République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Mémoire de Magister

Présenté à l'Université de Guelma
Faculté des Sciences et de l'Ingénierie

Département de : Génie Électrique
Spécialité : Informatique Industrielle et Imagerie

Présenté par : **Houria BOUDOUDA**

CLASSIFICATION AUTOMATIQUE NON SUPERVISÉE

Sous la direction de : **Dr. Hamid SERIDI**

Juin 2004

Soutenu le :.....devant la Commission d'Examen

JURY

Président :	H. TEBBIKH	Pr.	Université de Guelma
Rapporteur :	H. SERIDI	M.C.	Université de Guelma
Examinateurs :	B. BENSAKER	M.C.	Université de Annaba
	A/H. BOUKROUCHE	M.C.	Université de Guelma
	B. BENZELTOUT	PhD	Université de Guelma

Remerciements

L'ensemble des travaux présentés dans ce mémoire est réalisé au sein du Laboratoire d'automatique et Informatique de Guelma.

Je tiens à remercier Allah qui m'a donné la force pour accomplir ce travail.

J'exprime mes remerciements et ma gratitude à mon encadreur Docteur H.Seridi pour son suivi permanent durant le déroulement de ce mémoire.

Je remercie également le directeur du LAIG le professeur H.Tebbikh pour ses encouragements.

Mlle. H. Boudouda

Résumé

Devant la masse d'informations qui ne cesse de croître de manière exponentielle, l'expert humain est souvent confronté à des problèmes de classification de données qui rentrent dans le cadre de la reconnaissance des formes. Les méthodes de classification sont généralement le fruit d'un formalisme basé sur un raisonnement artificiel qui est plus au moins proche de celui d'un être humain. Il existe plusieurs méthodes de classification non supervisée qui se démarquent les unes des autres par le concept d'appartenance d'un objet à une classe. Dans ce même contexte, nous présentons une nouvelle approche de classification automatique non supervisée sous la famille des C-Moyennes. Cette nouvelle approche basée sur la fusion des théories du flou et des possibilités, permet d'une part de résoudre simultanément le problème de coïncidence et du bruit et d'autre part d'accélérer le temps de classification. La méthodologie d'initialisation utilisée dans cette nouvelle approche est basée sur le concept d'appartenance équiprobabiliste des objets aux différentes classes.

Abstract

In front of the mass of information which doesn't cease growing in an exponential way, the human expert is often confronted to data classification problems in the pattern recognition domain. The methods of classification are generally the result of a formalism based on an artificial reasoning, which is at least close to that of a human reasoning. There are several methods of unsupervised classification, which differ the ones from the others by the membership concept of an object to a class. In this same context, we present a new approach of unsupervised automatic classification under the C-Means family. This new approach based on the fusion of fuzzy and the possibility theory, allows on the one hand to solve, simultaneously the problem of coincidence and the noise and on the other hand to accelerate classification. The initialization methodology used in this new approach is based on equiprobabilist membership concept of the objects to the various classes.

الملخص

نظرا لضخامة المعلومات التي هي في تطور دائم وسريع، عادة ما يواجه الخبراء ضرورة تقسيمها من اجل تسهيل فهمها، هته العملية تدخل تحت إطار معرفة الأشكال حيث تعتبر طرق التقسيم ثمرة لتحصيل الذكاء الاصطناعي القريب من التفكير البشري. تقسم طرق التقسيم الأوتوماتيكي غير المراقب حسب كيفية انتماء العناصر إلى مختلف المجموعات. في نفس هدا السياق نقترح طريقة جديدة للتقسيم الأوتوماتيكي غير المراقب تحت عائلة C -Means التي تعتمد على الالتحام بين نضرتي الإمكان والغموض وهي قادرة من جهة وفي نفس الوقت على حل مشكلتي التصادف والتشويش ومن جهة أخرى على تسريع عملية التقسيم. طريقة التقسيم الأولي اللازمة لبدء خوارزميات التقسيم ترتكز أساسا على مفهوم للانتماء الاحتمالي المتكافئ لمختلف العناصر إلى مجموعاتها.

Table des Matières

Introduction Générale

La classification automatique est une étape fondamentale pour la reconnaissance des formes en Intelligence Artificielle.

En anglais et aussi parfois en français, classification veut dire classement : une méthode de classement est une façon d'associer à chaque objet une classe prédéterminée. C'est le mot clustring qui correspond à la démarche de détermination des classes à partir de données.

Classer est une activité qui consiste à regrouper des objets en un nombre limite de classes. La méthode de regroupement des objets peut être hiérarchique ou non.

Les méthodes de classification sont nombreuses et très utilisées dans différents domaines technologiques tels que : la reconnaissance des formes (caractères, visages,... etc.), l'analyse des images (médicales, radar, satellitaires,... etc.), le diagnostique (médical, de pannes,... etc.), l'inspection et contrôle de qualité dans l'industrie et la reconnaissance des cibles dans le domaine militaire.

D'une manière plus générale, les méthodes de classification permettent de regrouper des objets selon leur ressemblance. Elles placent alors certains objets dans une même classe et séparent d'autres en les plaçant dans des classes différentes. La classification est précisément une méthode qui a pour but de construire une partition d'un ensemble d'objets dont on sait calculer leurs distances deux à deux, les classes obtenues doivent être homogènes.

Il existe deux grandes familles de classification automatique:

- Classification non supervisée;
- Classification supervisée.

On utilise une classification non supervisée lorsque l'identité des différents types présentés dans l'ensemble des données n'est pas connue, cela résulte d'un manque d'information ou de l'incertitude sur la réalité de données à classifier.

Si on possède suffisamment d'informations sur l'ensemble à classifier, on peut effectuer une classification supervisée. Pour cela on doit au préalable définir des sites d'apprentissage. Toutes les méthodes de classification supervisée reposent sur l'hypothèse que les statistiques des données d'apprentissage de chaque classe sont distribuées selon la loi normale. C'est pour quoi une évaluation de la qualité des échantillons par le biais d'outils statistiques est nécessaire. Il existe plusieurs méthodes de classification supervisée, citons par exemple : le maximum de vrai semblance, distance minimum, parallélépipède... etc.

Dans ce mémoire, nous présentons la classification automatique non supervisée en quatre chapitres:

Dans le premier chapitre nous présentons les méthodes principales du raisonnement approximatif pour la gestion des incertitudes et des imprécisions en présence d'informations incomplètes et/ou inexactes.

Au dix-septième siècle Pascal et Fermat ont introduit la notion de probabilité afin de formaliser le concept de l'incertain. Cette théorie permet de raisonner dans un monde où l'observation expérimentale vient s'ajouter à la connaissance a priori de la fréquence d'occurrence d'événements. Cependant, la théorie des probabilités s'avère inadéquate à la gestion des connaissances de nature imprécises, d'autre part l'incertain peut provenir à partir des connaissances subjectives d'un expert humain, où l'on ne dispose d'aucune information de nature fréquentielle.

Face à ces imperfections Lotfi. A. Zadeh a introduit les deux théories suivantes:

1. La théorie des sous-ensembles flous [1] [3] pour modéliser l'imprécision de l'information.

2. La théorie des possibilités [2] [13] pour gérer les incertitudes d'événement de nature non probabiliste qui peuvent être précis ou imprécis.

Ces deux théories constituent le cadre de base de la logique floue.

L'introduction du flou dans la théorie des ensembles offre un nouveau cadre de raisonnement plus naturel comparativement au raisonnement classique qu'est plus dure. Ceci permet au raisonnement flou à occuper une place importante dans le domaine de classification

Dans le deuxième chapitre nous présentons les différentes approches de la classification automatique non supervisée à savoir : l'approche classique, floue et possibiliste.

Les méthodes de classification automatique non supervisée visent à répartir les objets en classes homogènes. On distingue deux grands types de méthodes de classification: celles dites non hiérarchiques qui produisent une partition des individus en un nombre fixe de classes et celles dites hiérarchiques qui fournissent une suite de partitions emboîtées qui définissent une hiérarchie.

Dans le troisième chapitre nous décrivons en premier lieu une méthode de classification automatique non supervisée non hiérarchique appelée C-Moyennes. Ensuite, nous présentons deux autres méthodes de classifications non supervisées hierarchiques ascendantes (CAH) et hiérarchiques descendantes (CDH). En dernier lieu, nous donnons une troisième variante appelée méthode d'agglomération compétitive (CA), qui combine les techniques des C-Moyennes et hiérarchiques.

Les méthodes de classification par C-Moyennes conduisent à une partition de l'ensemble de départ en un nombre C de classes de même niveau. Ces méthodes cherchent à trouver, pour chaque classe, un prototype (généralement le centre de gravité) le plus représentatif. Elles permettent aussi de traiter des ensembles d'effectifs assez élevés en optimisant des critères pertinents.

L'algorithme des C-Moyennes floues (FCM) repose à la fois sur le concept de la partition floue et sur la contrainte probabiliste qui est imposée sur la somme des degrés d'appartenance d'un objet aux différentes classes de l'ensemble grossier.

L'algorithme des C-Moyennes classiques (HCM) qui s'appuie sur le concept classique d'appartenance constitue un cas limite d'une partition floue lorsque le paramètre qui contrôle l'introduction du flou dans la partition vaut 1.

L'algorithme des C-Moyennes possibilistes (PCM) est basé sur le concept d'appartenance possibiliste.

Les méthodes de classification hiérarchique ascendante (CAH) partant des individus isolés assimilés à des classes et procédant, à chaque étape, par agrégation des deux classes les plus proches au sens de la norme choisie. Chaque niveau de hiérarchie représente une classe. Un arbre planaire hiérarchique permet de décrire de façon explicite la structure finale de la classification obtenue: "plus les individus se regroupent en bas de l'arbre, plus ils se ressemblent".

Inversement, les méthodes de classification hiérarchique descendantes (CDH) partent de l'ensemble des individus et procèdent par divisions successives de classes jusqu'à l'obtention de classes vérifiant certaines règles d'arrêt. On les appelle aussi méthodes divisives.

Les modifications qui ont été apportées sur l'algorithme de classification hiérarchique ascendante ont donné naissance à une nouvelle approche de classification appelée classification par agglomération compétitive (CA). Cette nouvelle variante combine les avantages des approches de la classification hiérarchique et ceux de la classification adaptative.

Le quatrième chapitre illustre une étude comparative entre les méthodes de classification par C-Moyennes.

Dans ce chapitre nous présentons une nouvelle approche de classification automatique non supervisée sous la famille des C-Moyennes. Cette nouvelle approche basée sur la fusion des théories du flou et des possibilités, permet d'une part de résoudre simultanément le problème de coïncidence et du bruit et d'autre part d'accélérer la classification [28].

Pour montrer les performances de la nouvelle approche par rapport aux autres, des tests ont été réalisés sur la base de données Iris et sur l'image des textures.

Enfin, nous terminons ce mémoire par une conclusion générale en indiquant quelques perspectives.

Chapitre I

Introduction Au Raisonnement Approximatif

I.1 Introduction

Dans le cadre de la logique classique une, proposition est soit vraie, soit fausse, soit inconnue ou indéterminée par apport à un corps de connaissances.

Dans la réalité, le raisonnement humain s'appuie fréquemment sur des connaissances inexactes, incertaines ou encore dont l'expression verbale est elle-même entachée d'imprécision.

L'adjectif « incertain » s'applique à des éléments de connaissances dont la valeur de vérité est connue avec plus au moins de précision; c'est le cas de l'énoncé: prendre l'avion doit être une bonne solution. L'affirmation dans cet énoncé est d'associée un certain degré de confiance.

Le terme d'« imprécis » s'applique à des éléments de connaissances dans le contenu est imprécis comme dans la proposition: La température est de 35^0 de façon floue.

Le présent chapitre traite des différentes techniques de traitement de l'information:

- Incertaine en s'appuyant sur les théories des probabilités, des possibilités et de nécessités;
- Imprécise à l'aide de la logique floue.

I.2 La Théorie des probabilités

La notion de probabilité a été introduite aux environs du dix-septième siècle par Pascal et Fermat afin de formaliser le concept de l'incertain. Le concept de probabilité est basé sur la fréquence d'occurrence d'un événement et permet de raisonner dans un monde ou

l'observation expérimentale vient s'ajouter à la connaissance a priori de la fréquence d'occurrence d'événements.

Logique (point de vue sémantique)	Théorie des ensembles classiques (point de vue évènement)
1. V proposition toujours vraie. 2. F proposition toujours fausse. 3. PROP ensemble des propositions. 4. p et q propositions élémentaires. 5. Négation. 6 Disjonction. 7. Déduction logique.	1. X référentiel (événement toujours présent). 2. Ø ensemble vide (événement toujours absent). 3. P(X) ensemble des parties de X (singleton) 4. A et B événement de P (X). 5. Complémentation (contraire). 6. Intersection. 7. Inclusion.

Tableau I.1: Parallèle entre proposition et ensemble.

I.2.1 Mesure de probabilité

Une mesure de probabilité P est une application de P(X) ensemble des parties de X dans [0,1] satisfaisant les axiomes suivants:

- $P(X)=1$;
- Si $A \subset X$ et $B \subset X$ et $A \cap B = \varnothing$ alors, $P(A \cup B) = P(A) + P(B)$.

Remarque:

Si q est une conséquence logique de p (déduction logique), alors $P(p) \leq P(q)$. De point de vue ensembliste, il est possible d'établir un parallélisme entre proposition logique et ensemble (voir tableau I.1). En effet il existe un isomorphisme entre proposition logique et ensemble, qui conserve les probabilités, ce qui permet d'assimiler la probabilité d'un évènement à celle de la vérité de la proposition qui atteste de la réalisation de cet événement.

I.2.2 Propriétés des probabilités

Il en résulte des probabilités les propriétés suivantes:

1. $P(\varnothing) = 0$
2. Si $A \subseteq B$ alors, $P(A) \leq P(B)$
3. $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
4. Si $A \cap B \neq \varnothing$ alors, $P(A \setminus B) = P(A) - P(B)$
5. Si $A \subset B$ et $B \subset A$ alors, $P(A) = P(B)$
6. $P(\overline{A \cup B}) = P(\overline{A} \cap \overline{B})$

7. $P(\overline{A \cap B}) = P(\overline{A} \cup \overline{B})$

8. $P(A)+P(\overline{A}) = 1$

I.2.3 Limites de l'approche probabiliste

a) La contrainte imposée sur les valeurs des probabilités qui indique que la somme des probabilités d'occurrence des différents événements soit égale à l'unité rend très difficile l'évaluation d'une base de connaissance.

b) Il est impossible de représenter l'ignorance d'occurrence d'un événement; l'exemple suivant illustre ce point:

Soit la règle experte suivante: Si A alors B.

Si on attribut une confiance de 0.5 à cette règle, cela signifie qu'on accorde une valeur de 0.5 à la probabilité conditionnelle P(B/A), mais cela ne nous permet pas du tout qu'on peut accorder la confiance à la règle: Si A alors $\overline{B}$, ce qui laisserait entendre $(P(B/A)+ P(\overline{B}/A)) = 1$.

Ce résultat est qualifié de paradoxe de l'ignorance totale. La seule façon d'exprimer l'ignorance sur $\overline{B}$ serait d'écrire:
$0 \leq P(\overline{B}) \leq 1$; ce qui n'apporte aucune information.

Cependant, d'autres théories ont été proposées afin de résoudre certains problèmes tels que le recouvrement, la modélisation de l'ignorance totale, la représentation et l'exploitation des connaissances approximatives.

I.3 La logique floue

Dans le paragraphe (§1), on aborde la théorie des probabilités qui est destinée à manipuler les incertitudes liées à la notion d'occurrence d'événement précis. Si cette approche est adaptée à ce type particulier d'incertitude son cadre est cependant trop limitatif pour rendre compte de l'ensemble des imperfections.

En effet la théorie des probabilités n'est pas adaptée à la gestion des connaissances de nature imprécises, d'autre part l'incertain peut provenir à partir des connaissances subjectives d'un expert humain, où l'on ne dispose d'aucune information de nature fréquentielle. Face à ces imperfections Lotfi. A. Zadeh a introduit les deux théories suivantes:

- La théorie des sous-ensembles flous [1], [3] pour modéliser l'imprécision de l'information;
- La théorie des possibilités [2], [13] pour gérer les incertitudes d'événement de nature non probabiliste qui peuvent être précis ou imprécis.

Ces deux théories constituent le cadre de base de la logique floue.

I.3.1 Théorie des sous-ensembles flous

La théorie des sous ensembles flous à été introduite par L. Zadeh en 1965[3] pour traiter le raisonnement en présence de connaissances imprécises, soit parce qu'elles sont exprimés en langage naturel par un observateur qui n'éprouve pas le besoin de fournir plus de précision (par exemple: à 100m du marché), ou n'en est pas capable (proche du marché), soit parce qu'elles sont obtenues avec des instruments d'observation qui produisent des erreurs de mesure.

La théorie des sous-ensembles flous offre un nouveau cadre pour aborder l'imprécis, elle permet de représenter l'imprécision à l'aide d'une fonction d'appartenance à un ensemble flou de telle sorte qu'elle autorise des éléments à n'appartenir complètement ni à une classe ni à une autre ou encore à appartenir partiellement à chacune.

I.3.1.1 Définition d'un sous-ensemble flou

Soit X un ensemble dénombrable ou non dénombrable qui représente l'univers de référence.

Un sous-ensemble flou A de X est défini par une fonction F_A à valeur dans [0,1], $F_A(x)$ est le degré d'appartenance de $x \in X$ à A.

$$F_A : X \rightarrow [0,1]$$
$$x \rightarrow F_A(x)$$

Prenons par exemple du domaine X des tailles possibles pour un homme. La proposition «Ali est de grande taille» contient le prédicat imprécis 'grand'. On peut alors considérer que l'ensemble des tailles qualifiées de grandes est un sous-ensemble flou défini par une fonction du domaine des tailles dans l'intervalle $[0,1]$. La figure (I.1) montre comment on attribut à chaque taille un degré d'appartenance à l'ensemble flou grand. Cette attribution reste une appréciation subjective laissée à la personne qui la définit.

D'après la figure (I.1), on peut définir un sous-ensemble flou A qui qualifie les grandes tailles comme étant l'ensemble des tailles supérieures à 1.7 m. En outre, on peut également définir un autre sous-ensemble flou B qui qualifie les petites tailles comme étant l'ensemble des tailles inférieures à 1.7 m.

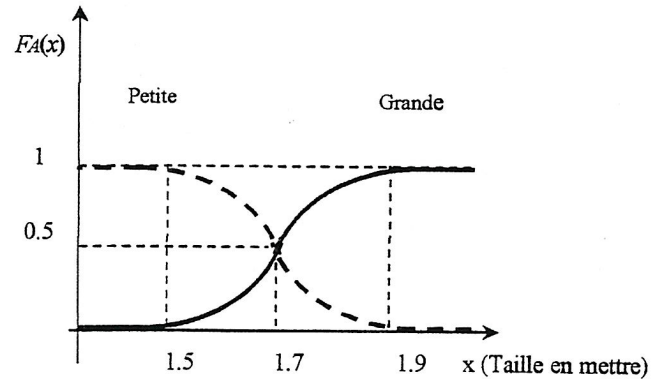


Figure I.1: Fonction d'appartenance d'une taille à l'ensemble des grandes tailles

Remarque:

Dans le cas particulier où $F_A(x)$ ne prend que des valeurs égales à 0 et 1; le sous-ensemble flou A est un sous-ensemble classique, est donc un cas particulier de sous -ensemble flou.

I.3.1.2 Caractéristiques d'un sous-ensemble flou

Dans ce paragraphe nous présentons les caractéristiques d'un sous-ensemble flou qui sont principalement liées à la forme de sa fonction d'appartenance et qui le démarquent des sous-ensembles classiques.

- ***Le support*:** est la partie de X sur la quelle le degré d'appartenance de A n'est pas nul.

  $\sup p(A)=(x\in X / F_A(x)>0)$,

- ***La hauteur*:** est la plus grande valeur prise par la fonction d'appartenance associée à A:

  $h(A)=\sup_{x\in X}(F_A(x))$, A est normalisée si $h(A)$ gale à 1.

- ***Le noyau:*** est l'ensemble des éléments de X pour lesquels la fonction d'appartenance de A vaut 1.

  $noy(A)=\{x\in X / F_A(x)=1\}$.

- ***Les α -coupes:*** est l'ensemble des éléments de X pour lesquels les degrés d'appartenances à A sont au moins égaux à α :

  $A_\alpha = \{x \in X / F_A(x) \geq \alpha\}$

La figure (I.2) illustre l'ensemble des caractéristiques sur un exemple simple:

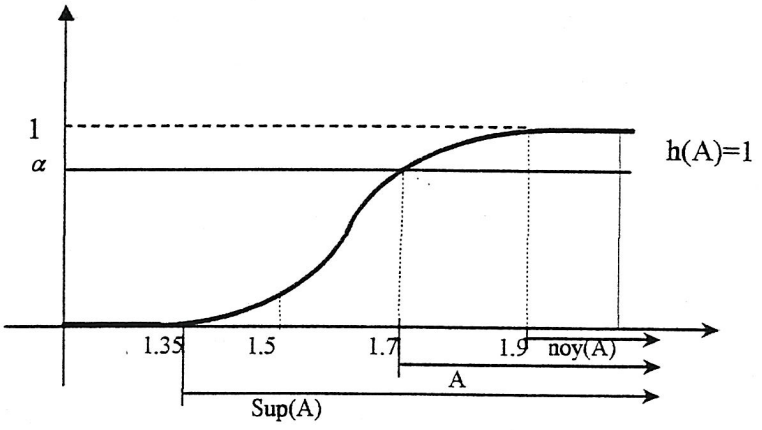


Figure I.2: Caractéristiques les plus utiles qui représentent un sous-ensemble flou.

I.3.1.3 Opérations élémentaires sur les sous-ensembles flous

Le concept de sous-ensemble flou de X comme nous l'avons signalé précédemment était une généralisation de la notion des sous-ensembles classiques de X. Les opérations sur les sous-ensembles flous décrites ci dessous sont choisies de telle sorte qu'elles sont équivalentes aux opérations classiques de la théorie des ensembles lorsque la fonction d'appartenance ne prend que les valeurs 1 et 0.

