Une nouvelle approche de la programmation DC et DCA pour la classification floue

Le Thi Hoai An*, Le Hoai Minh**, Pham Dinh Tao***

*LITA, UFR MIM, Université Paul Verlaine - Metz, Ile du Saulcy, 57045 Metz Cedex, France
lethi@univ-metz.fr,
http://www.lita;sciences.univ-metz.fr/ lethi/
**LITA, UFR MIM, Université Paul Verlaine - Metz, Ile du Saulcy, 57045 Metz Cedex, France
lehoai@univ-metz.fr
***LMI, INSA de Rouen,BP 08, Place Emile Blondel, 76131 Mont Saint Aignan Cedex, France
pham@insa-rouen.fr

Résumé. Dans cet article, nous nous intéressons à Fuzzy C-Means (FCM), une technique très connue pour la classification floue. Nous proposons un algorithme efficace basé sur la programmation DC (Difference of Convexe functions) et DCA (DC Algorithm) pour résoudre ce problème. Les expériences numériques comparatives avec l'algorithme standard FCM sur les données réelles montrent la robustesse, la performance de cet nouvel algorithme DCA et sa supériorité par rapport à FCM.

1 Introduction

Le problème de classification automatique (clustering) est considéré comme une des problématiques majeures en extraction des connaissances à partir de données. Parmi les techniques de classification, la classification floue (fuzzy) via Fuzzy C-Means (FCM) est très connue. FCM a été introduite par Jim Bezdek en 1981 (Bezdek (1981)) comme une amélioration des méthodes clustering précédentes, et a été beaucoup développée dans les années 90. Cette approche a été appliquée avec succès dans plusieurs problèmes (diagnostic médical (Whitwell (2005)) , classification de textes (Rodrigues and Sacks (2004))), et est de plus en plus utilisé dans le domaine du data mining.

Dans un travail récent (Le Thi et al. 3 (2006)) nous avons formulé le modèle de FCM pour la classification floue sous la forme d'un programme DC (Difference of Convexe functions) et développé un schéma de DCA (DC Algorithm) pour sa résolution numérique. La programmation DC et DCA ont été introduits par Pham Dinh Tao en 1985 et intensivement développés par Le Thi Hoai An et Pham Dinh Tao depuis 1994 (voir (Le Thi Hoai An (1997)) - (Le Thi et al. 2 (2006)), (Pham Dinh Tao and Le Thi Hoai An (1997)), (Pham Dinh Tao and Le Thi Hoai An (1998)) et leurs références) pour devenir maintenant classiques et de plus en plus populaire. Ils ont été appliqués avec succès à nombreux problèmes d'optimisation non convexe différentiable ou non de grande dimension dans différents domaines des sciences appliquées, en particulier aux problèmes du data mining (voir par exemple (Le Thi et al. 1 (2006)), (Le Thi et al. 2 (2006)), (Liu et al (2003)), (Neumann et al. (2004)), (Weber et al. (2005))). Les résultats numériques présentés dans (Le Thi et al. 3 (2006)) montrent que, comme pour les autres problèmes déjà traités en data mining, DCA est efficace pour FCM. Ils prouvent également la superiorité de DCA par rapport à K-means. Cet algorithme est itératif et consiste en la résolution d'un programme convexe à chaque itération. Le temps de calculs de DCA est donc proportionnel à celui de la méthode utilisée pour résoudre les programmes convexes générés. Dans (Le Thi et al. 3 (2006)) l'algorithme du gradient pro-

jeté a été utilisé du fait que la projection en question est explicite, même s'il est connu pour être lent. Sans doute qu'avec d'autres décompositions DC on peut améliorer DCA pour la résolution de FCM.

L'objectif de ce travail est de développer un nouveau schéma de DCA dans lequel la résolution des problèmes convexes générés est moins coûteux. Nous proposons une autre décomposition DC qui donne naissance à un DCA très simple dont les calculs sont explicites à chaque itération : le sous problème convexe est en fait la projection d'un point sur une boule. Les expériences numériques comparatives entre FCM et l'algorithme DCA étudié dans (Le Thi et al. 3 (2006)) sur les données réelles montrent la robustesse, la performance de cette nouvelle version de DCA et sa supériorité par rapport à FCM.

Le papier est organisé de la façon suivante. Dans la deuxième section, nous présentons la formulation du problème FCM. La résolution de ce probème par la programmation DC et DCA est étudiée dans la troisième section. Finalement, les résultats numériques de nos algorithmes DCA et FCM sont rapportés dans la dernière section.

2 Une nouvelle formulation du modèle de FCM

Soit $X := \{x_1, x_2, ..., x_n\}$ l'ensemble de n points à classer. Chaque point x_i est un vecteur dans l'espace $\mathbb{R}^p$. Nous avons à classer ces n points dans c $(2 \leq c \leq n)$ classes différentes.
Considérons une matrice de pourcentage U de taille $(c \times n)$ dont chaque élément $u_{i,k}$ définit le pourcentage d'appartenance d'un point x_k à la classe C_i. Il est clair que

$$u_{i,k} \in [0,1] \text{ pour } i = 1...c, k = 1...n \; ; \; \sum_{i=1}^{c} u_{i,k} = 1, \text{ pour } k = 1...n. \tag{1}$$

