Classification automatique et modèles

Yamina Bencheikh

LABORATOIRE DE RECHERCHE EN INFORMATIQUE
ET MATHEMATIQUE DE METZ

THESE

Présentée et soutenue publiquement à

L'UNIVERSITE DE METZ

Pour l'obtention du grade de

DOCTEUR DE L'UNIVERSITE DE METZ

SPECIALITE : MATHEMATIQUES APPLIQUEES

MENTION : ANALYSE DE DONNEES ET INFORMATIQUE

par

YAMINA BENCHEIKH

Sujet

CLASSIFICATION AUTOMATIQUE ET MODELES

Soutenue le 10 février 1992 devant la commission d'examen :

Président :	G. GOVAERT, Professeur à l'Université de COMPIEGNE.
Rapporteurs :	E. DIDAY, Professeur à l'Université de PARIS.
	J.M. PROTH, Directeur de recherches à l'INRIA de METZ.
Examinateurs :	D. ARNAL, Professeur à l'Université de METZ.
	A. ROUX, Professeur à l'Université de METZ

LABORATOIRE DE RECHERCHE EN INFORMATIQUE
ET MATHEMATIQUE DE METZ

THESE

Présentée et soutenue publiquement à

L'UNIVERSITE DE METZ

Pour l'obtention du grade de

DOCTEUR DE L'UNIVERSITE DE METZ

SPECIALITE : MATHEMATIQUES APPLIQUEES

MENTION : ANALYSE DE DONNEES ET INFORMATIQUE

par

YAMINA BENCHEIKH

Sujet

CLASSIFICATION AUTOMATIQUE ET MODELES

Soutenue le 10 février 1992 devant la commission d'examen :

Président :	**G. GOVAERT, Professeur à l'Université de COMPIEGNE.**
Rapporteurs :	**E. DIDAY, Professeur à l'Université de PARIS.**
	J.M. PROTH, Directeur de recherches à l'INRIA de METZ.
Examinateurs :	**D. ARNAL, Professeur à l'Université de METZ.**
	A. ROUX, Professeur à l'Université de METZ

A tous ceux qui me sont chers.

REMERCIEMENTS

A Monsieur G. Govaert, Professeur à l'Université de Compiègne.

Vous m'avez accueilli avec sympathie et bienveillance, vous avez bien voulu me conseiller et m'aider dans la réalisation de ce travail. Qu'il me soit permis de vous exprimer ma gratitude et ma profonde reconnaissance.

Soyez assuré de mon profond respect. Que votre droiture et votre compétence me servent d'exemple pour continuer mon chemin dans la recherche, qui ne fait que commencer.

A Monsieur E. Diday, Professeur à l'Université de Paris.

Avec beaucoup de bienveillance et malgré vos nombreuses obligations professionnelles, vous avez accepté de juger ce travail. Veuillez trouver ici l'expression de ma vive admiration et ma profonde reconnaissance.

A Monsieur J.M. Proth, Directeur de recherches à l'INRIA de METZ.

Permettez-moi, à l'occasion de cette soutenance de vous manifester toute mon admiration pour vos qualités humaines et professionnelles, votre dynamisme et votre gentillesse. Je vous remercie profondément de l'honneur que vous m'avez fait en acceptant de juger ce travail.

A Monsieur D. Arnal, Professeur à l'Université de METZ

Avec beaucoup de gentillesse et de sympathie, vous avez accepté de juger ce travail, je vous exprime à cette occasion ma profonde reconnaissance et mes vifs remerciements. Qu'il me soit permis de vous exprimer toute ma gratitude d'avoir accepté, malgré vos multiples tâches, d'honorer de votre présence ce jury de thèse.

A Monsieur A. ROUX, Professeur à l'Université de Metz.

Vous m'avez fait le grand honneur en participant à la commission d'examen de cette thèse, je vous prie de bien vouloir accepter l'expression de ma profonde gratitude, en vous remerciant pour l'intérêt que vous portez à ce travail.

Au gouvernement Algerien

Qui sans son aide matérielle, ce travail n'aurait pas vu le jour. Je vous exprime à cette occasion ma profonde reconnaissance et mes vifs remerciements.

A tous mes collègues de travail, F. Marchetti, M. Nadif et Y. Lemoine.

Vous étiez toujours présents aux moments difficiles, vous m'avez écouté et conseillé sans peine, ni relâche. Je ne saurais oublier votre aide et votre soutien moral qui ne m'ont jamais fais défaut.

Une pensée particulière à tous les membres du département d'informatique de Metz en particulier à messieurs B. Heuilluy chef de département et M. Grandmougin directeur des études pour leur gentillesse et leur disponibilité à toute épreuve.

TABLE DES MATIERES

TABLE DES MATIERES

INTRODUCTION 2

PARTIE A : CLASSIFICATION SIMPLE

CHAPITRE 1 :
CLASSIFICATION AUTOMATIQUE ET MELANGES

Introduction 11

1. Classification automatique 12
1.1. Notations 13
1.2. La méthode des Nuées Dynamiques 14
1.3. L'algorithme 15

2. Les mélanges 15
2.1. Formalisation du problème 16
2.2. La méthode de reconnaissance des composants d'un mélange 16
2.2.1. Définition de W 17
2.2.2. Définition de f 17
2.2.3. Définition de g 18

3. Généralisation 18
3.1. Etude des liens entre les critères métriques et les critères probabilistes 18
dans le cas continu 19
3.1.1. Critère métrique 19
3.1.2. Critère probabiliste 20
3.1.2.1. Identification d'un mélange 20
3.1.2.2. Approche classification 20
3.1.3. Etude des liens entre les deux critères 21
3.1.3.1. Critère métrique associé à un critère probabiliste 21
3.1.3.2. Critères probabilistes et métriques équivalents 22
3.1.3.3. Condition pour qu'un critère métrique soit associé à un critère probabiliste 22
3.1.3.4. Critère probabiliste équivalent à un critère métrique 22

3.1.4. Métrique quadratique et lois gaussiennes 23
3.1.4.1. Métrique quadratique fixe et identique pour toutes les classe 23
3.1.4.2. Métrique quadratique variable et dépendante de chaque classe 24
3.1.5. Métrique euclidienne et lois gaussiennes 25
3.1.6. Métrique de type L_1 25
3.1.6.1. Distance fixe et identique pour toutes les classes 25
3.1.6.2. Distance L_1 variable et dépendante de chaque classe 26
3.2. Etude des liens entre les critères métriques et les critères probabilistes dans le cas discret 26
3.2.1. Critère métrique associé à un critère probabiliste 27
3.2.2. Condition pour qu'un critère métrique soit associé à un critère probabiliste 27
3.2.3. Critère probabiliste équivalent à un critère métrique 27
3.2.4. Métrique L_1 et distribution de Bernoulli 28
3.2.4.1. Distance L_1 fixe et identique pour toutes les classes 28
3.2.4.1. Distance L_1 variable et dépendante de chaque classes 29
3.2.4.3. Distance adaptative L_1 identique pour toutes les classes 29
3.2.5. Données qualitatives nominales 29
3.2.5.1. Distance fixe et identique pour toutes les classes 30
3.2.5.2. Distance variable et dépendante de chaque classe 30
3.2.6. Données qualitatives ordinale 30

CHAPITRE 2 :
CLASSIFICATION ET MODELES SUR DONNEES QUALITATIVES

Introduction 32

1. Les tableaux disjonctifs complets 33
1.1. Exemple 33
1.2. Notations et définitions 34
1.3. Critères de classification 35
1.3.1. Critère d'information 35
1.3.2. Critère du Khi2 36
1.3.3. Etude du lien entre les deux critères 36
1.4. La méthode MNDQAL 37
1.4.1. L'algorithme 37
1.4.2. Autres expressions du critère 38
1.5. Approche modèle 39

1.5.1. Première approche 39
1.5.2. Deuxième approche 41
1.5.3. Troisième approche 42
1.5.3.1. Notations 42
1.5.3.2. Modèle de Celeux 42
1.6. Conclusion 43

2. Les tableaux de modalités 44
2.1. Notations et définitions 44
2.2. La méthode MNDDIJ 45
2.3. Approche modèle 47
2.3.1. La formule générale 47
2.3.2. Choix de la famille de distribution 48

CHAPITRE 3 :
CLASSIFICATION BINAIRE ET DISTANCE L_1 ADAPTATIVE

Introduction 50

1. La méthode MNDBIN 51
1.1. Exemple 52

2. Modèle associé aux données binaires 53
2.1. La formule générale 54
2.2. Choix de la famille de distribution 54
2.3. Etude du mélange M_2 55
2.4. Etude du mélange M_3 57

3. Problèmes de dégénérescence 59

4. Applications et comparaison des méthodes 62
4.1. Données réelles 62
4.1.1. Description des données 62
4.1.2. Première stratégie 63
4.1.3. Deuxième stratégie 63
4.1.4. Troisième stratégie 64
4.2. Données simulées 64
4.2.1. Le programme 64
4.2.2. Les trois fichiers de données 65

4.2.3. Résultats obtenus 65
4.2.3.1. Les données simul 1 66
4.2.3.2. Les données simul 2 66
4.2.3.3. Les données simul 3 66

5. Conclusion **69**

PARTIE B : CLASSIFICATION CROISEE

CHAPITRE 4 : CLASSIFICATION CROISEE ET MELANGES

Introduction **71**

1. La classification croisée **73**
1.1. Rappels et notations 73
1.2. Le principe de la classification croisée 73
1.3. L'algorithme 74

2. Modèle de mélange croisé **75**
2.1. Exemple illustratif 75
2.2. Modèle général 77
2.2.1. Identification d'un mélange "croisé" 77
2.2.2. Problème à resoudre 78

3. Approche classification **78**
3.1. Algorithme 79
3.2. Position intermédiaire 80

4. Transformation d'un modèle de mélange croisé en un modèle de mélange simple **82**

5. Applications pratiques **85**
5.1. Lois gaussiennes unidimensionnelles 85
5.2. Lois de Bernoulli 86

CHAPITRE 5 :
CLASSIFICATION CROISEE ET MODELES SUR DONNEES BINAIRES

Introduction 88

1. La méthode CROBIN 89
1.1. Le problème 90
1.2. L'algorithme 91
1.2.1. Les étapes intermédiaires 91
1.2.2. Convergence de l'algorithme 93
1.3. Exemple 93

2. Modèle associé aux données binaires 94
2.1. La formule générale 95
2.2. Choix de la famille de distribution 95

3. Extension du modèle binaire 96
3.1. Etude du mélange M_2 96
3.2. Etude du mélange M_3 98
3.3. Etude du mélange M_4 99

4. Problèmes de dégénérescence 101

5. Interprétation des bons résultats obtenus par la méthode CROBIN sur des données simulées 102

6. Conclusion 105

CHAPITRE 6 :
CLASSIFICATION CROISEE ET MODELES SUR DONNEES QUANTITATIVES

Introduction 106

1. La méthode CROEUC 107
1.1. Notations 107
1.2. Le problème 108
1.3. L'algorithme 109
1.4. Cas particulier 110

2. Modèle associé aux données quantitatives **111**
2.1. La formule générale 111
2.2. Choix de la famille de distribution 111

3. Conclusion **116**

CONCLUSION **118**

ANNEXE 1 **121**
ANNEXE 2 **128**
ANNEXE 3 **133**
ANNEXE 4 **136**

BIBLIOGRAPHIE **140**

INTRODUCTION

INTRODUCTION

A l'heure actuelle les modèles mathématiques ont reçu leurs lettres de noblesse dans de nombreux domaines des sciences et des techniques, mais certains esprits, notamment des médecins, des sociologues, des économistes affirment encore que les phénomènes étudiés sont trop complexes pour être adéquatement exprimés par des méthodes mathématiques.

Le modèle mathématique est une représentation simplifiée de la réalité. Tout l'art de la modélisation est de savoir quoi, où, quand et comment simplifier. L'étude d'un modèle probabiliste peut être conduite de deux façons :
Par les méthodes mathématiques issues de la théorie des probabilités et de la statistique, ou par la reproduction du fonctionnement de ce modèle sur ordinateur ; le second procédé s'appelle simulation statistique.

La simulation statistique est un puissant outil de manipulation des modèles probabilistes à toutes les étapes de la recherche. En théorie des files d'attente, par exemple, c'est la principale méthode de résolution des systèmes complexes. En statistique classique, c'est l'une des méthodes d'étude de la stabilité des estimations par rapport au fluctuations des hypothèses de base ; elle est utilisée seule ou conjointement à des méthodes analytiques asymptotiques.

Les modèles probabilistes sont un puissant instrument de description qualitative des relations liant les phénomènes et faits observés, d'étude des propriétés des systèmes envisagés, de choix d'un appareil statistique pour le traitement des données et l'organisation de la collecte des données. Tout modèle mathématique est une représentation simplifiée de la réalité et tout l'art du chercheur est de conjuguer la paramétrisation la plus simple du modèle à une description adéquate de la réalité, en d'autre termes, il doit " compresser " au maximum la réalité sous une forme mathématique simple.

La procédure de modélisation peut être conventionnellement partagée en cinq étapes principales :

1/ Définition des objectifs de la modélisation, des variables du modèle et de leur rôle .

2/ Analyse de la nature physique du phénomène étudié, collecte et formalisation de l'information initiale.

3/ Modélisation proprement dite (détermination de la forme générale du modèle).

4/ Analyse statistique du modèle : estimation des paramètres participant à la description du modèle.

5/ Vérification du modèle.

Une condition nécessaire pour le bon fonctionnement d'un modèle est la réalisation d'une analyse minutieuse de la nature du phénomène étudié afin de recueillir une information initiale fiable et d'en tirer le meilleur parti pour la déduction (ou le choix) de la forme générale du modèle cherché.

Un nombre croissant d'auteurs commencent à s'intéresser aux rapports de ces modèles avec les méthodes de classification automatique. Ces méthodes donnent, à partir d'un échantillon multidimensionnel, une description de la population qui doit ensuite être interprétée par le praticien et le statisticien ; selon la technique utilisée, cette description peut être un graphe d'analyse factorielle ou bien une partition ou encore une arborescence issue d'une classification. On distingue grossièrement deux techniques de classification :

- Techniques de classification hiérarchique (Johnson 1967, Lance et Williams 1967, Jardine et Sibson 1968, Sokal et Michner 1968, Lerman 1981).

- Techniques de classification non hiérarchique (Ball et Hall 1965 et 1967, Forgy 1965, Régnier 1965, Mac queen 1967, Diday 1972, Anderberg 1973).

Ces méthodes ont étés conçues dans un cadre géométrique sans référence en général à des modèles probabilistes.

D'autre part, le problème peut être posé, d'adapter par une technique convenable un modèle stochastique à un phénomène observé.

Par exemple, si K est le nombre de composants d'un mélange et ($f(., \lambda)/ \lambda \in L$) est la famille de lois de probabilité à laquelle appartiennent les distributions des différents composants, la densité du mélange s'écrit :

$$f(x) = \sum_{k=1}^{K} p_k \, f(x / \lambda_k)$$

où $f(x / \lambda_k)$ est la densité de la $k^{ème}$ composante du mélange et f (x) est la densité de la loi de probabilité résultante, p_k est la probabilité à priori d'apparition dans un échantillon aléatoire d'une observation de la loi $f(x/\lambda_k)$, (c'est à dire le poids spécifique de telle observation dans la population générale), k est le nombre de composantes du mélange .

L'analyste est confronté à de telles lois de probabilité lorsque, par exemple, il est amené à analyser une population générale composée de plusieurs sous-populations qui tout en étant homogènes dans un certain sens (ce qui peut s'exprimer par exemple, par la nature unique de la loi de probabilité $f(x/\lambda_k)$ sont fondamentalement différente l'une de l'autre (par exemple, par la valeur du paramètre λ_k). Le paramètre λ_k peut définir aussi bien le centre de regroupement des observations correspondantes (auquel cas il s'interprète comme un paramètre de localisation) que le degré de leur dispersion aléatoire (il est alors interprété comme un paramètre d'échelle). On peut trouver de plus amples informations sur les mélanges de lois dans (Diday E. et collaborateurs 1980).

Le problème consiste à estimer le nombre de composants du mélange et les paramètres inconnus $((p_k , \lambda_k)\ k = 1, K)$ au vu de l'échantillon. Ce problème a été étudié par de nombreux auteurs sous des hypothèses plus ou moins restrictives et sous deux approches foncièrement différentes.

L'approche la plus ancienne et la plus répandue consiste à voir là un simple problème d'estimation de paramètres, le problème ainsi posé est celui dans les articles anglo-saxons traitent sous le nom de " Mixtures Résolution ". Un nombre important de techniques existent pour résoudre les " Mixtures ". On distingue grossièrement deux types :

- Les techniques d'estimation, qui posant à priori le modèle ci-dessus, en estiment les paramètres à l'aide d'estimateurs calculés sur les observations : citons la méthode des moments (Pearson 1894) avec estimateurs du maximum de vraisemblance, (Rao 1948, Day 1969) avec estimateur du Khi2 minimum. Ces méthodes s'appliquent en général aux mélanges gaussiens et sont souvent restreintes aux distributions unidimensionnelles.

La méthode de Cooper et Cooper (1964), estiment les paramètres inconnus du modèle à partir des moments de la distribution globale observée, cette approche est sensiblement différente du problème d'estimation du modèle précédent.

- Les techniques de type bayesien, d'apprentissage, etc... qui procède par approximations successives, liées à l'introduction des observations pour estimer le modèle précédent. Citons les travaux de Patrick et Hancock (1966), Patrick et Costello (1970), Agrawala (1970) qui sont des techniques d'estimation bayesienne et les travaux de Agrawala (1970), Patrick (1972), Duda et Hart (1973) qui formalisent le problème de la résolution des mélanges en termes d'apprentissage avec ou sans maître.

Dans le cas particulier des mélanges gaussiens unidimensionnels, Benzécri (1972) propose une méthode basée sur une série de déconvolutions successives.

La deuxième approche considère qu'il s'agit d'un problème de classification, citons les travaux de Scott et Symons (1971). Wolfe (1970) formalise de façon originale le problème de la classification en termes d'analyses de mélanges, Schroerder (1974) propose une méthode itératif détectant parallèlement une partition en classes de l'échantillon observé et des distributions associées à ces classes.

Cette idée de la recherche simultanée d'une partition et de "noyaux" caractéristiques des classes de cette partition a été initialement utilisée en classification automatique non hiérarchique : il s'agit de la méthode des Nuées Dynamiques dûe à Diday ; les noyaux sont alors des éléments d'un échantillon à classer. Diday (1972) expose la méthode et propose l'utilisation du même schéma avec des noyaux de diverses types en vue de résoudre des problèmes spécifiques : par exemple, en prenant comme noyaux les éléments principaux d'inertie des classes, la méthode fournira des analyses factorielles locales à fortes inertie (Analyse factorielle typologique (Diday E, Schroeder A et OK Y 1974)). Si les noyaux sont des polynômes d'interpolation d'un point moyen des classes, l'algorithme permet de reconstituer les données manquantes d'un tableau en tenant compte des données présentes pour regrouper les observations en classes et réduire ainsi le nombre d'interpolation à effectuer. Les noyaux peuvent être des métriques (Classification avec distances adaptatives (Diday et Govaert 1977)) ou des distributions de probabilités (A new approach in mixed distributions detection (Diday et Schroeder 1976)).

L'algorithme proposé par Schroeder (1974) utilise des méthodes d'estimation classique, intervient en particulier celle du maximum de vraisemblance qui permet

l'optimisation d'un critère de vraisemblance. La méthode a été généralisé de façon à pouvoir optimiser ce même critère dans les mélanges de distributions dont les paramètres inconnus ne peuvent être calculés par le maximum de vraisemblance, par exemple les mélanges de lois gamma (Schroeder 1976).

Dans notre travail, nous insisterons particulièrement sur l'approche " Classification ". Cette approche présente bien des avantages car elle permet de voir d'un angle nouveau les méthodes de classification automatique et de justifier de manière rigoureuse des constatations faites de manières empirique. En revanche elle présente quelques inconvénients, car elle induit, en général un biais qui peut être important dans l'estimation des paramètres du fait de la connexité des classes. Ce biais persiste lorsque la taille de l'échantillon tend vers l'infini (Bryant et Williamson 1978, Marriott 1975). Pour que ce biais soit négligeable, il faut, d'une part, que les composants du mélanges soient assez séparés, d'autre part, que les fréquences d'apparition des composants du mélanges soient du même ordre.

Lorsqu'il est possible de trouver un modèle de lois de probabilité tel que l'estimation des paramètres du modèle par l'approche classification (Scott 1971, Schroeder 1976, Celeux 1988, Govaert 1988) conduisent à l'optimisation d'un critère numérique de classification, on obtient un éclairage nouveau de ce critère et de la métrique sous-jacente permettant de les justifier ou éventuellement de les rejeter ; par exemple Celeux (1988) a donné une signification au critère d'inertie interclasse, utilisé pour la classification d'individus décrits par des variables quantitatives, pour le modèle de mélange gaussien où les matrices de variances covariances ont toutes la même forme $\gamma.I_d$ où γ est un réel et I_d la matrice identité. Il a aussi apporté une interprétation en termes probabilistes pour le critère d'information utilisé pour la classification d'individus décrits par des variables qualitatives, pour le modèle des classes latentes. Dans le même cadre, Bock (1986) montre que les critères classiques d'information s'interprètent comme des vraisemblances classifiantes de modèles log-linéaires et Govaert (1988) montre que le critère optimisé par la méthode MNDBIN pour les données binaires correspond à un mélange issu de loi de Bernoulli ; en faisant varier le paramètre de tirage de cette loi, il propose une extension de l'algorithme MNDBIN qui utilise des **distances adaptatives de type L_1**. Govaert s'est aussi intéressé aux liens qui existent entre les critères métriques et les critères probabilistes et a vu que la comparaison de ces critères apporte un éclairage nouveau sur de nombreuses méthodes de classification. Cela à permis de justifier a posteriori certaines contraintes imposées souvent pour des raisons techniques d'optimisation, de proposer de nouveau critères, mais peut être encore plus, cette comparaison permet d'expliquer l'intérêt et la souplesse de la méthode des Nuées Dynamiques dont l'idée essentielle était l'utilisation

de la notion de noyau associé à une classe ; ce noyau correspond tout naturellement, avec le critère probabiliste, aux paramètres de la loi de probabilité associé à chaque classe.

Le travail que nous présentons dans cette thèse se situe à mi-chemin entre l'approche géométrique (méthodes de classification automatique) et l'approche probabiliste (les modèles). Nous proposons une application des liens existant entre ces deux types d'approches, sur quelques méthodes de classification automatique. Nous généralisons ces liens au cas où les données mettent en jeu deux ensembles ; c'est le cas de la classification croisée.

Dans le **premier chapitre**, nous rappelons le principe général de la méthode des Nuées Dynamiques (Diday 1972). Nous examinons ensuite une application de cette méthode aux problème des mélanges Schroeder (1974). Nous terminons ce chapitre par une généralisation des liens existant entre l'approche géométrique et l'approche probabiliste aux cas où les données sont continues Govaert (1989) ou discrètes Govaert (1990).