Soient A et B deux sous ensembles flous d'un même ensemble de référence X, on définit les opérations de base suivantes:

- ***Egalité:*** A et B sont dits égaux si leurs fonctions d'appartenance prennent les mêmes valeurs pour tout élément de X:

$$A = B \Leftrightarrow \forall x \in X, f_A(x) = f_B(x).$$

- ***Inclusion:*** On dit que A est inclut dans B si tout élément x de X appartient à B avec un degré d'appartenance au moins aussi grand que celui de son appartenance à A.

$$A \subseteq B \Leftrightarrow \forall x \in X, f_A(x) \leq f_B(x).$$

- ***Intersection:*** L'intersection de deux sous-ensembles flous A et B de X est le sous-ensemble flou C composé des éléments de X auquel on attribue la plus petite valeur des deux degrés d'appartenance à A et à B.

$$C = A \cap B \Leftrightarrow \forall x \in X, f_C(x) = \min(f_A(x), f_B(x))$$

Exemple: On prend l'exemple précédant (grand, petit)

$$A \cap B = \begin{cases} A \text{ si } x \in [0,1.7] \\ B \text{ si } x \in [1.7,\infty[\end{cases}$$

- ***Union:*** L'union de deux sous-ensembles flous A et B de X est le sous-ensemble flou C composé des éléments de X auquel on attribue la plus grande des deux degrés d'appartenance à A et B:

$$C = A \cup B \Leftrightarrow \forall x \in X, f_c(x) = \max(f_A(x), f_B(x))$$

Exemple: Prenons le même exemple:

$$A \cup B = \begin{cases} B \text{ si } x \in [0,1.7] \\ A \text{ si } x \in [1.7,\infty[\end{cases}$$

- ***La complémentation:*** Le complément A^c d'un sous-ensemble flou A de X est défini comme suit:

$$\forall x \in X, f_A^c(x) = 1 - f_A(x)$$

Exemple: Considérons le même exemple:
Dans ce cas: $A^c = B$

- ***Norme et conorme triangulaires:*** Il existe une multitude d'opérations que l'on peut utiliser pour mettre en œuvre les opérations d'unions et d'intersections. L. A. Zadeh a proposé, le premier, d'utiliser les opérations «min» pour l'intersection et «max» pour l'union de sous-ensembles flous comme nous l'avons défini précédemment.

Il existe deux familles d'opérations:

- Les normes triangulaires notées T-normes, qui définissent les opérations d'intersections (conjonctions).
- Les conormes triangulaires notées T-conormes qui définissent les opérations d'unions (disjonctions).

- ***Produit cartésien et projection des sous-ensembles flous:*** la description de tout système même peu complexe fait intervenir généralement plusieurs univers de référence, afin de caractériser ce système d'une manière explicite. Par exemple dans le domaine de la classification, la connaissance d'une classe d'objets est basée sur plusieurs caractéristiques définies dans différents espaces de référence pour constituer l'ensemble des vecteurs d'attributs. En effet, lorsqu'on considère plusieurs ensembles de référence simultanément on construit un univers global qui est le résultat du produit cartésien des différents univers de référence.

a. Le produit cartésien:

Soient des sous-ensembles flous $A_1, A_2, \ldots, A_n$ respectivement définis sur $X_1, X_2, \ldots, X_n$, on définit leur produit cartésien $A = A_1 \times A_2 \times \ldots A_n$ comme un sous-ensemble flou de E de fonction d'appartenance:

$$\forall x = (x_1, x_2, \ldots, x_n) \in X, f_A(x) = \min(f_{A_1}(x_1), f_{A_2}(x_2), \ldots, f_{A_n}(x_n))$$

b. La projection:

La projection d'un sous-ensemble flou A de $X_1 \times X_2 \times \ldots \times X_n$ sur $X_a \times X_b \times \ldots \times X_m$ $(m \leq n)$ est un sous-ensemble flou Proj(A) défini sur $X_a \times X_b \times \ldots \times X_m$ de fonction d'appartenance:

$$\forall x \in X_a \times X_b \times \ldots \times X_m : f_{proj(A)}(x) = \sup_{(i, 1 \leq i \leq n, i \neq a, \ldots, i \neq m)} f_A(\ldots, x_i, \ldots)$$

I.3.1.4 Mesure floue

Soit X un univers, et soient A, B, C des sous-ensembles. Notre but est d'affecter à chacun des sous-ensembles un coefficient indiquant le degré de certitude ou d'évidence avec lequel nous croyons qu'un élément x appartient à chacun de ces sous-ensembles.

Exemple:

Soit X un ensemble formé d'une association quelconque d'individus, et soient :

A: un sous-ensemble de X représentant les enfants.
B: un sous-ensemble de X représentant les jeunes.
C: un sous-ensemble de X représentant les vieux.

On attribut respectivement à chaque sous-ensemble les coefficients suivants:

0.10
0.60
0.30

Ces informations nous permettent de classer un membre donné de l'association X. Dans ce cas la valeur 0.60 attribuée à B indique le degré avec lequel nous pensons que l'individu considéré soit un jeune.

- ***Définition:***

  On appelle mesure floue, une représentation de l'incertitude attribuant à chaque sous-ensemble de l'univers X, un coefficient indiquant le degré de certitude avec lequel un élément quelconque de l'univers X appartient au sous-ensemble correspondant.

- ***Propriétés d'une mesure floue:***

  Une mesure floue est une application de P(X) dans [0,1]: f : $P(X) \rightarrow [0,1]$ avec P(X): ensemble des parties de X.

  1. f(Ø)=0 ,f(X)=1 (cas extrêmes)
  2. $\forall A, B \in X$ tels que $A \subseteq B$ alors $f(A) \leq f(B)$ (monotonie)
  3. Pour les chaînes $A_1 \subseteq A_2 \subseteq ... \subseteq A_n$, $ouA_n \subseteq A_{n-1} \subseteq ... \subseteq A1$, on a:
     $\lim_{n \to \infty} f(A_n) = f(\lim_{n \to \infty} A_n)$ (continuité).

  La monotonie de f et les propriétés classiques de la réunion et de l'intersection d'ensembles ordinaires nous permettent de déduire les relations suivantes:

  $$f(A \cup B) \geq \max[f(A), f(B)]$$

  $$f(A \cap B) \leq \min[f(A), f(B)]$$

I.3.1.5 Limites de la théorie des sous-ensembles flous

Nous avons étudié la théorie des sous-ensembles flous destinée pour manipuler les connaissances imprécises, à ce point nous remarquons que les incertitudes sur la véracité d'une information ne sont pas traitées par la théorie des sous-ensembles flous. Mais incertitude et précision sont intimement liées, notamment dans le cas où l'exigence de la précision engendre une perte de certitude et réciproquement. D'autre part la théorie des probabilités n'apporte pas des solutions dans les cas où les événements sont de nature non fréquentielles (incertitudes subjectives provient de l'appréciation d'un expert humain).

Pour ces raisons une nouvelle théorie appelée «théorie des possibilités» a été introduite par Lotfi. A. Zadeh pour répondre à ces besoins.

I.3.2 La théorie des possibilités

Apres avoir traiter l'incertain par la théorie des probabilités et l'imprécis par la théorie des sous-ensembles flous. La théorie des possibilités a été introduite pour constituer un cadre dont lequel imprécis et incertains sont conjointement lies pour manipuler des incertitudes de nature non probabilistes.

I.3.2.1 Mesure de possibilité

- ***Définition:***

  Une mesure de possibilité est une application $\Pi : p(X) \rightarrow [0.1]$ qui associe à chaque événement A un degré de confiance compris entre 0 et 1 évaluant à quel point cet événement est possible.

- ***Propriétés d'une mesure de possibilité :***

  Une mesure de possibilité vérifie les axiomes suivants:

  1. $\Pi(\phi)=0$;(Evènement impossible), $\Pi(X)=1$ (événement tout à fait possible);
  2. $\forall$ la famille d'ensemble $A_1, A_2, \ldots, A_n \in \Pi(X)$: $\Pi\left(\bigcup_{i=1,\ldots,n} A_i\right) = \sup_{i=1,\ldots,n}(\Pi(A_i))$
  3. $\forall$ (A, B) $P(X)^2$: $\Pi(A \cap B) \leq \min[\Pi(A), \Pi(B)]$.

  Cas particulier:

  Deux événements peuvent être possibles $(\Pi(A) \neq 0, \Pi(B) \neq 0)$ mais leur occurrence simultanément peut être impossible $(\Pi(A \cap B) = 0)$.
  De plus on a:

  Si $A \subseteq B$ alors $\Pi(A) \leq \Pi(B)$

De même:

Si B=Ā on a:

$$\Pi(A \cup \overline{A}) = \max[\Pi(A), \Pi(\overline{A})] = \Pi(X) = 1.$$

et par conséquent :

$$\Pi(A) + \Pi(\overline{A}) \geq 1$$

Contrairement à la théorie des probabilités qui impose la contrainte:

$$P(A) + P(\overline{A}) = 1$$

- ***Exemple:***

Soit X l'ensemble des heures comprissent entre 12 et 16 aux quelles est attendue l'arrive d'un ami mais comme l'heure de présence de cet ami n'a pas été précisée, on peut avoir différents événements évaluant à quel point il est possible et à quel point il est certain que l'ami serait présent.

Nous définissons ainsi, par exemple, 4 événements munis chacun d'une mesure de possibilité:

A = $\{12,13\} \rightarrow \Pi(A) = 1$ (tout à fait possible)
B = $\{14,15\} \rightarrow \Pi(B) = 0.8$ (relativement possible)
C = $\{15\} \rightarrow \Pi(C) = 0$ (impossible)
D = $\{15.16\} \rightarrow \Pi(D) = 0.2$ (peut possible)

On remarque bien que:

- $\Pi(A \cup B) = \max(1,0.8) = 1.$
- $\Pi(B \cap D) = \Pi(C) = 0 \leq \min(\Pi(B), \Pi(D)) = 0.2$
- $\Pi(\overline{A}) + \Pi(A) = \Pi(B \cup C \cup D) + \Pi(A) = \max(\Pi(B), \Pi(C), \Pi(D)) + \Pi(A) = 0.8 + 1 = 1.8 \succ 1$

I.3.2.2 Distribution de possibilité

- ***Définition:***

Une mesure de possibilité est totalement définie si on attribut un coefficient de possibilité à toute partie de l'ensemble de référence (X). Une distribution de possibilité est une application $\pi : X \rightarrow [0.1]$ telle que $\sup_{x \in X}(\Pi(x)) = 1$

On remarque que cette application est normalisée, on appelle également l'ensemble des éléments de X pour lesquels la distribution de possibilité π définie par:

$\forall A \in P(X)$ $\Pi(A) = \sup_{x \in A} \Pi(x)$ une mesure de possibilité.

Réciproquement toute mesure de possibilité Π affectée à toute partie de X (réduit à un seul élément) induit une distribution de possibilité Π, définie par:

$$\forall x \in X \quad \pi(x) = \Pi(\{x\})$$

- ***Exemple:***

À partir de l'exemple précédent, on définit $\pi(x)$ par:

$\pi(x) = 1$ si $x \in \{12.13\}$
$\pi(x) = 0.8$ si $x \in \{14.15\}$
$\pi(x) = 0$ si $x = 15$
$\pi(x) = 0.2$ si $x \in \{15.16\}$

Posons: A = [12,13], B = [14,16]

Il résulte: $\Pi(A) = 1$, $\Pi(B) = 0.8$

On voit qu'il est complètement possible que l'ami soit présent entre 12^h et 13^h.

I.3.2.3 Mesure de nécessité

Une possibilité donne une information sur l'occurrence d'un événement A mais ne suffit pas pour décrire complètement l'incertitude sur l'occurrence de cet événement. Pour compléter la connaissance sur A, on fait intervenir une autre mesure appelée également «mesure de nécessité».

Une mesure de nécessité est le duale d'une mesure de possibilité qui attribue un coefficient compris entre 0 et 1 à toute partie de X indiquant à quelle mesure la réalisation de A est certaine.

- ***Définition:***

Une mesure de nécessité est une fonction définie sur l'ensemble P(X) des parties de X à valeur dans [0,1], telle que :

i. N(Ø)=0, N (X)=1;

ii. $\forall A_1 \in P(X), \forall A_2 \in P(X), \ldots, \forall A_n \in P(X) : N\left(\bigcap_{i=1,\ldots,n} A_i\right) = \inf_{i=1,\ldots,n} A_i$.

- ***Conséquences:***

Les propriétés i) et ii) entraînent la monotonie de toute mesure de nécessité N c'est à dire si deux sous-ensembles sont tels que:
$A \subseteq B$ Alors $N(A) \leq N(B)$, en particulier, $N(A \cap B) \leq N(A)$.

$\forall(A,B)\in P(X)^2 \quad N(A\cap B)\geq \max(N(A),N(B))$

- ***Propriétés d'une mesure de nécessité :***

Tout sous-ensemble A de X vérifie :

1. $\min(N(A),N(A^c))=0 \quad (N(\phi)=0\ ,\ N(X)=1)$.
2. $N(A)+N(A^c)\leq 1$.

I.3.2.4 Dualité entre mesure de possibilité et de nécessité

Une mesure de possibilité $\Pi(A)$ traduit le degré avec lequel l'évènement A est susceptible d'être réalisée, alors que N(A) indique le degré de certitude que l'on peut attribuer à cette réalisation. En d'autre terme la réalisation d'un évènement A est certain (N(A) =1) si et seulement si celle de son complémentaire A^c est impossible $\Pi(A^c)=0$, donc: $\Pi(A)=1$, ce qui est exprimé par:

$$\forall A\in P(X): N(A)=1-\Pi(A^c).$$

I.3.2.5 Conséquences

D'après la propriété ($\Pi(A^c)+\Pi(A)\geq 1$) et la relation précédente on obtient:

- $\Pi(A)\geq N(A)$.

Il en résulte qu'un évènement certain (N(A) =1) est celui qui est complètement possible ($\Pi(A)=1$) et dont le complémentaire est impossible ($\Pi(A^c)=0$).

De même :

- $\max(\Pi(A),1-N(A))=1$
- Si $N(A)\neq 0$, alors $\Pi(A)=1$
- Si $\Pi(A)\neq 1$, alors $N(A)=0$

La deuxième propriété indique que tout évènement A un peu certain ($N(A)\neq 0$) est tout à fait possible ($\Pi(A)=1$), tant que la troisième propriété montre qu'on ne peut avoir le moindre de certitude ($N(A)=0$) sur un évènement relativement possible($\Pi(A)\neq 0$).

Exemple:

Reprenons l'exemple précédent : A=$\{12,13\}$, B=$\{14,15\}$

- N(A) =1-$\Pi(A^c)$=1-0.2 = 0.8 (parce que A est tout à fait possible).
- N(B) = 0 = 1-$\Pi(B^c)=1-1=0$(parce que B est relativement possible).

Il est donc certain que l'ami arrive à l'une des heures indiquées dans l'ensemble A.

I.4 Comparaison entre possibilité et probabilité

En matière de probabilité P(A) =1 signifie que A est certain. Ici $\Pi(A)=1$ ne signifie pas que A est certain car l'évènement complémentaire (A^c) peut aussi avoir une possibilité de 1 si la nécessité de A est nulle, en revanche N(A) =1 qualifie un évènement certain. D'autre part, si on dispose de la probabilité d'un évènement, on peut facilement déterminer celle de sa contraire (P(A)+P(A^c) =1), par contre l'axiome $\Pi(A \cup B) = \max[\Pi(A), \Pi(B)]$ qui implique $\max(\Pi(A), \Pi(A^c)) = 1$ ou $(\Pi(A) + \Pi(A^c)) \geq 1$ n'est autre qu'une liaison faible entre A et A^c.

En conclusion, pour caractériser l'incertitude sur A, on dispose des deux limites inférieures et supérieures [N (A), $\Pi(A)$], donc de plus grande souplesse pour modéliser l'incertain; dans ce cas la probabilité d'un évènement se situe à l'intérieur d'un intervalle [N(A), $\Pi(A)$] et celle de son contraire dans l'intervalle [1-$\Pi(A)$,1-N(A)].

I.5 Conclusion

Dans ce chapitre nous avons étudié quelques méthodes pour la représentation des connaissances incertaines ou imprécises ou incertaines et imprécises conjointement, ces méthodes constituent une introduction au raisonnement approximatif.

L'une des applications des possibilités et du flou est la génération d'un système expert capable de prendre une décision finale. Un système de classification des données est une des applications des théories étudiées.

Dans le chapitre qui suit, nous allons étudier la notion de classification automatique non supervisée et les différentes méthodes destinées à la classification en se basant sur les théories étudiées précédemment.

Chapitre II

La Classification Automatique Non Supervisée

II.1 Introduction

L'objectif principal d'une méthode de classification automatique est de répartir les éléments d'un ensemble en groupes c'est-à-dire d'établir une partition de cet ensemble. Différentes contraintes sont imposées; chaque groupe devant être le plus homogène possible, et les groupes devant être les plus différents possibles entre eux.

Deux grandes familles de classification automatique existent:

- Classification non supervisée;
- Classification supervisée.

On utilise une classification non supervisée lorsque on ne dispose d'aucune information concernant une appartenance d'un objet à une classe, on essaie alors de regrouper les individus (objets) en classes les plus homogènes possible de telles sorte que les distances entre les individus à l'intérieure d'une même classe soient le minimum possible et que simultanément les centres des classes générés devant être les plus éloignés possibles, le terme « clustring » désigne cette opération.

Il existe des algorithmes de classification composés de plusieurs itérations permettant de créer des regroupements d'objets ayant des caractéristiques similaires. Généralement ces algorithmes sont partagés en trois familles principales:

- Les algorithmes agrégatifs, qui correspondent aux méthodes dites de classification ascendante hiérarchique.
- Les algorithmes dichotomiques, qui correspondent aux méthodes dites de classification descendante hiérarchique.
- Les algorithmes de partitionnement, qui ne fournissent pas des hiérarchies mais simplement des partitions.

Contrairement, dans la classification non supervisée, on dispose des connaissances à priori sur l'appartenance de quelques objets à leurs classes ce qui permet de définir des sites d'apprentissage.

Dans ce mémoire nous nous intéressons à la classification non supervisée, pour détailler ce type de classification, nous donnons quelques connaissances de base.

II.2 Connaissances de base

II.2.1 Tableau individu-variable

Les variables sont supposées ici quantitatives, le tableau individu-variable est un tableau rectangulaire, noté ici X dont les lignes sont associées aux individus (éléments sur lesquels sont effectuées des mesures) et dont les colonnes sont associées aux variables. L'élément (i,j) de X noté x(i,j), est égal à la valeur prise j(j=1,…,p) par la variable pour l'individu i (i=1,…,n).

Remarque:

- Le transposé de la i-éme ligne de X est noté x_i, contient l'ensemble de mesures ou valeurs de variable pour cet individu appelé vecteur d'attribut, ce vecteur doit être aussi discriminant que possible afin de distinguer et de différencier les différents groupes présents dans l'ensemble d'individus (objets).
- La j-ème colonne x^j de X est la j-ème variable (ensemble des n valeurs prises par la j-ème mesure sur les individus).
- Si toutes les variables sont quantitatives, le tableau X est une matrice de nombres réels.

$$X = \begin{bmatrix} x(1,1) & \dots & x(1,j) & \dots & x(1,p) \\ \dots & \dots & \dots & \dots & \dots \\ x(i,1) & \dots & x(i,j) & \dots & x(i,p) \\ \dots & \dots & \dots & \dots & \dots \\ x(n,1) & \dots & x(n,j) & \dots & x(n,p) \end{bmatrix} = \begin{bmatrix} X^1 & \dots & X^j & \dots & X^p \end{bmatrix} = \begin{bmatrix} {}^tX_1 \\ . \\ {}^tX_i \\ . \\ {}^tX_n \end{bmatrix}$$

II.2.2 Distances

Un des objectifs de classification des objets est de décrire la proximité entre les individus (existe-il des groupes d'objets semblables qui se différencient d'autres groupes d'individus semblables ?), dans ce cas, il s'agit de mesurer la similarité des deux vecteurs d'individus.

Selon un grand nombre de critères, la comparaison des objets à classifier se ramène à une mesure de distance. Dans notre cas on dispose d'un tableau individus-variable X et on désire calculer une matrice de distance entre les individus. Trois distances de base peuvent être utilisées pour trois types de données de base:

- La distance euclidienne classique pour les données quantitatives;
- La distance du Chi2 pour les tableaux de dénombrements;
- L'indice de Jaccard pour les tableaux de présence-absence.

a. Distance euclidienne (variables quantitatives)

La distance euclidienne peut être utilisée pour calculer la distance entre deux individus sur lesquels on a mesuré des variables numériques quantitatives, si X(i,j) est la valeur de la variable j mesurée sur l'individu i, la distance entre les individus i et ì, est $d^2(i, ì)$:

$$d^2(i,ì) = \sum_{j=1,p} (x_{ij} - x_{ìj})^2 \qquad \text{(II.1)}$$

b. Distance du Chi2 (Tableau de contingences)

La distance de Chi2 induit une division par les effectifs marginaux qui lui permet de se ramener à des comparaisons de profils. Si $x(i,j)$ est l'effectif de la classe situé à la ligne i et à la colonne j de la table de contingence, la distance entre les lignes i, ì et $d^2(i, ì)$ est:

$$d^2(i,\ ì) = \sum_{j=1,p} \frac{1}{x_j \left(\frac{x(i,j)}{x_i} - \frac{x(ì,j)}{x_ì} \right)^2}$$

Avec $x_i = \sum_{j=1,p} x(i,j) \qquad x_ì = \sum_{j=1,p} x(ì,j) \qquad x_j = \sum_{i=1,n} x(i,j)$

c. Indice de Jaccard (tableau de présence- absence)

L'indice de Jaccard et un indice de ressemblance entre deux relevés écologiques, on note:

c le nombre d'espèces présentes dans les deux relèves simultanément.

p et q les effectifs d'espèces présentes dans chacun des deux relevés.

L'indice de Jaccard est $s = c/p+q-c$, il s'agit d'un indice de ressemblance, et on peut déduire facilement une mesure de distance d = 1-s.

Si deux espèces sont identiques, on a d = 0, et si deux relevés sont complètement différents (aucun espèce en commun), on a d =1.

II.2.3 Tableau de distance

Un tableau de distance est une matrice carrée symétrique à éléments positifs d'ordre n qui a pour élément (i,k) la distance entre les objets i et k. On note souvent Δ un tel tableau de distances.

a. Remarque

Le vecteur donnant les p mesures effectuées sur un même objet i est le vecteur individu (appelé souvent vecteur d'attribut) :

$X_i = (\mathrm{x(i,j)},\ j = 1,2,...,p)$.