Si la matrice de pourcentage U est déterminée, on en déduit la classification selon la règle suivante : le point x_k (pour $k = 1, \ldots, n$) est classé dans la classe C_i (pour $i = 1, \ldots, c$) si et seulement si

$$u_{i,k} = \max\{u_{j,k} : j \in \{1, \ldots, c\}\}.$$

Considérons la fonction J_m définie par :

$$J_m(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{i,k}^m ||x_k - v_i||^2, \tag{2}$$

où $\|.\|$ désigne, dans tout le papier, la norme Euclidienne de l'espace correspondant, V est une $(c \times p)$ - matrice dont chaque ligne v_i correspond au centre de la classe C_i, et $m \geq 1$ un paramètre entier qui définit le degré de flou du modèle.
Chercher une classification revient ainsi chercher la matrice de pourcentage U et les centres v_i. Le modèle mathématique de FCM s'écrit ainsi :

$$\begin{cases} \min J_m(U, V) := \sum\limits_{k=1}^{n} \sum\limits_{i=1}^{c} u_{i,k}^m ||x_k - v_i||^2 \\ \text{sous contraintes} \quad u_{i,k} \in [0,1] \text{ pour } i = 1, ..., c, \; k = 1, ..., n \\ \sum\limits_{i=1}^{c} u_{i,k} = 1, \; k = 1, ..., n \end{cases} \tag{3}$$

où seule la variable U est a priori bornée. En fait on peut aussi restreindre la variable V à un domaine borné. En effet, la condition nécessaire d'optimalité du premier ordre en (U, V) implique

$$\nabla_V J_m(U, V) = 0,$$

i.e.,

$$\partial_{v_i} J_m(U,V) = \sum_{k=1}^{n} u_{i,k}^m 2(v_i - x_k) \text{ pour } i = 1, ..., c,\ k = 1, ..., n$$

ou

$$v_i \sum_{k=1}^{n} u_{i,k}^m = \sum_{k=1}^{n} u_{i,k}^m x_k.$$

D'autre part, la non-vacuité des classes assure que $\sum\limits_{k=1}^{n} u_{i,k}^m > 0$, pour tout $i = 1, ..., c$. Par suite

$$\|v_i\|^2 \leq \frac{\left(\sum\limits_{k=1}^{n} u_{i,k}^m \max_{k=1,...,n} \|x_k\|\right)^2}{(\sum\limits_{k=1}^{n} u_{i,k}^m)^2} = \max^2{}_{k=1,...,n}\|x_k\| := r^2.$$

Considérons les nouvelles variables $t_{i,k}$ telles que $u_{i,k} = t_{i,k}^2$. La contrainte $\sum\limits_{i=1}^{c} u_{i,k} = 1$ devient $\sum\limits_{i=1}^{c} t_{i,k}^2 = 1$ ou $\|t_k\|^2 = 1$ avec $t_k \in \mathbb{R}^c$. Soient S_k la sphère de rayon 1 dans $\mathbb{R}^c$ et R_i la boule Euclidienne de rayon r dans $\mathbb{R}^p$, on peut reformuler le problème FCM comme :

$$\begin{cases} \min J_{2m}(T,V) := \sum\limits_{k=1}^{n}\sum\limits_{i=1}^{c} t_{i,k}^{2m}||x_k - v_i||^2 \\ \text{sous contraintes} \quad T \in \mathcal{S} := \Pi_{k=1}^{n} S_k,\ V \in \mathcal{C} := \Pi_{i=1}^{c} R_i \end{cases} . \tag{4}$$

Ce dernier est un problème d'optimisation non convexe dont la résolution sera décrite dans la suite.

Remarque. *Le changement de variables en $t_{i,k}$ nous amène à travailler sur $\mathcal{S}$, le domaine réalisable des variables $t_{i,k}$ qui est le produit des sphères et non le produit des simplexes comme dans le cas des variables initiales $u_{i,k}$. On pourrait penser alors que la non convexité de $\mathcal{S}$ rend le problème plus difficile qu'avec sa formulation initiale. Mais ce n'est pas le cas, comme on verra dans la suite, car le problème (4) sera reformulé sous une forme équivalente où $\mathcal{S}$ est remplacé par le produit des boules de rayon 1. Ce qui est intéressant, tant sur le plan algorithmique que numérique : la nouvelle formulation DC de (4) donne naissance à un schéma DCA extrêmement simple qui ne nécessite que des calculs explicites et donc non coûteux. Puisqu'il s'agit des calculs des projections d'un point sur une boule Euclidienne à chaque itération.*

3 La programmation DC et DCA pour la résolution de FCM

Pour faciliter la compréhension de notre approche, nous présentons, en premier lieu de cette section, une brève description de la programmation DC et DCA.

3.1 Introduction à la programmation DC et DCA

La programmation DC joue un rôle central en programmation non convexe (différentiable ou non) car la quasi totalité des problèmes d'optimisation de la vie courante est de nature DC. Elle connaît des développements spectaculaires au cours de cette dernière décennie. DCA est une méthode de descente (de type primal-dual sans recherche linéaire) pour la résolution d'un programme DC de la forme

$$\alpha := \inf\{f(x) := g(x) - h(x) \ :\ x \in \mathbb{R}^p\}, \tag{5}$$

où g, h sont les fonctions convexes semi-continues inférieurement et propres sur $\mathbb{R}^p$. Une telle fonction f est appelée fonction DC, et les fonctions convexes g et h les composantes DC de f. Il est à noter que la minimisation d'une fonction DC sur un ensemble convexe fermé C de $\mathbb{R}^p$ se ramène à un problème de type (5) car la contrainte $x \in C$ peut être incorporée dans la fonction objectif à l'aide de la fonction indicatrice χ_C définie par $\chi_C = 0$ si $x \in C$, $+\infty$ sinon. Lorsqu'une de ses composantes DC est polyédrale la fonction f est dite DC polyédrale et le programme DC correspondant DC polyédral. La programmation DC polyédrale joue un rôle crucial en programmation non convexe.

