on permet à l'utilisateur des possibilités d'interaction lui permettant d'améliorer le choix initial).



(a) (b) (c) (d)

FIG. 2 – *Exemple d'une classification visuelle et interactive de données sur la base "WINE". L'utilisateur obtient une visualisation satisfaisante (en (a) et (b)) à l'aide d'opérations interactives, puis sélectionne des données (en (c)) et obtient une classification (en (d)).*

3.4 Classification interactive

Une fois que l'utilisateur a fini de modifier ses POIs, il peut sélectionner des sous-ensembles de données (avec la souris) et définir ainsi un label de classe pour toutes les données sélectionnées. L'utilisateur étiquette les données à l'aide d'une couleur qui correspond à une classe. Il construit ainsi une classification des données, comme représenté sur la figure 2. Une fois toutes les données étiquetées, l'utilisateur peut exporter et sauvegarder sa base de données et ainsi la visualiser via d'autres méthodes que les POIs. Si l'on travaille sur des données supervisées, par exemple ayant été classifiées par une autre méthode, l'utilisateur peut alors dynamiquement comparer les classes réelles aux classes trouvées.

Grâce à cette méthode, l'utilisateur peut facilement détecter et définir des groupes, mais également avoir des informations sur la classification : le nombre de classes, les relations de voisinage entre classes, la densité des classes ou encore les points isolés.

4 Expérimentations

4.1 Protocole

Afin de déterminer de manière comparative l'efficacité de notre méthode, nous allons évaluer la classification obtenue à la fois en terme de nombre de classes trouvées C_T et de pureté des classes P_R. Pour cela, nous utilisons des bases réelles dont la classe des données est connue par avance, mais non utilisée dans le processus de classification.

Pour une classe donnée, la pureté représente le pourcentage de données bien classées et s'obtient à partir d'une matrice de confusion. Elle se calcule donc à partir de la classe réelle de chaque donnée. En effet, pour chacune des classes trouvées nous cherchons la cardinalité du groupe issu de la classe réelle qui est la plus représentée parmi les données de la classe trouvée considérée. Ainsi la somme de toutes les cardinalités pour toutes les classes trouvées divisée par le nombre total de données N représente la valeur de la pureté P_R (où M est la matrice de confusion, tableau croisé de la dispersion des données entre les classes trouvées et les classes réelles) :

$$P_R = \frac{1}{N} \times \sum_{i=1}^{C_T} maxM_i \tag{1}$$

4.2 Validation de la fonction d'efficacité

Nous avons commencé par tester l'efficacité de notre méthode d'optimisation dans l'amélioration du coût des POIs initiaux. Pour cela nous avons effectué 10 tests en moyenne par base et nous avons mesuré la progression de la fonction de coût en fonction du nombre d'itérations. Nous présentons le résultat de cette progression pour les bases Iris, Soybean et Thyroid sur la figure 3. On constate que le choix aléatoire peut être amélioré de manière efficace. De plus, nous avons généré tous les triplets de POIs sur la base Soybean afin de comparer le triplet retenu au triplet minimisant la fonction d'efficacité. La fonction d'efficacité est comprise dans [1.26383, 3.0] et la valeur $f(D_i, D_j, D_k)$ retenue par notre algorithme est en moyenne de 1.39962 avec un écart type de 0.22. Sur la figure 4, nous visualisons les différentes triplets choisis, en (a) les POIs sont choisis aléatoirement, en (b) le triplet retourné par la fonction d'efficacité et en (c) le triplet minimisant la fonction d'efficacité.

Ensuite, nous avons voulu vérifier que l'amélioration de la fonction de coût choisie correspond bien à l'amélioration de la visualisation. Pour cela, nous avons représenté dans la figure 5 la progression de la visualisation en fonction des itérations et du meilleur ensemble de POIs trouvé jusque là. On constate de manière visuelle que le critère choisi est pertinent.

Enfin, nous avons vérifié que les temps d'exécution de cet algorithme ne soient pas gênant pour l'utilisateur (la complexité de l'affichage est linéaire en fonction du nombre de données). D'après nos mesures, les temps d'affichage vont de 1 s (pour la base Iris) à 10 s (pour la base Pima), et de moins de 1 s pour le calcul des POIs.