Le deuxième et le troisième chapitre porte sur l'étude de la notion de modèle dans le cas de la classification simple.

Dans le **deuxième chapitre,** nous proposons des interprétations en termes probabilistes de quelques critères liés à la classification de données décrites par des variables qualitatives. Nous étudions, dans un premier temps, les tableaux disjonctifs complets et la méthode MNDQAL (Ralambondrainy 1988) qui est une méthode de classification sur tableau disjonctif complet utilisant la métrique du Khi2 pour classer les données. Nous proposons plusieurs approches pour cette méthode suivant l'optique statistique dans laquelle on se place ; si nous travaillons sur l'ensemble des profils que l'on plonge dans l'espace continu $\mathbf{R}^m$ (où m est le nombre total de modalités) munie de la métrique du Khi2 (que l'on considère comme une métrique quadratique), nous montrons que le critère du Khi2 est lié à un mélange de lois gaussiennes multidimensionnelles de même matrice de variances covariances ayant toutes la forme $\gamma.\mathbf{I_d}$ où γ est un réel et $\mathbf{I_d}$ est la matrice identité. Si maintenant nous travaillons directement sur les données du tableau qui sont des vecteurs binaires de modalités appartenant à l'espace discret $\{0, 1\}^m$, nous montrons qu'il n'existe pas de modèle probabiliste lié au critère du Khi2 minimisé par la méthode MNDQAL. Celeux (1988), en travaillant sur les mêmes données (initiales), à apporté une interprétation en termes probabilistes au critère d'information qui est une quantité proche de celle du Khi2. Nous étudions ensuite la méthode MNDDIJ (Marchetti 1989)

qui s'applique à un tableau de modalité, utilise la distance proposé par Marchetti (1989) qui permet de prendre comme distance entre deux modalités la valeur 0 si on a la même modalité et 1 sinon. Contrairement à la méthode MNDQAL, la méthode MNDDIJ utilise des noyaux ayant la même structure que les données initiales c'est-à-dire que nous imposons aux noyaux d'être des vecteurs binaires de modalités. Nous montrons alors que dans, ce cas nous pouvons supposer que les données du tableau proviennent d'un mélange de produit de p lois binomiales (où p est le nombre total de variables qualitatives que l'on suppose mutuellement indépendantes).

Le **troisième chapitre** comporte essentiellement une étude comparative entre les algorithmes adaptatifs et les algorithmes non adaptatifs. Cette comparaison sera faite en utilisant la notion de modèle probabiliste appliqué à un tableau binaire ; nous rappelons tout d'abord le modèle proposé par Govaert (1988) pour la méthode MNDBIN ; ce dernier à non seulement permis de justifier, d'une part le choix du critère, d'autre part l'utilisation de la distance L_1 et des noyaux binaires, mais aussi de proposer par son extension un nouvel algorithme utilisant des distances adaptatives de type L_1. Nous présentons donc ce nouvel algorithme appelé algorithme **MNDBIN adaptatif** qui n'est autre que l'ancien algorithme MNDBIN auquel s'ajoutent deux variantes pour la distance ; la première consiste à pondérer la distance par des coefficients dépendants des variables, la seconde par des coefficients dépendant des variables et des classes ; ce dernier système de pondérations favorise les variables déséquilibrées. Nous proposons ensuite d'appliquer les trois variantes de l'algorithme MNDBIN adaptatif sur deux types de données, des données réelles et des données simulées, et de comparer les partitions obtenues. Nous remarquons alors que quelques problèmes de dégénérescence apparaissent au niveau du calcul du critère. Nous proposons des méthodes pour les résoudre, et nous verrons l'avantage que présente l'algorithme MNDBIN adaptatif en particulier sur les données simulées.

Nous proposons dans les trois derniers chapitres de ce travail, d'étendre les liens qui existent entre les méthodes de classification et les modèles probabilistes au cas où les données mettent en jeu deux ensembles.

Dans le **quatrième chapitre**, nous nous intéressons aux liens qui existent entre les modèles probabilistes et les méthodes de classification croisée. Ces méthodes consistent à subdiviser la population des individus et la population des variables en un petit nombre de groupes ou classes homogènes dans un certain sens.
Nous montrons comment la méthode de classification croisée (Govaert 1983) peut être vue comme une solution à un problème d'estimation de paramètres d'un modèle de mélange croisé. Il s'en est suivi l'établissement des liens entre les méthodes de

classification croisée et les modèles probabilistes. Cette étude nous permettra d'apporter un éclairage nouveau sur les méthodes de classification croisées.

Le **cinquième chapitre** est consacré à l'étude de la notion de modèle lié à la classification croisée de données binaires. Nous montrons que la méthode CROBIN (Govaert 1983), qui est une méthode de classification croisée sur des tableaux binaires correspondant à un mélange de lois de Bernoulli ayant le même paramètre qui mesure l'écart d'une classe à son centre et ne tient compte ni de la partition en lignes ni de la partition en colonnes. ce qui, dans certaines situations, peut s'avérer irréaliste. Nous proposons une extension de ce modèle en considérant trois autres mélanges, le mélange M_2 (dont le paramètre dépend de la partition en lignes), le mélange M_3 (le paramètre dépend de la partition en colonnes) et le mélange M_4 (le paramètre dépend de la partition en lignes et en colonnes) ; en outre, en nous appuyons sur des variantes de ce modèle, nous proposons de nouveaux algorithmes de **classification croisée** utilisant des **distances adaptatives** binaires. Quelques problèmes de dégénérescence apparaissent alors au niveau du calcul des critères. Nous ferons une étude de ces problèmes et nous proposons des solutions pour les résoudre.

Dans le **sixième chapitre** nous interprétons la méthode CROEUC (Govaert 1983) qui est une méthode de classification croisée sur tableaux décrits par des variables quantitatives, une approche modèle est proposée où nous montrons que le critère d'inertie associé à la méthode CROEUC correspond à l'hypothèse d'une population issue d'un mélange de lois gaussiennes unidimensionelles.

PARTIE A

CLASSIFICATION SIMPLE

CHAPITRE 1

CLASSIFICATION AUTOMATIQUE ET MELANGES

INTRODUCTION

Jusqu'à présent, deux tendances parallèles se sont dégagées dans le développement et la pratique du traitement statistique des données analysées. La première met en jeu des méthodes qui envisagent la possibilité d'une **interprétation probabiliste** des données traitées et des résultats statistiques fournis par le traitement. La deuxième tendance fait intervenir une classe assez vaste de méthodes de traitement statistique de l'information initiale, plus exactement l'ensemble des méthodes qui à priori ne s'appuient pas sur la nature probabiliste des données traitées, telles les méthodes de **classification automatique** qui ont été conçues dans un cadre géométrique sans faire aucune référence à la notion de modèle.

Wolf (1970), Scott et Symons (1971), Diday et Schroeder (1976), Celeux (1988) ont exploités ces deux tendances pour transformer le problème de la classification automatique en un problème de statistique inférentielle.

Le problème de la reconnaissance des composants d'un mélange, s'il est constamment posé dans la pratique, est loin d'être résolu complètement. L'algorithme proposé par Schroeder (1974) présente vis à vis des techniques existantes une certaine souplesse dans le choix du nombre de composants, du type de lois recherchées dans le mélange, et dans la dimension de la population observée. Cet algorithme permet de détecter, dans un échantillon donné, l'existence possible de sous-ensembles qui seraient échantillons de lois de probabilité d'un type connu ; cette approche a été d'une grande utilité pour beaucoup de chercheurs du même domaine qui se sont servis de cet algorithme pour apporter des éclairages nouveaux sur de nombreuses méthodes de classification automatique.

Ce chapitre commence par un rappel des méthodes de classification automatique. Nous avons retenu la méthode des Nuées Dynamiques (Diday 1972) pour le reste de notre travail pour les nombreux avantages qu'elle présente.

Dans le deuxième paragraphe, nous montrons comment la méthode des Nuées Dynamiques à été utilisée par Schroeder pour proposer une solution à un problème d'estimation de paramètres d'un mélange, en proposant une méthode de reconnaissance des composants d'un mélange. Celle-ci nous à permis de remarquer que souvent il existe un lien étroit entre les méthodes de classification automatique et les modèles probabilistes concernant le choix des critères numériques optimisés par ces méthodes. Govaert (1989 et 1990) a exploité cette idée pour faire une étude détaillée de ces liens ; cette étude fera l'objet du dernier paragraphe de ce chapitre.

1. LA CLASSIFICATION AUTOMATIQUE

Par classification automatique, on entend essentiellement l'ensemble des techniques qui fournissent directement une ou plusieurs partitions d'un ensemble ; certaines d'entre elles, dites de classification hiérarchique, permettant d'obtenir des partitions qui sont présentées sous forme d'un arbre de classification. Les grands calculateurs ont été à l'origine de la prolifération des méthodes de classification automatique qui se révèlent très utiles pour appréhender les gros fichiers de données ; elles permettent de fractionner l'ensemble des individus considérés en lots grossièrement homogènes que l'on peut analyser ensuite plus finement à l'aide d'une analyse factorielle par exemple.

Le but de la classification automatique est de définir sur un ensemble d'objets une structure qui respecte au mieux les ressemblances entre ces objets. Les structures qui sont envisagées peuvent être très variées :

- Recherche de hiérarchie (Sokal et Sneath (1963), Roux (1968), Jambu (1971)).
- Recherche de partition (Ball et Hall (1965), Regnier (1965), Diday (1972)).
- Recherche de classes empiétantes.

Dans notre travail, nous nous sommes limités à la recherche de partitions. Les méthodes de classification automatique que nous envisageons sont des méthodes portant sur l'ensemble des individus (ou celui des variables). Nous nous intéressons en particulier à celles dont la mise en place nécessite la définition d'un critère mesurant la qualité de la partition obtenue.

Plusieurs méthodes ont été proposées pour résoudre le problème de la classification : des méthodes qui recherchent la partition qui optimise une fonction numérique définie sur l'ensemble des partitions, appelée en général **critère de classification** (Regnier (1965), Ruspini (1969), Jensen (1969)), ou encore des méthodes algorithmiques, telle que la méthode de Ball et Hall (1965) qui dépend d'un certain nombre de seuils donnés à priori. Ou celles de Forgy (1965) et Macc Queen (1967). Ces dernières méthodes mesurent la qualité d'une partition par la somme des inerties des classes par rapport à leur centre de gravité. Rappelons que ce critère ne permet pas de comparer des partitions n'ayant pas le même nombre de classes.

Sous le nom de méthode des Nuées Dynamiques, Diday (1972) a proposé une technique de classification qui présente de nombreux avantages. L'idée de base de cette méthode est la suivante :

Au lieu de regrouper les éléments de l'ensemble **I** à classifier autour d'éléments, qui n'appartiennent d'ailleurs pas nécessairement à l'ensemble **I** comme c'est le cas pour les méthodes proposées par les auteurs cités précédemment, on fait un regroupement autour d'ensemble d'éléments, appelés noyaux, qui seront des parties de **I**. Une classe d'une partition de **I**, au lieu d'être représentée par un seul élément, tel son centre de gravité, le sera par plusieurs de ses éléments (le noyau de la classe) ; s'ils sont bien choisis, ces éléments seront "typiques" de la classe et en formeront un résumé plus riche que peut l'être un centre de gravité. Cette façon de procéder, qui admet de nombreuses variantes, présente bien des avantages, principalement :

- Une grande souplesse : des contraintes peuvent être imposées aux noyaux dont les éléments par exemple peuvent être choisis parmi des éléments particuliers de **I**.

- Des facilitées au niveau de l'interprétation des résultats qui peut être faite en examinant les seuls noyaux.

Pour ces raisons, la plupart des méthodes de classification automatique proposées jusqu'à présent reposent sur le principe des Nuées Dynamiques. Ce principe a été repris par Diday et al (1980) sous la forme suivante :

1.1. NOTATIONS

On suppose dans tout ce travail que les données initiales sont fournies sous la forme d'un tableau rectangulaire de n lignes et p colonnes contenant les valeurs prises par n individus définis par p variables.

Soient :

I : un sous-ensemble fini de $\mathbf{R^p}$ contenant n éléments.

$\boldsymbol{P_k}$: L'ensemble des partitions de **I** en **K** classes, les éléments de $\boldsymbol{P_k}$ seront appelés **k**-partitions et notés $P = (P_1, ..., P_K)$.

L : L'espace des noyaux qui seront associés aux sous-ensembles de **I** comme une caractéristique de ces sous-ensembles variant selon l'application de l'algorithme.

$\mathbf{L_k}$: L'ensemble des **K**-uples d'élément de **L**, noté : $L = (\lambda_1, ..., \lambda_K)$

où $\forall\, k \in \{1, ..., K\}$ $\lambda_k \in L$

1.2. LA METHODE DES NUEES DYNAMIQUES

Considérons un ensemble **I** de n individus représentés par un ensemble de n points inclus dans un espace **E** (par exemple $\mathbf{R^p}$). On définit l'ensemble des noyaux **L**, une distance **D** entre les éléments de **E** et les noyaux de **L**. Le critère **W** de la classification est alors le suivant :

$$W(P, L) = \sum_{k=1}^{K} \sum_{x \in P_k} D(x, \lambda_k)$$

où $P = (P_1, ..., P_K)$ une partition de l'ensemble **I**.
$L = (\lambda_1, ..., \lambda_K)$ l'ensemble des noyaux des classes de la partition P.

L'algorithme construit itérativement une suite de P^o, L^o, P^1, L^1, ..., P^n, L^n de partitions et de noyaux en minimisant à chaque étape le critère. Cette construction repose sur la définition des deux fonctions suivantes :

La fonction d'affectation f : consiste à affecter chaque individu à l'une des classe de la partition de manière à optimiser, à chaque fois, le critère W(f(L), L). Elle dépend bien sûre du choix de la distance D.
Nous obtenons :

$$f(L) = f(\lambda_1, ..., \lambda_K) = P = (P_1, ..., P_K).$$

où $P_k = \{x \in I / D(x/\lambda_k) \leq D(x/\lambda_{k'})$ avec $k<k'$ en cas d'égalité$\}$

La classe P_k sera donc constituée des éléments de **I** qui seront plus proche de λ_k au sens de la distance D que de tout autre noyau de **L**.

La fonction de représentation g : permet de déterminer les noyaux de la partition de manière à optimiser, à chaque fois, le critère W(P, g(P)).

$$g(P) = g(P_1, ..., P_K) = (\lambda_1, ..., \lambda_K) = L.$$

1.3. L'ALGORITHME

L'algorithme utilisé dans la méthode des Nuées Dynamiques consiste en la construction de 2 suites :

$\{ V_n / \ n \in N \}$: suite de $L_k x P_k$, c'est à dire que : $\forall n \quad V_n = (L^n, P^n)$.
$\{ U_n / \ n \in N \}$: suite réelle de valeurs du critère sur les V_n, c'est à dire :

$$\forall n \quad U_n = W(L^n, P^n) = W(V_n).$$

Si P^o est une partition initiale quelconque prise au hasard ou choisie, et si L^o est l'ensemble des noyaux qui lui sont associés ($L^o = g(P^o)$) alors :

$$V_o = (L^o, P^o) = (g(P^o), P^o)).$$

La suite (V_n) est ensuite définie par récurrence :

si $V_n = (L^n, P^n)$ alors $V_{n+1} = (L^{n+1}, P^{n+1})$

où $P^{n+1} = f(L^n)$ et $L^{n+1} = g(P^{n+1}) = gof(L^n)$.

On montre que sous certaines conditions (Diday 1972, Schroeder 1974, Govaert 1975), la suite $U_n = W(V_n)$ décroît, converge et atteint sa limite :

$$\exists \ M \in N : \ \forall n \geq M \quad U_n = U^*.$$

le couple $V^* = (L^*, P^*)$ tel que $W(V^*) = U^*$ sera appelé optimum local.

Pour aborder le problème des mélanges de distributions de probabilités, le même schéma que celui des Nuées Dynamiques sera à nouveau utilisé en prenant comme noyaux des distributions de probabilités. Dans notre étude nous nous sommes limités à une forme particulière de l'algorithme utilisant la méthode d'estimation du **maximum de vraisemblance** et optimisant un **critère de vraisemblance.**

2. LES MELANGES

On désigne par I l'ensemble des n individus que nous considérons comme un échantillon de taille n à valeurs dans R^p. Nous cherchons donc à détecter dans cet échantillon l'existence possible de sous-ensembles qui seraient échantillons de lois de probabilité d'un type connu , dont la distribution globale aura la forme suivante :

$$f(x) = \sum_{k=1}^{K} p_k \, f(x / \lambda_k)$$

Dans laquelle $f(x / \lambda_k)$ et $f(x)$ sont les densités (dans le cas continu) ou les polygones de fréquence (dans le cas discret) respectivement de la $k^{ème}$ composante du mélange et de la loi de probabilité résultante.

p_k : La probabilité à priori d'apparition dans un échantillon aléatoire d'une observation de la loi $f(x/\lambda_k)$, (c'est à dire le poids spécifique de telle observation dans la population générale), k le nombre de composants .

2.1. FORMALISATION DU PROBLEME

L'ensemble I défini précédemment représente un ensemble de n observations sur lesquelles p mesures ont été effectuées ($I \subseteq \mathbf{R}^p$). On se donne une famille de densités de probabilités $f(., \lambda)_{\lambda \in L}$ à laquelle on suppose que les distributions des différents composants appartiennent : λ est un paramètre réel ou vectoriel et L son espace de définition $L \subseteq \mathbf{R}^s$ (par exemple si p = 1, la famille $f(., \lambda)$ peut être celle des distributions gaussiennes unidimensionnelles avec $\lambda = (\mu, \sigma)$; $s = 2$ $L = \mathbf{R} \times \mathbf{R}^+ \subsetneq \mathbf{R}^2$).

Le problème à résoudre est alors le suivant :

> *On cherche à trouver un couple (P, L) où* $L = (\lambda_1, ..., \lambda_K)$ *avec* $\lambda_k \in L$ *pour tout* k *et* $P = (P_1, ..., P_K)$ *où les* P_k *forment une partition de I tel que : Pour tout* $k \in \{1,..., K\}$ *;* P_k *puisse être considérée en un sens statistique à préciser comme un échantillon vraisemblable de la distribution de la loi* $f(., \lambda_k)$.

Ce problème peut être résolu par l'algorithme des Nuées Dynamiques étudié au paragraphe 1.1, en prenant comme noyaux les paramètres inconnus λ ; il suffit pour cela de se donner une fonction D mesurant la distance d'une observation $x \in I$ à une distribution $f(., \lambda)$. Le choix de cette fonction peut se faire de diverses façons selon l'optique statistique dans laquelle on se place.

2.2. LA METHODE DE RECONNAISSANCE DES COMPOSANTS D'UN MELANGE

Pour résoudre le problème posé ci-dessus, Schroeder (1974) propose de prendre comme définition de la fonction D la quantité suivante :

$$D(x, \lambda) = \text{Log} \left[\frac{f^*}{f(x, \lambda)} \right].$$

Cette définition exprime qu'une observation x sera d'autant plus proche du noyau λ que de la densité $f(., \lambda)$ sera grande en x. Pour que cette définition conduise à un ensemble de valeurs pour D qui soit borné inférieurement, il faut choisir la constante f* de façon à ce que :

$$f^* \geq \max \{f(x, \lambda) / \lambda \in L \text{ et } x \in I\}.$$

Nous verrons par la suite qu'une valeur explicite de f* n'est pas nécessaire au déroulement de l'algorithme.

Nous allons voir ce que deviennent les fonctions W, f et g :

2.2.1. Définition de W

L'expression du critère à optimiser devient :

$$W(P, L) = \sum_{k=1}^{K} \sum_{x \in P_k} D(x, \lambda_k) = n.\text{Log } f^* - \sum_{k=1}^{K} \text{LogL}(P_k/\lambda_k).$$

où $L(P_k / \lambda_k) = \prod_{x \in P_k} f(x/\lambda_k)$ qui est la vraisemblance de l'échantillon P_k pour la loi de probabilité $f(., \lambda_k)$.

La minimisation du critère W(P, L) revient donc à la maximisation du critère de **vraisemblance classifiante** suivant :

$$VC(P, L) = \sum_{k=1}^{K} \text{Log } L(P_k/ \lambda_k).$$

En utilisant les deux fonctions f et g définies ci-dessous, l'algorithme nous conduit à une solution locale du problème.

2.2.2. DEFINITION DE f

$f(L) = (P_1, ..., P_K)$.

où $P_k = \{x \in I / D(x/\lambda_k) \leq D(x/\lambda_{k'})$ avec $k \neq k'$ et $k<k'$ en cas d'égalité$\}$

$= \{x \in I / f(x/\lambda_k) \geq f(x/\lambda_{k'})$ avec $k \neq k'$ et $k<k'$ en cas d'égalité$\}$.

2.2.3. DEFINITION DE g

$g(P) = L = (\lambda_1, ..., \lambda_K)$

où λ_k minimise pour chaque classe la quantité : n_k.Log f* - Log $L(P_k/\lambda_k)$.

Log $L(P_k/\lambda_k) = \max_{\lambda \in L} L(P_k/\lambda)$; ce qui signifie exactement que λ_k est l'estimateur du maximum de vraisemblance de λ pour l'échantillon P_k.

Pour plus de détail sur l'existence de cet estimateur qui n'est pas toujours assuré on pourra consulter Schroeder (1974).

On est assuré que cet algorithme mène à un minimum local du critère et à un couple (L*, P*) tel que :

si $L^* = (\lambda_1^*, ..., \lambda_K^*)$ et $P^* = (P_1^*, ..., P_K^*)$ $\forall k \in \{1, ..., K\}$, λ_k^* est l'estimateur du maximum de vraisemblance de λ pour l'échantillon P_k^*.
$\forall x \in I$, $x \in P_k^* \Leftrightarrow f(x, \lambda_k^*) \geq f(x, \lambda_{k'}^*)$ avec $k \neq k'$ et $k < k'$ en cas d'égalité.

Les méthodes des Nuées Dynamiques reposent sur l'optimisation d'un critère numérique lui même défini à partir d'une distance. La méthode de reconnaissance des composants d'un mélange proposée par Schroeder à montré que souvent il existe un lien entre ces méthodes et les modèles probabilistes. Nous remarquons donc que le passage au critère probabiliste peut apporter une argumentation concernant le choix du critère numérique optimisé.

Nous proposons dans la dernière partie de ce chapitre une étude des liens qui existent entre les critères métriques et les critères probabilistes ; nous étudierons tout d'abord ces liens dans le cas continu (Govaert 1989) puis dans le cas discret (Govaert 1990) et nous montrons dans les deux cas comment ces deux critères peuvent se rejoindre.