La congugaison d'une fonction convexe g, notée g^* est définie par

$$g^*(y) := \sup\{\langle x, y\rangle - g(x) \ : \ x \in \mathbb{R}^p\}.$$

La dualité DC est définie via la conjugaison des composantes DC et le programme dual de (5) est donné par (ici l'espace dual de $\mathbb{R}^p$ est identifié à lui-même) :

$$\alpha_D := \inf\{h^*(y) - g^*(y) \ : \ y \in \mathbb{R}^p\}. \tag{6}$$

Puisque chaque fonction $h \in \Gamma_0(\mathbb{R}^p)$ est caractérisée comme le supremum d'une famille finie des fonctions affines, c.à.d.

$$h(x) := \sup\{\langle x, y\rangle - h^*(y) : y \in \mathbb{R}^p\},$$

on a

$$\alpha = \inf\{g(x) - \sup\{\langle x, y\rangle - h^*(y) : y \in \mathbb{R}^p\} : x \in \mathbb{R}^p\} = \inf\{\alpha(y) : y \in \mathbb{R}^p\},$$

où

$$\alpha(y) := \inf\{g(x) - [\langle x, y\rangle - h^*(y)] : x \in \mathbb{R}^p\} \quad (P_y).$$

Il est clair que (P_y) est un programme convexe et

$$\alpha(y) = h^*(y) - g^*(y) \text{ si } y \in \text{dom } h^*, \text{ et } +\infty \text{ sinon.} \tag{7}$$

Par suite

$$\alpha = \inf\{h^*(y) - g^*(y) : y \in \text{dom } h^*\}.$$

Finallement on obtient, avec la convention naturelle $+\infty - (+\infty) = +\infty$:

$$\alpha = \alpha_D := \inf\{h^*(y) - g^*(y) : y \in \mathbb{R}^p\}.$$

On observe ainsi la symétrie parfaite entre les programmes DC primal et dual : le dual de (6) est exactement (5).

Le transport des solutions optimales globales entre l'ensemble des solutions optimales $\mathcal{P}$ de (5) et celui de (6) noté $\mathcal{D}$ s'exprime de la manière suivante ((Le Thi Hoai An and Pham Dinh Tao (2005)), (Pham Dinh Tao and Le Thi Hoai An (1997))) :

$$\cup\{\partial h(x^*) : x^* \in \mathcal{P}\} \subset \mathcal{D} \text{ et } \cup\{\partial g^*(y^*) : y^* \in \mathcal{D}\} \subset \mathcal{P}. \tag{8}$$

La relation (8) indique que la résolution d'un programme DC implique celle de son dual. D'autre part, ce transport reste valable entre les ensembles des solutions locales de (5) et (6) sous certaines hypothèses techniques.

En analyse convexe, (Rockafellar (1976)),(Urruty and Lemarechal (1993))

$$\partial h(x^0) := \{y \in \mathbb{R}^p : h(x) \geq h(x^0) + \langle x - x^0, y\rangle, \forall x \in \mathbb{R}^p\}$$

est appelé le sous-différentiel de h au point x^0. Tout élément de $\partial h(x^0)$ est appelé gradient de h en x^0. Le sous-différentiel $\partial h(x^0)$ est une partie convexe fermée qui coincide avec le gradient $\nabla h(x^0)$ si et seulement h est différentiable en x^0. Pour un $\epsilon > 0$, le $\epsilon-$ sous-différentiel de h est défini par

$$\partial_\epsilon h(x^0) := \{y \in \mathbb{R}^p : h(x) \geq h(x^0) + \langle x - x^0, y\rangle - \epsilon, \forall x \in \mathbb{R}^p\}.$$

L'égalité des valeurs optimales des programmes primal et dual (5) et (6) peut être traduite de manière équivalente par

$$\mathcal{P} = \{x^* : \partial_\epsilon h(x^*) \subset \partial_\epsilon g(x^*),\ \forall \epsilon > 0\}.$$

Mais sauf des cas très rares, cette condition d'optimalité globale est impraticable. Nous nous intéressons dès lors aux conditions d'optimalité locale pour les programmes DC (voir (Le Thi Hoai An (1997)), (Le Thi Hoai An and Pham Dinh Tao (2005)), (Pham Dinh Tao and Le Thi Hoai An (1997)), (Pham Dinh Tao and Le Thi Hoai An (1998)),(Le Thi Hoai An and Pham Dinh Tao (2003)) et références inclues) :

$$\partial h(x^*) \subset \partial g(x^*), \tag{9}$$

et

$$\partial h(x^*) \cap \partial g(x^*) \neq \emptyset. \tag{10}$$

(Un tel point x^* vérifiant (10) est appelé *point critique* de $g - h$).

La condition nécessaire d'optimalité locale (9) est également suffisante dans plusieurs cas rencontrés en pratique - par exemple, quand la fonction objectif $f := g-h$ est DC polyédrale avec h polyédrale, ou quand f est localement convexe en x^*.