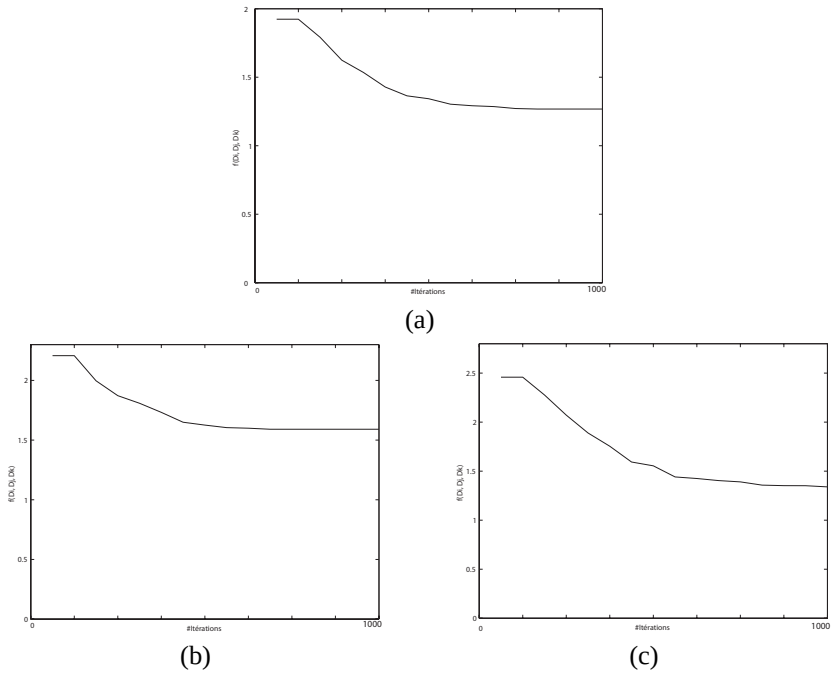


FIG. 3 – *Etude de la fonction de coût pour les bases Iris en (a), Soybean (b) et Thyroid (c).*

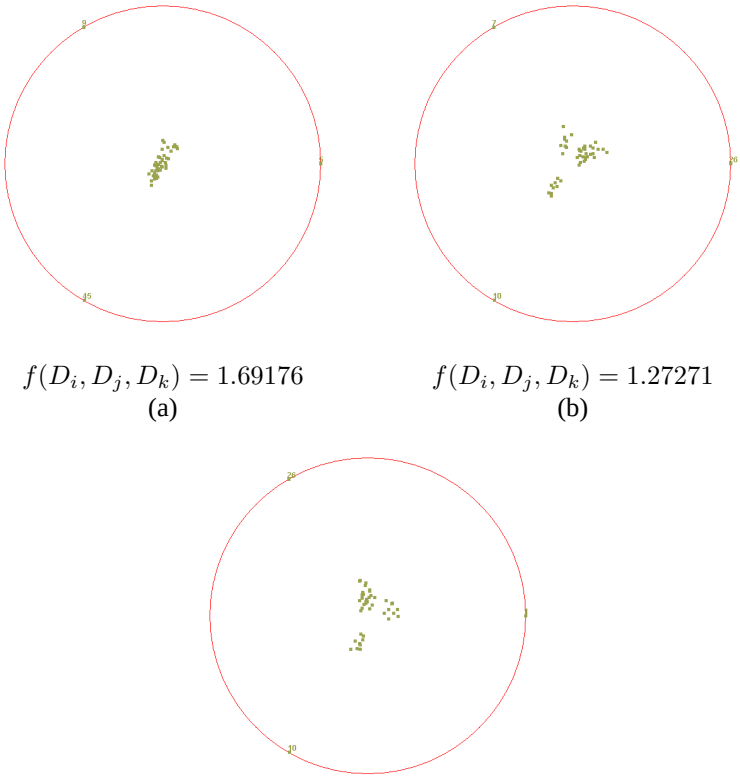
4.3 Comparaison entre classification interactive et classification automatique

Nous comparons les résultats de notre méthode à ceux obtenus avec la CAH (Lance et Williams, 1967)(Sneath et Sokal, 1973). Nous avons réalisé nos tests sur 7 bases numériques et symboliques. Chaque test consiste à visualiser les données, puis à regrouper les points comme indiqué dans la section précédente. Une fois la classification calculée, nous l'évaluons. Nous avons défini deux types de tests : les premiers présentés concernent un utilisateur expert des données (les auteurs de l'article) et ensuite des utilisateurs non experts (voir section suivante).