Si les mesures sont quantitatives, le vecteur X_i est un vecteur de $\Re^p$, on munit l'espace vectoriel $\Re^p$ de la distance euclidienne usuelle (II.1) et l'élément (i, k) de Δ s'écrit:

$$\Delta_{ik} = \sqrt{\sum_{j=1}^{p} (x(i,j) - x(k,j))^2} \qquad \text{(II.2)}$$

La formule ci-dessus permet de passer d'un tableau individus-variables à un tableau de distance.

b. Intérêt

Certaines méthodes de classification utilisent comme tableau d'entrée un tableau de distance plutôt qu'un tableau individu-variable. De plus, il existe des cas où on dispose d'un tableau de distance, sans disposer d'un tableau individu-variable.

c. Passage du tableau de distance à un tableau individu-variable

Dans certaines applications, on calcule pour chaque paire d'objet un indice de distance par une formule définie par l'utilisateur et mesurant une dissemblance entre les objets, sans savoir a priori si cela mène à une vraie mesure de distance ou à fortiori, si cela produit un tableau de distance euclidienne. Si on peut prouver qu'un tableau de distance Δ est euclidien, on peut trouver un tableau individus-variables tel que la formule (II.2) redonne le tableau Δ.

II.2.3 Equivalence entre partition et variable catégorielle

La donnée d'une variable catégorielle ou d'un facteur f défini sur un ensemble X d'individus et à valeurs dans un ensemble fini, dont les éléments sont appelés modalités, est équivalent à la donnée d'une partition: une classe est alors définie comme l'ensemble des individus qui sont associés à une même modalité.

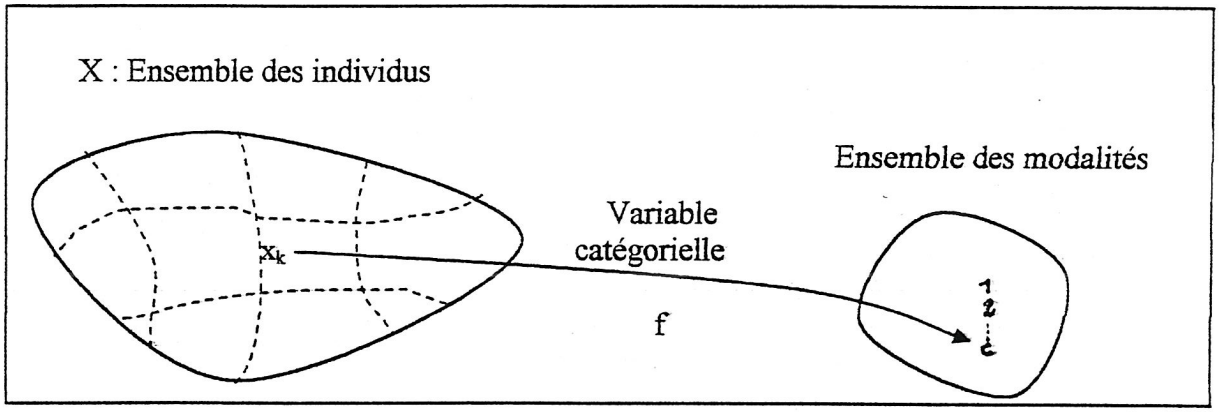


Figure II.1: Équivalence variable catégorielle-partition

II.3 Approches de classification automatique non supervisée

Les méthodes de classification automatique non supervisées visent à répartir des objets, dont on ne sait pas leurs appartenances, en classes homogènes telles que:

- Toutes les classes soient disjointes;
- L'union de toutes les classes soit l'ensemble de référence.

Plusieurs approches de classification automatique non supervisées ont été proposées dans la litérature pour obtenir des classes les plus naturelles possibles.

II.3.1 Approche classique

Dans le modèle classique de classification, l'affectation d'un objet à une classe est effectuée de manière «catégorique ». Cependant, les informations nécessaires à cette prise de décision sont la plupart du temps incomplètes. En effet, dans le cas où les distances entre le

point et les différentes classes candidates sont voisines le choix définitif d'une classe peut s'avérer difficile. En outre, si le choix de la classe est mauvais, la nature itérative du traitement fait que cette erreur se propage, ce qui ralentit la convergence de l'algorithme de classification.

Comment qualifier et comment quantifier cette information incomplète ?

Cette question se rapporte essentiellement à la nature incertaine ou imprécise de notre règle de décision. La décision n'est pas imprécise, car le résultat est le numéro de la classe connu avec précision. En effet si un objet est affecté précisément à la classe C_1, le choix de cette classe n'est pas absolument certain, il pourrait, par exemple appartenir en réalité à la classe C_2.

Comment prendre en compte cette incertitude ?

Nous nous plaçons dans le cadre de la théorie des sous-ensembles flous afin de gérer conjointement des informations imprécises et incertaines.

II.3.2 Approche floue

La classification floue se distingue de la classification classique principalement par des améliorations du taux d'identification pour les situations ambiguës situées aux frontières et intersections des classes, zones qui sont par essence celles où les erreurs de classification sont les plus fréquentes.

Cette amélioration est assurée par l'introduction de la notion d'appartenance partielle d'un objet aux différentes classes, cependant cette approche est sensible aux bruits [4] qui influent directement sur l'appartenance et finalement entraînent une mauvaise classification.

Pour remédier à cet inconvénient, une autre approche a été proposée par krishmapuran et Kaller, basée sur la théorie des possibilités.

II.3.3 Approche possibiliste

Dans cette approche, la contrainte d'inspiration probabiliste imposée sur les degrés d'appartenance dans l'approche floue a été brisée et dans ce cas le degré d'appartenance d'un objet à une classe est un degré d'appartenance absolu, contrairement à celui dans l'approche floue qui est relatif.

II.4 Exemple de classification

Si l'on considère le cas de deux clases (C_1, C_2):

- Avec la classification classique, l'information concernant un objet frontière (voir figure II.2) est notée symboliquement par le vecteur (1,1), alors qu'un objet proche de la frontière (voir figure II.3) est notée par (0,1). Ceci ne constitue pas une information significative, et on remarque qu'il existe une grande perte d'information.

- Avec la classification floue on peut représenter symboliquement, un objet frontière appartenant autant à une classe qu'à une autre (voir figure II.2), par le vecteur (0.5, 0.5). Si un objet presque frontière (voir figure II.3) appartenant presque autant à une classe qu'à une autre, alors on peut lui attribuer le vecteur (0.49, 0.51).

- Dans le cas de la classification possibiliste, un objet frontière est modélisé de la même façon que celle de la classification floue, par contre un objet proche de la frontière peut être représenter par le vecteur (0.49, 0.52) sans tenir compte de la contrainte d'inspiration probabiliste.

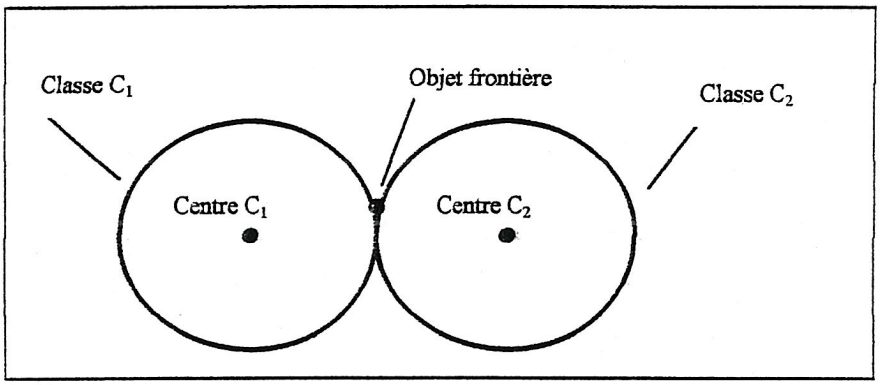


Figure II.2: Représentation symbolique d'un objet frontière entre deux classes.

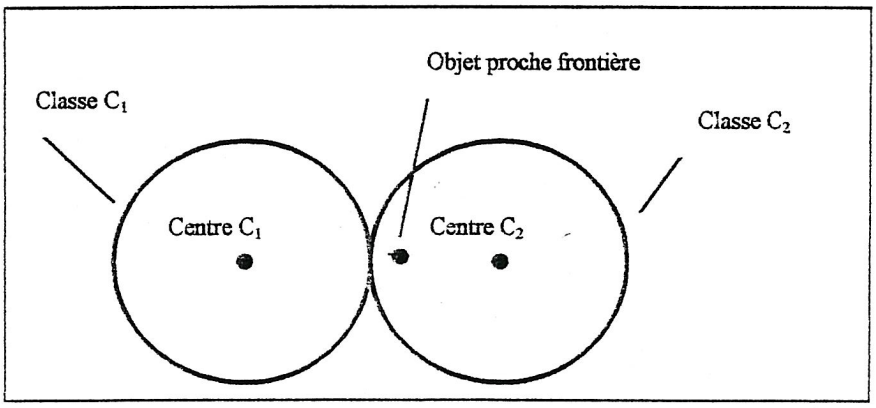


Figure II.3: Représentation symbolique d'un objet proche de la frontière entre deux classes.

II.5 Conclusion

Dans ce chapitre nous avons présenté les différents types de données d'entrées d'un classifieur qui peuvent être un tableau individus-variables ou un tableau de distance.

Les variables constitutives d'un tableau individus-variables sont soient: de nature quantitative, un tableau de dénombrement ou tableau de présence-absence.

Après avoir défini le critère de similarité qui permet de mesurer la distance, qui change selon le type de données entre les individus à classifier, nous avons aussi décrit les différentes approches sur lesquelles se greffe un système de classification automatique non supervisée, à savoir, l'approche classique basée sur la notion de partage rigide, l'approche floue et possibiliste qui sont basées sur la notion de partage partiel.

Dans le chapitre suivant, nous présenterons deux types d'algorithmes non supervisés : la famille de C-Means et les méthodes hiérarchiques.

Chapitre III

Méthodes De Classification Automatique Non Supervisée

III.1 Introduction

Les méthodes de classification ont pour but de regrouper les individus en classes homogènes. On distingue deux grands types de méthodes de classification : celles dites non hiérarchiques qui produisent une partition des individus en un nombre fixe de classes et celles dites hiérarchiques qui fournissent une suite de partitions en « classes de plus en plus vastes ».

Les méthodes de classification non hiérarchiques conduisent à une partition de l'ensemble de départ en un nombre C de classes de même niveau. Elles permettent de traiter des ensembles d'effectifs assez élevés en optimisant des critères pertinents.

La Classification hiérarchique produit des suites de classes emboîtées qui définissent une hiérarchie. À chaque étape, les deux classes les plus proches sont recherchées et fusionnées jusqu'à ce qu'il n'y ait plus qu'une seule classe. Cette méthode consiste à fournir un ensemble de partitions plus ou moins fines, obtenues par les regroupements successifs des parties [5].

Dans ce chapitre nous allons décrire en premier lieu une méthode de classification automatique non supervisée non hiérarchique appelée C-moyennes. Ensuite, nous présentons deux autres méthodes de classifications non supervisées hiérarchiques ascendantes (CAH) et hiérarchiques descendantes (CDH). En dernier lieu, nous donnons une troisième variante appelée méthode d'agglomération compétitive (CA), qui combine les techniques des C-moyennes et hiérarchiques.

Nous avons vu dans le chapitre précédant que les données d'entrées d'un classifieur sont de l'un des deux types suivants:

- Un tableau individus-variables quantitatifs et une distance permettant de calculer la dissemblance entre deux vecteurs individus. Lorsque les variables sont qualitatives, on utilise une méthode particulière (classification automatique à partir des variables qualitatives).

- Un tableau de distances entre individus.

Notre but est de trouver une partition des objets qui optimise un critère qui tend:

➢ À ne regrouper deux objets qui sont très semblables.
➢ À ne séparer que des objets qui sont suffisamment différents.

La démarche consiste dans un premier temps à choisir:

✓ Un critère de qualité d'une partition mesurant la qualité d'une partition.
✓ Un algorithme qui tend à trouver une partition qui optimise le critère (en général le nombre de partitions est fixé à priori).

III.2 Critère de qualité d'une partition

Une bonne partition doit avoir des classes homogènes, distinctes et compactes. En général, on détermine une partition de l'ensemble des individus en C-classes par l'intermédiaire des ensembles d'indices des groupes qui sont représentés par les centres de gravité.

On peut démontrer que pour une bonne partition (i.e : classes compactes dans les centres sont très éloignées les uns des autres), les algorithmes de classification automatique non supervisée possèdent la propriété d'optimiser le moment d'ordre 2 de la partition finale [5].

Soit: $X = \{x_1, x_2, ..., x_2\} = X_1 \cup X_2 \cup ... \cup X_1$ une C-partition quelconque de X avec:

$V = \left\{ v_i = \frac{\sum_{x_k \in X_i} x_k}{n_i} \right\}$, est l'ensemble des C-partition, avec n_i : l'effectif de la classe i (nombre d'objets dans la classe).

$\overline{V} = \frac{\sum_{x_k \in X} x_k}{n}$, est le centre de données X.

Le moment centré d'ordre 2 de l'ensemble X vaut:

$$M^2\left(\frac{X}{\overline{V}}\right) = \sum_{x_k \in X} (x_k - \overline{V})(x_k - \overline{V})^t = C_X.$$

Pour une partition donnée, on peut décomposer cette inertie par le théorème de Huygens:

$$M^2\left(\frac{X}{a}\right) = M^2\left(\frac{X}{\overline{V}}\right) + (\overline{V} - a)(\overline{V} - a)^t$$

Donc pour une partition Q de X, et q une classe de Q:

$$M^2\left(\frac{X}{\overline{V}}\right) = \sum_{q\in Q}\sum_{k\in q}(x_k - \overline{V})(x_k - \overline{V})'.$$

$$M^2\left(\frac{X}{\overline{V}}\right) = \sum_{q\in Q}\left[M^2\left(\frac{q}{V_q}\right) + (V_q - \overline{V})(V_q - \overline{V})'\right]$$

$$M^2\left(\frac{X}{\overline{V}}\right) = \sum_{q\in Q} M^2\left(\frac{q}{V_q}\right) + M^2\left(\frac{Q}{\overline{V}}\right)$$

$$M^2\left(\frac{X}{\overline{V}}\right) = \text{intra} + \text{inter} = C_X$$

On a alors:

Inertie totale = Inertie inter–classe + Inertie intra-classe.

Le moment d'ordre 2 qui représente la dispersion totale est une fonction de X, donc il ne dépend que de la partition générée, autrement dit, il est constant.

Ainsi pour ce critère, chercher une partition qui maximise l'inertie inter-classe (qui tend à disperser mieux les groupes), revient à chercher une partition minimisant l'inertie intra-classe (qui tend à obtenir des groupes plus compacts). La valeur du critère dépend du nombre de classes; l'inertie inter-classe est maximale pour la partition discrète, ayant comme classes les singletons et nulle pour la partition grossière ayant comme classe unique l'ensemble de données. On est donc amené à maximiser l'inertie pour un nombre de classe fixé à priori.

III.3 Les méthodes de classification automatique non supervisée par C-Moyennes (C-Means)

Les méthodes des C-Means, cherchent à trouver une partition de l'ensemble de données en affectant chaque objet à la classe la plus proche. La décision d'appartenance d'un objet à une classe ou à une autre se réalise à l'aide d'une mesure de distance entre cet objet et les classes candidates. En effet, chaque classe doit être identifier par un prototype (centre de classe) qui le différencie des autres classes. Les méthodes des C-Means cherchent à trouver ces prototypes représentatifs de l'ensemble d'objets de telle sorte que la partition finale minimise le critère de qualité.

Il existe un grand nombre de variantes d'algorithmes des C-Means qui permettent de donner une partition de l'ensemble d'objets, et qui se démarquent par les techniques de réaffectation des objets aux différentes classes ou par la méthode de calcul de distance.

Les algorithmes des C-Means sont des algorithmes itératifs (iterative relocation algorithms) dont les différentes étapes sont les suivantes:

- Choix d'une partition initiale;
- Calcul des centres de gravité des classes de la partition initiale;

- Réaffectation des objets à la classe dont le centre de gravité est le plus proche.

On itère cet algorithme en recalculant les centres de gravité des classes à chaque étape et en réaffectant les objets jusqu'à ce qu'aucun objet ne change de classe.

On distingue trois types d'algorithmes de classification automatique non supervisée par C-Means : classique, floue et possibiliste.

III.3.1 Algorithme des C-Moyennes classiques « Hard C-Means: HCM »

L'algorithme des C-Moyennes classiques cherche à trouver une partition de l'ensemble de données de telle sorte qu'un objet ne peut appartenir qu'à une seule classe. Soit $X=\{x_1,x_2,...x_n\}\subset R$, un ensemble d'objets à partitionner qui est représenté par un tableau individus-variables noté:

$$X=\begin{bmatrix} x(1,1) & \dots & x(1,j) & \dots & x(1,p) \\ \dots & \dots & \dots & \dots & \dots \\ x(i,1) & \dots & x(i,j) & \dots & x(i,p) \\ \dots & \dots & \dots & \dots & \dots \\ x(n,1) & \dots & x(n,j) & \dots & x(n,p) \end{bmatrix} \in \Re^n \times \Re^p$$

Soient: $v_1,v_2,...,v_c$ les C prototypes des classes qui caractérisent les C groupes ($c_1,c_2,...,c_c$) générés par l'algorithme.

Si on considère $U_{ik}=ui(x_k)$ une valeur qui modélise l'appartenance de l'objet x_k à la classe C_i, la partition classique donne une partition dure de l'ensemble X telle que:

$$\begin{cases} \forall i,k \quad 1\le i\le c, 1\le k\le n & u_{ik}\in\{0,1\} \\ \forall i,j \quad 1\le i\le c, 1\le j\le c, i\ne j, & (c_i\cap c_j=\phi) \\ \bigcup_{i=1}^{c} c_i=X \end{cases}$$

La fonction objective à minimiser (Within Group Sum of Squared Errors: **WGSS**) est définie de la manière suivante:

$$j_1(U,V;X)=\sum_{i=1}^{c}\sum_{x_k\in c_i}\|x_k-V_i\|^2$$

Dans ce cas, la matrice d'appartenance et les centres des classes sont déterminés par les relations suivantes:

$$U_{ik}=\begin{cases} 1 & \text{si} \quad d_{ik}^2=\|x_k-V_i\|^2=\min_{1\le j\le c}\{d_{jk}^2\} \\ 0 & \text{sinon} \end{cases} \qquad \text{(III1)}$$

$$v_i=\frac{\sum_{k=1}^{n} u_{ik}x_k}{\sum_{k=1}^{n} u_{ik}}$$

Ainsi, on obtient des groupements représentatifs des classes effectives :

$$c_i=\left\{x_k \in X / d_{ik}^2=\min_{1\le j\le c}\left\{\|x_k-V_j\|^2\right\}\right\}$$

III.3.1.1 Étapes de l'algorithme des C-Moyennes classiques

$X=\{x_1,x_2,...x_n\}\subset R^p$: un ensemble de données à partitionner.

1. **Initialisation**
   - Fixer un nombre de classes c : $2\le c\le n$;
   - Choisir une métrique de distance d ;
   - Initialiser le compteur d'itérations l=0 ;
   - Initialiser la matrice $U_{(0)}$;
   - Fixer le nombre ε >0.

2. **Adaptation des centres**
   - Calculer les C centres de classes $\{v_{i(l)}\}$ à l'aide de la formule suivante:

$$V_{i(l)}=\frac{\sum_{x_k\in ci} u_{ik(l-1)}x_k}{|X_i|} \text{ avec } |X_i|=\sum_{k=1}^{n} u_{ik}\ 1\le j\le c$$

3. **Évaluation des regroupements**
   - Mettre à jour $u_{ik(l)}$ à l'aide de la relation (III.1)

4. **Test d'arrêt**
   - Comparer $u_{ik(l-1)}$ et $u_{ik(l)}$: si $\|u_{ik(l)}-u_{ik(l-1)}\|<\varepsilon$ Alors **STOP**
     Sinon $l\leftarrow l+1$; Go to **(2)**
     Fin si

III.3.1.2 Remarques

1. La fonction objective à minimiser correspond à la distance intra-classe.

2. L'initialisation de l'algorithme peut être faite par:

   - Le choix d'une matrice d'appartenance qui doit vérifier la contrainte d'inspiration probabiliste ; dans ce cas, la seconde étape sera l'adaptation des prototypes.

   - Le choix des C-centres qui doivent être éloignés les uns des autres; dans ce cas, la seconde étape sera l'évaluation des regroupements.

3. On notera que les résultats changent selon le choix des conditions initiales.

4. L'algorithme minimise l'inertie intra-classe, mais son minimum dépend de la partition initiale, on n'est donc pas certain d'obtenir le minimum absolu. Cet inconvénient et commun à tous les algorithmes qui suivent le même principe. Une façon de remédier à ce problème est d'exécuter plusieurs fois l'algorithme en partant des partitions initiales différentes (méthode dite des « formes fortes »). Si on obtient de façon répétée des classes stables d'une répétition à l'autre, on peut considérer que ces classes sont fiables.

5. L'exécution de l'algorithme se termine lorsque le partitionnement n'évolue plus, c'est-à-dire lorsque la distance intra-classe tend à prendre une valeur fixe, ce qui est traduit par la différence $\Delta U = \|u_{(l)} - u_{(l-1)}\|$ qui est proche de zéro.

6. Dans la plupart des problèmes rencontrés dans la classification automatique non supervisée, on ne dispose pas à priori du nombre de classes effectives. Ce problème peut être partiellement surmonté en passant à une autre variante de HCM connue sous le nom ISODATA[7]. Cette dernière consiste d'une part à diviser une classe en deux si la distance intra-classe de cette classe est supérieure à un certain seuil T_1 et rassembler deux classes en une seule si la distance inter-classe est inférieure à une autre seuil T_2.

III.3.1.3 Problème de HCM

À cause de son concept d'appartenance, le HCM trouve une grande difficulté dans la séparation des classes à frontière mal définie particulièrement dans le cas d'une mauvaise initialisation des prototypes [7].

De plus, le HCM suppose que les classes sont identiques et ont une forme sphérique. Par conséquent, la présence des classes de différentes formes et/ou de tailles variables conduit à une mauvaise identification de ces dernières. Ce problème peut être partiellement surmonté en utilisant la distance de « Mahalambis » qui génère des classes ayant une forme hyper elliptique.

Dans le cas où le nombre de classes n'est pas connu à priori, on est amené à utiliser l'algorithme ISODATA (méthode des centres mobiles). Ce dernier consiste à rechercher une partition de l'ensemble de données en fonction de leur propre structure sans connaître le nombre des groupes existants dans la structure de départ. Dans ce cas, la procédure de l'ISODATA consiste à diviser une classe en deux si la distance intra-classe dépasse un maximum fixé, et que le nombre d'objets dans la classe est deux fois supérieures au minimum d'objets spécifiés. Les valeurs obtenues des centres des deux nouvelles classes sont égales à la valeur du centre de gravité de l'ancienne classe fractionnée plus au moins la distance intra-classe. Cette procédure permet de limiter la distance intra-classe. Inversement, si deux classes ont une distance inter-classe inférieure à une valeur spécifié, alors ces classes doivent être fusionnées et ainsi la distance inter-classe est maximisée [8].