Basé sur les conditions d'optimalité locale et la dualité DC, DCA consiste en la construction de deux suites $\{x^k\}$ et $\{y^k\}$, candidats respectifs aux solutions des problèmes primal et dual que l'on améliore à chaque itération (les deux suites $\{g(x^k) - h(x^k)\}$ et $\{h^*(y^k) - g^*(y^k)\}$ sont décroissantes) et qui convergent vers des solutions primale et duale x^* et y^* vérifiant des conditions d'optimalité locale. Le schéma général de DCA prend la forme :

$$y^k \in \partial h(x^k);\ \ x^{k+1} \in \partial g^*(y^k). \tag{11}$$

La première interprétation de DCA est simple : à chaque itération on remplace dans le programme DC primal la deuxième composante DC h par sa minorante affine $h_k(x) := h(x^k) + \langle x - x^k, y^k\rangle$ au voisinage de x^k pour obtenir le programme convexe suivant

$$\inf\{g(x) - h_k(x)\ :\ x \in \mathbb{R}^p\} \tag{12}$$

dont l'ensemble des solutions optimales n'est autre que $\partial g^*(y^k)$.

De manière analogue, la deuxième composante DC g^* du programme DC dual (6) est remplacée par sa minorante affine $(g^*)_k(y) := g^*(y^k) + \langle y - y^k, x^{k+1}\rangle$ au voisinage de y^k pour donner naissance au programme convexe

$$\inf\{h^*(y) - (g^*)_k(y)\ :\ y \in \mathbb{R}^p\} \tag{13}$$

dont $\partial h(x^{k+1})$ est l'ensemble des solutions optimales. DCA opère ainsi une double linéarisation à l'aide des sous-gradients de h et g^*. Il est à noter que DCA travaille avec les composantes DC g et h et non pas avec la fonction f elle-même. Chaque décomposition DC de f donne naissance à un DCA. Pour un programme DC donné, la question de décomposition DC optimale reste ouverte, en pratique on cherche des décompositions DC bien adaptées à la structure spécifiques du programme DC étudié pour lesquelles les suites $\{x^k\}$ et $\{y^k\}$

sont faciles à calculer, (si possible) explicites pour que les DCA correspondants soient moins coûteux en temps et par conséquent capables de supporter de très grandes dimensions.
La convergence de DCA : ((Le Thi Hoai An and Pham Dinh Tao (1997)), (Le Thi Hoai An and Pham Dinh Tao (2005)), (Pham Dinh Tao and Le Thi Hoai An (1997)), (Pham Dinh Tao and Le Thi Hoai An (1998)), (Le Thi Hoai An and Pham Dinh Tao (2003)))
Soient C (resp. D) l'ensemble convexe qui contient la suite $\{x^k\}$ (resp. $\{y^k\}$) et $\rho(g, C)$ (ou $\rho(g)$ si $C = \mathbb{R}^p$) défini par

$$\rho(g, C) = \sup\left\{\rho \geq 0 : g - \frac{\rho}{2}\|\cdot\|^2 \text{ soit convexe sur } C\right\}.$$

DCA est une méthodes de descente sans recherche linéaire, qui possède les propriétés suivantes :
i) Les suites $\{g(x^k) - h(x^k)\}$ et $\{h^*(y^k) - g^*(y^k)\}$ sont décroisantes et

- $g(x^{k+1}) - h(x^{k+1}) = g(x^k) - h(x^k)$ ssi
  $y^k \in \partial g(x^k) \cap \partial h(x^k), y^k \in \partial g(x^{k+1}) \cap \partial h(x^{k+1})$ et $[\rho(g, C) + \rho(h, C)]\|x^{k+1} - x^k\| = 0$. De plus, si g ou h est strictement convexe sur C alors $x^k = x^{k+1}$.
  Dans ce cas DCA se termine à l'itération k (convergence finie de DCA).
- $h^*(y^{k+1}) - g^*(y^{k+1}) = h^*(y^k) - g^*(y^k)$ ssi $x^{k+1} \in \partial g^*(y^k) \cap \partial h^*(y^k), x^{k+1} \in \partial g^*(y^{k+1}) \cap \partial h^*(y^{k+1})$ et $[\rho(g^*, D) + \rho(h^*, D)]\|y^{k+1} - y^k\| = 0$. De plus, si g^* ou h^* est strictement convexe sur D alors $y^{k+1} = y^k$.
  Dans ce cas DCA se termine à l'itération k (convergence finie de DCA).

ii) Si $\rho(g, C) + \rho(h, C) > 0$ (resp $\rho(g^*, D) + \rho(h^*, D) > 0$) alors la série $\{\|x^{k+1} - x^k\|^2$ (resp. $\{\|y^{k+1} - y^k\|^2\}$ converge.
iii) Si la valeur optimale α du problème *(5)* est finie et deux suites $\{x^k\}$ et $\{y^k\}$ sont bornées alors tout valeur d'adhérence $\widetilde{x}$ (resp. $\widetilde{y}$) de la suite $\{x^k\}$ (resp. $\{y^k\}$) est le point critique de $g - h$ (resp. $h^* - g^*$).
iv) DCA a la convergence linéaire pour les programmes DC généraux.
v) DCA a la convergence finie pour les programmes DC polyédraux.

Pour une étude complète de la programmation DC et DCA, se référer aux (Le Thi Hoai An (1997)) - (Le Thi Hoai An and Pham Dinh Tao (2005)), (Pham Dinh Tao and Le Thi Hoai An (1997)), (Pham Dinh Tao and Le Thi Hoai An (1998)) et références incluses. Il est à noter que *la recherche d'une décomposition DC adéquate et celle d'un bon point initial sont deux tâches importantes* dans la résolution d'un programme non convexe par DCA car elles conditionnent la réussite du résultant DCA.