3. GENERALISATION

3.1. ETUDE DES LIENS ENTRE LES CRITERES METRIQUES ET LES CRITERES PROBABILISTES DANS LE CAS CONTINU

On suppose toujours que les données initiales sont fournies sous la forme d'un tableau rectangulaire de n lignes et p colonnes contenant les valeurs prises par n individus pour p variables quantitatives. Nous envisageons ici deux types de critères : le premier que nous appelerons **critère métrique**, utilise la notion de mesure de

dissimilarité, le second que nous appelerons **critère probabiliste** utilise la notion de mélange probabiliste. Nous définissons tout d'abord ces deux types de critères, nous étudions ensuite les liens qui existent entre eux, puis nous montrons comment les mélanges de lois gaussiennes sont liés aux distances quadratiques et les lois exponentielles aux distances de type L_1.

3.1.1. Critère métrique

Dans cette approche, nous représentons le tableau de données sous la forme d'un ensemble I de n individus de $\mathbf{R}^p$. Chaque classe d'une partition va être représentée par un élément de l'ensemble L qui reste à préciser et qui sera appelé ensemble des "noyaux" ; enfin on se donne une fonction D de $\mathbf{R}^p \times L$ dans $\mathbf{R}^+$ qui mesurera la "dissimilarité" entre un élément de $\mathbf{R}^p$ et un noyau .

Le problème que l'on cherche à résoudre est de trouver la partition $P = (P_1, ..., P_K)$ de I en K classes et un K-uples $(\lambda_1, ..., \lambda_K)$ de noyaux (un par classe) minimisant le critère :

$$\sum_{k=1}^{K} \sum_{x \in P_k} D(x, \lambda_k)$$

Ce critère qui dépend de la mesure de dissimilarité D sera appelé **critère métrique** et noté **CM($\mathbf{R}^p$, L, D)**. Les méthodes des Nuées Dynamiques rappelées au début de ce chapitre proposent une solution à ce problème en construisant de manière itérative une suite de partitions-noyaux faisant décroître le critère en utilisant toujours les deux fonctions f et g de représentation et d'affectation défini au paragraphe 1.2.
On peut sans difficulté, en conservant le même critère, modifier le problème posé en ajoutant une contrainte au K-uples de noyaux $(\lambda_1, ..., \lambda_K)$ recherché. Par exemple, si le noyau est défini comme un couple (a, b), on peut imposer que le premier terme du couple soit identique pour tout les noyaux du K-uple recherché $\lambda = ((a, b_1),(a, b_2),..., (a,b_K))$.

Définition 1.1 (Govaert 1989)

On dira que deux critères métriques sont équivalents si et seulement s'ils sont définis sur les mêmes ensembles $\mathbf{R}^p$ et L et s'il existe une bijection ϕ de $\mathbf{R}^p$ strictement croissante vérifiant :

$$CM(\mathbf{R}^p, L, D_1) = \phi \circ CM(\mathbf{R}^p, L, D_2)$$

où D_1 et D_2 sont les mesures de dissimilarité associées aux deux critères. Si on remplace D par une fonction linéaire croissante de D, on obtient un critère métrique équivalent :

Proposition 1.1 (Govaert 1989)

$\forall\ \alpha \in R^+$ et $\beta \in R$, les critères $C\,M(R^p, L, D)$ et $C\,M(R^p, L, \alpha D+\beta)$ sont équivalents.

3.1.2. Critère probabiliste

On reprend ici la représentation de Celeux (1988).

3.1.2.1. Identification d'un mélange

Le tableau de données de départ de dimension (n, p) (où n est le nombre d'individus et p est le nombre de variables) est considéré comme un échantillon I de taille n d'une variable aléatoire à valeurs dans $\mathbf{R}^P$ dont la loi de probabilité admet la fonction de densité suivante :

$$f(x) = \sum_{k=1}^{K} p_k\, f(x/\lambda_k) \qquad (1.1)$$

$$\text{avec} \quad \forall\, k = 1,K \quad p_k \in\,]0,1[\quad \text{et} \quad \sum_{k=1}^{K} p_k = 1 \qquad (1.2)$$

où $f(./\lambda)$ appartient à une famille de fonctions de densité dépendant du paramètre λ élément de $\mathbf{R}^s$, où s est un entier supérieur ou égal à 1 et p_k est la probabilité qu'un point de l'échantillon suive la loi $f(./\lambda_k)$. On appelera ces p_k les proportions du mélange.

Le problème posé est l'estimation du nombre K de composants et des paramètres inconnus $\{p_k, \lambda_k / k = 1, K\}$ au vu de l'échantillon.

3.1.2.2. Approche classification

Dans l'approche classification (Scott et Symons 1971, Schroeder 1974), on remplace le problème initial d'estimation par le problème suivant :

Rechercher une partition $P = (P_1,..., P_K)$, K étant supposé connu, telle que chaque classe P_k soit assimilable à un sous-échantillon qui suit une loi $f(., \lambda_k)$.

Il s'agit alors de maximiser le critère de **vraisemblance classifiante :**

$$VC(P, L) = \sum_{k=1}^{K} \text{Log } L(P_k, \lambda_k) \qquad (1.3)$$

où λ est le p-uplet $(\lambda_1,..., \lambda_k)$ et $L(P_k, \lambda_k)$ est la vraisemblance du sous-échantillon P_k suivant la loi $f(./\lambda_k)$: $L(P_k, \lambda_k) = \prod_{x \in P_k} f(x/\lambda_k)$.

Ce critère qui dépend de la famille **F** de fonctions de densité définies sur $\mathbf{R^p}$ sera appelé **critère probabiliste** et noté **$CP(R^p, F)$.**

Pour maximiser ce critère, on utilise l'algorithme de type Nuées Dynamiques qui construit à partir d'une partition P^o en K classes une suite de partitions en appliquant les fonctions f et g décrites aux paragraphes 2.2.2 et 2.2.3.
On peut alors montrer que sous certaines hypothèses, cet algorithme est convergent. On obtient à la convergence une partition P et une estimation des paramètres λ_k. Les proportions p_k du mélange sont fournies par les fréquences des classes P_k.

De la même manière que pour les critères métriques, on peut modifier le problème en imposant une contrainte aux paramètres de la fonction de densité associées aux classes d'une partition ; par exemple, si la famille **F** est l'ensemble des lois gaussiennes sur $\mathbf{R^p}$, on peut imposer que toutes les lois gaussiennes associées aux classes d'une partition aient la même matrice de variances.

3.1.3. Etude des liens entre les deux critères

Govaert (1989) a défini deux types de liens entres les critères métriques et les critères probabilistes. Le premier permet d'associer à tout critère probabiliste un critère métrique appelé **critère métrique associé** au critère probabiliste, le second permet d'étendre la notion de critères équivalents définis dans le cas de critères métriques et probabilistes.

3.1.3.1. Critère métrique associé à un critère probabiliste :

Proposition 1.2 (Govaert 1989)

$$CP(R^p, F) = CM(R^p, L, D)$$

où L est l'ensemble de définition des paramètres de la famille F et D est définie par :

$$\forall\ x \in \mathbb{R}^p,\ \forall\ \lambda \in L \qquad D(x, \lambda) = -Log\ f(x, \lambda)$$

Le critère métrique ainsi défini est appelé **critère métrique associé.**
La démonstration de cette proposition est facile à faire. Il suffit d'utiliser la définition des deux critères. En outre, le lien existant entre les deux critères permet d'affirmer que la maximisation d'un critère probabiliste est équivalente à la minimisation du critère métrique associé. Ce résultat permet donc de considérer que tous les critères probabilistes sont des critères métriques, mais on peut s'interroger sur le problème inverse qui est le suivant : un critère métrique donné est-il associé à un critère probabiliste ? Cette propriété n'est pas vraie en général mais nous allons nous intéresser à l'étude des conditions nécessaires et suffisantes pour qu'elle soit vérifiée.

3.1.3.2. Critères probabilistes et métriques équivalents

Définition 1.2 (Govaert 1989)

Deux critères probabilistes sont équivalents si les critères métriques associés sont équivalents.

Un critère probabiliste CP_1 et un critère métrique CM_2 sont équivalents si le critère métrique CM_1 associé à CP_1 est équivalent au critère métrique CM_2.

3.1.3.3. Condition pour qu'un critère métrique soit associé à un critère probabiliste

Proposition 1.3 (Govaert 1989)

Un critère métrique $CM(\mathbb{R}^p, L, D)$ est associé à un critère probabiliste si et seulement si $\forall\ \lambda \in L$ la fonction $x \rightarrow e^{-D(x,\lambda)}$ est continue et vérifie $\int_{\mathbb{R}^p} e^{-D(x,\lambda)}\ dx = 1$.

3.1.3.4. Critère probabiliste équivalent à un critère métrique

En utilisant la proposition (1.1), on peut obtenir une condition plus faible permettant de montrer qu'un critère métrique est équivalent (et non associé) à un critère probabiliste.

Proposition 1.4 (Govaert 1989)

Etant donné le critère métrique CM(R^p, L, D), s'il existe un réel r>1 tel que la quantité $s = \int_{R^p} e^{-D(x,\lambda)}\, dx$ *soit indépendante de* λ, *alors le critère probabiliste CP(R^p, F) où F est définie par les fonctions de densité f :*

$$f(x, \lambda) = \frac{1}{s}\ r^{-D(x, \lambda)}$$

est un critère équivalent.

Preuve

Le critère métrique associé à la famille proposée est définie par la fonction D' :

$$D'(x, \lambda) = - \text{Log } f(x, \lambda) = - \text{Log } \{ \frac{1}{s}\ r\text{-}D(x, \lambda) \} = s + r.D(x, \lambda).$$

La proposition (1.1) permet d'affirmer que les critères métriques associés à D et D' sont équivalents. D'où le résultat annoncé.

Après avoir étudié les deux types de critères et les conditions dans lesquelles ces critères peuvent se rejoindre, nous nous intéressons maintenant aux liens existants entre les lois gaussiennes et les distances quadratiques et les lois exponentielles et les distances L_1.

3.1.4. Métriques quadratiques et lois gaussiennes

3.1.4.1. Métrique quadratique fixe et identique pour toutes les classes

L'ensemble à classifier est inclus dans $\mathbf{R}^p$, les noyaux sont aussi des éléments de $\mathbf{R}^p$ ($L = \mathbf{R}^p$), la fonction D est définie à partir d'une matrice M définie symétrique positive fixée à priori.

$\forall\ x \text{ et } \lambda_k \in \mathbf{R}^p \quad D(x, \lambda_k) = \alpha.\ {}^t(x - \lambda_k).M.(x - \lambda_k) \quad \forall\ \alpha \in \mathbf{R}^+ \text{ et } \forall\ \beta \in \mathbf{R}$

Quelles que soient les valeurs α et β, les critères seront tous équivalents (proposition 1.1) nous nous limiterons donc au critère le plus simple qui correspond à la fonction D' :

$$\forall\ x \text{ et } \lambda_k \in \mathbf{R}^p \qquad D'(x, \lambda_k) = {}^t(x - \lambda_k).M.(x - \lambda_k) \qquad (1.4a)$$

La proposition (1.4) permet d'affirmer que le critère métrique définie à partir de la distance (1.4a) est équivalent à un critère probabiliste car :

$\int_{\mathbf{R}^p} e^{-D(x,\lambda_k)}\, dx = \pi^{p/2} . |M|^{-1/2}$ est une quantité indépendante de λ_k , la fonction de densité s'écrit alors :

$$\forall\, x \text{ et } \lambda_k \in \mathbf{R}^p \quad f(x, \lambda_k) = \pi^{-p/2} . |M|^{1/2} . e^{-{}^t(x-\lambda_k).M.(x-\lambda_k)}$$

qui correspond à une loi gaussienne de centre λ_k et de matrice de variance $2.M^{-1}$.

3.1.4.2. Métrique quadratique variable et dépendante de chaque classe

La métrique M n'est pas fixe et dépend de chaque classe :

premier cas : $D(x, (a_k, M_k)) = {}^t(x - a_k).M_k.(x - a_k)$ (1.4b)

$\lambda_k = (a_k, M_k)$

ce critère métrique est associé à un critère probabiliste si la condition $|M_k| = \pi^p$ est vérifiée. Le critère probabiliste qui lui est associé est alors défini par la famille de fonction de densité **F** correspondant aux lois gaussiennes dont les matrices de variances sont de déterminant constant.

Deuxième cas : $D(x, (a_k, M_k, \alpha_k)) = \alpha_k + {}^t(x - a_k).M_k.(x - a_k)$ (1.4c)

$\lambda_k = (a_k, M_k, \alpha_k)$

si $\alpha_k = \frac{p}{2} \text{Log}\pi - \frac{1}{2} \text{Log}\,|M_k|$, le critère métrique définie à l'aide de la métrique (1.4c) est associé a un critère probabiliste dont la fonction de densité est définie par :

$$\forall\, x \text{ et } a_k \in \mathbf{R}^p \quad f(x, (a_k, \Gamma_k)) = (2\pi)^{-p/2} . |\Gamma_k|^{-1/2} .e^{-\frac{1}{2}{}^t(x-a_k).\Gamma_k^{-1}.(x-a_k)}$$

où $\Gamma_k = \frac{1}{2} . M_k^{-1}$. C'est le cas le plus général des lois gaussiennes.

Nous allons voir maintenant comment les métriques euclidiennes sont elles aussi liées aux lois gaussiennes :

3.1.5. Métrique euclidienne et lois gaussiennes

Premier cas : les noyaux sont de la forme (a, M) où $a \in \mathbf{R}^p$ et M est une matrice symétrique définie positive ; on impose aux noyaux d'avoir la même matrice M.

$$D(x, (a_k, M)) = {}^t(x - a_k).M.(x - a_k) \qquad (1.5a)$$

si $|M| = \pi^p$ alors le critère métrique définie à l'aide de la métrique euclidienne (1.5a) est associé à un critère probabiliste dont la fonction de densité s'écrit :

$$f(x, a_k) = e^{-\frac{1}{2}.{}^t(x-a_k).\Gamma^{-1}.(x-a_k)}$$

où $\Gamma = \frac{1}{2}.M^{-1}$.

Deuxième cas : $D(x, (a_k, M, \alpha)) = \alpha + {}^t(x - a_k).M.(x - a_k)$ (1.5b)

si $\alpha = \frac{p}{2}\text{Log}\pi - \frac{1}{2}\text{Log}|M|$ on obtient un critère probabiliste équivalent au critère métrique définie par (1.5b) en prenant comme fonction de densité :

$$f(x, (a_k, \Gamma)) = (2\pi)^{-p/2}.|\Gamma|^{-1/2}.e^{-\frac{1}{2}.{}^t(x-a_k).\Gamma^{-1}.(x-a_k)}$$

où $\Gamma = \frac{1}{2}.M^{-1}$; dans ce cas là on n'impose aucune contrainte au déterminant de la matrice M.

On remplace maintenant la métrique euclidienne par la distance L_1 ou la distance city-block.

3.1.6. Métrique de type L_1

Dans le cas de la distance L_1, le centre de gravité est remplacé par la notion de médiane.

3.1.6.1. Distance fixe et identique pour toutes les classes

$L = \mathbf{R}^p$ $$D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j |x^j - \lambda_k^j| \qquad (1.6a)$$

où les α^j sont des constantes réelles et positives.

si la condition $\prod_{j=1}^{p} \alpha^j = 2^p$ est vérifiée, le critère métrique définie par (1.6a) est associé au critère probabiliste définie par la fonction de densité suivante :

$$f(x, \lambda_k) = e^{-D(x,\lambda_k)} = \prod_{j=1}^{p} \frac{\alpha^j}{2} e^{-\alpha^j |x^j - \lambda_k^j|}$$

qui correspond pour chaque composante du mélange à un produit de p lois exponentielles bilatérales $L(\lambda_k^j, \alpha^j)$ (en supposant bien sûr l'hypothèse d'indépendance mutuelle vérifiée entre les p variables).

3.1.6.2. Distance L_1 variable et dépendante de chaque classe

$$D(x, (\lambda_k, \alpha_k, \beta_k)) = \sum_{j=1}^{p} \alpha_k^j |x^j - \lambda_k^j| + \beta_k \qquad (1.6b)$$

si $\prod_{j=1}^{p} \alpha_k^j = 2^p$ alors $\beta_k = p.\text{Log } 2 - \sum_{j=1}^{p} \text{Log } \alpha_k^j = 0$; on obtient ainsi une distance L_1 pondérée. Dans ce cas $f(x, \lambda_k) = \prod_{j=1}^{p} L(\lambda_k^j, \alpha_k^j)$

où $L(\lambda_k^j, \alpha_k^j)$ est une loi exponentielle bilatérale.

Nous venons donc de rappeler les différents liens qui existent entre les critères utilisés en classification automatique et les modèles probabilistes dans le cas où l'ensemble à classifier constitue un ensemble continu, nous allons voir ce que deviennent ces liens dans le cas où les données sont fini ou inclus dans un espace discret, c'est le cas des tableaux décrits par des variables binaires où qualitatives.

3.2. ETUDE DES LIENS ENTRE LES CRITERES METRIQUES ET LES CRITERES PROBABILISTES DANS LE CAS DISCRET

L'ensemble à classifier est maintenant inclus dans un espace fini **E**, nous allons reprendre toutes les définitions et propositions qui ont étés établi dans le cas continu mais cette fois-ci appliquées à un ensemble discret. Nous ne donnons ici que les principaux résultats.

Le critère métrique **CM(R^p, L, D)** défini dans le cas continu est remplacé ici par le critère métrique **CM(E, L, D)**, où **E** est un ensemble fini, par exemple **E = {0, 1}^{p}** dans le cas d'un tableau binaire à p variables.

La définition (1.1) et la proposition (1.1) restent les mêmes dans le cas discret où l'on remplace l'ensemble **R^p** par l'ensemble **E**. Le critère probabiliste que l'on note par **CP(E, F)** est lui aussi défini de la même manière que dans le cas continu, mais les fonctions de densités sont remplacées par des distributions de probabilités sur **E** ; les liens existant entre les critères métriques et probabilistes dans le cas discret sont les

mêmes que ceux obtenus dans le cas continu, mais notons que les conditions d'association et d'équivalence entre ces deux types de critères diffèrent dans le sens où l'on remplace l'intégrale par la sommation ; nous allons rappeler quelques uns d'entre eux.

3.2.1. Critère métrique associé à un critère probabiliste

Proposition 1.5 (Govaert 1990)

$$CP(E, F) = CM(E, L, D)$$

où L est l'ensemble de définition des paramètres de la famille $\overset{\bullet}{F}$ *et D est définie par :*
$\forall\ x \in E, \forall\ \lambda \in L \qquad D(x, \lambda) = -Log\ p(x, \lambda)$

3.2.2. Condition pour qu'un critère métrique soit associé à un critère probabiliste

Proposition 1.6 (Govaert 1990)

Un critère métrique CM(E, L, D) est associé à un critère probabiliste si et seulement si $\forall\ \lambda \in L$ *la fonction* $x \rightarrow e^{-D(x,\lambda)}$ *est continue et vérifie* $\sum_{x \in E} e^{-D(x,\lambda)}\ dx = 1.$

3.2.3. Critère probabiliste équivalent à un critère métrique

En utilisant la proposition (1.1) appliquée dans le cas discret, on peut obtenir une condition plus faible permettant de montrer qu'un critère métrique est équivalent (et non associé) à un critère probabiliste.

Proposition 1.7 (Govaert 1990)

Etant donné le critère métrique CM(E, L, D) , s'il existe un réel r>1 tel que la quantité $s = \sum_{x \in E} e^{-D(x,\lambda)}$ *soit indépendante de* λ*, alors le critère probabiliste CP(E, F) où F est définie par les distributions de probabilités suivantes :*

$$p(x, \lambda) = \frac{1}{s}\ r^{-D(x, \lambda)}$$

est un critère équivalent.

Proposition 1.8 (Govaert 1990)

Etant donné la fonction D de ExL dans R , le critère métrique CM(E, L, D') où D' vérifie :

$$D'(x, \lambda) = D(x, \lambda) + Log\left(\sum_{x \in E} e^{-D(x,\lambda)}\right)$$

est associé au critère probabiliste défini par la distribution $p(x, \lambda) = e^{-D'(x,\lambda)}$.

Si **E** est le produit cartésien d'ensemble finis $\mathbf{E_1}$, ..., $\mathbf{E_p}$ alors D peut se décomposer en une somme de p termes correspondant aux p ensembles $\mathbf{E_j}$.

$$\mathbf{L = L_1 x ... x L_p} \qquad D(x, \lambda) = \sum_{j=1}^{p} D^j(x, \lambda)$$

où x^j et λ^j correspondent à la décomposition de x et de λ sur les espace $\mathbf{E_j}$ et $\mathbf{L_j}$.

Proposition 1.9 (Govaert 1990)

S'il existe un réel positif tel que pour tout $j = 1, ..., p$ *, les quantités* $s_j = \sum_{x \in E_j} e^{-D^j(x,\lambda)}$ *sont indépendantes de* λ^j*, alors il existe un critère probabiliste* ***équivalent*** *au critère métrique défini par la fonction D. De plus, ce critère probabiliste correspond à un produit de distributions sur les* E^j *définies par* $p(x^j, \lambda^j) = \frac{1}{s^j} r^{-D^j(x,\lambda)}$. *Enfin si* $s_j = 1$*, le critère métrique est associé au critère probabiliste.*

3.2.4. Métrique L_1 et distribution de Bernoulli

Nous allons étudier un certain nombre de critères issus de la distance L_1 et définis sur un ensemble de données binaires.

3.2.4.1. Distance L_1 fixe et identique pour toutes les classes

$\mathbf{I \subseteq E = \{0, 1\}^p}$ $\qquad \mathbf{L = E}$

$$\forall x \text{ et } \lambda_k \in E \qquad D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j |x^j - \lambda_k^j| + \beta \qquad (1.7a)$$

où les α^j sont des constantes réelles positives et β un réel quelconque. On va se limiter à l'étude du critère défini par (1.7b) qui est équivalent au critère (1.7a) .

$$\forall x \text{ et } \lambda_k \in E \qquad D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j |x^j - \lambda_k^j| \qquad (1.7b)$$

Comme $\sum_{x^j \in E^j} e^{-\alpha^j |x^j - \lambda_k^j|} = 1 + e^{-\alpha^j}$, la proposition (1.9) permet d'affirmer qu'il existe un critère probabiliste équivalent. Celui-ci consiste à considérer que pour chaque composante du mélange les p variables sont indépendantes, et que chacune d'entres elles suit une loi de Bernoulli de distributions:

$$\{\varepsilon^j, 1-\varepsilon^j\} \text{ ou } \{1-\varepsilon^j, \varepsilon^j\} \quad \text{où} \quad \varepsilon^j = \frac{1}{1+e^{-\alpha^j}} .$$

3.2.4.2. Distance L_1 variable et dépendante de chaque classe

$$D(x, (\lambda_k, \alpha_k, \beta_k)) = \sum_{j=1}^{p} \alpha_k^j |x^j - \lambda_k^j| + \beta_k \qquad (1.7c)$$

si $\beta_k = \sum_{j=1}^{p} \text{Log}(1+e^{-\alpha_k^j})$, on retrouve une variante de Govaert (1988) qui suppose que pour chaque composante du mélange les p variables sont indépendantes et que chacune d'entres elles suit une loi de Bernoulli de distribution $\{\varepsilon_k^j, 1-\varepsilon_k^j\}$ ou $\{1-\varepsilon_k^j, \varepsilon_k^j\}$ où $\varepsilon_k^j = \frac{1}{1+e^{-\alpha_k^j}}$.