III.3.2 Algorithme des C-Moyennes floues « Fuzzy C-Means : FCM »

Le concept des sous-ensembles floues au sens de Zadeh offre un nouveau cadre pour aborder les problèmes des classes aux frontières mal définies, en autorisant un objet à appartenir avec un degré à plusieurs classes distinctes.

Nous allons présenter un algorithme de Bezdek [7] [9] [10] qui minimise la fonction des moindres carrés pondérée par le degré d'appartenance:

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

Où:

- C est le nombre de classes et n le nombre d'objets dans X;
- U est une partition floue de l'ensemble des objets dans X;
- $V=(v_1,v_2,...,v_c)\in\Re^c.\Re^p$, avec v_i, le centre ou prototype de la classe;
- m: la valeur qui caractérise le flou dans la partition $m\in]1,\infty[$.

Le concept d'une partition floue est défini de la manière suivante:

$$\text{Avec:}\begin{cases}\forall i,k;1\le i\le c,1\le k\le n: & u_{ik}\in[0,1]\\ \forall i;1\le i\le c: & 0<\sum_{k=1}^{n}u_{ik}<n\\ \forall k;1\le k\le n: & \sum_{i=1}^{c}u_{ik}=1\end{cases}$$

$\sum_{i=1}^{c}u_{ik}=1\ \forall k$ est appelée contrainte d'inspiration probabiliste.

III.3.2.1 Théorème de Hard/Fuzzy C-Means (HCM/FCM)

Soit X ayant au moins C ($<n$) objets distincts, définissons $\forall k$ les ensembles:

$$I_k=\{i/1\le i\le c,d_{ik}=0\}$$
$$\bar{I}_k=\{1,2,...,c\}-I_k$$

Alors (U,V) peut être globalement minimal pour J_m si et seulement si:

$m>1$:

$$I_k=\phi\Rightarrow U_{ik}=\frac{1}{\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{(m-1)}}}\quad 1\le i\le c;1\le k<n$$

Sinon

$$I_k \neq \phi \Rightarrow \begin{cases} u_{ik} = 0, \forall i \in \tilde{I}_k et \sum_{i=i}^{c} u_{ik} = 1 \\ u_{ik} = \frac{1}{j} \forall i \in I_k tq : j = card(I_k) \end{cases}$$

$\underline{m=1}$:

$$u_{ik} = \begin{cases} 1 \quad \text{si} \quad d_{ik} < d_{jk},\ j \neq i,\ \ 1 \leq i \leq c; 1 \leq k < n \\ \\ 0 \qquad\qquad \text{Ailleurs} \end{cases}$$

$$\underline{m \geq 1}: \quad v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m} \qquad \forall i;\ 1 \leq i \leq c$$

III.3.2.2 Preuve

Optimalité du FCM : $\min_{u,v} J_m(u,v) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik}^m d_{ik}^2$

u_{ik} indépendants $\Rightarrow \min J_m(u) = \sum_{k=1}^{n} \min_{u_k \in L_{fc}} \sum_{i=1}^{c} u_{ik}^m d_{ik}^2$

$$\begin{cases} \min_{u_k \in L_{fc}} \sum_{i=1}^{c} u_{ik}^m d_{ik}^2 \\ \sum_{i=1}^{c} u_{ik} = 1 \end{cases} \Rightarrow J_k(u_k, \lambda) = \sum_{i=1}^{c} u_{ik}^m d_{ik}^2 - \lambda \left(\sum_{i=1}^{c} u_{ik} - 1 \right)$$

$$\frac{dJ_k(u_k,\lambda)}{d(u_k,\lambda)} = 0 \Rightarrow \begin{cases} \frac{dJ(u_k,\lambda)}{d\lambda} = 0 \Leftrightarrow \sum_{i=1}^{c} u_{ik} = 1 \\ \frac{dJ_k(u_k,\lambda)}{du_{ik}} = 0 \Leftrightarrow m.u_{ik}^{(m-1)} d_{ik}^2 - \lambda = 0 \Leftrightarrow u_{ik} = \left(\frac{\lambda}{m.d_{ik}^2} \right)^{\frac{1}{(m-1)}} \end{cases}$$

Par substitution:

$$\sum_{j=1}^{c} u_{jk} = \sum_{j=1}^{c} \left(\frac{\lambda}{m.d_{jk}^2} \right)^{\frac{1}{(m-1)}} = 1 \Leftrightarrow \left(\frac{\lambda}{m} \right)^{\frac{1}{(m-1)}} \sum_{j=1}^{c} \left(\frac{1}{d_{jk}^2} \right)^{\frac{1}{(m-1)}} = 1$$

$$\Leftrightarrow \left(\frac{\lambda}{m} \right)^{\frac{1}{(m-1)}} = \frac{1}{\sum_{j=1}^{c} \left(\frac{1}{d_{jk}^2} \right)^{\frac{1}{(m-1)}}} \Leftrightarrow u_{ik} = \left(\frac{\lambda}{m.d_{ik}^2} \right)^{\frac{1}{(m-1)}} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{(m-1)}}} \qquad \text{(III.2)}$$

$$v_i \Rightarrow \min_{v_i} J_i(v_i) = \sum_{k=1}^{n} u_{ik}^m d_{ik}^2$$

$$\frac{d\sum_{k=1}^{n} u_{ik}^m d_{ik}^2}{dv_i} = 0 \Leftrightarrow \frac{d\sum_{k=1}^{n} u_{ik}^m \|x_k - v_i\|^2}{dv_i} = 0$$

$$\Leftrightarrow -2\sum_{k=1}^{n} u_{ik}^m (x_k - v_i) = 0$$

$$\Leftrightarrow \sum_{k=1}^{n} u_{ik}^m x_k = v_i \sum_{k=1}^{n} u_{ik}^m$$

$$\Leftrightarrow v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m} \qquad \text{(III.3)}$$

III.3.2.3 Étapes de l'algorithme FCM

$X = \{x_1, x_2, ..., x_n\} \subset R^p$: ensemble des données à partitionner.

1. **Initialisation**
   - Fixer $m \in]1, \infty[$
   - Fixer un nombre de classes c : $2 \le c \le n$;
   - Choisir une métrique de distance d;
   - Initialiser le compteur d'itérations l=0;
   - Initialiser la matrice $U_{(0)}$;
   - Fixer le nombre ε >0.

2. **Adaptation des centres**
   - Calculer les C centres de classes floues $\{v_{i(l)}\}$ à l'aide de l'équation (III.3).

3. **Evaluation des regroupements**
   - Mettre à jour $u_{ik(l)}$ en utilisant l'équation (III.2)

4. **Test d'arrêt**
   - Comparer $u_{ik(l-1)}$ et $u_{ik(l)}$: Si $\|u_{ik(l)} - u_{ik(l-1)}\| < \varepsilon$ Alors **STOP**
     Sinon $l \leftarrow l+1$; Go to **(2)**
     Fin si

III.3.2.4 Remarques

1. Le paramètre $m \in]1, \infty[$, pondère la portée du flou dans la partition.
2. La distance d_{ik}^2 entre le vecteur d'attributs x_k et le centre v_i de la classe C_i est définie de la façon suivante:
   $d_{ik}^2 = (x_k - v_i)^t A (x_k - v_i)$, où A et une matrice $p \times p$ définie positive.

3. La forme des classes obtenues dépend du choix d'une mesure de distance. En générale, les classes générées sont de formes ellipsoïdales et ayant des directions principales identiques. Dans le cas, où A=I, la structure des classes générées par l'algorithme est alors sphérique.

III.3.2.5 Problème de FCM

Les problèmes rencontrés lors de l'application du FCM sont générés par la contrainte d'inspiration probabiliste imposée sur les degrés d'appartenance d'un objet aux différentes classes. Cette dernière dégrade les performances du classifieur notamment en présence de bruit [11].

Les exemples suivants, illustrent les problèmes associés à la contrainte probabiliste utilisée dans le FCM.

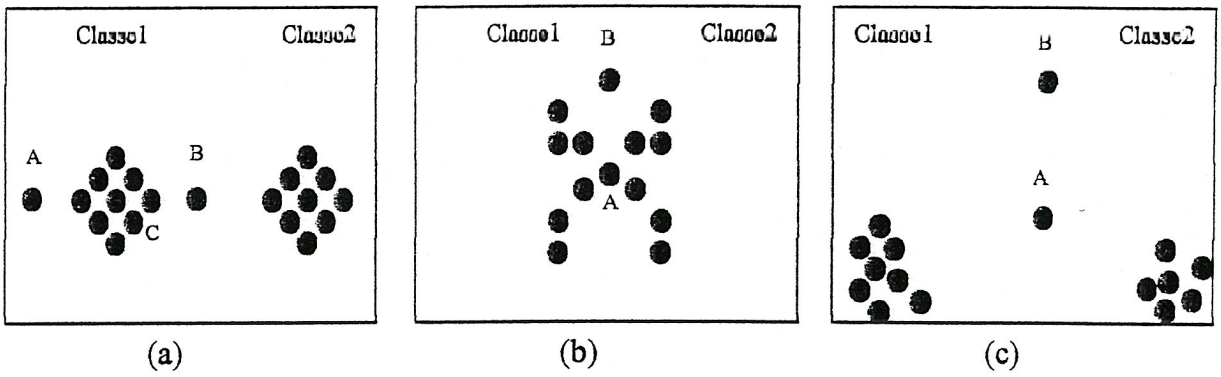


Fig.III.1. (a) exemple d'un ensemble de données constitué des deux classes dont lequel les degrés d'appartenance générés par le FCM pour les points A et B sont différents, alors qu'ils ont la même typicalité par apport à la classe1. Les points A et C ont le même degré d'appartenance, cependant ils n'ont pas la même typicalité par apport à la classe1.

(b) exemple d'un ensemble de données constitué des deux classes chevauchantes dont lequel les degrés d'appartenance des deux points A et B générés par le FCM sont de 0.5, malgré que le point A est un bon membre et le point B est un pauvre membre dans les deux classes.

(c) exemple d'un ensemble de données avec deux points bruités, A et B, dont lequel les degrés d'appartenance génères par le FCM sont presque de 0.5, alors que le point B est moins représentatif comparativement au point A dans les deux classes.

La Fig.III.1 (a) présente une situation contenant 2 classes, dans laquelle le FCM produit des degrés d'appartenance différents pour les points A et B dans la classe 1, alors qu'ils ont la même typicalité dans cette dernière (i.e., ils sont équidistants du prototype). Ce problème est apparu à cause de la contrainte probabiliste qui affaiblit l'appartenance du point B à la classe1 et augmente son appartenance à la classe 2. Similairement, le point A et le point C ont le même degré d'appartenance dans la classe1, alors que le point C est plus typique que le point A, ce qui signifie que le degré d'appartenance flou d'un point à une classe est un degré relatif, c'est à dire il dépend des autres appartenances aux différentes classes. L'appartenance du point B reflète le partage entre les 2 classes.

Fig.III.1 (b) montre une situation où il existe une intersection entre deux classes. Malgré que le point A est considéré comme un bon membre dans les deux classes et le point B

un pauvre membre, la contrainte probabiliste force les points A et B à appartenir aux deux classes avec un même degré d'appartenance qui est égale à 0.5. Ce cas de figure est contre notre intuition dans le sens de compatibilité avec le prototype.

La Fig.III.1(c) illustre une autre situation ayant deux classes et deux points A, B qui sont éloignés des prototypes. Intuitivement, le point A ne doit pas avoir un degré d'appartenance élevé (dans le sens de typicalité) dans chaque classe et le point B doit avoir un degré d'appartenance plus petit que celui de A dans chaque classe, du fait qu'il est très éloigné des deux classes. Cependant, le FCM assigne aux deux points A et B le même degré d'appartenance égal à 0.5 aux deux classes. Dans cet exemple, les degrés d'appartenance ne représentent pas la compatibilité, mais aussi ils ne peuvent pas distinguer entre un membre atypique modéré et un membre atypique extremum. Cette situation est apparue à cause de la contrainte probabiliste qui ne différencie pas entre l'égalité de l'évidence et ignorance. Pour surmonter cette difficulté, on fait recours à des théories plus récentes telles que la théorie de croyance [12] ou la théorie des possibilités [13], [14] qui sont capables d'apporter quelques améliorations.

Lorsque les donnés sont bruitées la partition finale générée par le FCM est mauvaise. Les points bruités qui sont éloignés de la classe fondamentale sont affectés aux différentes classes avec des degrés d'appartenance relativement importants, ce qui influe négativement sur l'estimation des prototypes, et finalement sur la partition.

Pour améliorer les performances du FCM et ses dérivatives en présence de bruit, nous devons tenir compte de la validité de classe. Dans l'exemple de la FigIII.1(c) si on utilise une mesure de validité convenable, on peut conclure qu'une bonne classification est obtenue en choisissant un nombre de classe égale à 3. Dans ce cas, les objets bruités sont englobés dans une seule classe séparée des deux autres. Cette manière d'agir donne un meilleur sens aux degrés d'appartenances des points bruités relativement aux deux autres classes. Cependant les mesures de validité restent difficiles à définir et le nombre des classes qui optimise une validité particulière n'est pas toujours correct. Dans n'importe quel cas, les méthodes basées sur la validité ne donnent pas des valeurs d'appartenance qu'on peut considérer comme des degrés de compatibilité.

Une autre approche pour résoudre le problème du bruit est d'introduire une classe bruitée [14], comme il a été proposé par Dave. Dans cette approche, tous les points sont considérés équidistants à une classe amorphe bruitée. La distance qui sépare la classe amorphe bruitée des classes non bruitées est relativement élevée par rapport aux autres distances (i.e. distances qui séparent les différents points appartenant aux classes non bruitées de leurs prototypes). Dans ce cas les points bruités sont classés dans la classe amorphe bruitée. Cependant, d'une part l'utilisation d'une seule valeur de distance entre la classe amorphe bruitée et tous les points non bruités peut devenir restrictive si les tailles des classes varient largement dans l'ensemble des données et d'autre part l'interprétation des degrés d'appartenance est moins significative. D'où l'on peut conclure que cette approche permet de réduire seulement l'effet de bruit.

Toutes les approches qui utilisent la validité de la classification sont efficaces dans certaines situations mais ne peuvent pas corriger le problème de degré d'appartenance relative.

III.3.3 Algorithme des C-Moyennes possibilistes (Possibilistic C-Means : PCM)

La représentation naturelle des classes floues et des concepts vagues par les méthodes des ensembles flous [1] a connu une panoplie d'applications dans le domaine du contrôle et du raisonnement à base de règles.

Dans la formulation de Zadeh [3], la représentation des classes floues ou des concepts vagues est obtenue au moyen des fonctions d'appartenances définies sur le domaine de discours [2] [1]. Ces appartenances sont absolues et représentent les degrés de typicalité. En d'autres termes, les valeurs d'appartenance d'un point dans le domaine de discours dans un ensemble flou, ne dépendent pas de ses valeurs d'appartenance dans les autres ensembles flous définis sur le même domaine de discours.

Zimmermann et Zysno ont présenté des études empiriques [15] qui donnent un bon model pour les fonctions d'appartenance correspondantes aux classes floues et aux concepts vagues:

$$u(x)=\frac{1}{1+d(x,x_0)}$$

Où: $d(x,x_0)$ est la distance entre un point x dans le domaine de discours et un prototype x_0 de la classe correspondante.

Dans cette formule les valeurs d'appartenance dépendent uniquement de la distance du point x au prototype x_0.

Le FCM et ses dérivatives ne sont pas réellement convenables pour générer de telles fonctions d'appartenance, du fait qu'ils ne peuvent pas générer des appartenances qui sont interprétés comme des degrés de compatibilité. Dans la plupart du temps, la nécessité d'une bonne méthode pour générer les fonctions d'appartenance automatiquement à partir des données d'apprentissage pose un problème.

Pour surmonter les problèmes rencontrés par le FCM en présence de bruit, Krishmapuran et Keller [16], [17] ont proposé une nouvelle approche appelée C-Moyennes possibilistes. Dans cette approche la contrainte probabiliste est relâchée et les degrés d'appartenance sont interprétés comme des degrés de compatibilité ou de possibilité. Cette nouvelle partition est définie de la manière suivante:

$$\begin{cases} \forall i,k;1\le i\le c,1\le k\le n: u_{ik}\in[0,1] \\ \forall i;1\le i\le c: 0<\sum_{k=1}^{n} u_{ik}\le n \\ \forall k;1\le k\le n: \max u_{ik}\succ 0 \end{cases}$$

Les C-partitions possibilistes résultantes définissent C-distributions possibilistes distinctes sur l'univers de discours de l'ensemble de données. Ainsi, cette nouvelle interprétation conduit aux appartenances possibilistes ou flous intrinsèques dans le sens où les degrés d'appartenance ne sont pas classiques même s'il existe une seule classe dans l'ensemble des données.

Les degrés d'appartenance possibilistes sont générés à partir de la minimisation de l'erreur quadratique, qui est redéfinie comme suit:

$J_m(U,V;X)=\sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m d_{ik}^2+\sum_{i=1}^{c}\eta_i\sum_{k=1}^{n}(1-u_{ik})^m$ Avec η_i un nombre homogène à une distance carrée.

Cette fonction objective est constituée de deux termes : le premier demande que les distances entre les objets à classifier et les prototypes soient les plus petites possible, cependant, le second terme force les u_{ik} pour devenir le plus large possible. Ceci permet de détourner la solution triviale où tous les degrés d'appartenance sont égaux à zéro. Le choix de η_i sera discuté plus tard.

III.3.3.1 Théorème

Soient $X=\{x_1,x_2,...x_n\}\subset R^p$ l'ensemble des objets à partitionner dans un univers de discours, $V=(v_1,v_2,...,v_c)\in\Re^c.\Re^p$ l'ensemble des C-prototypes, d_{ik}^2 la distance entre un point x_k et un prototype *vi* et $U=[u_{ik}]$ une matrice des degrés d'appartenance possibilistes.

$J_m(U,V)$ est globalement minimale si et seulement si:

$$u_{ik}=\frac{1}{1+\left(\frac{d_{ik}^2}{\eta_i}\right)^{\frac{1}{m-1}}} \qquad \text{(III.4)}$$

Les prototypes sont déterminés de la manière que ceux dans le cas du FCM

III.3.3.2 Preuve

Optimalité du PCM: $\min_{u,v} J_m(u,v,\eta)=\sum_{k=1}^{n}\sum_{i=1}^{c}u_{ik}^m d_{ik}^2+\sum_{i=1}^{c}\eta_i\sum_{k=1}^{n}(1-u_{ik})^m$

u_{ik} indépendantes: $\min J_m(u)=\sum_{k=1}^{n}\min_{u_k\in L_{pc}}\sum_{i=1}^{c}u_{ik}^m d_{ik}^2+\eta_i(1-u_{ik})^m$

$$\Rightarrow \min J_k(u_k)\Leftrightarrow \min_{u_k\in L_{pc}}\sum_{i=1}^{c}u_{ik}^m d_{ik}^2+\eta_i(1-u_{ik})^m \Rightarrow \frac{dJ_k(u_k)}{du_{ik}}=0$$

$$\Leftrightarrow m u_{ik}^{m-1}d_{ik}^2-m\eta_i(1-u_{ik})^{m-1}=0\Leftrightarrow u_{ik}=\frac{1}{1+\left(\frac{d_{ik}^2}{\eta_i}\right)^{\frac{1}{m-1}}}$$

$$v_i \Rightarrow \min_{v_i} J_i(v_i) = \sum_{k=1}^{n} u_{ik}^m d_{ik}^2$$

$$\frac{d\sum_{k=1}^{n} u_{ik}^m d_{ik}^2}{dv_i} = 0 \Leftrightarrow \frac{d\sum_{k=1}^{n} u_{ik}^m \|x_k - v_i\|^2}{dv_i} = 0 \Leftrightarrow v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m}$$

III.3.3.3 Etapes de l'algorithme possibiliste

$X = \{x_1, x_2, ...x_n\} \subset R^p$: ensemble des données à partitionner.

1. **Initialisation**
   - Fixer le nombre de classes $c \in]1, n]$;
   - Fixer $m \in]1, \infty[$; fixer le nombre $\varepsilon > 0$;
   - Initialiser le compteur d'itérations $l=0$;
   - Initialiser le C-partition possibiliste $U_{(0)}$;
   - Estimer les $\eta_{i(0)}$ en utilisant (III.5).

2. **Evaluation des regroupements**
   - Calculer u_{ik} à l'aide de l'équation (III.4).
   - Calculer les η_i à l'aide de l'équation (III.6).

3. **Adaptation des prototypes**
   - Evaluer les C prototypes $\{v_{i(l)}\}$ à l'aide de l'équation (III.2)

4. **Test d'arrêt**
   - Comparer $u_{ik(l-1)}$ et $u_{ik(l)}$ Si $\|u_{ik(l)} - u_{ik(l-1)}\| < \varepsilon$ Alors **STOP**

     Sinon $l \leftarrow l+1$; Go to **(2)**

     Fin si

La mise à jour des prototypes dépend de la mesure de distance choisie. Différentes mesures de distance existent qui conduisent aux différents algorithmes de classification.

- L'algorithme des C-Moyennes possibilistes (PCM) utilise comme mesure de distance la norme matricielle qui est définie par:

  $$d_{ik}^2 = (x_k - v_i)^T A_i (x_k - v_i),$$

  La mise à jour de prototypes est donnée par:

  $$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m}$$

- L'algorithme possibiliste de Gustafson-Kessel (PGK) utilise comme mesure de distance la distance de Mahalambis [18] qui est définie par:

$$d_{ik}^2 = |F_i|^{1/n}(x_k - v_i)F_i^{-1}(x_k - v_i)$$

Où :

F_i: est la matrice covariance floue d'une classe $\beta_i = (v_i, F_i)$, le centre v_i est calculé en utilisant l'équation (III.2) et la matrice covariance est mise à jour par:

$$F_i = \frac{\sum_{k=1}^{N} u_{ik}^m (x_k - v_i)(x_k - v_i)^T}{\sum_{k=1}^{N} u_{ik}^m}$$

Dans le cas d'une chaîne de classes sphériques une seule mesure de distance est utilisée [19] :

$$d_{ik}^2 = d^2(x_k - v_i) = \left(\|x_k - v_i\|^{1/2} - r_i \right)^2,$$

Où v_i est le centre, r_i est le rayon de la classe β_i et la mise à jour des prototypes est réalisée approximativement par :

$$P_i = -\frac{1}{2}(H_i)^{-1} w_i,$$

Où :

$$P_i = \begin{bmatrix} -2c_i \\ v_i^T v_i - r_i^2 \end{bmatrix} \quad H_i = \sum_{k=1}^{n} u_{ik}^m \begin{bmatrix} x_k \\ 1 \end{bmatrix} \begin{bmatrix} x_k^T & 1 \end{bmatrix} et$$

$$w_i = 2\sum_{k=1}^{n} u_{ik}^m \begin{bmatrix} x_k^T & x_k \end{bmatrix} \begin{bmatrix} x_k \\ 1 \end{bmatrix}.$$

L'algorithme résultant est appelé « possibilistic C-spherical shells » (PCSS). Les algorithmes "Possibilistic C quadric shells" (PCQS) [20] peuvent également être définies pareillement.