3.2 Nouvelle formulation DC de FCM

Dans toute la suite nous utilisons la présentation matricielle qui nous semble plus comode, sachant que l'on peut identifier une matrice et un vecteur (par ligne ou par colonne).
La fonction objectif de (4) peut s'écrire de la manière suivante :

$$J_{2m}(T, V) = \frac{\rho}{2}\|T\|^2 + \frac{\rho}{2}\|V\|^2 - \left[\frac{\rho}{2}\|(T, V)\|^2 - J_{2m}(T, V)\right].$$

Pour tout $(T, V) \in \mathcal{S} \times \mathcal{C}$ on a

$$J_{2m}(T, V) = \frac{\rho}{2}n + \frac{\rho}{2}\|V\|^2 - H(T, V).$$

avec

$$H(T, V) = \tfrac{\rho}{2}\|(T, V)\|^2 - J_{2m}(T, V) \tag{14}$$

Dans le lemme suivant nous donnerons les conditions pour que la fonction H soit convexe.

Lemme : soit $\mathcal{B} := \Pi_{k=1}^{n}\ B_k$, où B_k est la boule de centre 0 et de rayon 1 dans $\mathbb{R}^c$. La fonction $H(U,V)$ est convexe sur $\mathcal{B}\times\mathcal{C}$ pour toute valeur de ρ telle que

$$\rho \geq \frac{m}{n}(2m-1)\alpha^2 + 1 + \sqrt{\left[\frac{m}{n}(2m-1)\alpha^2+1\right]^2 + \frac{16}{n}m^2\alpha^2},$$

où

$$\alpha = r + \max_{1\leq k\leq n}\|x_k\|\,.$$

Preuve : on remarque tout d'abord que $\rho > 0$ car $m \geq 1$.

Puisque

$$H(T,V) = \sum_{k=1}^{n}\sum_{i=1}^{c}\left[\frac{\rho}{2}t_{i,k}^2 + \frac{\rho}{2}\|v_i\|^2 - t_{i,k}^{2m}\|x_k - v_i\|^2\right],$$

H est convexe si toutes les fonctions

$$h_{i,k}(t_{i,k},v_i) := \frac{\rho}{2}t_{i,k}^2 + \frac{\rho}{2}\|v_i\|^2 - t_{i,k}^{2m}\|x_k - v_i\|^2, i = 1,...c, k = 1,...n$$

sont convexes.

Considérons la fonction suivante :

$$\begin{array}{l} f: \quad \mathbb{R}\times\mathbb{R}\to\mathbb{R} \\ f(x,y) = \frac{\rho}{2}x^2 + \frac{\rho}{2}y^2 - x^{2m}y^2\,. \end{array} \tag{15}$$

Le Hessien de la fonction f est donné par :

$$J(x,y) = \begin{pmatrix} \rho - 2m(2m-1)y^2x^{2m-2} & -4mx^{2m-1}y \\ -4mx^{2m-1}y & \frac{\rho}{n} - 2x^{2m} \end{pmatrix}. \tag{16}$$

Pour tout $(x,y) : 0 \leq x \leq 1; \|y\| \leq \alpha$, on a :

$$\begin{array}{rl} \mid J(x,y) \mid & = \left(\rho - 2m(2m-1)y^2x^{2m-2}\right)\left(\frac{\rho}{n} - 2x^{2m}\right) - 16m^2x^{4m-2}y^2 \\ & \geq \frac{\rho}{n}^2 - \left[2\frac{m}{n}(2m-1)y^2x^{2m-2} + 2x^{2m}\right]\rho - 16m^2x^{4m-2}y^2\ . \\ & \geq \frac{1}{n}\rho^2 - 2\left(\frac{m}{n}(2m-1)\alpha^2+1\right)\rho - 16m^2\alpha^2 \end{array}$$

Par suite, si

$$\rho \geq \frac{m}{n}(2m-1)\alpha^2 + 1 + \sqrt{\left[\frac{m}{n}(2m-1)\alpha^2+1\right]^2 + \frac{16}{n}m^2\alpha^2} \tag{17}$$

alors $\mid J(x,y) \mid \geq 0$, pour tout $(x,y) \in \mathbb{R}^2$ tels que $0 \leq x \leq 1, \mid y \mid \leq \alpha$.
Ainsi avec ρ défini par (17), la fonction f est convexe sur $[0,1]\times[-\alpha,\alpha]$. Par conséquent les fonctions

$$\theta_{i,k}(t_{i,k},v_i) := \frac{\rho}{2}\ t_{i,k}^2 + \frac{\rho}{2}\|x_k - v_i\|^2\ - t_{i,k}^{2m}\|x_k - v_i\|^2$$

sont convexes sur $\{0 \leq t_{i,k} \leq 1, \|v_i\| \leq r\}$ avec ρ donné dans (17) et $\alpha = r + \max_{1\leq k\leq n}\|x_k\|$.
Il en est de même pour les fonction $h_{i,k}$, car

$$h_{i,k}(t_{i,k},v_i) = \theta_{i,k}(t_{i,k},v_i) + \rho\langle x_k, v_i\rangle - \frac{\rho}{2}\|x_k\|^2\,.$$

Ainsi, avec les valeurs données ci-dessus de ρ et α, la fonction $H(T,V)$ est convexe sur $\mathcal{B}\times\mathcal{C}$. Dans toute la suite nous travaillons avec ces valeurs de ρ et α.