3.2.4.3. Distance adaptative L_1 identique pour toutes les classes

$$D(x, (\lambda_k, \alpha, \beta)) = \sum_{j=1}^{p} \alpha^j |x^j - \lambda_k^j| + \beta \qquad (1.7d)$$

En prenant $\beta = \sum_{j=1}^{p} \text{Log}(1+e^{-\alpha^j})$ on obtient un critère probabiliste équivalent défini à l'aide de la loi de Bernoulli dont les distributions de probabilités sont $\{\varepsilon^j, 1-\varepsilon^j\}$ où $\{1-\varepsilon^j, \varepsilon^j\}$ où $\varepsilon^j = \frac{1}{1+e^{-\alpha^j}}$.

3.2.5. Données qualitatives nominales

$I \subseteq E = E_1 x \ldots x E_p$ où E_j est l'ensemble des q_j modalités de la variable qualitative j. Nous utilisons la distance proposée par Marchetti (1989) :

$$D(x, y) = \sum_{j=1}^{p} \delta^j(x, y)$$

où $\delta^j(x, y) = 0$ si $x^j = y^j$ et $\delta^j(x, y) = 1$ si $x^j \neq y^j$.

3.2.5.1. Distance fixe et identique pour toutes les classes

$$\mathbf{L = E} \qquad D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j \cdot \delta^j(x, \lambda_k) + \beta \qquad (1.8a)$$

$\forall \beta \in \mathbf{R}$ le critère métrique défini à l'aide de (1.8b) est équivalent à celui défini par (1.8a) :

$$D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j \cdot \delta^j(x, \lambda_k) \qquad (1.8b)$$

$$\forall \ \lambda_k^j \in \mathbf{E_j} \qquad \sum_{x^j \in E^j} e^{-\alpha^j . \delta^j(x, \lambda_k)} = \{ 1 + (m_j - 1). e^{-\alpha^j} \}.$$

Cette quantité est indépendante de λ_k^j ; il existe donc un modèle probabiliste qui consiste à considérer que pour chaque composante du mélange, les p variables sont indépendantes et chacune d'entre elles suit une loi multinomiale de paramètre $\{1-\varepsilon^j, \frac{\varepsilon^j}{m_j - 1}, ..., \frac{\varepsilon^j}{m_j - 1}\}$ où la modalité prenant la probabilité $1-\varepsilon^j$ étant celle qui est prise par le noyau λ_k^j avec $\varepsilon^j = \frac{m_j - 1}{m_j - 1 + e^{\alpha^j}}$.

3.2.5.2. Distance variable et dépendante de chaque classe

$$D(x, \lambda_k) = \sum_{j=1}^{p} \alpha_k^j \cdot \delta^j(x, \lambda_k) + \beta_k \qquad (1.8c)$$

si $\beta_k = \sum_{j=1}^{p} \text{Log} \{ 1 + (m_j - 1) + e^{-\alpha^j} \}$, il existe alors un critère probabiliste qui consiste à prendre pour chaque composante du mélange une distribution de probabilité formée d'un produit de p lois multinomiales dont les probabilités sont :

$\{1-\varepsilon_k^j, \frac{\varepsilon_k^j}{m_j - 1}, ..., \frac{\varepsilon_k^j}{m_j - 1}\}$ où la modalité prenant la probabilité $1-\varepsilon_k^j$ étant celle qui est prise par le noyau λ_k^j avec $\varepsilon_k^j = \frac{m_j - 1}{m_j - 1 + e^{\alpha_k^j}}$.

3.2.6. Données qualitatives ordinales

$\mathbf{L = E}$ $\quad \mathbf{I \subseteq E = E_1 x ... x E_p}$ où $\mathbf{E_j}$ est l'ensemble des q_j modalités de la variable qualitative j.

$$D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j |x^j - \lambda_k^j| + \beta \qquad (1.9a)$$

$$D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j |x^j - \lambda_k^j| \qquad (1.9b)$$

$\forall \beta \in \mathbf{R}$ le critère métrique défini à l'aide de (1.9b) est équivalent à celui défini par (1.9a) .

La quantité $\sum_{x^j \in E^j} e^{-\alpha^j |x^j - \lambda_k^j|}$ dépend de λ_k^j . Par conséquent il n'existe pas de critère probabiliste équivalent au critère définie par (1.9a) ou (1.9b) ; mais en utilisant la proposition (1.4) qui reste vraie dans le cas discret on peut se ramener au cas où :

$$D(x, \lambda_k) = \sum_{j=1}^{p} \alpha^j |x^j - a_k^j| + \text{Log} \left(\sum_{x^j \in E^j} e^{-\alpha^j |x^j - \lambda_k^j|} \right) \qquad (1.9c)$$

Le critère métrique définie à l'aide de cette distance est associé à un critère probabiliste qui correspond pour chaque composante du mélange où les p variables sont indépendantes au produit de p distributions qui ont cette fois-ci une forme assez compliquée.

Nous venons donc de voir que la comparaison des critères métriques et probabilistes permet d'apporter un éclairage nouveau sur certains critères optimisés par les méthodes de classification. Cette comparaison va nous permettre d'interpréter quelques méthodes de classification étudiées dans ce travail en termes probabilistes. Nous proposons cette étude dans le chapitre qui suit.

CHAPITRE 2

CLASSIFICATION ET MODELES SUR DONNEES QUALITATIVES

INTRODUCTION

Dans le cadre des liens existant entre les méthodes de classification et les modèles de statistique inférentielle, nous nous sommes intéressés dans ce chapitre à l'étude des modèles probabilistes liés à la classification de données qualitatives nominales.

Les tableaux étudiés ici sont les tableaux de modalités et les tableaux disjonctifs complets qui résultent de la transformation par le codage disjonctif complet d'un tableau de modalité, croisant individus et variables qualitatives nominales.

Plusieurs méthodes de classification pour ces données ont été proposées, souvent basées sur le principe des Nuées Dynamiques (Diday et all 1980). Citons par exemple la méthode MNDQAL (Ralambondrainy 1988) qui s'applique à des tableaux disjonctifs complets et utilise la métrique du Khi2 pour classer les données, mais les noyaux qu'elle fournit ne sont pas de la même forme que les éléments à classifier, et la méthode MNDDIJ (Marchetti 1989) qui s'applique à un tableau de modalité et qui permet de classifier les données en tenant compte de leur structure initiale particulière, dont les composantes codent les modalités.

La première partie de ce chapitre est consacrée à l'étude des tableaux disjonctifs complets, en particulier à l'étude du lien existant entre la méthode MNDQAL (Méthode des Nuées Dynamiques utilisant la métrique du Khi2 et des noyaux sous forme de profils) et la notion de modèle probabiliste. Nous étudierons dans un premier temps la méthode MNDQAL et l'algorithme de classification qui lui est associé ; nous montrons ensuite à l'aide de l'approche "classification" (Scott et Symons 1971, Schroeder 1976 et Celeux 1988) comment on peut identifier ce mélange. Nous proposons un modèle lié à cette méthode permettant d'interpréter le critère du Khi2 ; celui-ci consiste à considérer que les données du tableaux proviennent d'un mélange de lois gaussiennes

multidimensionnelles, nous rappelerons enfin le modèle proposé par G.Celeux (1988) pour la même méthode et permettant d'interpréter le critère d'information quantité proche de celle du Khi2.

La deuxième partie de ce chapitre traite la notion de modèle lié aux tableaux de modalités. Nous nous intéressons en particulier à la méthode MNDDIJ (Méthode des Nuées Dynamiques utilisant la distance proposée par Marchetti (1989) et des noyaux de modalités). En utilisant les liens existant entre les critères métriques et les critères probabilistes dans le cas discret qui ont étés étudiés par G.Govaert (1990) nous montrons que les données du tableaux proviennent d'un mélange de produit de lois binomiales ; nous apportons ainsi une interprétation en termes probabilistes au critère optimisé par la méthode MNDDIJ.

1. LES TABLEAUX DISJONCTIFS COMPLETS

1.1. EXEMPLE

Considérons un ensemble de 6 individus identifiés par les chiffres **1 à 6.** Ceux-ci sont décrits par 3 variables qualitatives nominales identifiées par les lettres **a, b** et **c.** Supposons que les modalités des variables soient {1, 2, 3} pour **a,** {1, 2, 3, 4} pour **b** et {1, 2, 3, 4, 5} pour **c.** Les valeurs observées sur les individus sont représentées sous la forme d'un tableau de modalité indiqué dans la figure 1. Le tableau de codage disjonctif complet est indiqué dans la figure 2. Celui-ci est obtenu à partir du tableau de la figure 1 en transformant les modalités en variables binaires codées par les valeurs 0 et 1 de sorte que : la valeur 1 signifie que la modalité a été choisie, la valeur 0 qu'elle ne l'est pas. On peut encore considérer que le tableau de codage est celui croisant l'ensemble des individus et l'ensemble des indicatrices de toutes les modalités. Nous avons choisi d'identifier une variable binaire (associée à une modalité) par l'identificateur de la variable suivi, en indice, de la modalité (par exemple, la variable binaire associée à la modalité **1** de **a** est identifiée par $\mathbf{a_1}$).

	a	b	c
1	3	2	3
2	2	4	1
3	3	1	5
4	3	3	2
5	3	3	4
6	1	3	3

figure 1
Tableau de modalités

	a_1	a_2	a_3	b_1	b_2	b_3	b_4	c_1	c_2	c_3	c_4	c_5
1	0	0	1	0	1	0	0	0	0	1	0	0
2	0	1	0	0	0	0	1	1	0	0	0	0
3	0	0	1	1	0	0	0	0	0	0	0	1
4	0	0	1	0	0	1	0	0	1	0	0	0
5	0	0	1	0	0	1	0	0	0	0	1	0
6	1	0	0	0	0	1	0	0	0	1	0	0

figure 2
Tableau de codage

1.2. NOTATIONS ET DEFINITIONS

Soit $\mathbf{X(I, J)_{(n, m)}}$ le tableau de codage disjonctif complet du tableau de modalité $\mathbf{Z(I, Q)_{(n, p)}}$.

La somme des éléments du tableau binaire ne dépend que du nombre d'individus n et du nombre de variables p puisque :

$$\sum_{i\in I}\sum_{j\in J} x_i^j = \sum_{i\in I}\sum_{q\in Q}\sum_{j\in J_q} x_i^{q(j)} = \sum_{i\in I}\sum_{q\in Q}(1) = np$$

où q(j) est l'indice de J correspondant à la modalité j de la variable q et J_q est l'ensemble des modalités de la variable q ; Q est l'ensemble des variables.

On définit également les valeurs suivantes :

$$f_{ij} = \frac{x_i^j}{np} \qquad f_{i.} = \sum_{j\in J} f_{ij} = \frac{1}{np}\sum_{j\in J} x_i^j = \frac{1}{n}$$

$$f_{.j} = \sum_{i\in I} f_{ij} = \frac{1}{n.p}\sum_{i\in I} x_i^j = \frac{n^j}{np} \quad \text{où} \quad n^j = \sum_{i\in I} x_i^j$$

Le profil de l'individu i, noté f_i (et appartenant à $\mathbf{R^m}$), est alors défini par :

$$f_i = \left(\frac{f_{i1}}{f_{i.}}, \frac{f_{i2}}{f_{i.}}, \ldots, \frac{f_{im}}{f_{i.}}\right)$$

Si on note x_i le représentant de l'individu i dans l'espace $B^m = \{0, 1\}^m$, nous avons :

$$f_i = \frac{1}{p} \ (x_i^1, x_i^2, \ldots, x_i^m)$$

$$f_i = \frac{1}{p} \ x_i$$

D'autre part, une pondération p_i est associée à chaque profil f_i ; elle est définie par :

$$p_i = f_{i.} = \frac{1}{n}$$

A partir du tableau X(I, J), nous définissons ainsi un nuage de n profils Nf(I) par :

$$Nf(I) = \{ \ (f_i, \frac{1}{n}), i \in I \ \}$$

1.3. CRITERES DE CLASSIFICATION

1.3.1. Critère d'information

L'information mutuelle d'un tableau X s'écrit par définition (cf Benzécri 1973) :

$$H(X) = \sum_{i \in I} \sum_{j \in J} \frac{x_i^j}{s} \ Log_2 \frac{s \ x_i^j}{x_i \ x^j} \ . \qquad (2.1)$$

où $\quad x_i = \sum_{j \in J} x_i^j \quad$ et $\quad x^j = \sum_{i \in I} x_i^j = n^j.$

Soit $P = (P_1, ..., P_K)$ une partition des individus. Si nous considérons P comme un résumé de l'information apportée par les données, nous pouvons substituer au tableau :

$$X = (x_i^j \ , i = 1, ..., n \quad \text{et} \quad j = 1, ..., p)$$

le tableau $\qquad (x_k^j \ , k = 1, ..., K \ \text{ et } \ j = 1, ..., p)$

$K = \{1, ..., K\}$.

L'information conservée par P s'écrit donc :

$$H(P) = \sum_{k \in K} \sum_{j \in J} \frac{x_k^j}{s} \ Log_2 \frac{s \ x_k^j}{x_k \ x^j} \ . \qquad (2.2)$$

avec $x_k^j = \sum_{i \in P_k} x_i^j$ et $x_k = \sum_{j \in J} x_k^j$.

Dans ce cadre, il est naturel (cf Benzécri 1973) de rechercher la partition qui maximise l'information H(P). Par des transformations algébriques simples, H(P) s'écrit :

$$H(P) = \frac{1}{s} \sum_{k \in K} \sum_{j \in J} x_k^j \ \mathrm{Log}_2 \frac{x_k^j}{x_k} + \mathrm{Log}_2 s - \frac{1}{s} \sum_{j \in J} x^j \ \mathrm{Log}_2 x^j .$$

Il s'en suit que maximiser H(P) revient à maximiser l'expression :

$$H(P) = \sum_{k \in K} \sum_{j \in J} x_k^j \ \mathrm{Log}_2 \frac{x_k^j}{x_k} . \qquad (2.3)$$

1.3.2. Critère du Khi2

L'algorithme utilisé est basé sur la méthode des Nuées Dynamiques ; il fournit une partition en K classes, K étant fixé à priori, de l'ensemble I des individus en optimisant le critère :

$$\begin{aligned} W(P, Lg) &= \sum_{k=1}^{K} \sum_{i \in P_k} f_{i.} d^2_{\chi 2}(f_i, g_k) \\ &= \sum_{k=1}^{K} \sum_{i \in P_k} f_{i.} \sum_{j \in J} \frac{1}{f_{.j}} (f_i^j - g_k^j)^2 \\ &= \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j \in J} \frac{x_i}{x^j} (\frac{x_i^j}{x_i} - \frac{x_k^j}{x_k})^2 \end{aligned} \qquad (2.4)$$

où $P = (P_1, P_2, \ldots, P_K)$ est une partition sur I en K classes et $Lg = (g_1, \ldots, g_K)$ est l'ensemble des noyaux des classes.

1.3.3. Etude du lien entre les deux critères

Nous avons tenu à préciser le lien qui existe entre le critère d'information et le critère du Khi2 car il n'existe pas d'algorithme permettant d'optimiser le premier critère qui est peu utilisé malgré son caractère naturel. On lui préfère le critère du Khi2 ; Benzécri (1973) en exprime très bien la raison : l'expérimentation montre que l'emploi du critère d'information ou du critère du Khi2 conduit pratiquement aux mêmes classifications et le deuxième induit des calculs beaucoup plus simples. Ce comportement quasi identique des deux critères a également été souligné par Govaert (1983), qu'il l'a introduit dans son programme de classification simultanée des lignes

et des colonnes d'un tableau de nombres positifs. Sur les nombreux exemples qu'il a analysé, il a constaté que l'information croissait systématiquement.

1.4. LA METHODE MNDQAL

La méthode MNDQAL (Ralambondrainy 1988) est une méthode de classification sur tableaux décrits par des variables qualitatives nominales. Les noyaux qu'elle fournit ne sont pas de la même forme que les éléments à classifier ; cette méthode travaille sur les tableaux disjonctifs complets et utilise la métrique du Khi2 qui convient particulièrement aux profils. Les noyaux fournit sont les centres de gravité des classes et chacun d'eux est caractérisé par un vecteur de $\mathbf{R}^m$ (où m est le nombre total de modalités).

Le problème à résoudre est alors le suivant :

Trouver une partition $P = (P_1, \ldots, P_K)$ de l'ensemble I en K classes, K fixé a priori, tel que le critère :

$$W(P, Lg) = \sum_{k=1}^{K} \sum_{i \in P_k} f_i . d^2_{\chi 2}(f_i, g_k)$$

soit minimal.

Explicitons l'expression du critère :

$$W(P, Lg) = \sum_{k=1}^{K} \sum_{i \in P_k} f_i . d^2_{\chi 2}(f_i, g_k)$$

$$= p \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j \in J} \frac{1}{n^j} \left(\frac{x_i^j}{p} - g_k^j \right)^2 \qquad (2.5)$$

où $P = (P_1, P_2, \ldots, P_K)$ est une partition sur I en K classes,

et $Lg = (g_1, g_2, \ldots, g_K)$ est l'ensemble des noyaux des classes.

1.4.1. L'algorithme

Les fonctions caractérisant l'algorithme MNDQAL sont les suivantes :

-La fonction d'affectation (f) : qui minimise le critère W (f(Lg), Lg) en affectant chaque individu à la classe P_k du noyau g_k duquel il est le plus proche (au sens de la distance du Khi2).

-***la fonction de représentation (g)*** : qui permet de déterminer les K noyaux minimisant le critère W(P, g(P)). On peut facilement montrer que ces noyaux sont les centres de gravité des classes. Si nous notons $\{g_1, g_2, \ldots, g_K\}$ l'ensemble de ces centres, on a :

$$\forall \quad k = 1, \ldots, K \quad \forall \quad j \in J \qquad g_k^j = \frac{n_k^j}{n_k \, p}$$

où n_k est le nombre d'individus appartenant à la classe P_K et $n_k^j = \sum_{i \in P_k} x_i^j$ est le nombre d'individus de P_k ayant choisi la modalité j.

Ces noyaux ne sont pas ici directement interprétables par rapport aux données initiales.

1.4.2. Autres expressions du critère

On reprend ici l'expression (2.5) du critère minimisé par l'algorithme MNDQAL :

$$W(P, Lg) = p \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j \in J} \frac{1}{n^j} \left(\frac{x_i^j}{p} - g_k^j \right)^2$$

$$= p \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j \in J} \frac{1}{n^j} \left(f_i^j - g_k^j \right)^2 \qquad (2.6)$$

où $f_i^j = \frac{x_i^j}{p}$.

On définit une matrice M diagonale de terme général $\frac{1}{n^j}$ qui représente la métrique des poids. C'est une matrice définie positive et de déterminant constant.

$$|M| = \prod_{j=1}^{p} \frac{1}{n^j} = cte \qquad car \qquad n^j = \sum_{i \in I} x_i^j$$

L'expression (2.6) peut alors se mettre sous la forme :

$$W(P, Lg) = p \sum_{k=1}^{K} \sum_{i \in P_k} t(f_i - g_k).M.(f_i - g_k) \qquad (2.7)$$

En posant $\lambda_k^j = \frac{n_k^j}{n_k}$ où λ_k^j est la fréquence de la variable j dans la classe P_k

alors $g_k^j = \frac{\lambda_k^j}{p}$; si on remplace maintenant la nouvelle expression de g_k^j dans

l'expression du critère (2.5), on obtient :

$$W(P, L_\lambda) = \frac{1}{p} \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j \in J} \frac{1}{n^j} (x_i^j - \lambda_k^j)^2$$

qui peut encore se mettre sous la forme suivante :

$$W(P, L_\lambda) = \frac{1}{p} \sum_{k=1}^{K} \sum_{i \in P_k} t(x_i - \lambda_k).M.(x_i - \lambda_k) \qquad (2.8)$$

1.5. APPROCHE MODELE

Ce paragraphe comporte une étude des liens s'ils existent entre les critères optimisés par la méthode MNDQAL et les modèles probabilistes en s'appuyant essentiellement sur l'étude faite par G.Govaert (1989 et 1990) et rappelée dans le chapitre 1. Cette étude sera faite en trois étapes. La première consiste à travailler sur l'ensemble des profils associés aux données du tableau que l'on plonge dans l'espace **continu R^m** (où m est le nombre total de modalités) muni de la métrique du Khi2. La deuxième approche travaille directement sur les données du tableau qui forment un ensemble inclus dans l'espace **discret $\{0, 1\}^m$** que l'on munit de la métrique du Khi2.
Dans la dernière approche on rappelle le modèle proposé par G.Celeux (1988) ; celui-ci est directement lié au critère d'information. Nous étudierons ensuite le lien qui existe entre ces trois approches.

1.5.1. Première approche

Les données du tableau sont prises sous la forme d'un ensemble de profils inclus dans l'espace **continu R^m**. Nous munissons cet espace de la métrique du Khi2 qui est une métrique quadratique et qui se présente pour les tableaux disjonctifs complets sous la forme d'une métrique quadratique fixe et identique pour toutes les classes (voir l'expression (2.7)). En se basant sur l'étude faite par G.Govaert (1989) sur les liens entre les critères métriques et les critères probabilistes dans le cas continu, on pourra

dire que le critère métrique (2.7) est équivalent à un critère probabiliste si la quantité $\int_{R^p} e^{-D(f,g_k)}\, df$ est indépendante de g_k.

Calculons cette intégrale :

$$\int_{R^p} e^{-D(f,g_k)}\, df = \int_{R^p} e^{-{}^t(f-g_k).M.(f-g_k)}\, df = \pi^{-p/2}\,.\,|M|^{1/2}.$$

Cette quantité est constante car $|M| = \prod_{j=1}^{p} \frac{1}{n^j} = cte$ où $n^j = \sum_{i \in I} x_i^j$. Les conditions de la proposition (1.4) sont vérifiées. Il existe alors un critère probabiliste équivalent au critère métrique (2.7) ; ce critère probabiliste consiste à prendre comme fonction de densité sur l'espace $\mathbf{R}^m$ une loi gaussienne de centre μ_k et de matrice de variance $\Gamma = 2.M^{-1}$, où M est la matrice définie précédemment et de terme général $\frac{1}{n^j}$.

$$f(f_i / \mu_k) = \pi^{-p/2}\,.\,|M|^{1/2}.\, e^{-{}^t(f_i-\mu_k).M.(f_i-\mu_k)}$$

Dans le cas d'un mélange gaussien, les paramètres $(\lambda_k, k = 1, ..., K)$ s'écrivent : $\lambda_k = (\mu_k, \Gamma_k)$; avec :

μ_k : espérance du composant numéro k.
Γ_k : matrice de variance du composant numéro k.