III.3.3.4 Signification et choix du paramètre η_i

Le paramètre η_i, appelé largeur de bande, résolution ou balance [17] détermine la zone d'influence d'une classe, et sa valeur correspond à la distance dont laquelle les degrés d'appartenances sont égaux à 0.5.

Dans le cas où l'ensemble des données est relativement exempte de bruit, les résultats d'une partition floue (FCM ou ses dérivatives) donnent une excellente estimation des η_i ; par exemple on peut estimer les η_i par:

$$\eta_i = \frac{k\left(\sum_{k=1}^{n}(u_{ik})^m d_{ik}^2\right)}{\sum_{k=1}^{n}\left(u_{ik}^m\right)} \qquad \text{(III.5)}$$

La valeur de η_i est proportionnel à la distance intra-classe moyenne floue d'une classe définie par son centre de gravité v_i. En pratique la valeur de k est choisie égale à 1.

Une grande valeur de η_i signifie que les objets de la classe correspondante ont plus de mobilité d'une itération à une autre. En d'autres termes les objets d'une classe vide qui a une extension large (i.e., distance intra-classe importante) ont une grande liberté pour se déplacer, comparativement avec ceux des classes compactes qui ont une difficulté dans le déplacement.

La règle suivante peut être aussi utiliser pour estimer les valeurs de η_i :

$$\eta_i = \frac{\sum_{x_k \in (\Pi_i)_\alpha} d_{ik}^2}{\left|(\Pi_i)_\alpha\right|} \qquad \text{(III.6)}$$

Où $(\Pi_i)_\alpha$ est un α-coupe de (Π_i) , dans ce cas, η_i est la distance intra-classe moyenne de tous les vecteurs des attributs dont lesquels les degrés d'appartenance sont supérieurs ou égal à α.

Lorsque les donnés sont bruités, la partition initiale produite par le FCM est mauvaise, ce qui implique une mauvaise estimation des η_i .

La valeur de η_i peut être fixée pour toutes les itérations ou elle peut être changée pour chaque itération. Quand η_i est changée dans chaque itération, ceci peut mener aux instabilités. Dans leur expérience, Krishmapuran et kaller ont prouvé que la classification finale est tout à fait insensible aux grandes variations des valeurs de η_i , bien que les formes finales du Π_i dépendent des valeurs exactes des η_i [17]. En effet, la meilleure approche pour estimer les valeurs des η_i basée sur une première partition floue, on utilise l'équation (III.5), et après la convergence de l'algorithme, on doit recompter les valeurs des η_i avec précision en utilisant l'équation (III.6) et exécuter l'algorithme une deuxième fois. Le second passage au moyen de l'algorithme avec des valeurs de raffinage pour les η_i permet aux appartenances résultantes dans un environnement bruyant d'être presque identiques à ceux obtenues en état d'absence de bruit. N'importe quelle valeur d'alpha entre 0,1 et 0,4 semble donner des résultats compatibles [21].

III.3.3.5 Problème de PCM

Le mauvais comportement du PCM est sa tendance à générer des classes ayant le même centre de gravité est dû à l'indépendance des classes, puisque aucun lien n'existe entre les sous fonctions objectives; la minimisation globale du $J_m(U,V)$ peut être obtenue en en minimisant indépendamment chaque sous fonctions objectives. Pour chacune de ces sous fonctions objectives, la quantité suivante doit être réduite au minimum:

$$J^{(i)}{}_m\left(u^{(i)}, v^{(i)}\right) = \sum_{k=1}^{n} (u_{ik})^m d_{ik}^2 + \eta_i \sum_{k=1}^{n} (1-u_{ik})^m$$

Où :

$u^{(i)}$ est la $i^{ème}$ rangé de la matrice de partition U et $v^{(i)}$ est le centre de la classe i. Si i est le même pour toutes les classes, alors le couple $\left(u^{(i)}, v^{(i)}\right)$ minimise $J^{(i)}{}_m\left(u^{(i)}, v^{(i)}\right)$ indépendamment de i. Dans ce cas, le PCM génère un (U,V) ayant des rangés égaux c'est dire des centres identiques. Cette tendance a été signalée par Barni et Al. [22]. Des tests ont réalisé sur des images satellitaires, qui sont composées de quatre classes bien distinctes, les résultats obtenus avec le PCM montrent que trois des quatre classes ont des centres quasiment identiques.

III.3.4 Comparaison entre les méthodes de classification par C-Means

Nous avons vu que l'approche classique est basée essentiellement sur le principe « tout pour le gagnant », puisque à chaque itération de l'algorithme, il y'a attribution totale de chaque élément à la classe dont il est le plus proche, ce qui conduit à une mauvaise classification des données dans le cas où les classes sont à frontières mal définies.

Contrairement à l'approche classique, la méthode des C-Moyennes Floues qui est basée sur le concept des sous-ensembles flous, est robuste en présence d'ambiguïté entre les frontières des classes. Cette robustesse est atteinte avec la notion de partage de chaque élément aux déférentes classes existantes.

Les méthodes possibilistes non supervisées qui ont été introduites par Krishmapuran et Keller ont été créées également pour surmonter certains problèmes rencontrés lors de l'application des méthodes de type flou C-Means, qui sont dûs à la contrainte d'inspiration probabiliste, notamment en présence de bruit. Cette contrainte est relaxée dans les approches possibilistes, et dans ce cas les degrés de partage sont remplacés par ceux de typicalité. Nous avons montré que les approches possibilistes conduisent à des classes denses ayant le même centre de gravité, ce qui implique un échec dans la classification.

III.3 Les méthodes de classification hiérarchiques

L'objectif principal d'une méthode de classification automatique est de répartir les éléments d'un ensemble en groupes, c'est à dire d'établir une partition de cet ensemble. Différentes contraintes sont imposées, chaque groupe doit être le plus homogène possible, et les groupes doivent être les plus différents possibles entre eux.

Les méthodes de classification hiérarchique produisent des suites de classes emboîtées qui définissent une hiérarchie. Dans ce cas, notre but n'est pas de trouver une partition, mais plutôt une hiérarchie de partitions qui constituent un arbre binaire appelé le dendrogramme.

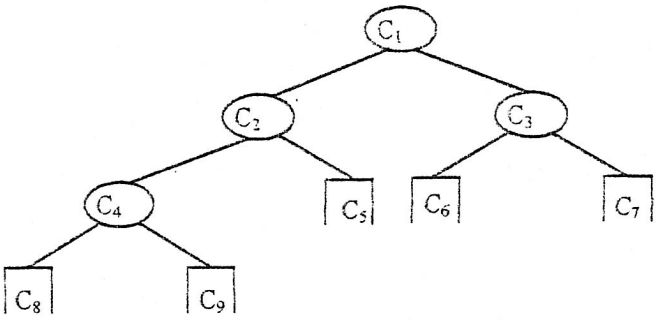
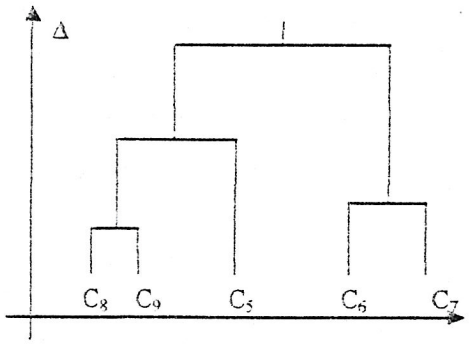


Figure III.2 : Arbre de décision d'une partition en 5 classes



FigureIII.3 : Hiérarchie indicée d'une partition en 5 classes

La figure III.3 représente un dendrogramme d'une partition à 5 classes (C_5, C_6, C_7, C_8, C_9)

Pour juger l'homogénéité des groupes, il est nécessaire de disposer d'un critère de comparaison des individus (distance, inertie, etc...) à partir de ce critère, on peut construire une hiérarchie, par exemple en regroupant deux à deux les individus qui se ressemblent le plus. On pourrait aussi envisager de comparer toutes les hiérarchies possibles, et choisir celle qui optimise le critère choisi. Malheureusement, le nombre total des hiérarchies possibles est beaucoup trop grand, même pour un petit nombre d'objets. Pour n objets, il existe $(2n-3)!/2^{n-2}(n-2)!$ hiérarchies possibles, soit pour 10 objets, 3.10^7 hiérarchies, et pour 20 objets 8.10^{21}, ce qui rend l'exploitation impossible. On est donc conduit à utiliser des heuristiques, c'est à dire des algorithmes empiriques dont les propriétés d'optimalité sont relatives. Ceci a deux conséquences importantes : d'une part, on n'est pas sur que la hiérarchie obtenue est celle qui optimise le critère, et d'autre part, les algorithmes utilisés sont souvent coûteux en temps de calcul et/ou en place mémoire.

On distingue deux types des méthodes de classification hiérarchique :

- Classification hiérarchique ascendante (CAH).
- Classification hiérarchique descendante (CDH).

III.4.1 Classification hiérarchique ascendante (CAH)

Les méthodes de classification hiérarchique ascendante partant des individus isolés assimilés à des classes et procédant, à chaque étape, par agrégation des deux classes les plus proches au sens de la norme choisie. Chaque niveau de hiérarchie représente une classe. Un arbre planaire hiérarchique permet de décrire de façon explicite la structure finale de la classification obtenue : "plus les individus se regroupent en bas de l'arbre, plus ils se ressemblent".

Lorsque la norme choisie correspond à une mesure de distance entre objets, on parle dans ce cas de classification hiérarchique ascendante simple. Dans le cas où le critère utilisé pour mesurer la ressemblance entre objets est le moment d'inertie d'ordre deux, on définit la classification hiérarchique ascendante sur le critère du moment d'ordre deux.

III.4.1.1 La CAH simple

La CAH simple repose sur un algorithme itératif à deux étapes : dans la première étape, on réunit les deux objets les plus proches pour former un nouveau groupe, et dans la seconde étape on recalcule la distance entre le groupe qui vient d'être crée et tous les autres objets. On recommence ensuite jusqu'à ce qu'il n'y ait plus qu'un seul objet, constitué de la réunion de tous les éléments initiaux de l'ensemble de départ [5].

Il faut donc disposer d'une méthode de calcul de la distance entre deux objets. Dans le cas où au moins un des deux objets est un groupe, il existe plusieurs façons de calculer cette distance. En CAH, on peut utiliser classiquement trois types de méthodes de calcul de distance:

La plus utilisée est celle qu'on appelle la méthode du lien moyen ou distance moyenne. Elle correspond à la méthode de classification automatique la plus employée, aussi appelée UPGMA (Unweighted Pair Group Method of Aggregation). Elle consiste à calculer la moyenne des distances avec les éléments du groupe, pondérée par l'effectif du groupe. La réunion des objets i et i' nécessite le calcul de la distance entre $i \cup i'$ et l'élément k :

$$D(i \cup i', k) = \frac{p(i)d(i,k) + p(i')d(i',k)}{[p(i) + p(i')]}$$

Les deux autres méthodes sont opposées : celle du lien simple ou saut minimum consiste à prendre le minimum de la distance avec les objets du groupe, alors que celle du lien complet, ou diamètre consiste à prendre le maximum de ces deux distances:

$$D(i \cup i', k) = Min(d(i,k), d(i',k))$$
$$D(i \cup i', k) = Max(d(i,k), d(i',k))$$

Ces trois méthodes de calcul de la distance à un groupe donnent des arbres qui peuvent avoir des formes très différentes : la méthode du saut minimum conduit en général à des arbres très aplatis, avec accrochages successifs des objets un à un, ce qui conduit à la formation de chaînes. La méthode du diamètre a au contraire tendance à former des arbres très éclatés, avec formation des groupes isolés. Dans le cas général, il vaut donc mieux utiliser la méthode de la distance moyenne (UPGMA).

III.4.1.2 CAH sur le critère du moment d'ordre deux

La CAH sur le critère du moment d'ordre deux est, comme la CAH simple, un algorithme itératif. La différence vient du critère utilisé pour mesurer la ressemblance entre les objets. Au lieu d'utiliser une distance simple, on utilise l'augmentation du moment d'inertie de la classe résultante de la fusion des deux objets. Ce critère fait intervenir les notions de dispersion inter- classes [23].

Le moment d'inertie résultant de la fusion des deux classes c_1 et c_2 se calcule de la façon suivante:

$$M^2(c_1 \cup c_2) = M^2(c_1) + M^2(c_2) + m_{c_1} d^2(g, g_{c_1}) + m_{c_2} d^2(g, g_{c_2})$$

m_{c_1} et m_{c_2} sont les masses des classes c_1 et c_2. On a donc une décomposition de la variabilité totale, $M^2(c_1 \cup c_2)$ en une somme due à l'inertie intra-classe de chacune des deux classes $(M^2(c_1)+M^2(c_2))$ plus un terme correspondant à l'augmentation du moment d'inertie par fusion des deux classes $(m_{c_1} d^2(g, g_{c_1}) + m_{c_2} d^2(g, g_{c_2}))$.

Dans l'algorithme de la CAH sur le moment d'ordre deux, on fusionne les deux classes pour lesquelles cette augmentation du moment d'inertie intra-classe résultante est minimale. Ceci correspond bien à la notion intuitive selon laquelle on doit regrouper les classes de façon à ce que la classe résultante soit la plus homogène possible.

De point de vue de la réalisation des calculs, l'augmentation du moment d'inertie peut être considérée comme une pseudo-distance, et on peut donc utiliser le même algorithme que celui du CAH simple, grâce à la formule suivante qui exprime l'augmentation du moment d'inertie intra –classe comme une distance[5].

$$d(i \cup i', k) = [(m_i + m_k) d(i,k) + (m_{i'} + m_k) d(i',k) - m_k d(i,i')] / (m_i + m_{i'} + m_k)$$

m_i est l'effectif du groupe i.

Il existe une autre façon de réaliser les calculs, ne nécessitant pas de passer par le calcul de la matrice de distance. Elle fournit le même résultat, mais en travaillant directement sur le tableau de données, qui est souvent plus petit que la matrice de distances. Cette méthode est basée sur l'algorithme des plus proches voisins réciproques. Elle nécessite le calcul de distances à partir de données, mais cet inconvénient est compensé par le fait qu'on peut agréger à chaque étape tous les couples de plus proche voisins réciproques (au lieu des deux voisins le plus proche seulement), et elle est donc beaucoup plus rapide.

III.4.2 Classification hiérarchique descendante (CDH)

Les méthodes de classification hiérarchique descendantes sont des méthodes de classification divisives. Elles partent de l'ensemble des individus et procèdent par divisions successives de classes jusqu'à l'obtention de classes vérifiant certaine règle d'arrêt. On les appelle aussi méthodes dichotomiques. La complexité d'un algorithme descendant est généralement exponentielle, en effet il se base sur l'énumération complète qui évalue toutes les divisions des n individus en deux sous-ensembles non vides, soit 2^{n-1}-1 possibilités. Cette stratégie de l'énumération complète, adoptée par Cavalli-Sforzal [24] pour le critère de la variance intra-classe, et bien sur difficilement applicables dés que le nombre n d'individus est supérieur à 20. Différentes approches ont été envisagées pour palier à ce problème de complexité.

III.4.2.1 CDH sur le critère du moment d'ordre deux

Cette méthode est basée sur un algorithme itératif, mais contrairement aux précédents, il s'agit d'un algorithme descendant, c'est à dire qu'il procède par dichotomies successives. À chaque itération, il y a deux choix à faire :

1. Choix de la classe à scinder en deux : on peut choisir par exemple la classe de plus grande effectif, la classe de plus grand diamètre, la classe de plus proches ou minimiser un critère de dispersion, etc.

2. Affectation des objets de la classe à scinder en deux : on peut choisir par exemple la classe de plus périphériques, les objets les plus proches ou minimiser un critère de dispersion, etc.

Il existe donc pour cette méthode aussi un grand nombre d'algorithmes différents, qui donnent des résultats plus au moins différentes en fonction de leurs propriétés. Un des principes de la CDH sur le critère du moment d'inertie d'ordre deux est le suivant :

1. Pour choisir la classe à scinder en deux, on recherche la classe dont le moment d'inertie et maximal.

2. Pour affecter les objets de la classe à scinder aux deux sous-classes résultantes, on recherche deux «point noyaux » susceptibles de servir de centre de gravité à ces deux futures sous-classes (les deux sous-classes étant inconnues a priori, on ne peut en effet pas prendre directement leur centre de gravité).

On cherche alors à sélectionner ces deux points noyaux de façon telle que l'accroissement du moment d'inertie d'ordre deux soit maximal quand on affecte les éléments à la classe dont le noyau et le plus proche. Il faut donc examiner successivement toutes les paires de points possibles, en calculant à chaque fois l'augmentation du moment d'inertie obtenue en affectant à chacune des deux futures sous-classes correspondantes les points les plus proches. Cet algorithme et donc très long s'il existe au départ un grand nombre de points à classer.

III.4.2.2 La méthode DIVOP

La méthode DIVOP [25] [26], présentée dans le cas particulier des variables qualitatives, divise à chaque étape une classe en fonction d'une question binaire et du critère d'inertie.

Dans le cas de variables qualitatives ordinales, la méthode utilise soit la distance euclidienne usuelle sur le tableau individus-variable où les modalités d'une variable sont codées par leur rang (un pour la première modalité, deux pour la seconde,..., etc.) soit la distance du khi-2 sur le tableau disjonctif complet. À chaque étape, la méthode définit la question binaire qui induit la bipartition d'inertie intra-classe minimum.

Rappelons la formule de Huygens qui relie l'inertie de l'union de deux classes $I(c_1 \cup c_2)$ à l'inertie intra-classe $W(c_1, c_2)$et à l'inertie inter-classe $B(c_1, c_2)$:

$$M^2\left(\frac{X}{\overline{V}}\right) = \sum_{x_k \in X}(x_k - \overline{V})(x_k - \overline{V})^t = C_X.$$

$$C_x = W(c_1, c_2) + B(c_1, c_2)$$

L'inertie intra-classe W est égale à l'inertie de c_1 plus l'inertie de c_2 :

$$W(c_1, c_2) = I(c_1) + I(c_2)$$

On cherche tout d'abord une bipartition (c_1, c_2) de l'ensemble des individus de plus petite inertie intra-classe. Pour cela, on évalue toutes les bipartitions induites par toutes les questions binaires. Si la variable X est ordinale, on évalue au maximum $m-1$ bipartitions, m étant le nombre de modalités de X. Dans le cas d'une variable qualitative nominale, on se heurte à un problème de complexité, le nombre de dichotomies des domaines d'observation étant alors égal à : $2^{m-1} - 1$.

Dans la pratique, le temps de calcul devient important dés que m est supérieur à 10. On sélectionne la variable X et la dichotomie $(S, \overline{S})$ du domaine d'observation de X qui induit la bipartition de plus petite inertie intra-classe. Dans un deuxième temps, il faut choisir la classe que l'on va diviser à l'étape suivante. On peut en effet choisir de diviser la classe c_1 en deux classes c_1^1 et c_1^2. On obtient alors une partition en trois classes (c_1^1, c_1^2, c_2). On peut également choisir de diviser la classe c_2 en deux classes c_2^1 et c_2^2. On obtient alors la partition en trois classes (c_1, c_2^1, c_2^2). La méthode DIVOP divise la classe c_i qui induit la partition en trois classes de plus petite inertie intra-classe. On montre que cela revient à choisir la classe c_i qui maximise:

$$\Delta(c_i) = I(c_i) - I(c_i^1) + I(c_i^2)$$

On répète ensuite les deux étapes précédentes jusqu'à ce que le nombre de classes fixé au départ soit atteint. Les divisions sont arrêtées après k itérations, ce nombre étant fixé au départ par l'utilisateur. En conséquence les feuilles de l'arbre de discision sont les K+1 classes de la dernière partition construite et les nœuds ont les questions binaires sélectionnées par la méthode. L'arbre de décision est une hiérarchie sur le k+1 classes. Cette hiérarchie indicée par Δ ne possède pas d'inversion. Ainsi, une classe divisée avant une autre est représentée plus haut dans l'arbre hiérarchique et les partitions de 2 à k+1 classes obtenues après chaque division sont bien les partitions de la hiérarchie indicée.

Par exemple, l'arbre de décision de la figure III.2 représente une partition en 5 classes (C_5, C_6 ,C_7 ,C_8 ,C_9). Dans cette représentation on ne connaît pas l'ordre de découpage. On ne sait pas si C_2 a été découpé avant C_3. En revanche, en associant à cet arbre de décision la hiérarchie indicé par Δ de la figure III.2 l'indétermination est levée.

III.5 Classification par agglomération compétitive (CA)

Au départ l'algorithme d'agglomération compétitive (CA) partitionne l'ensemble de données en un grand nombre de petites classes et progressivement les classes faibles sont exclues de la compétition. Au long des itérations l'algorithme génère une série de partitions avec décroissance progressive du nombre de classes et la partition finale est prise pour celle qui donne la valeur optimale du nombre de classes au sens de la fonction objective [30].

Des modifications ont été apportées sur cet algorithme par Frigui et Krishnapuram [27], ont abouti à la création d'une nouvelle approche de classification appelée classification par agglomération compétitive. Cette nouvelle variante combine les avantages des approches de la classification hiérarchique et ceux de la classification adaptative de type C-Means.

Comme dans le cas de la classification par C-Means, l'algorithme d'agglomération compétitive minimise une fonction objective définie par:

$$\min_{(U,V)}\left(J(U,V;X)\right)=\sum_{i=1}^{C}\sum_{k=1}^{n}(u_{ik})^2 d_{ik}^2-\alpha\sum_{i=1}^{C}\left[\sum_{k=1}^{n}u_{ik}\right]^2$$

avec $\sum_{i=1}^{C}u_{ik}=1 \qquad \forall k\in[1,n]$

$d_{ik}^2=d^2(x_k,v_i)$ est la distance de l'objet x_k du centre v_i de la classe C_i.
$U=[u_{ik}]$ est une matrice $C\times n$ qui représente la matrice des degrés d'appartenance.