Il est clair que, pour tout $T \in \mathcal{B}$ et un $V \in \mathcal{C}$ fixé, la fonction $J_{2m}(T,V)$ est concave en variable T (car $H(T,V)$ est convexe), par suite son minimum sur $\mathcal{B}$ est atteint sur la frontière $\mathcal{S}$ de $\mathcal{B}$,i.e.,

$$\begin{aligned}&\min\left\{\tfrac{\rho}{2}\|V\|^2 - H(T,V) : (T,V) \in \mathcal{B}\times\mathcal{C}\right\}\\ &= \min\left\{\tfrac{\rho}{2}\|V\|^2 - H(T,V) : (T,V) \in \mathcal{S}\times\mathcal{C}\right\}.\end{aligned}$$

Le problème (4) peut être alors reformulé comme

$$\min\left\{\frac{\rho}{2}\|V\|^2 - H(T,V) : (T,V) \in \mathcal{B}\times\mathcal{C}\right\},$$

ou encore

$$\min\left\{\ \chi_{\mathcal{B}\times\mathcal{C}}(T,V) + \frac{\rho}{2}\|V\|^2 - H(T,V) : (T,V) \in \mathbb{R}^{c\times n}\times\mathbb{R}^{c\times p}\right\} \tag{18}$$

qui est un programme DC avec la décomposition DC suivante :

$$\chi_{\mathcal{B}\times\mathcal{C}}(T,V) + \frac{\rho}{2}\|V\|^2 - H(T,V) := G(T,V) - H(T,V),$$

où

$$G(T,V) := \chi_{\mathcal{B}\times\mathcal{C}}(T,V) + \tfrac{\rho}{2}\|V\|^2 \tag{19}$$

est bien évidemment une fonction convexe grâce à la convexité de $\mathcal{B}$ et $\mathcal{C}$.

3.3 Résolution de (18) par DCA

Selon la description de DCA dans la section 2.1, la résolution de FCM via la formulation (18) par DCA consiste en la détermination de deux suites $(Y^l, Z^l) \in \partial H\ (T^l, V^l)$ et $(T^{l+1}, V^{l+1}) \in \partial G^*(Y^l, Z^l)$.

La fonction H est différentiable et son gradient au point (T^l, V^l) est calculé de la manière suivante :

$$\nabla H(T^l, V^l) = \rho(T^l, V^l) - (2mt_{i,k}^{2m-1}\|x_k - v_i\|^2, 2\sum_{k=1}^{n}(v_i - x_k)t_{i,k}^{2m}). \tag{20}$$

Le calcul de $(T^{l+1}, V^{l+1}) \in \partial G^*(Y^l, Z^l)$ se ramène à la résolution du problème suivant (voir Section 2.1)

$$\min\left\{\frac{\rho}{2}\|V\|^2 - \langle (T,V), (Y^l, Z^l)\rangle : (T,V) \in \mathcal{B}\times\mathcal{C}\right\}.$$

Il s'en suit que (Proj étant l'application de projection)

$$T^{l+1} = \mathrm{Proj}_{\mathcal{B}}(Y^l), V^{l+1} = \mathrm{Proj}_{\mathcal{C}}(\frac{1}{\rho}Z^l).$$

Plus précisément :

$$V_{i,.}^{l+1} = \begin{cases}\frac{(Z^l)_{i,.}}{\rho} \text{ si } \|(Z^l)_{i,.}\| \le \rho r\\ \frac{(Z^l)_{i,.}r}{\|(Z^l)_{i,.}\|} \text{ sinon}\end{cases}, i = 1, ...c, \tag{21}$$

et

$$T_{.,k}^{l+1} = \begin{cases}Y_{.,k}^l \text{ si } \|Y_{.,k}^l\| \le 1\\ \frac{(Y^l)_{.,k}}{\|(Y^l)_{.,k}\|} \text{ sinon}\end{cases}, k = 1, ...n. \tag{22}$$

3.3.1 Schéma DCA

Initialisation :

- Choisir $T^0 \in \mathbb{R}^{c\times n}$ et $V^0 \in \mathbb{R}^{c\times p}$. Soit $l = 0$.
- Choisir une tolérance $\epsilon > 0$.

Répeter

- Calculer $(Y^l, Z^l) \in \nabla H(T^l, V^l)$ à l'aide de (20) ;
- Calculer $(T^{l+1}, V^{l+1}) \in \partial G^*(Y^l, Z^l)$ à l'aide de (21) et (22) ;
- $l + 1 \longleftarrow l$

Jusqu'à $\|(T^{l+1}, V^{l+1}) - (U^l, V^l)\| \leq \epsilon(\|(T^{l+1}, V^{l+1})\|)$.

Construction des classes Soient (T^*, V^*) la solution calculée par DCA et $u_{i,k} = t^{*2}_{i,k}$. Le point x_k appartient à la classe C_i si $u_{ik} = \max_{j=1..c} u_{j,k}$.