Dans notre cas on prend $\Gamma_k = \Gamma \quad \forall\ k = 1, ..., K$ et Γ connu .
Le critère de vraisemblance classifiante s'écrit alors :

$$VC(P, L_\mu) = \sum_{k=1}^{K} \sum_{i \in P_k} Log\ \{\pi^{-p/2}\,.\,|M|^{1/2}.\, e^{-{}^t(f_i-\mu_k).M.(f_i-\mu_k)}\}$$

$$VC(P, L_\mu) = -\frac{n.p}{2} Log\pi + \frac{n}{2} Log|M| - \sum_{k=1}^{K} \sum_{i \in P_k} {}^t(f_i - \mu_k).M.(f_i - \mu_k)$$

Les deux premiers termes de cette expression sont constants. Maximiser $VC(P, L\mu)$ revient donc a minimiser l'expression suivante :

$$C(P, L_\mu) = \sum_{k=1}^{K} \sum_{i \in P_k} {}^t(f_i - \mu_k).M.(f_i - \mu_k)$$

Il est facile de vérifier que l'estimateur du maximum de μ_k n'est autre que le centre de gravité des profils f_i de la classe P_k. Il est estimé par :

$$\mu_k = g_k = \frac{1}{n_k} \sum_{i \in P_k} f_i$$

d'où $\quad \forall\ j = 1,\ldots, p \qquad \mu_k^j = g_k^j = \frac{1}{n_k} \sum_{i \in P_k} \frac{x_i^j}{p} = \frac{n_k^j}{n_k\ p}$

Le critère à minimiser prend alors la forme suivante :

$$W(P, Lg) = \sum_{k=1}^{K} \sum_{i \in P_k} {}^t(f_i - g_k).M.(f_i - g_k)$$

qui correspond, à une constante multiplicative près, à l'expression (2.7) du critère minimisé par la méthode MNDQAL ; ainsi nous avons apporté dans cette première approche une interprétation en termes probabilistes du critère du Khi2 minimisé par l'algorithme MNDQAL, et cela en travaillant sur l'ensemble des profils que l'on considère comme ensemble continu. Mais il serait intéressant dans ce travail d'essayer de voir si on peut trouver un modèle qui s'appuie directement sur les données initiales, lesquelles se présentent sous la forme d'un ensemble discret. Nous allons essayer de voir ce que devient la notion de modèle dans ce cas.

1.5.2. Deuxième approche

Cette approche travaille directement sur les données initiales du tableau en prenant comme ensemble à classifier les vecteurs binaires appartenant à l'espace **discret** $\{0,1\}^m$ que l'on munit toujours de la métrique du Khi2. En utilisant l'étude faite par Govaert (1990) sur les liens entre les critères métriques et les critères probabilistes dans le cas discret, on va essayer de voir si les conditions de la proposition (1.7) sont remplies.

L'ensemble à classifier appartient à l'espace $E = \{0, 1\}^m$. Les noyaux sont ici des éléments de $\mathbf{R}^m$ et leur nature est différente de celle des données à classifier.

$E = \{0, 1\}^m \qquad E = E_1 x \ldots x E_p \quad$ où $\quad E_j = \{0, 1\}.\ \forall\ j = 1, \ldots, p$

$L = \mathbf{R}^m$, la fonction D n'est autre que la métrique du Khi2 définie par :
$\forall\ x \in E\ \ \forall\ \lambda_k \in \mathbf{R}^m$

$$D(x, \lambda_k) = d^2_{\chi 2}(f, g_k) = \sum_{j \in J} \frac{1}{f_{.j}} (f^j - g_k^j)^2 = \sum_{j \in J} \frac{n}{n^j.p} (x^j - \lambda_k^j)^2$$

Le critère métrique (2.8) est équivalent à un critère probabiliste si la quantité $\sum_{x \in E} e^{-t(x-\lambda_k).M.(x-\lambda_k)}$ est indépendante de λ_k (proposition 1.7) ou encore si la quantité $s_j = \sum_{x^j \in E^j} \exp\left\{ -\frac{n}{n^j.p} (x^j - \lambda_k^j)^2 \right\}$ est indépendante de λ_k^j (proposition 1.9).

or $\sum_{x^j \in E^j} \exp\left\{ -\frac{n}{n^j.p} (x^j - \lambda_k^j)^2 \right\} = \exp\left[-\frac{n}{n^j.p} .(\lambda_k^j)^2 \right].\left[1 + \exp\left\{ -\frac{n}{n^j.p} (1 - 2\lambda_k^j) \right\} \right]$.

Cette quantité dépend de λ_k^j . La proposition (1.9) n'est pas vérifiée. Par conséquent il n'existe pas de modèle probabiliste qui soit lié directement au critère du Khi2 optimisé par la méthode MNDQAL dans le cas où l'on travaille directement sur les données initiales plongées dans l'espace discret $\mathbf{E = \{0, 1\}^m}$.

G.Celeux (1988), en travaillant sur les mêmes données (initiales), a montré qu'il existe un modèle sous-jacent permettant d'apporter un éclairage nouveau au critère d'information. L'étude de ce modèle fera l'objet de la troisième approche.

1.5.3. Troisième approche

1.5.3.1. Notations

Soit **X** le tableau disjonctif complet à n lignes et m colonnes où $m = \sum_{q \in Q} m_q$ (m_q est le nombre de modalités de la variable q).

On note x_i^{qj} le terme générique du tableau **X**. x_i^{qj} vaut 1 si l'individu i présente la modalité j de la variable q et 0 sinon. On pose :

$$x^{qj} = \sum_{i \in I} x_i^{qj} \quad ; \quad x_i = \sum_{q \in Q} \sum_{j \in m_q} x_i^{qj} = p \qquad \text{et} \qquad s = \sum_{i \in I} \sum_{q \in Q} \sum_{j \in m_q} x_i^{qj} = np.$$

De plus, si P_k est une classe d'une partition $P = (P_1, \ldots, P_K)$ des individus, on note :

$$x_k^{qj} = \sum_{i \in P_k} x_i^{qj} \text{ et } x_k = \sum_{q \in Q} \sum_{j \in m_q} x_k^{qj} = p.n_k \qquad \text{où} \qquad n_k = \text{card } P_k.$$

1.5.3.2. Modèle de Celeux

Le modèle proposé par Celeux (1988) consiste à considérer que les données du tableau proviennent d'un mélange de produit de p lois multinomiales multivariées soit :

$$f(x/\lambda_k) = \prod_{q=1}^{p} \prod_{j=1}^{m_q} (\lambda_k^{qj})^{x^j}$$

où $\lambda_k = (\lambda_k^{qj} , j = 1, ..., m_q \text{ et } q = 1, ..., p)$

avec $\sum_{j=1}^{m_q} \lambda_k^{qj} = 1 \qquad \forall \quad k= 1, ..., K \quad \text{et} \quad q = 1, ..., p.$

Le critère de vraisemblance classifiante s'écrit alors :

$$VC(P, L) = \sum_{k=1}^{K} \sum_{x \in P_k} \text{Log} \prod_{q=1}^{p} \prod_{j=1}^{m_q} (\lambda_k^{qj})^{x^j}$$

$$= \sum_{k=1}^{K} \sum_{q=1}^{p} \sum_{j \in m_q} x_k^{qj} \text{Log} (\lambda_k^{qj}).$$

En maximisant le critère VC(P, L) en tenant compte de la contrainte $\sum_{j \in m_q} \lambda_k^{qj} = 1$ on obtient : $\lambda_k^{qj} = \frac{x_k^{qj}}{n_k}$

autrement dit $\lambda_k = (\lambda_k^1 , ..., \lambda_k^p)$ est le centre de gravité de la classe P_k. Il s'en suit que le critère de vraisemblance classifiante s'écrit :

$$VC(P, L) = \sum_{k=1}^{K} \sum_{q=1}^{p} \sum_{j \in m_q} x_k^{qj} \text{Log} \frac{x_k^{qj}}{n_k}. \qquad (2.9)$$

or $n_k = \frac{x_k}{p}$. Le critère (2.9) se réduit :

$$W(P, L) = \sum_{k=1}^{K} \sum_{q=1}^{p} \sum_{j \in m_q} x_k^{qj} \text{Log} \frac{x_k^{qj}}{x_k}$$

qui correspond à la constante multiplicative près $\frac{1}{\text{Log}2}$ au critère d'information donnée par l'expression (2.3).

1.6. CONCLUSION

Nous venons donc, par l'étude de ces trois approches d'apporter des éclairages nouveaux sur les deux types de critères optimisés par la méthode MNDQAL. Selon

l'optique statistique dans laquelle on se place, on retrouve à chaque fois l'un des deux critères ; dans la première approche, où l'on travaille sur l'ensemble des profils, on montre que la méthode MNDQAL est liée à un mélange de lois gaussiennes multidimensionnelles ; cette méthode n'est qu'un cas particulier de la méthode MNDQAN (Méthode des Nuées Dynamiques sur des données quantitatives) dans le sens où elle s'applique à des données particulières (Données binaires) ; or Celeux (1988) à montré que la méthode MNDQAN était elle aussi liée à un mélange gaussien. On retrouve donc un résultat déjà connu. D'autre part, on montre que si l'on travaille sur les données initiales du tableau qui constituent un ensemble discret , qu'il n'existe pas de modèle équivalent au critère métrique défini à l'aide de la métrique du Khi2, mais on peut trouver une interprétation en termes probabilistes du critère d'information qui est une quantité proche du Khi2.

Nous proposons maintenant de voir ce que devient la notion de modèle dans le cas où les données sont décrites par un tableaux de modalités et où les noyaux ont la même structure que les données initiales.

2. LES TABLEAUX DE MODALITES

2.1. NOTATIONS ET DEFINITIONS

Soit **Z(I, Q)** le tableau de modalités croisant un ensemble **I** = {1, 2, ..., n} de n individus et un ensemble Q = {1, 2, ..., p} de p variables qualitatives nominales. On note :

$$Z(I, Q) = (z_i^q)$$

où z_i^q représente la modalité de la variable q choisie par l'individu i.

A chaque variable q correspond l'ensemble de modalités $J_q = \{1, 2, ..., m_q\}$. Nous définissons alors l'espace **E** comme le produit $J_1 x J_2 x ... x J_p$, que nous munissons de la distance d_E, égale au nombre de composantes différentes entre les deux points considérés ; son expression sur **E** est la suivante :

$$\forall (x, y) \in E^2 \quad d_E(x, y) = \sum_{q \in Q} \delta^q(x, y)$$

$$\text{où} \quad \delta^q(x, y) = \begin{cases} 1 \text{ si } x^q \neq y^q \\ 0 \text{ sinon} \end{cases}$$

A partir du tableau **Z(I, Q)**, nous définissons le nuage $N_Z(I)$, inclus dans l'espace **E**, par :

$$N_Z(I) = \{ z_i, i \in I \}$$

$$\text{où} \quad z_i = (z_i^1, z_i^2, \ldots, z_i^p)$$

Soit **X(I, J)** le tableau de codage disjonctif complet du tableau de modalités **Z(I, Q)**. C'est un tableau binaire d'ordre (n, m), où m est le nombre total de modalités. L'ensemble **J** = {1, 2, ..., m} contient les indices des colonnes de **X(I, J)**. On note :

$$X(I, J) = (x_i^j)$$

$$\forall\, q \in Q, \forall\, j \in J_q \quad x_i^{q(j)} = \begin{cases} 1 \text{ si } j = z_i^q \\ 0 \text{ sinon} \end{cases}$$

où q(j) est l'indice de **J** correspondant à la modalité j de la variable q.

Nous définissons l'espace **F** comme l'espace des vecteurs binaires de modalités. Les éléments de **F** résultent du codage des vecteurs de modalités de l'espace **E**. De même, à tout élément de **E** correspond un élément de **F**.

Nous munissons **F** de la distance L_1, notée d. Les distances d et d_E sont liées par une relation très simple. Si X, Y sont deux vecteurs de modalités (appartenant à **E**) de codage x, y (appartenant à **F**), cette relation s'exprime par :

$$d(x, y) = 2\, d_E(X, Y)$$

A partir du tableau **X(I, J)**, nous définissons le nuage **N(I)**, inclus dans **F**, par :

$$N(I) = \{ x_i, i \in I \} \quad \text{où} \quad x_i = (x_i^1, x_i^2, \ldots, x_i^m)$$

Ainsi, à tout point x_i du nuage **N(I)** correspond un unique point z_i de $N_Z(I)$ et réciproquement.

2.2. LA METHODE MNDDIJ

La méthode MNDDIJ (Marchetti 1989) permet de déterminer une partition de l'ensemble **I** des individus en K classes, K étant fixé a priori. Chaque individu i est représenté par un point x_i du nuage **N(I)**. Celui-ci est inclus dans **F**. Les noyaux doivent appartenir à ce même espace **F**.

Le problème à résoudre est alors le suivant :

Trouver une partition $P = (P_1, \ldots, P_K)$ *de l'ensemble* I *et un ensemble* $L = (\lambda_1, \ldots, \lambda_K)$ *de* K *noyaux de* F *tels que le critère :*

$$W(P, L) = \sum_{k=1}^{K} \sum_{i \in P_k} d(x_i, \lambda_k) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j \in J} / x_i^j - \lambda_k^j /$$

soit minimal.

On peut encore représenter chaque individu i par un point z_i du nuage $N_Z(I)$ inclus dans l'espace E, les noyaux doivent aussi appartenir à ce même espace.

Le problème à résoudre peut aussi s'énoncer de la façon suivante :

Trouver une partition $P = (P_1, \ldots, P_K)$ *de l'ensemble* I *et un ensemble* $L = (A_1, \ldots, A_K)$ *de* K *noyaux de* E *tels que le critère :*

$$W(P, L_E) = \sum_{k=1}^{K} \sum_{i \in P_k} d_E(z_i, A_k)$$

soit minimal.

Il est clair que : $W(P, L) = 2\, W(P, L_E)$.

Il s'agit maintenant de définir la fonction d'affectation f, telle que W(f(L), L), ou encore $W(f(L_E), L_E)$, soit minimal, et la fonction de représentation g telle que W(P, g(P)) soit minimum.

Fonction d'affectation f :

L'individu i est affecté à la classe P_k du noyau λ_k duquel il est le plus proche , au sens de la distance L_1 sur le tableau binaire, ou au sens de la distance d_E sur le tableau des modalités.

Fonction de représentation g :

Il s'agit de déterminer l'ensemble L des K noyaux optimisant le critère W(P, g(P)). Pour cela, il suffit de rechercher pour toute classe P_k, le noyau λ_k appartenant à F et minimisant la quantité :

$$\sum_{i \in P_k} d(x_i, \lambda_k)$$

Soit A_k appartenant à **E**, le vecteur de modalités codé par λ_k. Les distances d_E sur **E** et d sur **F** étant liées par la relation suivante :

$$d(x_i, \lambda_k) = 2\, d_E(z_i, A_k)$$

Le problème est alors de trouver le point A_k de **E** minimisant :

$$\sum_{i \in P_k} d_E(z_i, A_k) = \sum_{i \in P_k} \sum_{q \in Q} \delta^q(z_i, A_k)$$

ce qui revient à déterminer, pour tout q, la composante A_k^q minimisant la quantité :

$$\sum_{i \in P_k} \delta^q(z_i, A_k)$$

qui représente le nombre d'individus de P_k n'ayant pas choisi la modalité A_k^q.

La solution est de choisir pour composantes q du point A_k, la modalité de la variable q qui est majoritaire relative dans la classe considérée. Cette composante est donc déterminée de la façon suivante :

$$A_k^q = \text{modalité majoritaire relative de } \{ z_i, i \in P_k \}.$$

Finalement, le noyau λ_k recherché est le transformé du vecteur de modalités dont les composantes sont les modalités majoritaires relatives des variables dans la classe P_k.

L'algorithme ainsi construit fournit un ensemble de noyaux du même type que les éléments à classifier. Il représente la modalité la plus souvent choisie par les individus de la classe considérée.

Le critère représente le nombre d'éléments binaire du tableau **X(I, J)** qui sont différents des éléments du tableau correspondant à la situation "idéale" (situation où tous les individus sont identiques aux noyaux des classes auxquelles ils appartiennent).

2.3. APPROCHE MODELE

2.3.1. La formule générale

Le modèle que l'on va proposer portera sur le tableau de modalités. Pour cela on considère que les données initiales forment un échantillon de taille n d'une variable aléatoire à valeurs dans N^p dont la distribution de probabilité f est toujours définie par

(1.1) et (1.2) ; mais ici $p(./\lambda_k)$ est une distribution de probabilités sur N^p appartenant à une famille paramétrée de distributions de probabilités.
En suivant l'approche "classification", rappelée dans le chapitre 1 pour identifier le mélange, on se ramène à maximiser le critère de vraisemblance classifiante VC(P, L) défini par (1.3).

On peut alors utiliser le même algorithme que dans le cas des données continues. A partir d'une partition P^o en K classes de l'échantillon, on applique successivement les deux fonctions g et h définies aux paragraphes 2.2.2 et 2.2.3 (chapitre 1) jusqu'à l'obtention d'une partition stable.

2.3.2. Choix de la famille de distribution

Le modèle que l'on propose pour interpréter la méthode MNDDIJ suppose que pour chaque composant du mélange, les p variables sont indépendantes, et que chacune d'entre elles suit la loi binomiale suivante :

1- ε est la probabilité d'avoir la modalité du noyau, ε est la probabilité d'avoir une modalité différente de celle du noyau.

avec $\varepsilon \in]0, \frac{1}{2}[$

On peut alors écrire :

$$p(z/A_k) = \prod_{q=1}^{p} p(z/A_k^q)$$

où $$p(z/ A_k^q) = (1- \varepsilon)^{1- \delta^q(z, A_k)}. (\varepsilon)^{\delta^q(z, A_k)}$$

$$\text{avec} \quad \delta^q(z, A_k) = \begin{cases} 1 & \text{si } z^q \neq A_k^q \\ 0 & \text{sinon} \end{cases} \qquad \varepsilon \in]0, \frac{1}{2}[$$

Le critère de vraisemblance classifiante s'écrit alors :

$$VC(P, L) = \sum_{k=1}^{K} \sum_{z \in P_k} \text{Log } p(z/A_k)$$

$$VC(P, L) = \sum_{k=1}^{K} \sum_{z \in P_k} \text{Log} \{ \prod_{q=1}^{p} (1-\varepsilon)^{1-\delta^q(z, A_k)} . \varepsilon^{\delta^q(z, A_k)} \}$$

$$= \text{Log} \left(\frac{\varepsilon}{1-\varepsilon} \right) \sum_{k=1}^{K} \sum_{z \in P_k} \sum_{q=1}^{p} \delta^q(z, A_k) + n.p \sum_{q=1}^{p} \text{Log} (1-\varepsilon)$$

Pour ε fixé appartenant à $]0, \frac{1}{2}[$, $\text{Log}\frac{\varepsilon}{1-\varepsilon}$ est négatif. La maximisation du critère $VC(P, L, \varepsilon)$ est équivalente à la minimisation du critère $W(P, L)$ associé à la méthode MNDDIJ, ce qui montre l'équivalence des deux approches.

Il est facile de voir que la valeur ε maximisant le critère $VC(P, L, \varepsilon)$ est $\frac{e}{n.p}$, où e est la valeur du critère $W(P, L)$ obtenue à la convergence. Notons que l'estimateur du maximum de vraisemblance classifiante de A_k^q est atteint pour la valeur qui correspond pour chaque classe et chaque variable à la modalité majoritaire.

CHAPITRE 3

CLASSIFICATION BINAIRE ET DISTANCE L_1 ADAPTATIVE

INTRODUCTION

Dans la plupart des méthodes de classification proposées jusqu'alors, la mesure de ressemblance est supposée connue au départ et fixée définitivement. Par contre, dans les méthodes de discrimination, on cherche une mesure de ressemblance bien adaptée à une partition donnée au départ et qui variera suivant les classes de cette partition. Govaert (1975) a exploité cette idée pour la recherche de partitions. Il a proposé un algorithme original où la mesure de ressemblance n'est pas la même pour toutes les classes et de plus évolue au cours du déroulement de l'algorithme en s'adaptant aux classes.

On s'intéresse dans ce chapitre à une étude comparative entre les algorithmes utilisant une mesure de ressemblance fixe pendant leur déroulement et les algorithmes utilisant une mesure de ressemblance qui varie lorsque les données sont binaires. Cette dernière fait intervenir un système de pondération pour classer les données.

L'extension du modèle proposé par G.Govaert (1988) pour les données binaires a permis de proposer un algorithme nouveau utilisant une distance de type L_1 munie d'un système de pondération qui varie avec les variables puis avec les variables et les classes. Nous rappelons, dans un premier temps, le modèle associé aux données binaires (Govaert 1988), puis nous examinons comment un système de pondération peut intervenir pour classer les données. Nous proposons ensuite une étude théorique des éventuels problèmes de dégénérescence. Nous terminons par des applications pratiques faites sur deux types de données et nous montrons l'avantage que présente l'algorithme **MNDBIN adaptatif** en particulier sur des données simulées qui suivent à chaque fois une des variantes du modèle.

1. LA METHODE MNDBIN

La méthode de classification des Nuées Dynamiques (Celeux et alt. 1989) repose essentiellement sur l'utilisation de la notion de noyau associé à chaque classe. La nature de ce noyau peut être très diverse. Dans le cas le plus simple, et si les variables sont quantitatives, le noyau est un élément de l'espace $\mathbf{R}^p$ contenant l'ensemble à classifier. Lorsque les variables ne sont pas quantitatives, les noyaux fournis par la méthode des Nuées Dynamiques habituellement utilisés dans ce cas ont alors une structure différente des données initiales. La méthode MNDBIN que nous allons étudier utilise des noyaux de même nature que les données à classifier.

Les n individus de l'ensemble **I** à classifier sont mesurés par p variables binaires et notés :

$$x_i = (x_i^1 , ..., x_i^P) \quad \text{avec} \quad x_i^j \in \{0, 1\} \quad \forall\ i = 1, ..., n \quad \forall\ j = 1, ..., p.$$

On note $\boldsymbol{B} = \{0,1\}^p$ l'ensemble des vecteurs binaires à p composantes. Tous les éléments x_i de **I** appartiennent donc à $\boldsymbol{B}$. Les noyaux sont aussi des vecteurs binaires, mais ne sont pas nécessairement des éléments de **I**. Il suffit donc de prendre une distance sur $\boldsymbol{B}$. On a retenu celle qui consiste à prendre le nombre de fois où les coordonnées ne sont pas identiques. On peut l'exprimer à l'aide de la distance L_1, ou distance " City Block ", sur l'espace $\{0,1\}^p$.

Le problème que l'on cherche à résoudre est alors le suivant :

*Trouver une partition $P = (P_1, ..., P_K)$ de **I** et un ensemble de K éléments de $\boldsymbol{B}$, soit $L = (\lambda_1, ..., \lambda_K)$, tels que :*

$$W(P, L) = \sum_{k=1}^{K} \sum_{x \in P_k} D(x, \lambda_k)$$

soit minimal..