La fonction objective est composée de deux termes :

Le premier terme est semblable à celui de FCM dans le cas où le facteur de fuzzification m est choisi égale à 2. Cette composante contrôle la forme et la taille des classes pour obtenir des classes compactes. Le deuxième terme est une somme des carrées des degrés d'appartenance pour chaque classe pondérée par le coefficient α.

La minimisation du premier terme est atteinte dans le cas où chaque classe est formée d'un singleton. Cependant le deuxième terme exige que tous les objets soient assignés à une seule classe. Pour cela le coefficient α doit être choisi de tel sorte que le second terme de la fonction objective ne soit pas négligeable et en même temps ne soit pas important par apport au premier afin de ne pas générer un nombre important de classes ou une partition constituée d'un nombre de classes très faible.

Frigui et Krishnapuram ont proposé [27] la formule suivante pour le calcul des valeurs de α :

$$\alpha=\alpha(k)=\eta(k)\frac{\left[\sum_{i=1}^{C}\sum_{k=1}^{n}(u_{ik})^2 d_{ik}^2\right]}{\left[\sum_{i=1}^{C}\left[\sum_{k=1}^{n}u_{ik}\right]^2\right]}$$

$$\eta(k)=\eta_0\exp\left(\frac{-k}{\tau}\right)$$

k : L'itération de rang k de l'algorithme
τ :Le nombre maximal d'itérations
η_0 :Valeur initiale.

La minimisation de la fonction objective nous permet de définir les degrés d'appartenance :

$$u_{ik} = u_{ik}^{bias} + u_{ik}^{FCM}$$

Avec $u_{ik}^{FCM} = \left[\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{(m-1)}}\right]^{-1}$ et $u_{ik}^{bia} = \frac{\alpha\left(N_i - \overline{N}_k\right)}{d_{ik}^2}$

Où

$N_i = \sum_{k=1}^{n} u_{ik}$: La cardinalité de la classe C_i

$\overline{N}_k = \frac{\left[\sum_{i=1}^{C}\frac{1}{d_{ik}^2}N_i\right]}{\left[\sum_{i=1}^{C}\frac{1}{d_{ik}^2}\right]}$: La moyenne pondérée des cardinalités de classes

Dans le cas où la cardinalité N_i de la classe C_i est très faible et l'objet x_k est proche d'une autre classe dont la cardinalité est élevée ($\overline{N}_k$ est grande), le degré d'appartenance devient négatif. Dans ce cas on force u_{ik} à zéro. De même u_{ik} peut prendre des valeurs supérieures à 1, dans le cas où la cardinalité N_i de la classe C_i est grande et l'objet x_k est proche d'une classe de faible cardinalité($\overline{N}_k$ est petite), u_{ik} est mis à 1.

III.5.1 Étapes de l'algorithme d'agglomération compétitive (CA)

Les étapes de l'algorithme d'agglomération compétitive (CA) sont les suivantes:

- Fixer le nombre de classes C=C_{masx}, fixer ε>0, Initialiser $U^{(0)}$, l à 0
- Calculer les cardinalités initiales N_i : $N_i = \sum_{k=1}^{n} u_{ik}$
- Répéter
  - Calculer d_{ik}^2 ;
  - Mise à jour du nombre de classes ;
  - Mise à jour des centres de classes :

$$v_i = \frac{\sum_{k=1}^{n}(u_{ik})^2 x_k}{\sum_{k=1}^{n}(u_{ik})^2}$$

  - $l = l+1$

  Jusqu'à $\Delta U<\varepsilon$.

III.6 Conclusion

Dans ce chapitre, nous avons présenté trois grands types d'algorithmes en classification automatique non supervisée :

- Les algorithmes de partitionnement (HCM, PCM, PCM), qui fournissent une partition

conduisant à une décomposition de l'ensemble de départ en un nombre C de classes fixé a priori de même niveau et permet de traiter des ensembles de données en optimisant des critères pertinents qui sont dans le cas général le moment d'inertie d'ordre deux.

- Les algorithmes agrégatifs, qui correspondent aux méthodes dites de classification ascendante hiérarchique.

- Les algorithmes dichotomiques, qui correspondent aux méthodes dites de classification descendante hiérarchique.

La méthode d'agglomération compétitive proposée par Frigui et Krishnapuram [27] combine les avantages des approches de la classification hiérarchique et ceux de la classification adaptative de type C-Means.

Dans le chapitre IV, nous allons présenter une étude comparative entre les méthodes de classification automatique non supervisée par C-Means, ensuite nous proposons une nouvelle approche [28] basée sur la fusion d'algorithmes FCM et PCM avec résolution du problème d'initialisation.

Chapitre IV

Comparaison Entre Les Méthodes De Classification Par C-MEANS

IV.1 Introduction

Le but principal de l'approche possibiliste est de trouver la solution des problèmes associés à la contrainte imposée sur les degrés d'appartenances [1] qui sont utilisés dans les algorithmes de classification floue tels que le flou C-Means (FCM); la contrainte génère dans le FCM des degrés d'appartenance qui peuvent être interprétées comme degrés de partage mais pas comme degrés de typicalité.

Ainsi, dans une classe donnée constituée d'un ensemble de points, si deux points sont équidistants du prototype, alors leurs degrés d'appartenance peuvent être sensiblement différents; par contre s'ils sont répartis autrement, leurs degrés d'appartenance peuvent être égaux. Ces deux cas affaiblissent les performances du FCM en présence de bruit.

L'introduction du terme η_i [17] dans la fonction objective du FCM, a permis d'établir un nouveau algorithme de classification appelé PCM (possibilistic C-Means). Dans l'approche possibiliste, l'appartenance d'un point à une classe représente la typicalité ou la possibilité de ce point d'appartenir à cette dernière. La relaxation de la contrainte imposée sur les degrés d'appartenance peut engendrer des appartenances qui représentent la typicalité des objets aux différentes classes, ce qui permet de réduire l'effet de bruit et d'améliorer les résultats du classifieur. En revanche, l'approche possibiliste génère des centres de gravité identiques [22].

Compte tenu des modifications introduites dans le PCM, nous présentons une étude comparative entre les différents algorithmes de classification par C-Means, ensuite, nous proposons une nouvelle approche [28] qui est basée sur la fusion des algorithmes flou et possibiliste.

Pour mener à bien cette tâche, nous réalisons des tests sur les bases de données Iris et Textures, qui sont décrites dans le paragraphe suivant:

1. La base de données Iris

La base de données Iris est constituée de 150 fleurs décrites par 4 variables (longueur et largeur de sépales, et de pétales), le nombre de classes est égal à 3 (IRIS SETONA, IRIS VERSICOLOR, IRIS VERGINIA), les objets sont uniformément répartis en trois classes, les classes 2 et 3 sont facilement séparables de la classe 1, mais difficilement séparables entre elles.

2. L'image texturée

La figure (IV.1) montre une image qui est constituée des deux microtextures différentes. Un prétraitement (le calcul des différentes corrélations locales) de l'image initiale a donné naissance à une série de 8 images dont chacune est le résultat d'une détection d'un attribut particulier. Un pixel est alors décrit par un vecteur à 8 attributs. Ainsi, l'image des deux textures de la figure (IV.1) est représentée par deux classes dont chacune est décrite par 8 fichiers contenant chacun les valeurs de pixels pour l'une des 8 composantes. L'échantillon obtenu est constitué de 400 pixels de chaque classe.

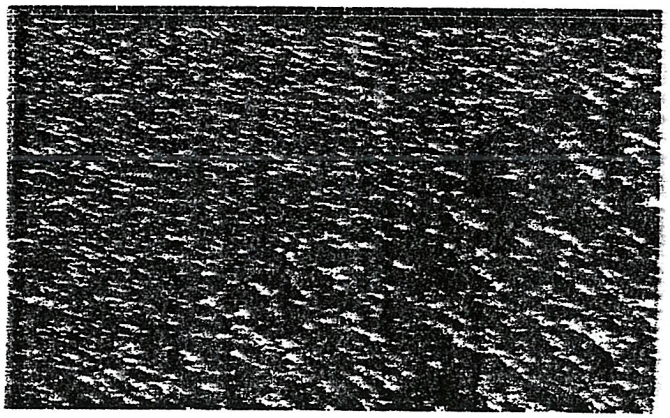


Figure IV.1: Image des textures.

IV.2 L'algorithme de classification possibiliste

Rappelons la formulation originale du FCM qui minimise la fonction objective:

$$J(V,U)=\sum_{i=1}^{C}\sum_{j=1}^{N}u_{ik}{}^{m}d_{ik}^{2} \qquad \text{(IV.1)}$$

Sachant que: $\sum_{i=1}^{C}u_{ik}=1$ pour tout $k=1,2,...,N$ (IV.2)

Dans (IV.1), $V=(v_1,v_2,...,v_c)$ est l'ensemble des C-prototypes, d_{ik}^2 est la distance du point x_k au prototype v_i, N le nombre total des vecteurs à classifier, C est le nombre de classes, et $U=[u_{ik}]$ une $C\times N$ matrice à partition floue qui doit satisfaire la condition (IV.2). u_{ik} est le degré d'appartenance d'un point x_k dans la classe v_i, et $m\in[1,\infty[$ est le facteur de

fuzzification qui contrôle l'introduction du flou dans la partition.

Une simple relaxation de la contrainte (IV.2) engendre la solution triviale (i.e. tous les degrés d'appartenance sont mis à 0). Or notre objectif est normalement d'assigner:

- Aux points représentatifs des degrés d'appartenance forts.
- Aux points non représentatifs des degrés d'appartenance faibles.

La fonction objective qui répond à nos exigences est donnée par :

$$J_m(U,V;X)=\sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m d_{ik}^2+\sum_{i=1}^{c}\eta_i\sum_{k=1}^{n}(1-u_{ik})^m \qquad \text{(IV.3)}$$

Cette reformulation montre bien que le premier terme de la fonction objective exige que la distance du point au prototype soit la plus petite possible, tandis que le deuxième terme force le u_{ik} pour être le plus grand possible.

La dérivation de (IV.3) nous permet de définir les u_{ik} comme suit :

$$u_{ik}=\frac{1}{1+\left(\frac{d_{ik}^2}{\eta_i}\right)^{\frac{1}{m-1}}} \qquad \text{(IV.4)}$$

Ainsi, dans chaque itération, la valeur de mise à jour de u_{ik} dépend seulement de la distance de x_k à v_{ik}, qui est un résultat intuitivement satisfaisant. L'assignation d'un point à une classe est maintenant déterminée seulement à partir de la distance qui le sépare du prototype de la classe.

IV.2.1 Signification de η_i

La valeur de η_i est égale à la distance dont le cas où les degrés d'appartenance sont égaux à 0.5. En outre, le η_i détermine la zone d'influence d'une classe. En effet, un point x_k aura moins d'influence sur l'estimation d'un prototype v_i si $d^2(x_k, v_i)$ est supérieure à η_i. Un bon choix de η_i est donné par la variance (distance intra-classe moyenne):

$$\eta_i=\frac{k\left(\sum_{k=1}^{n}(u_{ik})^m d_{ik}^2\right)}{\sum_{k=1}^{n}(u_{ik}^m)} \qquad \text{(IV.5)}$$

IV.2.2 Propriété d'une partition possibiliste

La fonction objective correspondante à une classe i peut être formulée comme suit :

$$J_i(Ui,Vi;X)=\sum_{k=1}^{n}(u_{ik})^m d_{ik}^2+\eta_i\sum_{k=i}^{n}(1-u_{ik})^m \qquad \text{(IV.6)}$$

Les degrés d'appartenances générés avec le PCM ne sont pas liés par la contrainte d'inspiration probabiliste $\sum_{i=1}^{c}u_{ik}=1 \quad \forall k=1,\ldots, n$. La mise à jour de l'équation qui calcule les degrés d'appartenance dans le PCM est donnée par (IV.4).

À partir de l'équation (IV.4), on obtient :

$$d_{ik}^2 = \eta_i \left(\frac{1-u_{ik}}{u_{ik}} \right)^{m-1} \qquad \text{(IV.7)}$$

En éliminant $d^2(x_k, v_i)$, la fonction objective définie dans (IV.6) devient :

$$J_i(Ui, Vi; X) = \eta_i \sum_{k=1}^{n} (1-u_{ik})^{m-1}$$

Pour une valeur donnée de η_i, minimiser $J_m(Ui, Vi; X)$ est équivalent à minimiser:

$$J_i'(Ui, Vi; X) = \eta_i \sum_{k=1}^{n} \left[1-(1-u_{ik})^{m-1} \right] = \eta_i \sum_{k=1}^{n} u_{ik}' \qquad \text{(IV.8)}$$

Où $u_{ik}' = 1-(1-u_{ik})^{m-1}$ peut être interprété comme étant un degré d'appartenance modifié. Il est obtenu à partir de u_{ik} via un tracé monotone :

$$\frac{d}{du_{ik}} u_{ik}' = (m-1)(1-u_{ij})^{m-2} \succ 0 \quad \text{Pour} \quad m > 1.$$

D'où, u_{ik}' varie de la même façon que u_{ik}, i.e., $u_{ik} = 0 \Rightarrow u_{ik}' = 0$; $u_{ik} = 1 \Rightarrow u_{ik}' = 1$; ces derniers sont des fonctions monotones décroissantes de d_{ik}^2. De plus, dans le cas spécial où m=2, l'équation (IV.8) se réduit à :

$$J_i'(Ui, Vi; X) = \eta_i \sum_{k=1}^{n} u_{ik} \qquad \text{(IV.9)}$$

À partir des deux équations (IV.8) et (IV.9), on remarque que pour une valeur donnée de η_i, tous les C sous-fonctions objectives sont maximisées par le choix des positions des prototypes de tels sorte que la somme des degrés d'appartenance modifiés soit maximisée. Ceci est vérifié lorsque les prototypes sont localisés dans des régions denses, du fait que la fonction d'appartenance est une fonction monotone et décroissante de la distance. S'il existe réellement C-régions denses, alors avec une bonne initialisation, chaque prototype converge vers une région dense. Dans une telle situation, même si tous les η_i sont égaux chacun d'eux aura C-distincts minimums correspondants aux C-régions denses.

IV.2.3 Problème de coïncidence dans le PCM

Le mauvais comportement du PCM, est sa tendance à générer des classes ayant des centres identiques, est dû au fait que les degrés d'appartenance générés par le PCM sont généralement très voisins les uns des autres pour tous les objets dans toutes les classes. Cette situation engendre un glissement progressif des centres des classes vers le centre de données à partitionner.

Plusieurs modifications ont été appliquées sur le PCM pour contourner le problème de coïncidence.

Krishnapuram et Keller ont proposé une autre fonction objective indépendante de 'm' définie par :

$$J_m(U, V; X) = \sum_{i=1}^{C} \sum_{k=1}^{n} (u_{ik}) d_{ik}^2 + \sum_{i=1}^{C} \eta_i \sum_{k=1}^{n} u_{ik} (\log u_{ik} - 1) \qquad \text{(IV.10)}$$

Alors la mise à jour des u_{ik} est donnée par :

$$u_{ik} = \exp\left(-\frac{d_{ik}^2}{\eta_i}\right) \qquad 1 \le i \le C, 1 \le k \le n \tag{IV.11}$$

L'équation de mise à jour des prototypes reste inchangeable. Dans ce cas, la fonction exponentielle descend plus rapidement pour les grandes valeurs de $d^2(x_k, v_i)$. Cette formulation est mieux adaptée lorsque les classes deviennent probablement compactes.

Il est à noté que $u_{ik}(\log u_{ik} - 1)$ est une fonction monotone décroissante dans $[0.1]$. Si on utilise la distance de Mahalambis, alors on peut éliminer η_i (i.e., $\eta_i = 1$).

Khodja [29] a proposé aussi d'autres modifications pour corriger ce problème. Il affaiblit la valeur de 'm' en modifiant l'équation qui permet de calculer les degrés d'appartenance:

$$u_{ik} = \frac{1}{1 + \left(\frac{d_{ik}^2}{\eta_i}\right)^{\frac{m}{m-1}}} \tag{IV.12}$$

Qui est obtenue par dérivation de la fonction objective suivante :

$$J_m(U,V;X) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m d_{ik}^{2m} + \sum_{i=1}^{c}\eta_i^m \sum_{k=1}^{n}(1-u_{ik})^m \tag{IV.13}$$

Dans ce cas pour les valeurs de m<2 les degrés d'appartenance des objets qui sont éloignées d'une classe seront affaiblis et ceux qui lui appartiennent seront amplifiés.

Ouled Ahmedou [30] à proposé une approche appelée PCMM (Possibilistic C-Means Modifier). L'algorithme consiste à trouver une partition de telle manière que le degré d'appartenance d'un objet à la classe gagnante représente un degré d'appartenance renforcé par un élargissement de l'extension, alors que ce degré est abaissé en affaiblissant l'extension de chaque classe restante. Il a proposé une nouvelle méthode de calcul de η_i :

$$\eta_{ik} = \begin{cases} \eta_1 & \text{si} \quad i = \arg\min_j \left\{\|x_k - v_j\|^2\right\} (\text{i = gagnant}) \\ \eta_2 & \text{sinon} \end{cases}$$

avec $\eta_1 \succ \eta_2$

Pour le calcul de η_1 et η_2, on peut choisir par exemple:

Où
$$\eta_1 = D^2 et \eta_2 = \frac{\eta_1}{2}$$
$$D = \max_{i \neq j}\|v_i - v_j\|$$

Le choix de η_1 et η_2 est donné par:

$$\eta_1 = \begin{cases} D^2 & \text{si } D^2 \geq 1 \\ \sqrt{D} & \text{sinon} \end{cases}$$

$$\eta_2 = \begin{cases} \sqrt{D} & \text{si } D^2 \geq 1 \\ D^2 & \text{sinon} \end{cases}$$

IV.3 L'approche proposée

IV.3.1 Présentation du problème

Nous avons vu dans le chapitre précédent que le FCM repose sur l'appartenance relative en s'appuyant sur la contrainte d'inspiration probabiliste. Dans ce cas, les degrés d'appartenance sont interprétés comme des degrés de partage et non pas de typicalité. En effet, les degrés d'appartenance des deux points équidistants d'un prototype d'une classe peuvent devenir différents. Réciproquement on peut trouver le cas où les deux points ont le même degré d'appartenance alors que ces derniers sont arbitrairement éloignés l'un de l'autre. Ceci dégrade les performances du FCM en présence de bruit.

Pour surmonter ce problème, le PCM relâche la contrainte et la typicalité est remplacée par le partage ; signalons encore une fois que les degrés d'appartenance générés par le PCM sont très voisins et peuvent engendrer des centres identiques.

IV.3.2 La solution Proposée

Pour résoudre le problème de coïncidence, il est convenable que les objets d'une même classe doivent avoir des degrés d'appartenance plus forts que ceux des autres classes. Dans ce cas, nous attribuons à chaque objet qui est susceptible d'être dans une classe un degré flou au sens possibiliste et un degré possibiliste aux classes restantes [28].

IV.3.3 Conséquences

Si on considère le cas où l'ensemble à partitionner contient deux classes distinctes qui sont définies par leurs centres C1 et C2 et par leurs extensions η_1 et η_2. Un point p quelconque de l'ensemble à partitionner peut avoir quatre positions possibles:

- Soit il appartient à la zone d'influence de la première classe;
- Soit il appartient à la zone d'influence de la deuxième classe;
- Soit il appartient à l'intersection des deux zones d'influence des deux classes;
- Soit il n'appartient pas ni à la zone d'influence de la première classe ni à la deuxième.

Dans le cas où le point p appartient à la première zone d'influence, ce dernier est assigné à la première classe avec un degré flou et à la deuxième avec un degré possibiliste. Réciproquement, si p appartient à la deuxième zone d'influence, il est assigné à la deuxième classe avec un degré flou et à la première avec un degré possibiliste.

Si p appartient à l'intersection des deux zones d'influence, chaque classe attribue à p un degré flou. Dans le cas où p n'appartient pas ni à la première zone d'influence ni à la deuxième, on affecte p aux deux classes avec des degrés possibilistes.

IV.3.4 Exemple de quelques situations typiques

Le tableau IV.1 présente la satisfaction quelques règles typiques :

Algorithme de classification	$u_A > u_B$	$u_{C1} = u_{D1}$	$u_{C1} > u_{F1}$	$U_{E1} = u_{E2}$	$u_{G1} > u_{H1}$
FCM	Non	Non	Non	Oui	Non
PCM	Oui	Oui	Oui	Oui	Oui
FPCM	Oui	Oui	Oui	Oui	Oui

Tableau IV.1 : Satisfaction de quelques règles typiques par le FCM, PCM et le FPCM

1. $u_A > u_B$:

À cause de la contrainte imposée sur les degrés d'appartenance le FCM assigne les deux points A et B aux deux classes (1 et 2) avec le même degré d'appartenance, néanmoins le point B est moins typique que le point A. Dans le cas du PCM, le point B est affecté aux deux classes avec des degrés moins typiques que ceux du point A. Avec le FPCM, la situation est la même que celle dans le PCM du fait que les points A et B sont considérés comme des points hors de la zone d'influence de chaque classe.

2. $u_{C1} = u_{D1}$:

Avec le FCM, le degré d'appartenance à la première classe du point C est inférieur à celui du point D, parce que le point C est partagé entre les deux classes. Contrairement, le PCM attribut aux deux points le même degré d'appartenance. Le FPCM assigne les points C et D à avec un degré flou, dans ce cas le point C est affecté à la première classe avec un degré flou et à la deuxième avec un degré possibiliste ce qui élimine l'opération de partage qui apparaît dans le FCM.

3. $u_{C1} > u_{F1}$:

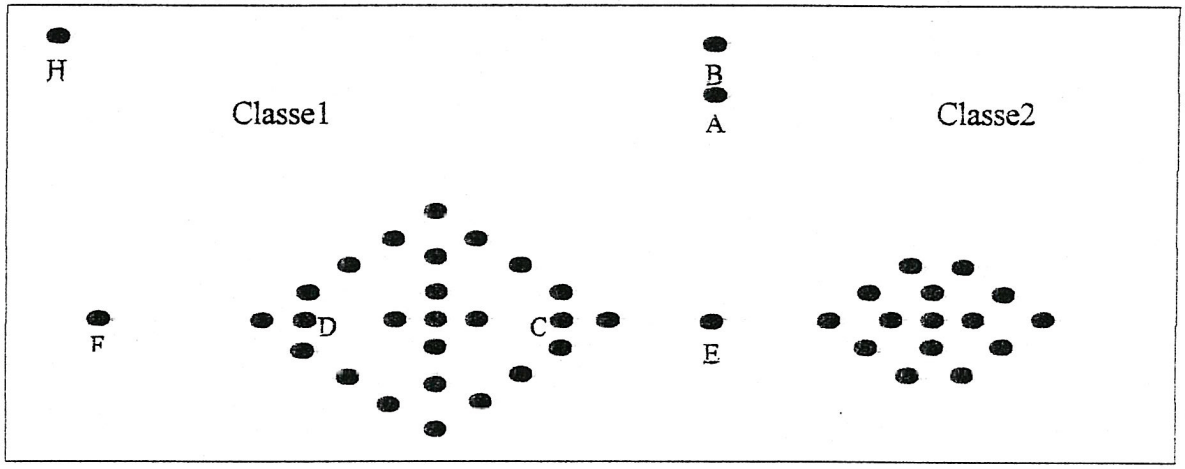
Le FCM assigne le point F à la première classe avec un degré d'appartenance plus élevé que celui du point C. Avec le PCM, le point C prend un degré d'appartenance plus fort que celui du point F. Dans le cas du FPCM, le point C est assigné à la première classe avec un degré flou plus fort que celui du point F.