Les premiers résultats "experts" sont présentés dans le tableau 1. Pour chaque base, le nombre de classes réelles (CR), la dimension de l'espace des données (M) et le nombre de données total (N) sont rappelés.

Nous avons représenté les résultats obtenus par notre méthode selon trois colonnes : à partir des POIs choisis aléatoirement (et sans opérations interactives pour les améliorer), à partir des POIs optimisés automatiquement (toujours sans autres opérations interactives), et à partir des POIs optimisés automatiquement suivis des opérations interactives : ces opérations (modifications des points d'intérêt, zooms sans perte de contexte) permettent en effet d'améliorer encore la visualisation et de distinguer encore plus clairement les groupes.

On constate ainsi que le choix aléatoire des POIs, comme on peut s'y attendre, est moins efficace que les autres. L'utilisation de l'optimisation des POIs améliore les résultats au moins



$f(D_i, D_j, D_k) = 1.69176$
(a)

$f(D_i, D_j, D_k) = 1.27271$
(b)

$f(D_i, D_j, D_k) = 1.26383$
(c)

FIG. 4 – *Visualisation de la base Soybean, POIs choisis aléatoirement en (a), POIs retournés par la fonction d'efficacité en (b) et POIs optimaux en (c).*

pour les bases *Soybean* et *Hayes Roth*. Enfin, les meilleurs résultats sont obtenus lorsque l'utilisateur bénéficie de toutes les opérations possible sur la visualisation. A titre de comparaison, nous avons présenté les performances de la CAH sur les mêmes données. Pour déterminer le nombre de classes, nous utilisons la valeur du saut maximal du critère de Ward. On constate que notre méthode obtient des résultats compétitifs avec ceux de la CAH, tout en offrant à l'expert des informations additionnelles sur les classes.

Pour certaines bases comme *Pima*, il y a beaucoup plus de difficultés à étiqueter les données (ces données sont connues pour être très fortement bruitées, une caractéristique que l'on retrouve également avec la CAH qui obtient de mauvaises performances). C'est le cas également de la base *Vehicle*. Ces bases correspondent à des cas où les classes sont mal séparées les unes des autres, où les données sont superposées, une propriété que notre méthode détecte facilement. Pour cette dernière base, le fait de couper le dendogramme au niveau du saut maximum

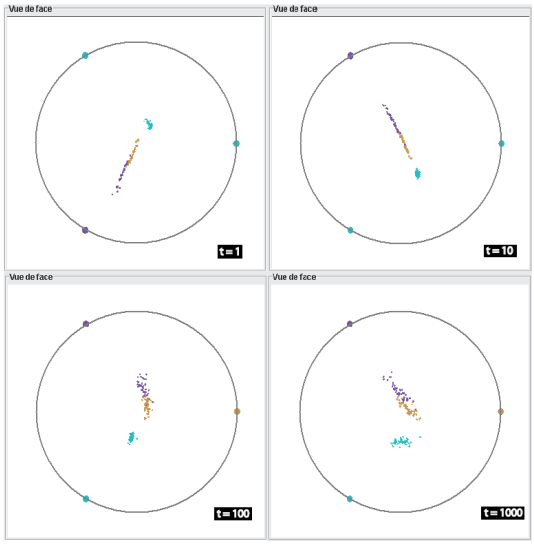


FIG. 5 – *Progression de la visualisation de la base Iris en fonction du nombre d'itérations pour l'optimisation des POIs.*

du critère de Ward engendre de mauvais résultats pour la CAH.

Cependant, nous avons aussi joué le rôle d'expert du domaine dans cette expérimentation. Afin de ne pas biaiser l'évaluation, nous avons demandé à d'autres utilisateurs novices de tester notre approche.

4.4 Test avec d'autres utilisateurs

Nous avons demandé à des utilisateurs (doctorants en informatique), ne connaissant ni les données ni la méthode et son implémentation, de tester notre outil afin de comparer leurs résultats à ceux obtenus par l'expert. Nous avons réalisé nos tests sur les mêmes bases numériques et symboliques. Les résultats sont présentés dans le tableau 2. Pour chaque base, le nombre de classes réelles (CR), la dimension de l'espace des données (M) et le nombre de données total (N) sont de nouveau rappelés ainsi que l'ensemble des classes trouvées (CT), la pureté et l'écart type de la pureté et du temps. Deux personnes ont été sollicitées : les deux premiers essais pour chaque base leur servent à apprendre (ils n'obtiennent jamais la valeur de pureté et les vrais nombres de classes). Les deux essais suivants sont notés. L'objectif annoncé est la création de groupes de points.