L'algorithme se construit alors de la manière habituelle suivante :

-***Construction des classes (fonction f)*** : Cette fonction ne pose aucun problème. On range chaque élément de **I** dans la classe du noyau duquel il est le plus proche.

-***Construction des noyaux (fonction g)*** : La fonction g va dépendre du choix de la mesure de dissimilarité. On associe à chaque classe P_k le vecteur binaire λ_k minimisant :

$$\sum_{x \in P_k} d(x, \lambda_k) = \sum_{x \in P_k} \sum_{j=1}^{p} |x^j - \lambda_k^j| = \sum_{j=1}^{p} \sum_{x \in P_k} |x^j - \lambda_k^j|$$

λ_k^j est donc l'élément majoritaire pour la variable j dans la classe P_k (on retrouve la notion de médiane). La construction du noyau est donc très simple.

A la convergence, le noyau étant fonction de la partition, on peut exprimer le critère uniquement par rapport à la partition. On obtient :

$$W'(P) = W(P, L) = W(P, g(P))$$

$$= \sum_{k=1}^{K} \sum_{j=1}^{p} |x^j - \lambda_k^j| = \sum_{k=1}^{K} \sum_{j=1}^{p} A_k^j$$

où A_k^j est le nombre d'éléments minoritaires dans la classe P_k pour la variable j.

Ce critère représente le nombre de fois où la solution obtenue s'écarte de la situation "idéale".

Pour illustrer le principe de cet algorithme, nous proposons de l'appliquer sur l'exemple suivant :

1.1. EXEMPLE

Soit un ensemble de 10 micro-ordinateurs identifiés par les lettres de a à j et caractérisés par un ensemble de 10 propriétés identifiées par les nombres de 1 à 10. On représente les données initiales (figure 1) sous forme d'un tableau binaire où un 1 indique que la propriété est vérifiée et un 0 qu'elle ne l'est pas.
Si on applique l'algorithme précédent en demandant 3 classes, on obtient comme partition de l'ensemble des micro-ordinateurs l'ensemble des classes {{a,d,h},{b,e,f,j},{c,g,i}}. On peut représenter cette partition sur les données initiales (figure 2) en réordonnant simplement les lignes de manière à respecter la partition. Les noyaux obtenus sont indiqués dans la figure 3 et le tableau des écarts A_k^j à la valeur idéale dans la figure 4. La valeur du critère est égale à 15, ce qui indique que sur 100 valeurs initiales du tableau, 15 ne sont pas égales à la valeur idéale représentée par le noyau correspondant.

	1	2	3	4	5	6	7	8	9	10
a	1	0	1	0	1	0	0	1	0	1
b	0	1	0	1	0	1	1	0	1	0
c	1	0	0	0	0	0	0	1	1	0
d	1	0	1	0	0	0	0	1	0	0
e	0	1	0	1	1	1	1	0	1	0
f	0	1	0	0	1	1	1	0	1	0
g	0	1	0	0	0	0	0	1	0	1
h	1	0	1	0	1	1	0	1	1	1
i	1	0	0	1	0	0	0	0	0	1
j	0	1	0	1	0	0	1	0	0	0

fig. 1
Tableau initial

	1	2	3	4	5	6	7	8	9	10
a	1	0	1	0	1	0	0	1	0	1
d	1	0	1	0	0	0	0	1	0	0
h	1	0	1	0	1	1	0	1	1	1
b	0	1	0	1	0	1	1	0	1	0
e	0	1	0	1	1	1	1	0	1	0
f	0	1	0	0	1	1	1	0	1	0
j	0	1	0	1	0	0	1	0	0	0
c	1	0	0	0	0	0	0	1	1	0
g	0	1	0	0	0	0	0	1	0	1
i	1	0	0	1	0	0	0	0	0	1

fig. 2
Tableau réordonné

	1	2	3	4	5	6	7	8	9	10
A	1	0	1	0	1	0	0	1	0	1
B	0	1	0	1	0	1	1	0	1	0
C	1	0	0	0	0	0	0	1	0	1

fig. 3
Les noyaux

	1	2	3	4	5	6	7	8	9	10
A	0	0	0	0	1	1	0	0	1	1
B	0	0	0	1	2	1	0	0	1	0
C	1	1	0	1	0	0	0	1	1	1

fig. 4
Tableau des écarts

Les données initiales sont résumées par K vecteurs binaires très facilement interprétables. La qualité du résultat, qui est fournie par la valeur du critère à la convergence, est simple à comprendre puisqu'il représente, nous l'avons vu, le nombre de différences entre les vecteurs binaires de départ et les vecteurs binaires caractérisant leur classe.

On rappelle maintenant comment l'algorithme MNDBIN que nous venons de décrire peut être interprété et justifié en terme de modèle probabiliste.

2. MODELE ASSOCIE AUX DONNEES BINAIRES

En suivant l'approche classification rappelée dans le chapitre 1, Govaert (1988) a proposé le modèle suivant.

2.1. LA FORME GENERALE

On considère dans ce modèle que les données initiales forment un échantillon de taille n d'une variable aléatoire à valeurs dans $\{0,1\}^p$ dont la distribution de probabilité f est toujours définie par les expressions (1.1) et (1.2) ; mais ici $p(./\lambda_k)$ est une distribution de probabilités sur $\{0,1\}^p$ appartenant à une famille paramétrée de distributions de probabilités.

En suivant l'approche "classification", rappelée dans le paragraphe 3, chapitre 1 pour identifier le mélange, on se ramène à maximiser le critère de vraisemblance classifiante VC(P, L) défini par l'expression (1.3).

On peut alors utiliser le même algorithme que dans le cas des données continues. A partir d'une partition P^o en K classes de l'échantillon, on applique successivement les deux fonctions f et g jusqu'à l'obtention d'une partition stable.

2.2. CHOIX DE LA FAMILLE DE DISTRIBUTION

On suppose que, pour chaque composant du mélange, les p variables sont indépendantes, et que chacune d'entre elles suit une des deux lois de Bernoulli suivantes :

$$\begin{cases} 1 \text{ avec la probabilité } 1-\varepsilon \text{ et } 0 \text{ avec la probabilité } \varepsilon \\ 1 \text{ avec la probabilité } \varepsilon \text{ et } 0 \text{ avec la probabilité } 1-\varepsilon \end{cases}$$

où $\varepsilon \in]0, \frac{1}{2}[$, c'est-à-dire la loi de Bernoulli de paramètre $(1-\varepsilon)$ et la loi de Bernoulli de paramètre ε.

Le critère de vraisemblance classifiante s'écrit alors :

$$VC(P, L, \varepsilon) = \sum_{k=1}^{K} \sum_{x \in P_k} \text{Log} \{ \prod_{j=1}^{p} \varepsilon^{|x^j - \lambda_k^j|} . (1-\varepsilon)^{1-|x^j-\lambda_k^j|} \}$$

$$VC(P, L, \varepsilon) = \left(\text{Log} \frac{\varepsilon}{1-\varepsilon} \right) \sum_{k=1}^{K} \sum_{x \in P_k} \sum_{j=1}^{p} |x^j - \lambda_k^j| + n.p \text{ Log}(1-\varepsilon) \qquad (3.1)$$

Donc, pour ε fixé appartenant à $]0, \frac{1}{2}[$, $\text{Log} \frac{\varepsilon}{1-\varepsilon}$ est négatif. La maximisation du critère $VC(P, L, \varepsilon)$ est équivalente à la minimisation du critère W(P, L) associé à la méthode MNDBIN, ce qui montre l'équivalence des deux approches.

Il est facile de voir que la valeur ε maximisant le critère VC (P, L, ε) est $\frac{e}{n.p}$, où e est la valeur du critère W(P, L) obtenue à la convergence.

Ce premier mélange (noté M_1) de lois de Bernoulli multivariées dépend du paramètre ε qui mesure l'écart d'une classe à son centre et ne dépend ni des variables ni des classes ce qui, dans certaines situations, peut s'avérer irréaliste ; ce fait est à rapprocher de celui des distances adaptatives dans le cas continu, lorsque le choix de la métrique M est fixé ($M = I_d$). Mais ce choix à tendance à donner des classes sphériques de même volume, ce qui limite son champ d'application. On se retrouve devant un critère simple utilisant une distance non pondérée. Certains auteurs ce sont attachés à dépasser cette limitation (Chernoff 1970, Friedman et Rudin 1967, Govaert 1975) en supprimant la contrainte pour la métrique M d'être fixée tout au long de l'algorithme tout en restant dans le cadre euclidien de façon à obtenir une dissemblance qui s'adapte aux données traitées. Ces auteurs ont été obligés de fixer des contraintes sur la métrique M ; la solution proposée est l'utilisation de la méthode des Nuées Dynamiques avec la distance adaptative d_M unique telle que $|M| = 1$; en effet, cette contrainte est assez naturelle, néanmoins on pourrait en envisager d'autres.
Dans le cas où les classes ont le même type de dispersion mais possédant des directions d'allongement inconnues, ils utilisent à chaque itération la métrique $M=|W|^{1/p}\cdot W^{-1}$ où $W = \sum_{k=1}^{K} \sum_{i\in P_k} {}^t(x_i - g_k)(x_i - g_k)$ et g_k est le centre de gravité de la classe P_k.
Par ailleurs, le critère de l'algorithme des distances adaptatives (Govaert 1975) élaboré dans un cadre géométrique pour permettre de reconnaître des classes de "formes" différentes, associe à chaque classe P_k et pour chaque itération la métrique suivante $|W_k|^{1/p}\cdot W_k^{-1}$ où $W_k = \sum_{i\in P_k} {}^t(x_i - g_k)(x_i - g_k)$.
Ainsi Govaert (1988) s'appuie toujours sur le modèle précédent pour proposer d'autres critères où le paramètre ε peut dépendre des variables et des classes. Il considère deux autres mélanges de Bernoulli (noté respectivement M_2 et M_3) que l'on propose d'étudier :

2.3. Etude du mélange M_2

Le mélange M_2 consiste à prendre des mélanges de Bernoulli dont le paramètre ε est remplacé par un vecteur ε formé de p valeurs ε^j dépendant de chaque variable. La vraisemblance classifiante associé à ce mélange s'écrit :

$$VC_2(P, L, \varepsilon) = \sum_{k=1}^{K} \sum_{x \in P_k} \sum_{j=1}^{p} \{ \left(\text{Log} \frac{\varepsilon^j}{1-\varepsilon^j}\right) |x^j - \lambda_k^j| + \text{Log}(1-\varepsilon^j) \} \qquad (3.2)$$

Maximiser le critère (3.2) revient à minimiser l'expression :

$$C_2(P, L, \varepsilon) = \sum_{k=1}^{K} \sum_{x \in P_k} d_\varepsilon(x, \lambda_k) - A \qquad (3.3)$$

où d_ε est une distance de type L_1 pondérée par les coefficients $\text{Log}\frac{1-\varepsilon^j}{\varepsilon^j}$ qui sont des quantités positives et A est la quantité $n. \sum_{j=1}^{p} \text{Log}(1-\varepsilon^j)$.

♦ ***fonction d'affectation (recherche des classes)***

Le second terme A est constant lors de cette étape. Puisqu'on cherche à maximiser le critère $VC_2(P, L, \varepsilon)$, on affectera les points aux"centres" les plus proches au sens d'une distance L_1 pondérée par les valeurs $\text{Log}\frac{1-\varepsilon^j}{\varepsilon^j}$.

♦ ***fonction de représentation (recherche des λ_k^j et des ε^j)***

Quelles que soient les valeurs ε^j obtenues, il est facile de voir que les λ_k^j sont nécessairement les valeurs majoritaires de chaque classe pour chaque variable. Il ne reste plus qu'à déterminer les ε^j. Il faut donc maximiser $VC_2(P, L, \varepsilon)$.

$$VC_2(P, L, \varepsilon) = \sum_{k=1}^{K} \sum_{x \in P_k} \sum_{j=1}^{p} \{ \left(\text{Log} \frac{\varepsilon^j}{1-\varepsilon^j}\right) |x^j - \lambda_k^j| + \text{Log}(1-\varepsilon^j) \}$$

$$= \sum_{j=1}^{p} \{ \left(\text{Log} \frac{\varepsilon^j}{1-\varepsilon^j}\right).e^j + n.\text{Log}(1-\varepsilon^j) \}$$

où $e^j = \sum_{k=1}^{K} \sum_{x \in P_k} |x^j - \lambda_k^j|$. On peut facilement vérifier que la valeur $\varepsilon^j = \frac{e^j}{n}$, correspond à un maximum qui appartient bien à l'intervalle $]0, \frac{1}{2}[$ sauf dans le cas très particulier où $e^j = 0$. e^j est le nombre de fois où la valeur majoritaire n'a pas été prise pour la variable j.

2.4. Etude du mélange M_3

Cette fois les paramètres ε_k^j vont dépendre à la fois des classes et des variables. Le critère que l'on cherche à minimiser se met alors sous la forme :

$$C_3(P, L, \varepsilon) = \sum_{k=1}^{K} \sum_{x \in P_k} \left\{ d_{\varepsilon_k}(x, \lambda_k) - A_k \right\} \qquad (3.4)$$

où d_{ε_k} est une distance de type L_1 pondérée par les quantités $\text{Log}\frac{1-\varepsilon_k^j}{\varepsilon_k^j}$ qui dépendent du vecteur ε_k des ε_k^j $(1 \leq j \leq p)$ et A_k est la quantité $\sum_{j=1}^{p} \text{Log}(1-\varepsilon_k^j)$ qui dépend de k.

♦ ***fonction d'affectation (recherche des classes)***

Cette fois le second terme dépend de k et aura donc une influence. On affectera x à la classe k qui minimise la quantité suivante :

$$d_{\varepsilon_k}(x, \lambda_k) - \sum_{j=1}^{p} \text{Log}(1 - \varepsilon_k^j)$$

REMARQUE 1

Comme dans le cas du modèle gaussien où, rappelons-le, la différence entre les distances adaptatives et l'approche modèle proposé par Anne Schroeder (1976) reposait sur l'existence d'un terme additif, on pourrait s'intéresser à l'annulation de ce terme pour obtenir un critère qui ne dépend que de la distance d.

Sachant que : $\sum_{j=1}^{p} \text{Log}(1 - \varepsilon_k^j) = \text{Log} \prod_{j=1}^{p} (1 - \varepsilon_k^j)$, si on impose aux coefficients de la distance la contrainte $\prod_{j=1}^{p} (1 - \varepsilon_k^j) = \text{Cte}$, on obtient alors une affectation qui dépend uniquement de la distance d_{ε_k}, et on se retrouve exactement dans la situation habituelle des Nuées Dynamiques. Pour déterminer les nouvelles valeurs des ε_k^j, on procède comme nous l'expliquons ci-dessous.

Il s'agit d'un problème classique d'optimisation sous contrainte. Le Lagrangien de ce problème s'écrit :

$$Lag = \sum_{k=1}^{K} \sum_{x \in P_k} \sum_{j=1}^{p} \{ Log \frac{1-\varepsilon_k^j}{\varepsilon_k^j} |x^j - \lambda_k^j| - Log(1-\varepsilon_k^j) \} - \mu \left(\prod_{j=1}^{p} (1 - \varepsilon_k^j) - Cte \right)$$

où μ est le multiplicateur de Lagrange.
La résolution de cette équation est équivalente à la résolution des p.K équations :

$$\frac{\partial Lag}{\partial \varepsilon_k^j} = 0 \quad \text{pour} \quad j = 1, ..., p \quad \text{et} \quad k = 1, ..., K.$$

$$\frac{\partial Lag}{\partial \varepsilon_k^j} = \frac{\partial}{\partial \varepsilon_k^j} \left[Log \frac{1-\varepsilon_k^j}{\varepsilon_k^j} . e_k^j - n_k \, Log(1-\varepsilon_k^j) \right] - \mu \left(\prod_{j=1}^{p} (1 - \varepsilon_k^j) - Cte \right) = 0$$

où $e_k^j = \sum_{x \in P_k} |x^j - \lambda_k^j|$.

On obtient $\varepsilon_k^j = \frac{e_k^j}{n_k + \mu . Cte}$. Si on applique la contrainte, on obtient :

$$\prod_{j=1}^{p} (1 - \varepsilon_k^j) = \prod_{j=1}^{p} \left(\frac{n_k + \mu . Cte - e_k^j}{n_k + \mu . Cte} \right) = Cte \qquad (3.5)$$

On peut remarquer qu'il est impossible de trouver par des méthodes analytiques une solution à l'équation (3.5). Seules des solutions obtenues par des méthodes numériques peuvent être proposées. ces dernières sortent du cadre de notre travail. Nous nous limiterons donc au cas où la fonction d'affectation dépend du second terme $\sum_{j=1}^{p} Log(1 - \varepsilon_k^j)$.

♦ ***<u>fonction de représentation</u> (recherche des λ_k^j et des ε_k^j)***

Comme dans les situations précédentes, quels que soient les valeurs ε_k^j, les meilleurs λ_k^j sont les médianes par classe et par variable. On peut alors montrer que :

$$\varepsilon_k^j = \frac{e_k^j}{n_k}.$$

où ϵ_k^j est le nombre de fois où la valeur majoritaire n'a pas été prise dans la classe k pour la variable j, et n_k le cardinal de la classe P_k. Le paramètre ϵ_k^j appartient bien à l'intervalle $]0, \frac{1}{2}[$ sauf dans le cas très particulier où $\epsilon_k^j = 0$.

Proposition :

Le système de pondération $\{\text{Log}\frac{1-\epsilon_k^j}{\epsilon_k^j} \quad j = 1, ..., p \text{ et } k = 1, ..., K\}$ correspondant au mélange M_3 favorise pour chaque classe, les variables déséquilibrées.

Preuve

Posons n_{1k}^j : le nombre de 1 se trouvant dans la classe P_k pour la variable j.

et n_{0k}^j : le nombre de 0 se trouvant dans la classe P_k pour la variable j.

Si j0 est l'indice correspondant à une variable déséquilibrée alors : $n_{1k}^{j0} << n_{0k}^{j0}$, et si j est l'indice d'une variable équilibrée alors $n_{1k}^j \approx n_{0k}^j$ (i.e proche de $\frac{n_k}{2}$). Les coefficients de pondération s'écrivent :

$$\alpha_k^{j0} = \text{Log}\frac{n_k - e_k^{j0}}{e_k^{j0}} \qquad \text{et} \qquad \alpha_k^j = \text{Log}\frac{n_k - e_k^j}{e_k^j}$$

comme $n_{1k}^{j0} << n_{0k}^{j0}$ et $n_{1k}^j \approx n_{0k}^j$ alors $e_k^{j0} < e_k^j$ d'où $\frac{e_k^j}{e_k^{j0}} > 1$ et $\frac{n_k - e_k^{j0}}{n_k - e_k^j} > 1$. Or $\frac{\exp \alpha_k^{j0}}{\exp \alpha_k^j} = \frac{e_k^j}{e_k^{j0}} \cdot \frac{n_k - e_k^{j0}}{n_k - e_k^j} > 1$ d'où $\exp \alpha_k^{j0} > \exp \alpha_k^j$

par conséquent $\alpha_k^{j0} > \alpha_k^j$ (c.q.f.d).

3. PROBLEMES DE DEGENERESCENCE

Nous venons de voir que les coefficients de pondérations obtenus sont tous exprimés en fonction de e^j (pour le mélange M_2) et ϵ_k^j (pour le mélange M_3) ; on a remarqué

que certains d'entre-eux ne sont pas définis pour les e^j nuls où e_k^j nuls. On peut donc s'interroger sur la validité des critères $VC_2(P, L, \varepsilon)$ et $VC_3(P, L, \varepsilon)$ dans ce cas de figure.

Soit :
$$\alpha^j = \text{Log}\frac{1-\varepsilon^j}{\varepsilon^j} = \text{Log}\frac{n-e^j}{e^j} \tag{3.7}$$

$$\alpha_k^j = \text{Log}\frac{1-\varepsilon_k^j}{\varepsilon_k^j} = \text{Log}\frac{n_k-e_k^j}{e_k^j} \tag{3.8}$$

i) Supposons que pour j_0, tous les éléments de chaque classe P_k (k = 1, ..., K) prennent la même valeur (0 ou 1), alors e_k^{j0} est nul pour tout k ce qui entraîne que e^{j0} est nul, et par conséquent la pondération α^{j0} n'est plus définie par la formule (3.7).

ii) Supposons que pour j_0, il existe $k_0 \in \{1, ..., K\}$ tel que tous les éléments de P_{k0} prennent la même valeur (0 ou 1). Alors e_{k0}^{j0} est nul et par conséquent la pondération α_{k0}^{j0} n'est pas définie par la formule (3.8).

Pour éviter ces problèmes de dégénérescence, on propose à chaque fois qu'on se retrouve dans la situation (i) ou (ii) de garder les anciennes valeurs de α^{j0} ou α_{k0}^{j0} (c'est-à-dire les valeurs qui correspondent à l'itération précédente) ; en procédant ainsi on fait augmenter les valeurs des critères mais on ne les optimise pas. On peut remarquer que le fait de prendre ces valeurs n'a aucune influence sur la convergence des critère $VC_2(P, L, \alpha(\varepsilon), \varepsilon)$ et $VC_3(P, L, \alpha(\varepsilon), \varepsilon)$ dont le calcul devient possible.

i) si $e^{j0} = 0$ alors $\varepsilon^{j0} = 0$ et α^{j0} non définie

$$C_2(P, L, \alpha(\varepsilon), \varepsilon) = \sum_{j=1}^{p} \{\alpha^j \cdot e^j - n.\text{Log}(1-\varepsilon^j)\}$$
$$= \sum_{j\neq j0} \{\alpha^j \cdot e^j - n.\text{Log}(1-\varepsilon^j)\} + \{(\alpha^{j0} \cdot e^{j0} - n.\text{Log}(1-\frac{e^{j0}}{n})\} \tag{3.9}$$

quelque soit la valeur que l'on donne à la pondération α^{j0}, la deuxième partie de l'expression (3.9) est toujours nulle, et le calcul du critère est possible.

ii) si $e_{k0}^{j0} = 0$ alors $\varepsilon_{k0}^{j0} = 0$ et α_{k0}^{j0} non définie.

$$C3(P, L, \alpha(\varepsilon), \varepsilon) = \sum_{j=1}^{p} \{ \alpha_k^j . e_k^j - n.\mathrm{Log}(1-\varepsilon_k^j) \}$$

$$= \sum_{j \neq j0} \{\alpha_k^j . e_k^j - n.\mathrm{Log}(1-\varepsilon_k^j)\} + \{(\alpha_{k0}^{j0} . e_{k0}^{j0} - n.\mathrm{Log}(1- \frac{e_{k0}^{j0}}{n_k})\} \quad (3.10)$$

quelque soit la valeur que l'on donne à la pondération α_{k0}^{j0}, la deuxième partie de l'expression (3.10) est toujours nulle, le calcul du critère est alors possible.