4. $u_{E1} = u_{E2}$:

Le point E est assigné avec le même degré d'appartenance aux deux classes (1 et 2) que se soit avec le FCM ou avec le PCM. Dans le cas du FPCM, le point E est assigné avec un degré flou à la première et à la deuxième classe, bien que le point E appartient à la zone d'intersection des deux classes le FPCM applique la notion de partage et élimine ainsi le problème de chevauchement.

5. $u_{E1} > u_{H1}$:

Le FCM assigne le point H à la première classe avec un degré d'appartenance plus élevé qu celui du point E. Le PCM attribut à chacun d'eux un degré d'appartenance en fonction des distances réelles. Tandis que le FPCM considère le point H comme étant un point étranger et il l'affecte alors avec un degré possibiliste.



FigureIV.2 : Représentation de quelques situations typiques.

IV.3.5 Avantages de l'approche FPCM

La nouvelle approche que nous avons proposée (FPCM) qui se base sur la fusion de la théorie du flou et de la possibilité permet de résoudre le problème de chevauchement, d'éliminer le problème de bruit et de générer des centres disjoints.

IV.4 Résultats de la classification sur la base de données Texture et Iris

Pour estimer les performances de la nouvelle approche (FPCM) par rapport aux autres décrites précédemment, nous avons réalisé des tests sur les bases de données Texture et Iris avec le HCM, FCM, PCM et le FPCM.

IV.4.1 Résultats de la classification sur l'image texturée

Les résultats obtenus avec les trois algorithmes HCM, FCM, PCM et FPCM changent selon le choix de l'initialisation de l'algorithme.

On distingue deux méthodes d'initialisation :

- Initialisation par centres de gravité;
- Initialisation par matrice d'appartenance.

IV.4.1.1 Initialisation par centres de gravité

L'initialisation de l'algorithme par centres de gravité consiste à exécuter plusieurs fois l'algorithme en partant des centres initiaux différents. Lorsqu'on obtient de façon répétée des centres stables d'une répétition à l'autre, alors on peut les considérer comme fiables. En appliquant cette procédure, on obtient les centres suivants:

$$V_{FCM} = \begin{pmatrix} 145.12 & 148.41 & 115.36 & 185.51 & 131.42 & 153.82 & 145.81 & 89.70 \\ 95.91 & 145.90 & 135.55 & 114.88 & 118.22 & 198.72 & 123.29 & 146.61 \end{pmatrix}$$

$$V_{HCM} = \begin{pmatrix} 175.50 & 141.82 & 114.45 & 134.47 & 147.74 & 147.72 & 157.08 & 147.01 \\ 174.19 & 147.30 & 114.77 & 179.27 & 171.92 & 157.20 & 114.28 & 124.14 \end{pmatrix}$$

Les différentes partitions de l'image des textures générées par les quatre algorithmes de classification sont représentées dans la figure (IV.3).

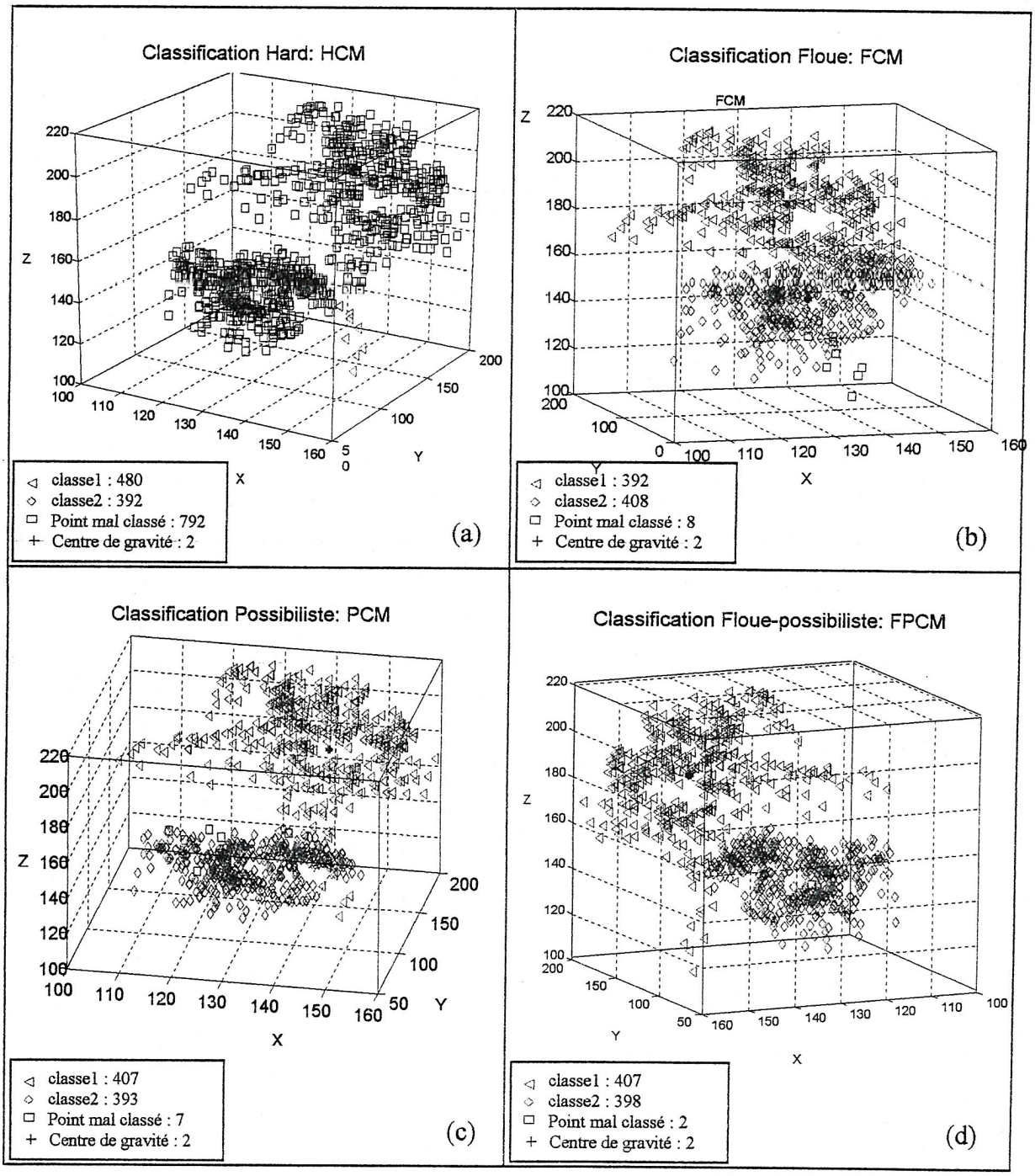


Figure IV.3 : a), b), c), d) Représentation dans l'espace des attributs les différentes partitions de l'image des textures générées respectivement par le HCM, le FCM, le PCM et le FPCM.

Le tableau IV.2 montre les taux de réussite de la classification de l'image texturée en appliquant les différents algorithmes de classification.

Algorithme de classification	Taux de réussite (%)	Indice de satisfaction (J_m)	Nombre d'itérations
HCM	01.00	$1.27.10^6$	05
FCM	99.00	$1.20.10^6$	10
PCM	99.50	$3.34.10^5$	40
FPCM	99.75	$5.96.10^9$	9

Tableau IV.2 : Comparaison entre les différents résultats des classifications obtenus avec les quatre algorithmes HCM, FCM, PCM et FPCM.

À la lecture du tableau IV.2, on remarque que HCM atteint un taux de réussite seulement de 1% après 05 itérations avec un coût minimal de $1.27.10^6$; le taux de classification obtenu par le FCM est de 99.00% après 10 itérations avec un coût minimal de $1.20.\ 10^6$, tandis que le taux de classification avec le PCM est égal à 99.50% avec un nombre d'itérations qui vaut 40 et un coût minimal de $3.34\ .10^5$, cependant le FPCM atteint un taux de réussite de 99.75% après 9 itérations et avec un coût minimal de $1.30.10^6$.

Le tableau ci-dessous présente les matrices de confusion obtenues avec les quatre algorithmes de classification de l'image texturée.

	C1	C2
C1	8	392
C2	400	0

HCM
Erreur totale 792

	C1	C2
C1	392	8
C2	0	400

FCM
Erreur totale 8

	C1	C2
C1	400	0
C2	7	393

PCM
Erreur totale 7

	C1	C2
C1	400	0
C2	2	398

FPCM
Erreur totale 2

Tableau IV.3: Matrices de confusion

À partir des résultas du tableau IV.3, on remarque qu'avec le HCM : 392 objets de la première classe (C_1) sont classés dans la deuxième tandis que 400 objets de la deuxième classe (C_2) sont classés dans la première, le HCM commet alors une erreur total de 792 objets parmi 800. Avec le FCM, les 400 objets de la deuxième classe sont complètement reconnus, tandis que la première classe ne reconnaît que 392 objets parmi 400. Dans le cas du PCM et du FPCM la première classe est complètement reconnue, alors que lors de la classification des objets de la deuxième classe, le PCM et le FPCM commettent respectivement 7 et 2 erreurs.

IV.4.2.2 Initialisation par matrice d'appartenance

On cherche à trouver une matrice d'appartenance initiale de telle sorte que les degrés d'appartenance sont partagés équiprobablement entre les classes tout en respectant la contrainte d'inspiration probabiliste.

Les résultats présentés par le tableau IV.4 donnent les taux de réussite et les coûts de classification ainsi que le nombre d'itérations correspondants à chaque algorithme de classification de l'image des textures.

Algorithme de classification	Taux de réussite (%)	Indice de satisfaction (J_m)	Nombre d'itérations
HCM	01.00	$1.27.10^6$	05
FCM	99.00	$1.20.10^6$	08
PCM	99.50	$3.34.10^5$	40
FPCM	99.75	$1.30.10^6$	09

Tableau IV.4: Comparaison entre les différents résultats des classifications obtenus avec les quatre algorithmes HCM, FCM, PCM et FPCM.

A la lecture du tableau IV.4, on remarque que le taux de réussite ainsi que le coût obtenus avec chaque algorithme de classification restent inchangeable, alors que le nombre d'itérations a diminué de deux avec le FCM et reste le même avec le PCM et le FPCM.

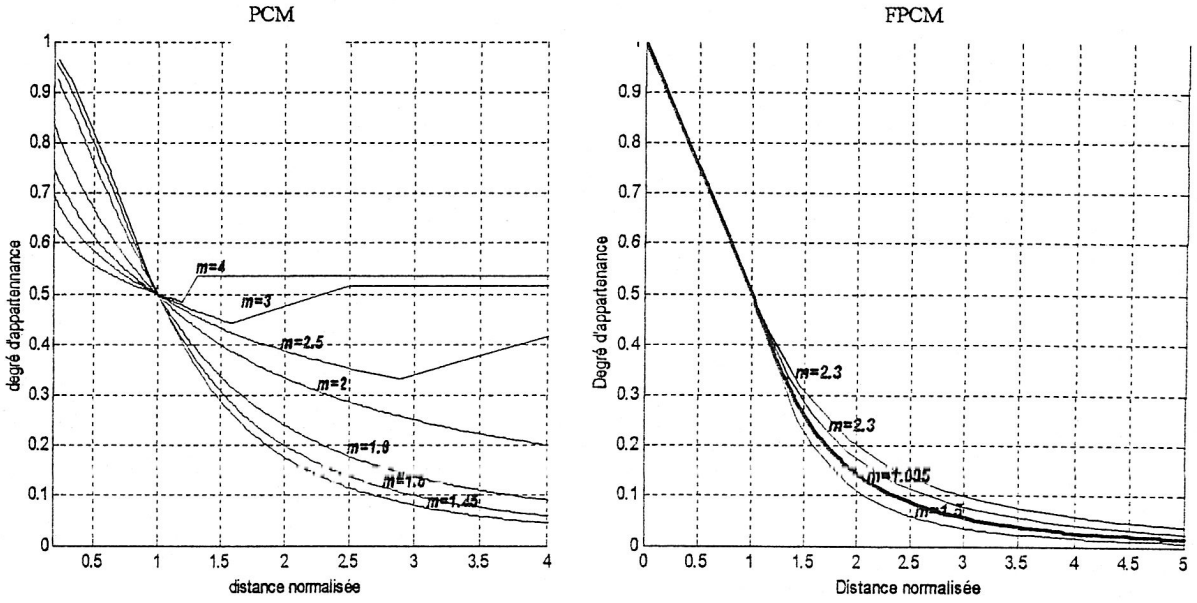
Le tableau IV.5 donne les centres finaux générés par les différents algorithmes de classification :

centres	HCM	FCM	PCM	FPCM
V_1	118.62	163.42	164.12	161.41
	87.09	180.58	179.20	190.78
	149.02	180.59	174.69	177.59
	133.01	137.49	140.19	136.83
	112.26	119.58	128.84	118.64
	138.27	145.01	147.43	161.97
	118.96	149.23	142.50	144.57
	121.62	131.76	128.03	131.21
V_2	163.22	118.82	119.55	118.37
	180.27	87.74	87.18	87.74
	180.26	149.58	146.52	148.63
	137.13	132.95	128.43	125.84
	119.21	112.19	104.46	104.99
	144.35	138.19	135.17	133.36
	148.76	119.29	112.34	109.65
	131.57	121.57	122.63	120.00

Tableau IV.5: Les centres finaux générés par les différents algorithmes de classification

En analysant le tableau ci-dessus, on remarque que tous les centres générés par les quatre algorithmes sont plus au moins différents; les centres obtenus avec le HCM sont très distincts par contre ceux obtenus avec le FCM, PCM et le FPCM sont très voisins.

Les courbes représentées par la figure IV.4 illustrent la fonction d'appartenance en fonction de la distance normalisée $\frac{d_{ik}}{\eta_i}$ avec différentes valeurs de m pour le PCM et le FPCM.



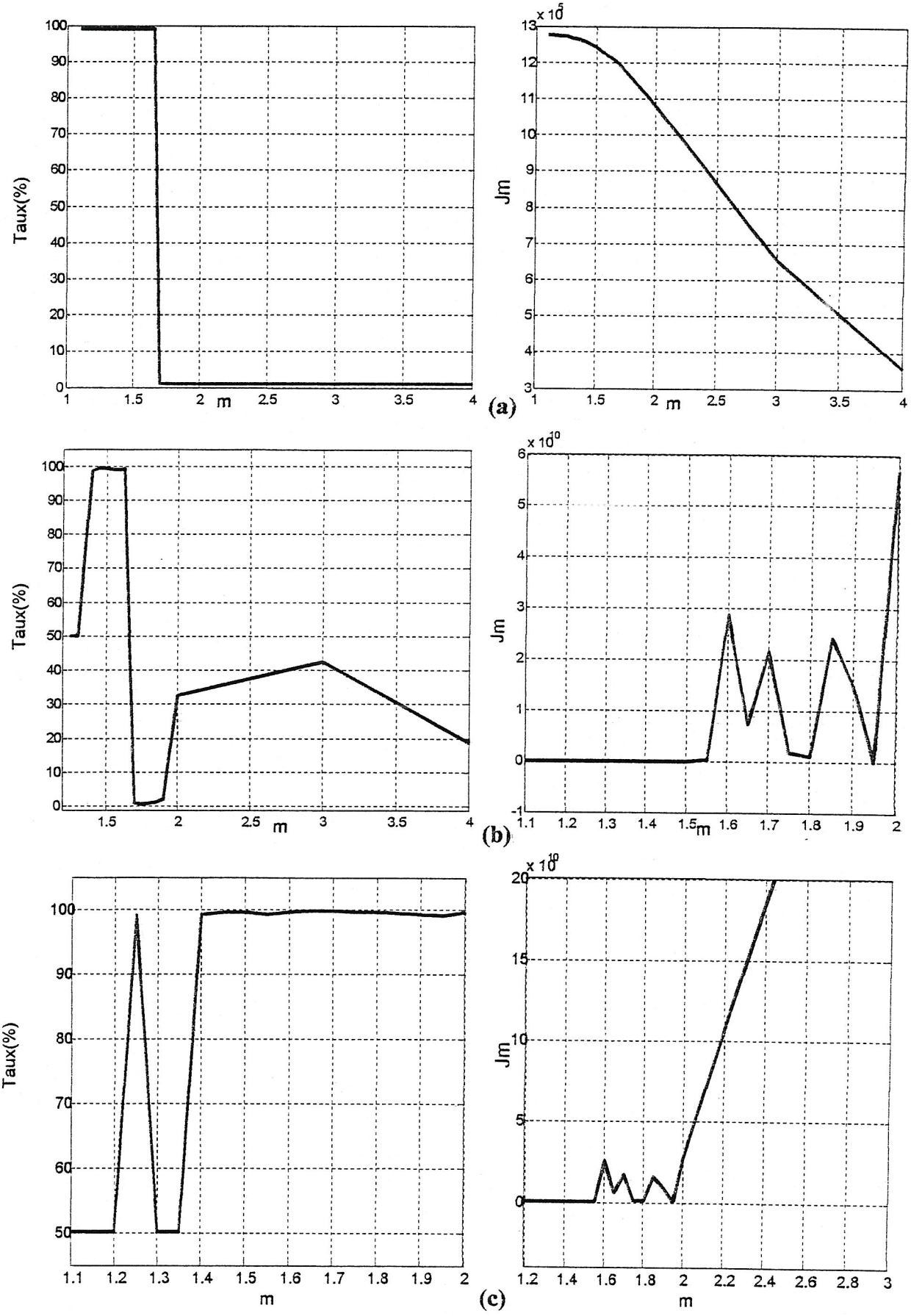
(a) (b)

Figure IV.4: Variation des degrés d'appartenance en fonction de la distance normalisée $\left(d_{ik}^2/\eta_i\right)$ dans le cas du PCM et du FPCM.

En analysant les courbes de la figure IV.4, on remarque que pour une valeur de m spécifié, chaque courbe présente deux parties différentes : la première partie correspond à des degrés d'appartenance compris entre 0.5 et 1 et la deuxième partie correspond à des degrés compris entre 0.5 et 0.

La première partie des courbes correspond à des valeurs de $\left(d_{ik}^2/\eta_i\right) \leq 1$, tandis que la deuxième correspond à des valeurs de $\left(d_{ik}^2/\eta_i\right) > 1$ .De plus lorsque $\left(d_{ik}^2/\eta_i\right)$ tend vers 0, u_{ik} tend vers 1, et lorsque $\left(d_{ik}^2/\eta_i\right)$ tend vers ∞, u_{ik} tend vers 0. Dans la figure IV.4.a, pour une valeur de m au voisinage de 1.45, la fonction d'appartenance obtenue avec le PCM descend rapidement, par conséquent les degrés d'appartenance obtenus avec des valeurs de $\left(d_{ik}^2/\eta_i\right) \leq 0.5$ sont très élevés par rapport à ceux obtenus avec $\left(d_{ik}^2/\eta_i\right) \succ 0.5$. Si la valeur de m augmente, alors la vitesse de décroissance de la fonction d'appartenance ralentit. Dans la figure IV.4.b, toutes les courbes des fonctions d'appartenance obtenues avec le FPCM descendent rapidement comparativement à celles obtenues avec le PCM. Pour une valeur de m au voisinage de 1.635 le degré d'amortissement de la fonction d'appartenance est très élevé par rapport aux autres.

Les courbes représentées par la figure IV.5, illustrent la variation du taux de réussite de la classification et de l'indice de satisfaction en fonction du facteur de fuzzification m pour le FCM et le PCM et le FPCM.



FigureIV.5 a), b), c) : Variation du taux de réussite de la classification et de l'indice de satisfaction en fonction du facteur de fuzzification pour le FCM, le PCM et le FPCM.

Les courbes de la figure IV.5.a montrent que pour une valeur de m inférieur 1.65 et supérieur à 1, le FCM atteint un taux de classification égale à 99% avec un coût qui varie entre $1.2.10^6$ et $1.27.10^6$, tandis que pour des valeurs de m supérieur à 1.7 le taux de classification vaux 1%.

La valeur de m est choisie de telle sorte que le taux de classification soit maximal est le coût soit minimal [31], en effet, le choix d'une valeur de 1.65 est celle qui répond à nos exigences.

Dans la figure IV.5.b pour une valeur de m de 1.45 le PCM atteint son maximum des taux de réussite de 99.5% et son minimum des coûts de $3.34.10^5$.

Dans la figure IV.5.c pour une valeur de m de 1.653 le FPCM atteint son maximum des taux de réussite de 99.75% et un coût de $5.96.10^9$.

À partir des deux figures IV.4 et IV.5, on remarque la compatibilité des résultats obtenus avec le PCM et aussi avec le FPCM. Dans le cas du PCM, et avec une valeur de m estimée à 1.65, le degré d'affaiblissement de la fonction d'appartenance obtenue pour les différentes valeurs de la distance normalisée est justifiable, du fait que les points qui sont proches de la classe correspondante sont affectées à cette dernière avec des grands degrés d'appartenance, alors que ceux qui sont éloignés sont assignés à cette même classe avec des degrés d'appartenance faibles, de plus pour cette même valeur de m le taux de réussite de la classification atteint sa valeur maximale et le coût atteint sa valeur minimale.

Avec le FPCM, la situation est encore mieux ; une valeur de m estimée à 1.635 donne le meilleur taux de réussite qui vaut 99.75% et pour cette même valeur de m la fonction d'appartenance décroît rapidement, ce qui permet aux points situés au voisinage d'une classe d'appartenir à cette dernière avec des degrés plus forts comparativement aux point éloignés qui sont assignés à cette classe avec des degrés plus faibles.

IV.4.2 Résultats de la classification sur la base de donnée IRIS

IV.4.2.1 Initialisation par centres de gravité

On choisit une partition initiale de Iris en partant des centres de gravité plus au moins représentatifs des classes.