Nous pouvons constater que les utilisateurs non-experts obtiennent des résultats convenables sur ces données, résultats qui s'approchent des performances de l'expert. Cela nous conforte dans l'idée que notre méthode est plutôt intuitive. Par ailleurs, nous avons pu observer que le temps passé par les utilisateurs novices à classifier manuellement les données est lui aussi acceptable (moins de deux minutes).

Base de données	N	M	CR	POI						CAH	
				Sans interaction				Avec interaction			
				Aléatoire		Efficacité		Efficacité			
				C_T	P_R	C_T	P_R	C_T	P_R	C_T	P_R
IRIS	150	4	3	3	0,81	3	0,84	3	0,88	3	0,88
PIMA	768	8	2	3	0,69	3	0,66	3	0,70	3	0,65
SOYBEAN	47	35	4	3	0,66	3	0,78	3	0,79	6	1,00
THYROID	215	5	3	3	0,85	3	0,76	3	0,86	5	0,84
VEHICLE	846	18	4	3	0,41	3	0,40	2	0,70	3	0,35
WINE	178	12	3	3	0,70	3	0,60	3	0,64	6	0,84
HAYES ROTH	132	5	3	3	0,44	3	0,64	3	0,47	4	0,42

TAB. 1 – *Résultats obtenus par la CAH et notre méthode (POI) sur des bases de données numériques et symboliques, en considérant que l'utilisateur est un expert des données.*

Base de données	N	M	CR	Efficacité		
				C_T	P_R [σ_{P_R}]	Temps (s) [σ_{Temps}]
IRIS	150	4	3	{2, 3, 4}	0,78 [0,1323]	88 [63]
PIMA	768	8	2	{2, 3, 4}	0,6 [0,1018]	94 [16]
SOYBEAN	47	35	4	{3}	0,79 [0,010]	47 [18]
THYROID	215	5	3	{3}	0,87 [0,0455]	63 [27]
VEHICLE	846	18	4	{2, 3}	0,32 [0,0954]	82 [37]
WINE	178	12	3	{2, 3}	0,77 [0,1383]	74 [27]
HAYES ROTH	132	5	3	{2, 3}	0,43 [0,0804]	82 [43]

TAB. 2 – *Résultats obtenus par des utilisateurs "non experts".*

5 Conclusion

Nous avons proposé dans cet article une nouvelle méthode de classification non supervisée qui est interactive et qui s'appuie sur une visualisation des données. L'intérêt de cette méthode est de faire intervenir directement l'expert du domaine qui peut ainsi valider les résultats obtenus sans avoir à interpréter les sorties d'une méthode automatique.

Cependant, toutes les applications de classification ne se prêtent pas a un tel traitement et à une découverte interactive des classes. En effet, les experts peuvent déjà avoir des méthodes qui leur sont familières. En considérant d'une part des données où les classes ne sont pas trop recouvrantes (et ne vont donc pas rendre difficile ou impossible la classification interactive, voir la base Pima par exemple qui est difficile pour toutes les méthodes), et d'autre part un utilisateur plutôt novice dans les méthodes de classification, alors nous pensons que notre approche peut être très utile.

Nous n'avons pas effectué de comparaisons avec d'autres méthodes de classification inter-

actives, même si notre approche a été évaluée en comparaison avec une approche classique. Néanmoins, nous avons traité des données un peu plus volumineuses que les approches citées. De plus, on peut remarquer que l'apprentissage de notre représentation visuelle est peut être plus simple que pour les autres méthodes. En effet, il est connu que l'utilisation de certaines méthodes, comme les coordonnées parallèles, n'est pas facilement apprise par les utilisateurs. Il serait cependant intéressant de continuer les recherches dans cet axe et de valider notre approche par rapport aux autres sur les mêmes problèmes (recherche de classes en connaissant d'avance un résultat possible) mais également sur d'autres problèmes et d'autres tâches. Il peut s'agir par exemple de détecter des points aberrants, ou encore de traiter des données de volume beaucoup plus important (comme Forest Cover Type mentionnées au début de cet article).