4 Expériences numériques

Pour comparer la performance de notre algorithme, nous avons réalisé les tests numériques sur deux ensembles des données : le premier ensemble de données contient 4 exemples très connus et beaucoup utilisés dans le domaine de classification pour l'évaluation des algorithmes :

- **PAPILLON** : un jeu de données connu sous le nom "jeux de papillon".
- **IRIS** : IRIS est peut-être le plus connu jeu de test dans le domaine de classification. Il contient 3 classes, chacune a 50 objets.
- **VOTE** : Congressional Votes dataset (Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL : Congressional Quarterly Inc. Washington, D.C., 1985).
- **GENE** : L'ensemble de 384 gènes disponible sur http ://faculty.washington.edu/kayee/cluster/
- **ADN** : L'ensemble de 3186 gènes disponible sur ftp ://genbank.bio.net, chaque gène est présenté par une séquence de 60 éléments. Ces gènes sont classés dans 3 clusters différents : donors (767 objets), acceptors (765 objets) et le rest.

Le deuxième ensemble de données est composé de deux jeux de données de biopuces : "Yeast" et "Serum" téléchargeables sur

http ://genomics.stanford.edu/.

"Yeast" : 2945 points (gènes) dans l'espace de dimension 15 ;

"Serum" : 517 points dans l'espace de dimension 12 (voir (Dembele et al. (2003)) pour la description de ces données).

Les tests ont été réalisés sur un ordinateur de 2.8MHz, 512Mb Ram. La valeur de ϵ est fixée à 10^{-7}. La valeur de m est égale à 2 pour le premier ensemble de données. Pour les données de biopuces nous considérons différentes valeurs de m dans l'intervalle $(1, 2)$ (il a été prouvé dans (Dembele et al. (2003)) que le choix de $m = 2$ n'est pas convenable à ces données).

Dans le Tableau 1, nous comparons notre nouvelle méthode DCA (DCA2) avec l'algorithme DCA développé dans (Le Thi et al. 3 (2006)) (DCA1) et une implémentation de la méthode FCM (téléchargeable sur http ://www-igbmc.u-strasbg.fr/projets/fcm/. Les critères de comparaison sont : le temp de calcul en seconds (Time), le nombre d'iterations (N^{o}it) et POBC (Pourcentage de Objets Bien-Classés). Nous constatons que dans tous les 4 jeux de données, DCA2 donne toujours le meilleur résultat.

Dans les Tableaux 2 et 3, nous présentons les résultats comparatifs de ces trois méthodes sur les données biopuces. Les critères de comparaison sont : la valeur de $J_m(U,V)$ (J_m), le coût du cluster (CC), soit $\sum_{i=1}^{n} \min_{k=1\ldots c} \|x_i - v_k\|^2$, le temps de calcul en seconds (Time) et le nombre d'itération (N^{o}it).

Data				DCA1			DCA2			FCM		
Name	n	p	c	N^{o}it	Time	POBC	N^{o}it	Time	POMC	N^{o}it	Time	POBC
PAPILLON	23	4	4	10	0.002	95.7	2	0.001	95.7	18	0.002	95.7
IRIS	150	4	3	23	0.03	92.77	4	0.01	92.77	15	0.03	92.25
VOTE	435	2	2	16	0.05	89.9	4	0.01	92.6	19	0.06	83.7
GENE	384	17	5	16	0.67	88.3	7	0.20	88.9	35	0.73	85.8
ADN	3186	60	3	8	0.78	92	6	0.55	94	25	1.95	89.8

Tableau 1 : Résultats comparatifs du premier ensemble de données

	DCA1				DCA2				FCM			
m	J_m	CC	N^{o}it	Time	J_m	CC	N^{o}it	Time	J_m	CC	N^{o}it	Time
1.1	23115	64831	33	331	**21615**	**64061**	189	33	21712	65868	179	564
1.3	17554	64144	357	64	**16789**	**62681**	225	99	16819	**62681**	543	1886
1.5	10531	43398	54	301	**10432**	**43389**	543	167	10652	44367	143	269
1.7	64129	44981	47	197	**6126**	**43792**	102	43	6294	43939	44	110
1.9	3676	45012	65	84	**3497**	**43956**	101	35	3607	44643	32	77

Tableau 2 : Résultats comparatifs de données de biopuces "Yeast".

	DCA1				DCA2				FCM			
m	J_m	CC	N^{o}it	Time	J_m	CC	N^{o}it	Time	J_m	CC	N^{o}it	Time
1.1	**1511.11**	12237	56	2.3	**1511.11**	**12234**	72	1.02	1511.56	13265	198	1.98
1.3	1523.6	**9572**	78	5.4	**1478.6**	**9572**	102	3.4	1554.50	10231	176	5.3
1.5	1645	8642	52	6.7	**1586**	**8034**	543	12	1755	9432	69	3.1
1.7	**1404.5**	6034	41	4.1	**1404.5**	**6021**	102	3.2	1415.3	6068	29	1.4
1.9	935	6079	85	2.2	**926**	**6022**	101	1.2	935	6079	14	0.7

Tableau 3 : Résultats comparatifs de données de biopuces "Serum".

Conclusion. Nous avons introduit une nouvelle formulation DC du modèle de FCM pour la classification floue et développé un schéma de DCA pour sa résolution numérique. Avec cette décomposition DC notre algorithme itératif est extrêmement simple, il consiste en la détermination de la projection d'un point sur une boule Euclidienne, qui est explicite et non coûteux. Les résultats numériques montrent que, comme pour les autres problèmes déjà traités en data mining, DCA est efficace pour FCM. Ils prouvent la superiorité de DCA non seulement par rapport à l'algorithme FCM standard mais aussi par rapport au schéma de DCA proposé dans (Le Thi et al. 3 (2006)). Dans l'étape suivante nous devront exploiter cet algorithme pour la classification des données biologiques, en particulier des donnéées de biopuces.