Nous venons de voir que l'identification d'un mélange de Bernoulli avec le même paramètre pour toutes les classes et toutes les variables correspond au critère de classification optimisé par la méthode MNDBIN. La généralisation de ce modèle en considérant différents paramètres permet de proposer un nouvel algorithme utilisant une distance adaptative que nous appelerons algorithme **MNDBIN adaptatif** qui n'est autre que l'ancien algorithme MNDBIN auquel s'ajoute deux variantes pour la distance ; la première consiste à pondérer la distance par des coefficients dépendants des variables (mélange M_2), la seconde par des coefficients dépendants des variables et des classes (mélange M3), mais à cette dernière distance s'ajoute le terme :
$- \sum_{j=1}^{p} \mathrm{Log}(1 - \varepsilon_k^j)$ qui dépend de k.

Au cours des différentes itérations de l'algorithme, la distance évolue en s'adaptant localement à la structure de l'espace dans lequel on travaille jusqu'à la convergence en faisant décroître un certain critère qui exprime bien cette adaptation. L'utilisation de ce type de distance améliore la partition et permet par conséquent de mettre l'ancien algorithme MNDBIN à défaut. L'extension de ce modèle pour les données binaires a permis aussi d'expliquer les bons résultats que nous avions obtenus à l'aide de cet algorithme sur des données simulées qui suivent à chaque fois une des variantes du modèle. On propose maintenant de comparer les résultats obtenus en appliquant successivement les trois variantes de l'algorithme sur des données réelles et des données simulées ; pour ces dernières on a réalisé un programme de simulation de données binaires que nous étudierons en détail dans le paragraphe 5.2.
Pour pouvoir comparer les différentes partitions obtenues à l'aide des trois variantes de l'algorithme MNDBIN adaptatif sur les même données, nous avons procédé de deux manières :

1/ Appliquer successivement les trois variantes de l'algorithme aux mêmes données en demandant à chaque fois 20 tirages ; l'algorithme retient donc le meilleur tirage pour chaque variante.

2/ Appliquer les variantes 2 et 3 de l'algorithme en les initialisant par le meilleur tirage obtenu par la variante 1. Cette deuxième méthode est beaucoup plus intéressante dans le sens où toutes les variantes sont initialisées par la même partition.

On s'intéresse par la suite et dans les deux cas aux variations des partitions, des noyaux, et des coefficients de pondérations, du critère et de l'écart.

4. APPLICATION ET COMPARAISON DES METHODES

4.1. DONNEES REELLES

4.1.1. Description des données

On a appliqué l'algorithme MNDBIN adaptatif sur les données nommées : " Information and store choice " traité par Goldstein et Dillon (1978). Il s'agit de 412 individus caractérisés par 4 variables binaires.
Les fréquences observées sont :

1111 : 91
1110 : 24
1100 : 15
1000 : 8
0000 : 23
1101 : 38
1011 : 6
1010 : 7
1001 : 7
0111 : 37
0100 : 17
0011 : 10
0010 : 9
0001 : 40
0110 : 14
0101 : 56

Pour ces données réelles, nous avons procédé de trois manières différentes que nous présentons.

4.1.2. Première stratégie

Le premier essai consiste à appliquer d'abord la variante 1 de l'algorithme MNDBIN adaptatif en demandant deux classes et 20 tirages, puis on applique successivement les variantes 2 et 3 en les initialisant par la meilleure partition obtenue par la variante 1 ; on procède ensuite à une comparaison des partitions des variantes 2 et 3 par rapport à la partition de la variante 1. On peut résumer les résultats obtenus par les tableaux de confusions ci-dessous :

Variante1-Variante2

	P_1	P_2
P'_1	208	0
P'_2	68	136

Variante1-Variante3

	P_1	P_2
P''_1	242	0
P''_2	34	136

où (P_1, P_2), (P'_1, P'_2), (P''_1, P''_2) sont respectivement les partitions obtenues par les variantes 1, 2 et 3 ; on peut voir dans cet essai que 68 éléments ont été classés différemment dans la variante 2 par rapport à la variante 1 ; de même 34 éléments mal rangés dans la variante 3 (toujours par rapport à la partition de la variante 1). Les valeurs des écarts entre le tableau réordonné et le tableau idéal obtenus pour chaque variantes sont respectivement 375, 375 et 351 ; on peut remarquer que les variantes 2 et 3 ont donnés des écarts plus faibles ou égaux à celui de la variante 1.
Les résultats détaillés de ce premier essai se trouvent en annexe 1.

4.1.3. Deuxième stratégie

Cette fois on applique la variante 2 de l'algorithme en demandant 20 tirages. La variante 3 est initialisée par la meilleure partition de la variante 2. On compare là aussi la partition de la variante 3 et celle de la variante 2, on obtient :

Variante2 - Variante3

	P'_1	P'_2
P''_1	208	19
P''_2	0	185

De la même manière que dans l'essai précédent , 19 éléments ont été classés différemment dans la variante 3 par rapport à la partition de la variante 2. L'écart entre le tableau réordonné et le tableau idéal étant de 375 pour la variante 2 et de 366 pour la variante 3. Les résultats détaillés de cet essai figurent en annexe 2

4.1.4. Troisième stratégie

Le troisième essai consiste à appliquer la variante 3 de l'algorithme en demandant 20 tirages. On a retenu le meilleur tirage possédant un écart de 375 (voir annexe 3 pour plus de détails sur cet essai)

Si maintenant on procède à une comparaison entre les meilleures partitions obtenues parmi 20 tirages pour chaque variante, on remarque que la variante 2 compte 68 éléments classés différemment par rapport à la variante 1, que la variante 3 compte 60 éléments classés différemment par rapport à la variante 1 et que la variante 3 compte 8 éléments classés différemment par rapport à la variante 2 ; cette comparaison est moins intéressante que celle faite ci-dessus. cela est dû au problème du choix de la partition initiale qui en général n'est pas le même pour les trois variantes ; l'écart entre le tableau réordonné et le tableau initial étant le même pour les trois variantes.

Nous remarquons tout de même, sur les quelques applications qui ont étés faites sur des données réelles ; que les variantes deux et trois arrivent toujours à modifier la partition obtenue par la variante 1. La comparaison des partitions obtenues dans les trois approches est délicate. Il est donc probable que l'algorithme MNDBIN adaptatif ait ainsi mis en évidence une structure difficile à découvrir. Pour cette raison nous proposons de nous appuyer sur des modèles probabilistes pour simuler des données sur lesquelles on à pu approfondir notre comparaison.

4.2. DONNEES SIMULEES

La simulation des données à été faite en utilisant les trois modèles probabilistes décrits précédemment, ces derniers nous ont permis d'écrire le programme suivant.

4.2.1. Le programme

Ce programme construit un tableau de données plus ou moins proches d'un tableau idéal structuré en colonne de 1 et 0. Pour cela il faut fournir le nombre de classes en lignes, le nombre d'éléments de chacune des classes, le tableau des valeurs idéales et

enfin la probabilité avec laquelle le tableau ainsi défini va s'approcher du tableau idéal. Nous proposons aussi trois variantes pour cet algorithme :

Variante 1 : La même probabilité de tirage ε pour tout les éléments du tableau.

Variante 2 : Probabilité différente pour chaque variable (colonne) du tableau ε^j ;
$j = 1, ..., p,$ où p est le nombre de variables.

Variante 3 : Probabilité différente pour chaque couple variable-classe : ε_k^j pour
$k = 1,..., K$ et $j = 1, ..., p,$ où K est le nombre de classes.

Chaque valeur du tableau est obtenue en faisant un tirage de Bernoulli des deux nombres 0 et 1 avec les probabilités ε et $(1- \varepsilon)$, si la valeur idéale est 1 ; $(1 - \varepsilon)$ et ε si la valeur idéale est 0 ($\varepsilon \in]0, \frac{1}{2}[$) ; à la fin du programme le tableau est permuté de manière aléatoire en ligne.

4.2.2. Les trois fichiers de données

Les paramètres ayant servi à la simulation de trois fichiers de données que l'on note par simul 1, simul 2 et simul 3 correspondant respectivement aux trois variantes citées ci-dessus se trouvent en annexe 4.

4.2.3. Résultats obtenus

Pour ces données simulées on procède pratiquement de la même manière que pour les données précédentes (données réelles) mais cette fois le premier essai consiste à appliquer les trois variantes de l'algorithme en demandant 20 tirages pour chaque variante. On compare ensuite ces trois meilleures partitions à la partition de la simulation. Le deuxième essai consiste à appliquer les variantes 2 et 3 de l'algorithme en les initialisant à la meilleure partition obtenue parmi 20 tirages de la variante 1 et on compare toujours ces trois partitions avec la partition de la simulation. Le dernier essai consiste à appliquer la variante 3 de l'algorithme en l'initialisant à la meilleure partition obtenue parmi 20 tirages de la variante 2, on compare aussi ces deux partitions avec la partition de la simulation, ensuite on sélectionne dans chaque essai la variante qui a fourni la partition qui se rapproche le plus de celle qui à servi à la simulation. Ces trois essais ont été faits sur les trois types de données simulées (Simul 1, simul 2 et simul 3)

4.2.3.1. Les données simul 1

On constate que pour les données simul 1, toutes les partitions obtenues par les différentes variantes et dans les trois essais sont toutes les mêmes et différent de deux éléments par rapport à la partition de la simulation ; il est clair que les variantes 2 et 3 de l'algorithme n'ont pas amélioré cette partition puisque les données ont été simulées suivant le modèle 1, la variante 1 de l'algorithme est censée donner la partition qui se rapproche le plus de la partition simulée.

4.2.3.2. Les données simul 2

Première stratégie

Dans cette première stratégie on applique les trois variantes de l'algorithme MNDBIN adaptatif en demandant 20 tirages , on retient ensuite la meilleure partition pour chaque variante ensuite on procède à une comparaison de ces partitions avec la partition ayant servi à la simulation, on obtient :

Variante 1 : W = 178
10 éléments ont étés classés différemment .
Les éléments 18, 95, 98 de la classe 3 sont dans la classe 1.
Les éléments 9, 39, 66 et 89 de la classe 3 sont dans la classe 2.
L'élément 34 de la classe 2 est dans la classe 1.
L'élément 70 de la classe 1 est dans la classe 3.
L'élément 71 de la classe 2 est dans la classe 3.

Variante 2 : W=399.61
3 éléments ont été classés différemment
L'élément 66 de la classe 3 est dans la classe 2.
L'élément 96 de la classe 3 est dans la classe 1.
L'élément 51 de la classe 1 est dans la classe 3.

Variante 3 : W=404.55
4 éléments classés différemment.
Les éléments 18 et 98 de la classe 3 sont dans la classe 1.
L'élément 66 de la classe 3 est dans la classe 2.
L'élément 6 de la classe 1 est dans la classe 3.

On peut voir ici que la variante 2 a donné la partition qui se rapproche le plus de celle qui à servi à la simulation.

Deuxième stratégie

On initialise cette fois-ci les variantes 2 et 3 de l'algorithme à la meilleure partition de la variante 1 ; on compare toujours les partitions obtenues avec celle de la simulation.

Variante 2 : W=404.15
3 éléments classés différemment.
L'élément 66 de la classe 3 est dans la classe 2.
Les éléments 18 et 98 de la classe 3 sont dans la classe 1.

Variante 3 : W=409.00
Dans cette variante on à obtenu la même partition que celle de la variante 2. Les données sont simulées suivant le modèle 2, la variante 3 n'a pas amélioré cette partition.

Troisième stratégie

On initialise maintenant la variante 3 de l'algorithme à la meilleure partition obtenue parmi 20 tirages de la variante 2. On obtient :

Variante 3 : W=444.99
4 éléments classés différemment.

Conclusion

On peut remarquer que la partition qui se rapproche le plus de celle qui à servi à la simulation est obtenue par la variante 2 de l'algorithme. Ceci vient du fait que les données simulées suivent aussi le modèle 2. On peut voir ici déjà l'avantage que présente l'algorithme MNDBIN adaptatif sur les données simulées. Il permet en effet de reconnaître la nature des données (données simulées de type M_1, de type M_2 ou encore de type M_3).
Dans le dernier exemple qui reste à étudier (celui qui correspond au modèle 3) on verra encore mieux l'amélioration des partitions à partir de la variante 1 jusqu'à ce qu'on arrive à la variante 3 qui donne la partition la plus proche de celle de la simulation.

4.2.3.3. Les données simul 3

Première stratégie

On demande 20 tirages pour chaque variante et l'on retient la meilleure partition pour chacune d'entre elle ; on procède ensuite à la comparaison avec la partition simulée.

Variante 1 : W=165

8 éléments classés différemment.

Les éléments 9, 38, 39, 46, 48, 58, 93, 100 de la classe 2 sont dans la classe 3.

Variante 2 : W=417.03

6 éléments classés différemment

Les éléments 9, 38, 39, 46, 58 et 93 de la classe 2 sont dans la classe 3

Variante 3 : W=360.47

1 seul élément à été rangé différemment.

L'élément 94 de la classe 3 est dans la classe 2.

On peut voir là aussi la décroissance du nombre d'éléments classés différemment ce qui indique le rapprochement des partitions à la partition de la simulation.

Deuxième stratégie

Les variantes 2 et 3 sont initialisées à la meilleure partition de la variante 1:

Variante 2 : W=417.03

On a obtenu la même partition que celle de la variante 2 dans l'essai 1.

Variante 3 : W = 386.83

4 éléments mal rangés .

Les éléments 17, 48, 58, 100 de la classe 2 sont dans la classe 3.

Troisième stratégie

La variante 3 est initialisée à la meilleure partition de la variante 2.

Variante 3 : W=401.18

On obtient exactement la même partition que celle de la simulation.

Nous allons généraliser les résultats obtenus sur les trois fichiers de données simul 1, simul 2 et simul 3 sur d'autres données simulées. Pour cela nous avons simulé 20 fichiers de données de chaque variante du programme de simulation. Nous avons ensuite appliqué la méthode MNDBIN adaptatif en procédant de la même manière que pour les données simul 1, simul 2 et simul 3.

Les tableaux ci-dessous résument l'ensemble des résultats obtenus. En colonne nous avons les données M_1, M_2, M_3 qui suivent respectivement les modèles 1, 2 et 3 ; en

ligne les trois variantes de l'algorithme MNDBIN adaptatif (V_1, V_2, V_3). Les valeurs des tableaux (situés à gauche) indiquent la moyenne des 20 valeurs qui sont pour chaque fichier le nombre d'éléments classés différemment par rapport à la partition ayant servi à la simulation ; les valeurs des tableaux situés à droite indiquent l'écart - type de ces mêmes valeurs.

1/ On demande 20 tirages pour chaque variante :

	M_1	M_2	M_3		M_1	M_2	M_3
V_1	2.3	12.1	8.9	V_1	0.9	1.7	1.5
V_2	3.1	3.4	5.7	V_2	1.4	1.3	1.2
V_3	1.9	4.2	1.1	V_3	0.9	1.7	1.1

2/ Les variantes 2 et 3 sont initialisées à la meilleure partition de la variante 1 :

	M_1	M_2	M_3		M_1	M_2	M_3
V_1	2.3	12.1	8.9	V_1	0.9	1.9	1.5
V_2	2.5	3.2	5.7	V_2	0.8	1.6	1.5
V_3	1.9	3.2	4.0	V_3	0.1	1.9	1.9

3/ La variante 3 est initialisée à la meilleure partition de la variante 2:

	M_1	M_2	M_3		M_1	M_2	M_3
V_2	2.2	3.4	5.7	V_2	0.8	1.3	1.2
V_3	2.2	3.3	0.2	V_3	0.6	1.0	0.3

5. CONCLUSION

On remarque que sur les deux types des données utilisées, l'algorithme MNDBIN adaptatif modifie plus ou moins la partition obtenue par la variante 1 de l'algorithme (ancien MNDBIN), les valeurs des différents coefficients de pondérations sont raisonnables pour les deux cas de pondérations (variable, variable-classe), ; en revanche l'algorithme MNDBIN adaptatif présente un grand avantage pour les données simulées, car chaque variante de l'algorithme permet de donner la partition la plus proche de celle qui a servi à la simulation du tableau de données suivant le modèle de la variante en question. Sur plusieurs essais qui ont été faits sur des données simulées, on a constaté ce qui suit.

En appliquant les trois variantes de l'algorithme sur des données simulées suivant le modèle 1, les trois partitions sont en général identiques ; sur des données simulées suivant le modèle 2, la variante 2 nous donne le tirage le plus proche de celui qui à servi à la simulation ; par contre sur les données simulées suivant le modèle 3 , la variante 1 fournit une certaine partition, la variante 2 l'améliore et la variante 3 nous donne la partition qui se rapproche le plus de celle qui a servi à la simulation.

L'algorithme MNDBIN adaptatif permet donc d'améliorer les résultats obtenus par l'ancien MNDBIN (variante 1 de l'algorithme MNDBIN adaptatif) dans lequel on ne tient pas compte des pondérations, c'est à dire qu'on donne à toutes les variables et les variables-classes le même poids, ce qui statistiquement peut être vu comme un défaut.

PARTIE B

CLASSIFICATION CROISEE

CHAPITRE 4

CLASSIFICATION CROISEE ET MELANGES

INTRODUCTION

Jusqu'à présent les liens qui existent entre les méthodes de classification automatique et les modèles de statistiques inférentielles ont surtout été étudiés lorsque les données mettent en jeu un seul ensemble (Scott et Symons 1971, Schroeder 1974, Celeux 1988, Govaert 1988). Nous proposons ici de le faire lorsque les données mettent en jeu deux ensembles. Nous étudierons alors les liens qui existent entre les méthodes de **classification croisée** et les **modèles de statistiques inférentielles** en proposant une méthode de reconnaissance des composants d'un mélange dans le cas de la classification croisée.

Nous rappelons dans le premier paragraphe, le principe général de la classification croisée et l'un des algorithmes qui lui est associé (Govaert 1983). Dans le second paragraphe nous introduisons la notion de mélange "croisé" et nous posons le problème de ces mélanges. Nous essayons dans le paragraphe trois de résoudre ce problème par l'approche classification, nous remarquons alors que la résolution de ce problème correspond exactement à un problème de classification croisée. Nous nous intéressons dans le paragraphe quatre à la transformation d'un problème de mélange "croisé" en un problème de mélange simple ; nous remarquons que ce passage correspond exactement à celui qui permet de transformer un problème de classification croisée en un problème de classification simple qui a été étudié par Govaert (1983). Cette analogie entre ces deux transformations s'exprime par le fait que la connaissance d'une partition de l'un des deux ensembles, pour le problème de la classification correspond à la connaissance de tout les composants de l'un des deux échantillons pour le problème des mélanges.
Nous terminons ce chapitre par une application des résultats obtenus dans le paragraphe quatre sur deux types de fonctions de densités. Nous montrons alors que lorsque le nombre de composants du mélange d'un échantillon coincide avec la taille

de l'échantillon, on retrouve exactement le modèle de la classification simple (Celeux 1988 et Govaert 1988).

1. LA CLASSIFICATION CROISEE

1.1. RAPPELS ET NOTATIONS

Les données sont toujours fournies sous la forme d'un tableau rectangulaire X de dimension (n, p) où n est le nombre d'individus et p est le nombre de variables décrivant les n individus.
Les deux ensembles I et J sont respectivement l'ensemble des individus et l'ensemble des variables, la valeur x_i^j se trouvant à l'intersection de la ligne i avec la colonne j représente la valeur prise par l'individu i pour la variable j.

Soient :
- I, un sous ensemble fini de $\mathbf{R}^p$, contenant n éléments.
- J, un sous ensemble fini de $\mathbf{R}^n$, contenant p éléments.
- P_k, l'ensemble des partitions de I en K classes, les éléments de P_k seront appelés K-partitions et notés $P = (P_1, ..., P_K)$.
- Q^m, l'ensemble des partitions de J en M classes, les éléments de Q^m seront appelés M-partitions et notés $Q = (Q^1, ..., Q^M)$.
- L, l'ensemble des noyaux, ces noyaux seront associés aux partitions des deux ensembles I et J ; $L = \{(\lambda_k^m), k = 1, ..., K \text{ et } m = 1, ..., M)\}$.

1.2. LE PRINCIPE DE LA CLASSIFICATION CROISEE

La position générale du problème de la classification croisée consiste à subdiviser la population des individus et la population des variables en un petit nombre de groupes ou classes homogènes dans un certain sens. Nous ne ferons pas de distinction entre les objets et les variables à classer en raison de l'identité de la position des problèmes et de la principale méthode d'étude ; le problème à résoudre en classification croisée est alors le suivant : essayer de trouver une partition P de l'ensemble I des individus en K classes et une partition Q de l'ensemble J en M classes, tels qu'on réordonnant les lignes et les colonnes suivant les deux partitions, on obtient des classes homogènes.
La recherche de ce meilleur couple de partitions (P, Q) correspond à l'optimisation d'un certain critère noté W(PxQ, L).
Lorsque le tableau envisagé est un tableau de contingence, un tableau de variables qualitatives ou encore un tableau de variables quantitatives, la méthode de classification croisée repose sur la notion de mesure d'information associée à un

tableau. Cette mesure peut être le Khi2 de contingence, dans le cas de tableau de contingence, ou l'inertie, dans le cas de données quantitatives ; la méthode fournit un tableau résumant le tableau initial et maximisant cette mesure d'information. Elle procède de façon itérative en recherchant, à chaque étape, une classification sur l'ensemble I ou l'ensemble J. Dans les deux cas l'algorithme intermédiaire est alors le même et il maximise, à chaque fois, la mesure d'information.

1.3. L'ALGORITHME

Plusieurs algorithmes de classification croisée existent, on a retenu l'algorithme suivant développé par Govaert (1983) ; celui-ci utilise deux algorithmes voisins l'un de l'autre et tout deux basés sur le principe des Nuées Dynamiques.
Cet algorithme se déroule en trois étapes :

(0) Partition initiale P^0 et Q^0 (quelconque).
$n = 0$
(1) On associe le noyau L^n minimisant $W(P^n x Q^n, L^n)$.
(2) Q^n et L^n fixés, P^{n+1} est la partition minimisant $W(P^{n+1} x Q^n, L^n)$
(3) P^{n+1} et L^n fixés, Q^{n+1} est la partition minimisant $W(P^{n+1} x Q^{n+1}, L^n)$
si $P^n \equiv P^{n+1}$ et $Q^n \equiv Q^{n+1}$ alors fin.
sinon $n = n+1$ et aller en 1.

Le principe général de cet algorithme est le suivant : a partir d'une partition initiale $(P^o x Q^o)$ en K.M classes, on construit une suite de partitions en appliquant successivement les trois fonctions suivantes :

La fonction de représentation g :

Cette fonction permet de déterminer les K.M noyaux minimisant le critère W(PxQ), g(PxQ)).

$$g(PxQ) = L = \{(\lambda_k^m),\ k = 1, ..., K \text{ et } m = 1, ..., M)\}.$$

La fonction d'affectation f

Cette fonction permet de déterminer, à Q et L fixés une partition P de l'ensemble I améliorant le critère W(f(PxQ), L), en affectant chaque individu $i \in I$ à la classe P_k du noyau de laquelle il est le plus proche.