Enfin, une extension importante de notre méthode est en cours d'étude : elle consiste à généraliser cette méthode en 3D et à utiliser du matériel de réalité virtuelle (écran stéréoscopique pour la visualisation et capteurs 3D pour les interactions).

Références

Ankerst, M., C. Elsen, M. Ester, et H.-P. Kriegel (1999). Visual classification : an interactive approach to decision tree construction. In *KDD '99 : Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, pp. 392–396. ACM Press.

Berthold, M. R., B. Wiswedel, et D. E. Patterson (2002). Neighborgram clustering interactive exploration of cluster neighborhoods. In *ICDM '02 : Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02)*, Washington, DC, USA, pp. 581. IEEE Computer Society.

Blake, C. et C. Merz (1998). UCI repository of machine learning databases.

Breiman, L., J. H. Friedman, R. A. Olshen, et C. J. Stone (1984). *Classification and Regression Trees*. Wadsworth : Belmont.

Broekens, J., T. Cocx, et W. A. Kosters (2006). Object-centered interactive multi-dimensional scaling : Ask the expert. In P.-Y. Schobbens, W. Vanhoof, et G. Schwanen (Eds.), *Proceedings of the 18th Benelux Conference on Artifical Intelligence (BNAIC 2006)*, Namur, Belgium, pp. 59–66.

Da Costa, D. et G. Venturini (2006). Visualisation interactive de données avec des points d'intérêt. In *Actes de la 18ème conférence francophone sur l'Interaction Homme-Machine, IHM'06*, pp. 219–222. AFIHM : ACM Press.

desJardins, M., J. MacGlashan, et J. Ferraioli (2007). Interactive visual clustering. In *International Conference on Intelligent User Interfaces (IUI07)*.

Do, T.-N. et F. Poulet (2005). Svm et visualisation pour la fouille de grands ensembles de données. In *Extraction et Gestion des Connaissances (EGC 2005)*, pp. 545–556.

Fruchterman, T. et E. Reingold (1991). Graph drawing by force-directed placement. In *Software - Practice and Experience*, Volume 21, pp. 1129–1164.

Hoffman, P., G. Grinstein, et D. Pinkney (1999). Dimensional anchors : a graphic primitive for multidimensional multivariate information visualizations. In *NPIVM '99 : Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation*

in conjunction with the eighth ACM internation conference on Information and knowledge management, New York, NY, USA, pp. 9–16. ACM Press.

Jain, A. et R. Dubes (1988). *Algorithms for Clustering Data*. Upper Saddle River, NJ, USA : Prentice Hall Advanced Reference Series.

Kandogan, E. (2000). Star coordinates : A multidimensional visualization technique with uniform treatment of dimensions. In *Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics*.

Lance, G. et W. Williams (1967). A general theory of classificatory sorting strategies : I. hierarchical systems. *Computer journal 9*(4), 373–380.

McCrickard, S. et C. Kehoe (1997). Visualizing search results using sqwid. In *Proceedings of the Sixth International World Wide Web Conference*, Santa Clara.

McQueen, J. (1967). Some methods of classification and analysis of multivariate observations. In *Proceedings of 5th Berkley Symposium on Mathematical Statistics and Probability*, pp. 281–297.

Quinlan, J. R. (1990). Induction of decision trees. In J. W. Shavlik et T. G. Dietterich (Eds.), *Readings in Machine Learning*. Morgan Kaufmann. Originally published in *Machine Learning* 1 :81–106, 1986.

Sneath, P. H. et R. R. Sokal (1973). *Numerical Taxonomy*. San Francisco : W.H. Freeman.

Teoh, S. T. et K.-L. Ma (2003). Paintingclass : Interactive construction, visualization and exploration of decision trees. In *Proceedings of ACM KDD 2003 Conference*, New York, pp. 667–672. ACM Press.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. New York : Springer Verlag.

Summary

The aim of this work is to visually represent a data set and to let the domain expert interactively define clusters among these data. Our approach is based on the existence of a similarity function in order to deal with data of any type (numeric, symbolic, images, texts, etc). Thanks to a visualization based on points of interest, the expert can define clusters using graphical operations. We compare our interactive approach with the ascending hierarchical clustering (AHC) on traditional bases. We show that a domain expert car reach similar or better performances than a completely automatic algorithm, and that he may benefit of additional information on the classes.