References

Bezdek (1981). Pattern Recognition with Fuzzy Objective Function Algorithm. *New York, NY. Plenum Press. 1981*

Dembélé, D., Kastner, P.. Fuzzy C-means Clustering method for clustering microarray data. *Bioinformatics. Vol. 19, No 8, pp 573-580, 2003*.

F. Höppner, F. Klawonn. Obtaining Interpretable Fuzzy Models from Fuzzy Clustering and Fuzzy Regression. *Proc. of the 4th Int. Conf. on Knowledge-Based Intelligent Engineering Systems and Allied Technologies (KES), Brighton, UK, pp. 162-165, 2000.*

F. Höppner, F. Klawonn. Fuzzy Clusteringof Sampled Functions. *Proc. of the 19th Int. Conf. of the North American Fuzzy Information Processing Society (NAFIPS), Atlanta, USA, pp. 251-255, 2000.*

F. Klawonn, F. Höppner. What is Fuzzy About Fuzzy Clustering? – Understanding and Improving the Concept of the Fuzzifier. *Advances in Intelligent Data Analysis , Berlin (2003), 254-264, Springer.*

J. Neumann, C. Schnörr, G. Steidl. SVM-based Feature Selection by Direct Objective Minimisation. *Pattern Recognition, Proc. of 26th DAGM Symposium, pp. 212 - 219, LNCS Volume 3175, 2004*.

Le Thi Hoai An. Contribution à l'optimisation non convexe et l'optimisation globale : Théorie, Algorithmes et Applications. *Habilitation à Diriger des Recherches, Université de Rouen, (1997).*

Le Thi Hoai An and Pham Dinh Tao. Solving a class of linearly constrained indefinite quadratic problems by DC algorithms. *Journal of Global Optimization, Vol 11, No 3, pp 253-285, 1997.*

Le Thi Hoai An and Pham Dinh Tao. Large Scale Molecular Optimization from distances matrices by a DC Optimization approach. *SIAM J. Optimization, Vol. 14. No1, pp. 77-117, 2003.*

Le Thi Hoai An and Pham Dinh Tao. The DC (difference of convex functions) Programming and DCA revisited with DC models of real world nonconvex optimization problems. *Annals of Operations Research 2005, Vol 133, pp. 23-46.*

Le Thi Hoai An, T. Belghiti and Pham Dinh Tao. A new efficient algorithm based on DC programming and DCA for Clustering. *In Press, Available July 2006, Journal of Global Optimization*.

Le Thi Hoai An, Le Hoai Minh and Pham Dinh Tao. Optimization based DC programming and DCA for Hierarchical Clustering. *n Press, Available online June 2006, European Journal of Operational Research*.

Le Thi Hoai An, Le Hoai Minh, Pham Dinh Tao. Une approche de la programmation DC pour la Classification floue. *Actes de XIIIème Rencontres de la Société Francophone de Classification SFC'06, Metz 6-9 Septembre, 2006*.

Pham Dinh Tao and Le Thi Hoai An. Convex analysis approach to DC programming : Theory, Algorithms and Applications. *Acta Mathematica Vietnamica, dedicated to Professor Hoang Tuy on the occasion of his 70th birthday, Vol.22, Number 1 (1997), pp. 289-355*.

Pham Dinh Tao and Le Thi Hoai An. DC optimization algorithms for solving the trust region subproblem. *SIAM J.Optimization, Vol. 8, pp. 476-505 (1998)*.

B. T. Polyak. Introduction to optimization. *Inc., Publications Division, 1987*.

Susana Nascimento, Boris Mirkin, and Fernando Moura-Pires. Modeling Proportional Membership in Fuzzy Clustering. *IEEE Transactions on Fuzzy Systems, Vol. 11, Nř 2, April 2003*.

Rochafellar, R.T. Convex Analysis (Princeton Landmarks in Mathematics and Physics). *Reprint Princeton University Press.*

M.E.S. Mendes Rodrigues and L. Sacks. A scalable hierarchical fuzzy clustering algorithm for text mining. *In : Proc. of the 4th International Conference on Recent Advances in Soft Computing, RASC'2004, pp.269-274, Nottingham, UK, Dec. 2004* .

Yufeng LIU, Xiaotong SHEN, and Hani DOSS, Multicategory. Multicategory ψ-Learning and Support Vector Machine :Computational Tools. *Journal of Computational and Graphical Statistics,14(1) : 219-236*.

J. B. Hiriart Urruty, and C. Lemaréchal. Convex Analysis and Minimization Algorithms. *Springer Verlag berlin Heidelberg (1993).*

S. Weber, T. Schüle, C. Schnörr. Prior Learning and Convex-Concave Regularization of Binary Tomography. *Electr. Notes in Discr. Math., 20 :313-327, 2005*.
Glenn Whitwell, Xiao Ying Wang, Jonathan M. Garibaldi. The Application of a Simulated Annealing Fuzzy Clustering Algorithm for Cancer Diagnosis. *SIP 2005, Japan*.

Summary

In this paper, a model of Fuzzy C-means Clustering (FCM), one of the most popular and best studied fuzzy clustering measures, is discussed. A fast and robust algorithm based on DC (Difference of Convex functions) programming and DCA (DC Algorithms) is investigated. Preliminary numerical solutions on real-world databases show the efficiency and the superiority of the appropriate DCA in both the running-time and quality of solutions with respect to the standard FCM algorithm.