Les centres de gravité qui servent à l'initialisation de chaque algorithme sont les suivants :

$$V_{FCM} = \begin{pmatrix} 5.11 & 5.22 & 1.71 & 1.41 \\ 5.53 & 6.51 & 4.63 & 2.15 \\ 4.80 & 3.53 & 5.40 & 7.50 \end{pmatrix} \qquad V_{HCM} = \begin{pmatrix} 5.84 & 3.04 & 3.72 & 1.18 \\ 5.82 & 3.12 & 3.73 & 1.22 \\ 5.86 & 2.99 & 3.82 & 1.19 \end{pmatrix}$$

Les différentes partitions de la base de donnée Iris générées par les quatre algorithmes de classification sont représentées dans la figure IV.6.

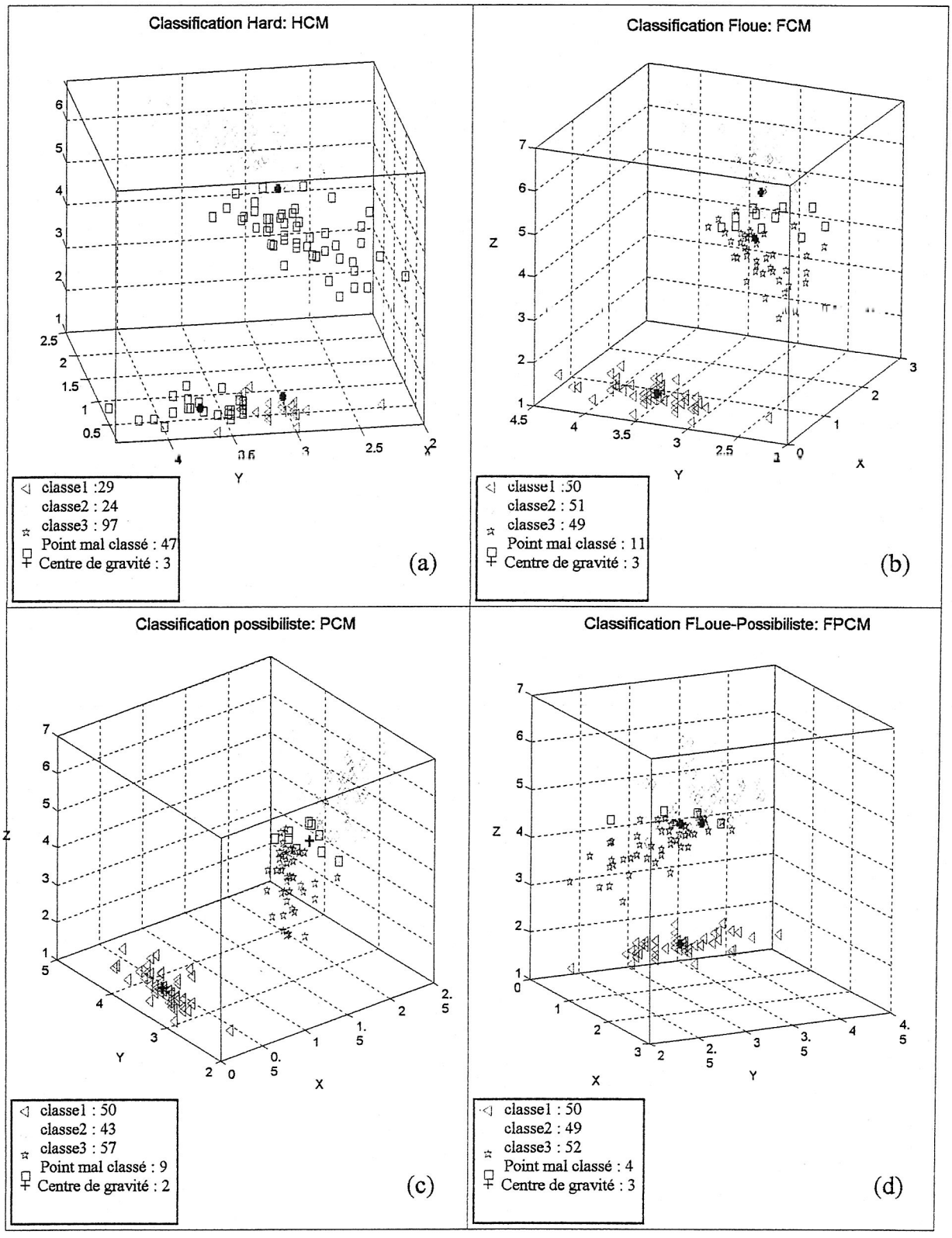


Figure IV.6 : a), b), c), d) Représentation dans l'espace des attributs les différentes partitions de la base de données Iris générées respectivement par le HCM, le FCM, le PCM et le FPCM.

Le taux de réussite ainsi que l'indice de satisfaction et le nombre d'itérations correspondants à chaque algorithme de classification sont donnés dans le tableau suivant :

Algorithme de classification	Taux de réussite (%)	Indice de satisfaction (J_m)	Nombre d'itérations
HCM	50.66	$1.43 .10^2$	06
FCM	92.66	$4.32.10^{-6}$	35
PCM	94.00	9.42.10	51
FPCM	97.33	0.24	13

Tableau IV.6: Comparaison entre les résultats des classifications obtenus avec les quatre algorithmes HCM, FCM, PCM, FPCM.

À la lecture du tableau IV.6, on remarque que le taux de classification obtenu avec le FPCM est égal à 97.33% après 13 itérations, cependant le PCM atteint 94.00% après 51 itérations, tandis que le FCM et HCM atteignent respectivement des taux de 92.66% et 50.66% pendant 35 et 6 itérations.

IV.4.1.2 Initialisation par matrice d'appartenance

Les résultats de classification de la base de donnée Iris dans le cas de l'initialisation par matrice d'appartenance pour les différents algorithmes sont donnés dans le tableau IV.7.

Algorithme de classification	Taux de réussite (%)	Indice de satisfaction (J_m)	Nombre d'itérations
HCM	50.66	$1.43\ 10^2$	06
FCM	92.66	$4.32.10^{-6}$	34
PCM	94.00	9.42.10	51
FPCM	97.33	0.24	13

Tableau IV.7 : Comparaison entre les différents résultats des classifications obtenus avec les quatre algorithmes HCM, FCM, PCM et le FPCM.

En analysant les résultats du tableau IV.7, on remarque que les taux et les coûts de classification obtenus dans le cas de l'initialisation par matrice d'appartenance restent inchangeables comparativement au cas de l'initialisation par centres de gravité. Cependant le nombre d'itérations a diminué dans le cas du FCM d'une seule itération et reste le même dans le cas du HCM, PCM et FPCM.

La matrice de confusion correspondante aux résultats de classification pour chaque algorithme est donnée par le tableau IV.8.

	C1	C2	C3
C1	26	24	0
C2	3	0	47
C3	0	0	50

HCM
Erreur totale 47

	C1	C2	C3
C1	50	0	0
C2	0	45	8
C3	0	6	44

FCM
Erreur totale 11

	C1	C2	C3
C1	50	0	0
C2	0	42	8
C3	0	1	49

PCM
Erreur totale 9

	C1	C2	C3
C1	50	0	0
C2	0	47	3
C3	0	1	49

FPCM
Erreur totale 4

Tableau IV.8 : Matrices de confusion

Les matrices de confusion qui sont représentées par le tableau IV.8, montrent que le HCM reconnaît à 100% la troisième classe (C3) et commet une erreur totale de 47 lors de la classification des objets de la première et de la deuxième classe. Cependant les trois algorithmes restants reconnaissent à 100% la première classe (C1) et commettent des erreurs lors de la classification des objets des deux classes restantes (C2 et C3). L'erreur totale qui correspond respectivement aux FCM, PCM, et le FPCM est de 47, 11, 9 et 4.

Les centres finaux générés par le HCM, FCM, PCM et le FPCM dans le cas de l'initialisation par matrice d'appartenance sont donnés par le tableau IV.9.

centres	V1	V2	V3
HCM	[4.78 3.07 1.62 0.29]	[5.28 3.71 1.49 0.28]	[6.30 2.89 4.96 1.67]
FCM	[5.02 3.39 1.51 0.25]	[6.02 2.87 4.48 1.46]	[6.49 2.99 5.24 1.89]
PCM	[5.06 3.43 1.46 0.24]	[6.17 2.88 4.76 1.60]	[6.17 2.88 4.76 1.60]
FPCM	[5.10 3.45 1.45 0.20]	[6.10 2.95 4.65 1.40]	[6.05 3.00 4.85 1.80]

Tableau IV.9: Comparaison entre les différents résultats des classifications obtenus avec les quatre algorithmes HCM, FCM, PCM et le FPCM.

À partir du tableau IV.9, on remarque d'une part la coïncidence des centres des deux dernières classes si on effectue la classification avec le PCM (i.e. V_2=V_3), alors qu'avec le HCM, FCM et le FPCM on aboutit à une bonne séparation des centres. D'autre part, le HCM génère des centres qui sont très éloignés comparativement à ceux obtenus avec le FCM, PCM et le FPCM qui sont très voisins.

Les courbes représentées par la figure IV.7, illustrent la variation du taux de réussite de la classification et de l'indice de satisfaction en fonction du facteur de fuzzification m pour le FCM, le PCM et le FPCM.

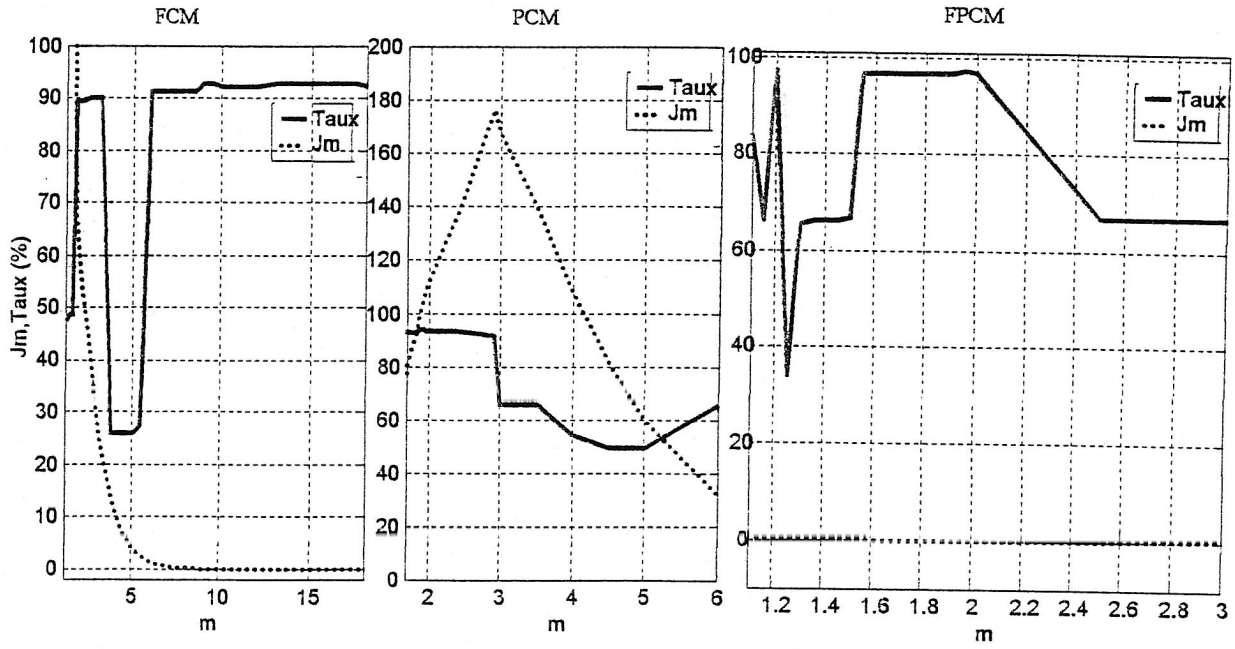


Figure IV.7: Variation du taux de réussite de la classification et de l'indice de satisfaction en fonction du facteur de fuzzification pour le FCM, le PCM et le FPCM.

En analysant les courbes de la figure IV.7, la valeur de m qui correspond à un indice de satisfaction minimal (J_m) et un taux de réussite maximal pour chaque algorithme de classification est donnée dans le tableau suivant :

Algorithme	m	Taux	J_m
FCM	17.697	92.66	$4.32.10^{-6}$
PCM	1.823	94.00	9.42.10
FPCM	1.230	97.33	$2.40.10^{-1}$

Tableau IV.10: Valeurs optimales de m pour le FCM, PCM et le FPCM.

IV.5 Conclusion

Les résultats de la classification de la base de données Iris ont permis de déduire les conclusions suivantes :

Avec un taux de réussite de 97.33%, avec un nombre d'itérations égal à 13 et un coût minimal de 0.24, obtenus par l'approche proposée à savoir le FPCM, on peut conclure que cette dernière est nettement meilleure par rapport aux FCM, PCM et HCM. Le HCM qui est basé essentiellement sur la détermination d'une partition dure, se trouve confronté à des classes chevauchantes (la deuxième et la troisième) et conduit à une mauvaise classification de 50.33%. Pour surmonter ce type de contrainte, le FCM propose la notion d'appartenance d'un objet à une classe, de telle sorte qu'il est partagé entre les différentes classes tout en respectant la contrainte d'inspiration probabiliste [32]. Ces nouvelles notions apportent des améliorations considérables avec un taux égal à 92.66%; malgré cette supériorité, le FCM

rencontre des difficultés en présence des données bruitées ce qui dégrade les performances du classifieur. Pour résoudre le problème des données bruitées. Le PCM propose la substitution de la notion de partage par celle de typicalité absolue; dans ce cas, on aboutit à des classes ayant les mêmes centres, et finalement un échec dans la classification. L'approche proposée (FPCM), construit sur la base des deux algorithmes le PCM et le FCM, est robuste au bruit, génère des centres séparables, résout le problème de chevauchement et converge rapidement.

Les résultas de classification obtenus avec l'image des textures et Iris confirment la supériorité de notre approche. Le FPCM atteint un taux de réussite de 99.75% et un nombre d'itérations égale à 6, tandis que le PCM présente des résultas meilleurs comparativement au FCM avec un taux de réussite de 99.5 % et un coût minimal de $3.34.10^{5}$. Cependant, l'approche classique montre son échec dans la classification avec un taux réussite de 1%.

Malgré que l'initialisation par centres de gravité donne des résultas satisfaisants, elle reste une méthode aléatoire, par contre l'initialisation par matrice d'appartenance est une méthode qui se base sur la notion de partage équiprobabiliste vu qu'on ne dispose pas connaissances a priori sur l'appartenance des objets aux différentes classes, De plus elle est valable pour toute base de donnée.

Le paramètre m qui contrôle l'introduction du flou dans la partition est choisi de telle sorte que le degré d'affaiblissement de la fonction d'appartenance est justifiable. Cette valeur change d'un algorithme de classification à un autre. Dans la classification des textures et avec le PCM, le taux de réussite atteint son maximum et le coût son minimum pour une valeur de m égale à 1.45 qui correspond à une bonne fonction d'appartenance. Le FPCM génère des degrés d'appartenance plus raisonnables, la meilleure valeur de m est de 1.635. Avec le FCM une valeur de m de 1.65 correspond à un coût minimal et un taux de réussite maximal. Dans le cas de la classification d'Iris, les valeurs optimale de m qui correspondent aux FCM, PCM, FPCM sont respectivement égales à 17.697, 1.823 et 1.230.

Le choix du paramètre m dépend de la nature de l'ensemble à partitionner; ainsi, si cette dernière présente beaucoup d'ambiguïté, alors on doit introduire beaucoup du flou dans la partition pour arriver à une bonne séparation des classes. Ceci nécessite le renforcement du paramètre m qui contrôle l'introduction du flou dans la partition.

Conclusion Générale

Dans ce mémoire, nous avons présenté quelques approches de classification automatique non supervisée par les méthodes des C-Moyennes et hiérarchiques.

La méthode des C-Moyennes classiques qui est basée essentiellement sur la partition dure a conduit à une mauvaise classification des données en particulier dans le cas où les frontières sont mal définies.

Contrairement à l'approche classique, les C-Moyennes Floues qui sont basés sur la notion de partage probabiliste sont robustes en présence de classes chevauchante mais sensible aux bruits.

Pour surmonter le problème du bruit, l'approche des C-Moyennes possibilistes utilise la notion de typicalité ; cependant il génère des centres identiques.

Toutes les méthodes décrites précédemment ont pour objectif de trouver des partitions de même niveaux, par contre les méthodes hiérarchiques ont pour but de construire une hiérarchie de partition qui constitue un arbre de partition.

Pour contourner les problèmes rencontrés dans le contexte de l'application des C-Moyennes, nous avons proposé une nouvelle approche basée sur la fusion des théories du flou et des possibilités avec une nouvelle méthode d'initialisation des algorithmes qui est basée sur le concept d'appartenance équiprobabiliste des objets aux différentes classes.

La nouvelle approche permet de résoudre d'une part le problème de chevauchement, de bruit, de la coïncidence et d'autre part d'accélérer le temps de classification.

En perspective, nous envisageons :

1. De modifier le critère d'optimisation dans le sens où la partition générée doit vérifier une bonne propriété de clustring.

2. De déterminer un bon critère pour la validation de la partition.

Bibliographie

[1] L. A. Zadeh, "Fuzzy Sets", Inform. Control, 1965, 8, 338-353.

[2] L. A. Zadeh, 'Fuzzy sets as basic for the theory of possibility', Fuzzy sets and Systems, 1978, vol. 1, 3-28

[3] L. A. Zadeh, 'La logique floue et ces applications', traduit de l'américain par A. benmakhlouf.

[4] J. C. Bezdek "A Review of Probabilistic, Fuzzy, and Neural Models for Pattern Recognition", *Journal of Intelligent and Fuzzy Systems*, vol.1, pp. 1-25, 1993.

[5] J. Thioulous, D. Chessel & A.B. Dufour «Classification automatique » fiche de Biostatique - Stage 7.

[6] Classification automatique et reconnaissance de formes, fichier Internet, http://www.inria.fr/rapportsactivite/RA95/colorec.html.

[7] Fuzzy models for pattern recognition, J. C. Bezdek and S. K. Pal. Eds. New York: IEEE Press. 1992.

[8] Gabriel Dos Santos, « partie III classification automatique, carte d'occupation des sols dans la vallée de Biriatou », Rapport de stage, avril/juin 2001.

[9] J. C. Bezdek, "Optimisation of Clustering Criteria by Reformulation", *IEEE Transaction of Fuzzy Systems*, vol.3, pp. 241-245, 1995.

[10] J. C. Bezdek, N. R. Pal, "Two Soft Relatives of Learning Vector Quantization", *Neural Networks*, vol. 8, pp. 729-743, 1995.

[11] R. N. Dave, "characterization and detection of noise in clustring," *pattern recognition.*

[12] G. Shafer, *A mathematical theory of evidence*. Princeton, NJ: Princeton University Press. 1979.

[13] D. Dubois and H. Prade, *Possibility theory: An approach to computerized Processing of uncertainty*, New York: Plenum Press, 1988.

[14] G.klir and T.Folger, *Fuzzy Sets, uncertainty, and information. Englewood Cliffs, NJ: Prentice-Hall* , 1988, chap.4.*Lett.*, vol 12, no. 11, pp. 657-664, 1992.

[15] H.-J. Zimmerman and P. Zysno, "Quantifying vagueness in decision models," *European J. Operational Res.*, vol.22, pp.148-158, 1985.

[16] R. Krishnapuram and J. M. Keller, "The Possibilistic C-Means Algorithm: Insights and Recommendations", *IEEE Trans. Fuzzy Syst*ems, vol. 4, pp. 385-396, August 1996.

[17] R. Krishnapuram and J. M. Keller, "A Possibilistic Approach to Clustering", *IEEE Trans. Fuzzy Systems,* vol. 1, pp. 98-110, May, 1993.

[18] R. Krishnapuram and C.-P. Freg, "Fitting an unknown number of lines and planes to image data through compatible

[19] R. N. Dave "Fuzzy -shell clustring and application to circle detection in digital images," *Int .J.General Sytems*, vol.16, pp.343-355, 1990.

[20] R. Krishnapuram, H. Frigui, and O. Nasraoui, "New fuzzy shell clustring algorithms for boundary detection and pattern recognition," in *proc. SPIE Conf. Intelligent Robots and computer vision X: Algorithms and techniques* (Boston), Nov.1991, pp. 458-465.

[21] Chavent M. (1998), A monotonic clustring method, pattern Recognition Letters 19, pp.989-996.

[22] M.Barni, V. Cappellini, end A. Mecocci, "Comments on 'A Possibilistic Approach to Clustering,'"IEEE Conf, Fuzzy Syst., Orlando, FL, july 1994, pp.902-908.

[23] algorithmes de construction hiérarchiques cherchent à optimiser le critère des moindres carrés, thèse, université Aix-Marseille III

[24] Edwards et Cavalli-SforzaL.L(1965), A method for cluster analysis, Biometrics 21, pp.362-375.

[25] Chavant M. (1997), analyse des données symboliques; une méthode division de classification, Thèse, Université Pris IX-Dauphine.

[26] Chavant M. (1998), A monotonic clustring method, pattern recognition Letters 19, pp.989-996.

[27] H. Frigui, R. Krishnapuram, "Clustring by competitive agglomeration" Pattern recognition letters, 1997, vol.30, 1109-1119.

[28] H.Boudouda et H.Seridi "Une Nouvelle Approche De Classification Automatique Non Supervisée Par C-Means : Fusion Des Algorithmes Flou Et Possibiliste ", *proc. 2 ème conférence internationale.* SETIT, Tunisie, 15-20 Mars 2004, p. 163.

Résumé

Devant la masse d'informations qui ne cesse de croître de manière exponentielle, l'expert humain est souvent confronté à des problèmes de classification de données qui rentrent dans le cadre de la reconnaissance des formes. Les méthodes de classification sont généralement le fruit d'un formalisme basé sur un raisonnement artificiel qui est plus au moins proche de celui d'un être humain. Il existe plusieurs méthodes de classification non supervisée qui se démarquent les unes des autres par le concept d'appartenance d'un objet à une classe. Dans ce même contexte, nous présentons une nouvelle approche de classification automatique non supervisée sous la famille des C-Moyennes. Cette nouvelle approche basée sur la fusion des théories du flou et des possibilités, permet d'une part de résoudre simultanément le problème de coïncidence et du bruit et d'autre part d'accélérer le temps de classification. La méthodologie d'initialisation utilisée dans cette nouvelle approche est basée sur le concept d'appartenance équiprobabiliste des objets aux différentes classes.

Mots clés : Classification, logique floue, reconnaissance des formes, raisonnement approximatif, apprentissage supervisé.