La fonction d'affectation h

Cette fonction permet de déterminer, à P et L fixés, une partition Q de l'ensemble J améliorant le critère W(h(PxQ), L), en affectant chaque individu j∈ J à la classe Q^m du noyau de laquelle il est le plus proche.

La définition de ces trois fonctions entraîne que la suite $W(P^n x Q^n, L^n)$ est décroissante ; on retrouve les propriétés habituelles de convergence des Nuées Dynamiques. Nous développons cet algorithme dans les chapitres cinq et six où son application est faite sur deux tableaux bien particuliers. On verra alors mieux le rôle des trois fonctions g de représentation, f et h d'affectations.

2. MODELE DE MELANGE "CROISE"

2.1. EXEMPLE ILLUSTRATIF

Nous désignons par $P = \{p_1, ..., p_n\}$ un ensemble de n produits que peut fabriquer un système, lequel est composé d'un ensemble $M = \{m_1, ..., m_P\}$ de p machines ; nous supposons que les échantillons P et M sont tous les deux formés respectivement de R et S sous-échantillons (appelés respectivement famille de produits et de machines).
Soit h une fonction permettant d'associer à chaque couple $(p_i, m_j) \in P x M$ une valeur $h(p_i, m_j)$ qui mesure la "ressemblance" entre les deux objets p_i et m_j. Par exemple si $h(p_i, m_j) \in R$, $h(p_i, m_j)$ peut indiquer le nombre d'heures passé par la machine m_j pour la fabrication du produit p_i. Si maintenant $h(p_i, m_j) \in \{0, 1\}$, la fonction h peut associer au couple (p_i, m_j) la valeur 1 si le produit p_i est fabriqué par la machine m_j, la valeur 0 sinon.
La fonction h est définie sur l'ensemble P x M qui est le produit cartésien des deux ensembles P et M. On remarque que l'ensemble P x M est lui aussi formé de R.S sous-échantillons (appelés famille de produits-machines).
Supposons maintenant qu'un produit p_i soit issu d'une certaine famille r (noté CPr), de même, on suppose connu la famille s à laquelle la machine m_j est issue (on note cette famille par CMs), la fonction h définie précédemment suit alors la densité de probabilité $f(h / \lambda_r^s)$ où λ_r^s est le paramètre caractérisant la densité de probabilité de la

$(r.s)^{ème}$ famille du mélange auquel le couple (p_i, m_j) appartient. Le modèle correspondant à cet exemple s'écrit alors :

$$f(h) = \sum_{r=1}^{R} \sum_{s=1}^{S} p_r^s \, f(h / \lambda_r^s) \qquad (4.1)$$

$f(h / \lambda_r^s)$: représente la densité de la $(r.s)^{ème}$ famille du mélange.

$f(h)$: représente la loi de probabilité résultante.
p_r^s : probabilité d'appartenance à priori à chacune des familles.

Si on reprend l'exemple précédent où h $(p_i, m_j) \in \mathbf{R}$, $f(h / \lambda_r^s)$ peut représenter la densité d'une loi gaussienne unidimensionnelle, le paramètre λ_r^s s'écrit :

$$\lambda_r^s = (\mu_r^s , \sigma_r^s)$$

où μ_r^s : espérance du composant (r. s)

σ_r^s : l'écart-type du composant (r.s).

Posons $V_r^s = (\sigma_r^s)^2$: variance du composant (r. s).

$$f(h / \lambda_r^s) = (2\pi)^{-1/2}. (\sigma_r^s)^{-1}. \exp - \frac{1}{2} \left[\frac{(h - \mu_r^s)}{\sigma_r^s} \right]^2$$

Le modèle (4.1) s'écrit alors :

$$f(h) = \sum_{r=1}^{R} \sum_{s=1}^{S} p_r^s . (2\pi v_r^s)^{-1/2}. \exp\left\{ - \frac{(h - \mu_r^s)^2}{2.V_r^s} \right\} \qquad (4.2)$$

Il s'agit de résoudre un problème classique d'estimation des paramètres. Nous avons retenu la méthode d'estimation du maximum de vraisemblance qui permet d'estimer les nombres R et S de composants du mélange et des paramètres inconnus ($q_r^s = (p_r^s , (\mu_r^s , V_r^s))$; r = 1, ..., R et s = 1, ..., S) au vu de l'échantillon PxM.

Après résolution du problème on obtient :

$$\mu_r^s = \frac{1}{n_r.q_s} \sum_{p_i \in CPr} \sum_{m_j \in CMs} h(p_i, m_j)$$

$$V_r^s = \frac{1}{n_r.q_s} \sum_{p_i \in CPr} \sum_{m_j \in CMs} (h(p_i, m_j) - \mu_r^s)^2$$

$$p_r^s = \frac{n_r.q_s}{n.p}$$

μ_r^s est la moyenne de la distribution $f(h / \lambda_r^s)$, σ_r^s représente sa variance.

p_r^s = représente la probabilité d'apparition du composant r.s dans le mélange.

n_r : représente le nombre de produits constituant la famille CMr.
n_s : représente le nombre de machines constituant la famille CMs.

2.2. MODELE GENERAL

L'exemple que nous venons de voir peut nous conduire à considérer que les tableaux de données habituels (contingence, binaire, qualitatif, quantitatif) qui sont tous décrits par deux ensembles que l'on note par **I** et **J** peuvent être replacés dans le même cadre que celui défini pour l'exemple, c'est-à-dire que l'on peut toujours définir une variable aléatoire **Z** qui permet d'associer à chaque couple $(i, j) \in IxJ$ la valeur se trouvant à l'intersection de la ligne i avec la colonne j.

Dans la suite de ce travail, on suppose que l'ensemble **I** constitue un échantillon de taille n d'une population Ω, de même on suppose que l'ensemble **J** constitue un échantillon de taille p d'une population Ω'. Soit $T = IxJ$ le produit cartésien des deux ensembles **I** et **J** ; l'ensemble **T** peut être considéré comme un échantillon de taille (n, p) d'une population $\Omega x\Omega'$.

2.2.1. IDENTIFICATION D'UN MELANGE "CROISE"

Le tableau de données de départ de dimension (n, p) est considéré comme un échantillon $T = IxJ$ de taille (n, p) d'une variable aléatoire à valeur dans **R** dont la loi de probabilité admet la fonction de densité :

$$f(x) = \sum_{k=1}^{K} \sum_{m=1}^{M} p_k^m \, f(x / \lambda_k^m) \qquad (4.3)$$

$$\forall x \in \mathbf{R} \quad \forall k = 1, ..., K \text{ et } m = 1, ..., M \quad 0 \le p_k^m \le 1 \text{ et } \sum_{k=1}^{K} \sum_{m=1}^{M} p_k^m = 1.$$

L'espace probabilisé sur lequel la variable aléatoire **Z** est définie, est de la forme $(IxJ, P(IxJ), p)$, la probabilité p définie sur l'espace probabilisé $P(IxJ)$ est déterminée par la donnée des nombres $\{p_k^m\}$ pour $k = 1, ..., K$ et $m = 1, ..., M$.

La formule (4.3) décrit le modèle d'un mélange de type donné $f(., \lambda_k^m)$ qui est une fonction de densité sur **R** appartenant à une famille paramétrée de fonctions de densité dépendant du paramètre λ, et p_k^m est la probabilité d'apparition de l'observation $f(., \lambda_k^m)$ dans le mélange.

2.2.2. PROBLEME A RESOUDRE

Problème 1

Le problème consiste à estimer les nombres K et M de composants du mélange et les paramètres inconnus ($a_k^m = (p_k^m , \lambda_k^m)$; $k = 1, ..., K$ *et* $m = 1, ..., M$) *au vu de l'échantillon* $T = IxJ$.

Il s'agit d'un problème d'estimation de paramètres. Nous ne traitons pas par l'approche "estimation" du problème, nous nous concentrerons sur l'approche "classification" pour l'identification d'un mélange "croisé".

3. APPROCHE CLASSIFICATION

Dans cette approche on remplace le problème 1 d'estimation par le problème 2 suivant :

Problème 2

Rechercher une partition $PxQ = \{ P_kxQ^m \; k=1,..., K \text{ et } m=1, ..., M \}$, *K et M étant supposés connus, telle que chaque classe* P_kxQ^m *soit assimilable à un sous-échantillon qui suit une loi* $f(., \lambda_k^m)$.

En suivant l'approche modèle proposée par Schroeder (1974) et la représentation de Celeux (1988) qui transforme le problème d'optimisation de **critère de vraisemblance** en un problème d'optimisation de **critère de vraisemblance classifiante**, on se ramène également, dans le cas de mélange "croisé", à la maximisation du critère de vraisemblance classifiante suivant :

$$VC(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \text{Log } L(P_kxQ^m, \lambda_k^m) \qquad (4.4)$$

où L est le K.M-uple (λ_k^m , k=1, ..., K et m=1, ..., M) et $L(P_kxQ^m, \lambda_k^m)$ la vraisemblance du sous-échantillon P_kxQ^m qui suit la loi $f(., \lambda_k^m)$.

On peut alors écrire :

$$L(P_kxQ^m, \lambda_k^m) = \prod_{x \in P_kxQ^m} f(x/ \lambda_k^m) \qquad (4.5)$$

Enfin le critère de **vraisemblance classifiante** s'écrit :

$$VC(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \text{Log } f(x_i^j / \lambda_k^m) \qquad (4.6)$$

On peut remarquer que la résolution du problème (4.6) correspond exactement à la résolution d'un problème de classification croisée (Govaert 1983) rappelé au début de ce chapitre. Nous proposons d'utiliser le même algorithme que celui défini pour la classification croisée et rappelé dans le paragraphe 1.3. Cet algorithme, que nous allons reprendre pour l'adapter à notre problème, utilise deux algorithmes voisins l'un de l'autre, et tout deux basés sur le principe des Nuées Dynamiques.

3.1. ALGORITHME

Le principe de cet algorithme est le suivant.
En partant de deux nombres K et M et d'une partition initiale $(PxQ)^0$ en K.M classes, l'algorithme construit une suite de partitions jusqu'à l'obtention d'une partition stable en appliquant successivement les trois fonctions suivantes.

Une fonction de représentation g définie comme suit :

Cette fonction permet de déterminer les K.M noyaux minimisant le critère VC(PxQ, g(PxQ)). On peut facilement voir que ces noyaux sont les estimateurs du maximum de vraisemblance des paramètres associés aux sous échantillons $\{P_k x Q^m\}$; k =1,..., K et m=1, ..., M).
$g(PxQ)=g(\{P_k x Q^m\}$; k =1,..., K et m=1, ..., M $)=(\lambda_k^m$; k =1,..., K et m=1, ..., M) =L.

Une fonction d'affectation f définie comme suit :

Cette fonction permet de déterminer, à Q et L fixés, une partition P de l'échantillon I améliorant le critère VC(PxQ, L). Le critère (4.6) s'écrit alors :

$$VC(PxQ, L) = \sum_{k=1}^{K} \sum_{i \in P_k} \text{Log } F(x_i / \lambda_k) \qquad (4.7)$$

$$\text{où } F(x/\lambda_k) = \prod_{m=1}^{M} \Big(\prod_{j \in Q^m} f(x^j, \lambda_k^m) \Big) \qquad (4.8)$$

On retrouve ainsi la forme du critère de vraisemblance classifiante dans le cas de la classification simple. Les éléments de chaque classe P_k seront définis comme suit :

$$P_k = \{ i \in I \ / F(x_i / \lambda_k) \geq F(x_i / \lambda_{k'}) \text{ avec } k<k' \text{ en cas d'égalité } \}$$

Une fonction d'affectation h définie comme suit :

Cette fonction permet de déterminer, à P et L fixés, une partition Q de l'échantillon **J** améliorant le critère VC(PxQ, L). De façon symétrique, le critère de vraisemblance classifiante s'écrit :

$$VC(PxQ, L) = \sum_{m=1}^{M} \sum_{j \in Q^m} \text{Log } F(x^j/\lambda^m) \qquad (4.9)$$

$$\text{où } F(x^j/\lambda^m) = \prod_{k=1}^{K} \left(\prod_{i \in P_k} f(x_i^j, \lambda_k^m) \right) \qquad (4.10)$$

Les éléments de la partition Q de l'échantillon **J** seront déterminés comme suit :
$Q^m = \{ j \in \mathbf{J} / F(x^j/\lambda^m) \geq F(x^j/\lambda^{m'})$ avec $m<m'$ en cas d'égalité$\}$.

On peut montrer que sous certaines hypothèses, cet algorithme est convergent. On obtient à la convergence une partition PxQ et une estimation des paramètres λ_k^m. Les proportions p_k^m du mélange sont fournies par les fréquences des classes $P_k x Q^m$.

Nous proposons maintenant de développer sur l'échantillon $\mathbf{T} = \mathbf{I}x\mathbf{J}$, l'approche mélange simple et de voir sous quelles conditions cette approche et l'approche mélange "croisé" coincident.

3.2. POSITION INTERMEDIAIRE

Nous supposons toujours que l'échantillon $\mathbf{T} = \mathbf{I}x\mathbf{J}$ est le produit cartésien des deux ensembles **I** et **J** de tailles respectives n et p. L'échantillon **T** de taille (n, p) provient d'une variable aléatoire à valeurs dans **R**, dont la loi de probabilité admet cette fois-ci la fonction de densité suivante :

$$f(x) = \sum_{n=1}^{N} p_n f(x/\lambda_n)$$

$$\text{avec} \quad \forall n = 1,N \quad p_n \in]0,1[\quad \text{et} \quad \sum_{n=1}^{N} p_n = 1$$

où $f(./\lambda)$ appartient à une famille de fonctions de densité dépendant du paramètre λ, élément de **R**, et p_n est la probabilité qu'un point de l'échantillon suive la loi $f(./\lambda_n)$. On appellera ces p_n les proportions du mélange.

Problème 1'

Le problème posé est l'estimation du nombre N de composants du mélange et des paramètres inconnus $\{p_n, \lambda_n / n = 1, N\}$ *au vu de l'échantillon.*

Dans l'approche classification (Scott et Symons 1971, Schroeder 1974), on remplace le problème 1' d'estimation par le problème 2' suivant :

Problème 2'

Rechercher une partition $R = (R_1,..., R_N)$, *N étant supposé connu, telle que chaque classe* R_n *soit assimilable à un sous-échantillon qui suit une loi* $f(., \lambda_n)$.

Il s'agit alors de maximiser le critère de **vraisemblance classifiante :**

$$VC(R, L) = \sum_{n=1}^{N} \text{Log } L(R_n/\lambda_n)$$

où L est le N-uplet $(\lambda_1,..., \lambda_N)$ et $L(R_n, \lambda_n)$ est la vraisemblance du sous-échantillon R_n suivant la loi $f(./\lambda_n)$: $L(R_n, \lambda_n) = \prod_{x \in R_n} f(x/\lambda_n)$.

Pour maximiser ce critère, on utilise l'algorithme de type Nuées Dynamiques qui construit à partir d'une partition R^o en N classes une suite de partitions en appliquant les fonctions f et g décrites aux paragraphes 2.2.2 et 2.2.3. du chapitre 1. Nous remarquons qu'en utilisant cet algorithme, on obtient une partition R de l'échantillon T = IxJ en N classes, cette partition ne correspond à aucune partition sur l'ensemble I ni sur l'ensemble J. Ce problème ne correspond pas à un problème de classification croisée, là est la différence entre le modèle de mélange "croisé" proposé dans le paragraphe 2 et le modèle de mélange simple que l'on vient de développer. Ces deux modèles sont définis sur le même ensemble T formé de couples d'éléments. La différence essentielle entre ces deux approches vient du fait que si l'on a une partition P de l'ensemble I et une partition Q de l'ensemble J, on à automatiquement une partition PxQ de l'ensemble T = IxJ, mais la réciproque n'est pas vraie. Pour cette dernière raison on est amené à poser une condition sur la recherche de la partition R en N classes posé dans le problème 2'. Cette condition s'exprime par la recherche séparée de deux partitions P et Q correspondant respectivement aux deux ensembles de départ I et J en K et M classes.
On peut alors écrire R = PxQ et N = K.M. Le problème 2' est donc remplacé par le problème 2. Ainsi nous nous retrouvons dans le cas de l'identification d'un mélange "croisé", et nous pouvons utiliser le même algorithme que celui développé dans le paragraphe 3.1.

Nous venons de voir, comment la méthode de classification croisée (Govaert 1983), peut être vu comme une solution à un problème d'estimation de paramètres d'un modèle de mélange "croisé". Ce problème est à rapprocher de celui de la méthode des Nuées Dynamique (Diday 1972) qui a été utilisée par Schroeder (1974) pour proposer une solution à un problème d'estimation de paramètres d'un modèle de mélange simple. Mais il serait intéressant dans la suite de ce travail d'établir s'il y en des liens entre le problème de mélange "croisé" et le problème de mélange simple ; nous savons que Govaert (1983), s'est intéressé aux rapports qui existent entre les méthodes de classification croisée et les méthodes de classification simple, il a remarqué que la comparaison des partitions obtenues par ces deux types de méthodes était très délicate. Néanmoins l'algorithme qu'il a proposé lui permettait de passer d'un problème de classification croisée à un problème de classification simple. Nous allons essayer de suivre le même chemin que celui de Govaert (1983), pour transformer le problème de mélange "croisé" en un problème de mélange simple. Nous verrons alors comment s'interprète le passage d'un problème de classification croisée à un problème de classification simple en termes de modèle probabiliste.

4. TRANSFORMATION DU MODELE DE MELANGE "CROISE" EN UN MODELE DE MELANGE SIMPLE

Pour éviter d'introduire de nouvelles notations pour indiquer les composants du mélange **IxJ**, qui risquerait de rendre difficile la compréhension de ce paragraphe, nous avons tenu de garder la notation $\{P_k x Q^m\}$ pour indiquer le composant du mélange ayant pour fonction de densité la loi $f(. / \lambda_k^m)$. Rappelons que cette notation à

été utilisée dans le paragraphe trois pour désigner une classe. Dans ce paragraphe, l'écriture $\{P_k x Q^m\}$indique plutôt le composant k.m du mélange **IxJ**. De même P_k et Q^m indiquent respectivement les composants k du mélange **I** et m du mélange **J**.

Rappelons rapidement le modèle associé à un mélange "croisé", **Z** étant une variable aléatoire définie sur le produit cartésien **I x J**, dont la loi de probabilité f admet la fonction de densité suivante :

$$f(x) = \sum_{k=1}^{K} \sum_{m=1}^{M} p_k^m \; f(x / \lambda_k^m)$$

ou encore $\forall \; (i, j) \in IxJ$

$$f(\, z(i, j)) = \sum_{k=1}^{K} \sum_{m=1}^{M} p_k^m \; f(\, z(i, j) / \lambda_k^m) \qquad (4.11)$$

La valeur $Z(i, j)$ ne peut provenir que de l'un des K.M composants suivants $\{P_k x Q^m)$ pour $k = 1, ..., K$ et $m = 1, ..., M$; les p_k^m sont les proportions du mélange :

$$p_k^m = \frac{n_k . q_m}{n . p} = p\,(P_k x Q^m).$$

n_k : représente le nombre d'éléments $i \in I$ constituant le composant P_k.
q_m : représente le nombre d'éléments $j \in J$ constituant le composant Q^m.

i) Supposons que l'on connaisse le composant $Q^{m(j)}$ auquel un élément $j \in J$ appartient, que devient alors le modèle donné par la formule (4.11) ?

Dans ce cas la valeur $Z(i, j)$ sachant que $j \in Q^{m(j)}$ ne peut provenir que de l'un des K composants suivants $\{P_k x Q^{m(j)}\}_{k=1, ..., K}$. Le modèle (4.11) s'écrit alors :

$$f(z(i, j) / Q^{m(j)}) = \sum_{k=1}^{K} p_k^{m(j)}\, f(z(i, j) / \lambda_k^{m(j)}) \qquad (4.12)$$

avec $p_k^{m(j)} = \frac{n_k . q_{m(j)}}{n . q_{m(j)}} = \frac{n_k}{n} = p_k$

ii) On suppose maintenant que l'on connait tout l'échantillon J de Ω' de taille p. Le composant dont chaque élément $j \in J$ est issu est connu. Notons par $Q^{m(j)}$ ce composant. On peut définir une variable aléatoire ξ de dimension p telle que :

$$\xi : I \longrightarrow R^p$$
$$i \longrightarrow \xi(i) = (\xi^1(i), \ldots, \xi^p(i)) = (x_i^1, \ldots, x_i^p) = x_i$$

On peut alors écrire $(\xi^1(i), \ldots, \xi^p(i)) = (Z(i, 1), Z(i, 2), ..., Z(i, p))$.

Comme $f(z(i, j) / Q^{m(j)}) = \sum_{k=1}^{K} p_k\, f(z(i, j) / \lambda_k^{m(j)})$ $\quad \forall\ j = 1, ..., p.$

alors $\quad f(x_i / Q) = f((Z(i, 1), ..., Z(i, p)) / Q^{m(1)}, ..., Q^{m(p)})$

$$= \sum_{k=1}^{K} p_k\, f((Z(i, 1), ..., Z(i, p)) / (\lambda_k^{m(1)}, ..., \lambda_k^{m(p)}))$$

$$= \sum_{k=1}^{K} p_k\, f(x_i / (\lambda_k, Q)) \qquad (4.13)$$

En suppose qu'à l'intérieur de chaque classe, les p variables aléatoires ($\xi^1, ..., \xi^p$) sont mutuellement indépendantes. On peut écrire :

$$f(x_i / (\lambda_k, Q)) = \prod_{j=1}^{p} f(x_i^j / \lambda_k^{m(j)}) \qquad (4.14)$$

on considère deux cas :

Premier cas

Tout les composants auxquels les éléments j de l'échantillon J appartiennent sont distincts, dans ce cas, le nombre de composants du mélange J coincide avec la taille de l'échantillon. On obtient :

$$\lambda_k^{m(j)} \neq \lambda_k^{m(j')} \qquad \forall\, j \neq j'$$

on peut alors écrire : $\lambda_k^{m(j)} = \lambda_k^j \qquad \forall\, j = 1, ..., p$

La formule (4.13) s'écrit : $$f(x_i) = \sum_{k=1}^{K} p_k\ f(x_i / \lambda_k) \qquad (4.15)$$

et la formule (4.14) devient : $$f(x_i / \lambda_k) = \prod_{j=1}^{p} f(x_i^j / \lambda_k^j) \qquad (4.16)$$

Deuxième cas

Passons maintenant au cas le plus général où l'on suppose que plusieurs éléments de l'échantillon J peuvent provenir d'un même composant ; soit M le nombre de composants dans lesquels sont répartis tout les éléments de l'échantillon J. On suppose toujours connu le composant auquel appartient chaque élément j de l'échantillon J que l'on note par $Q^{m(j)}$.

Posons $\lambda_k^{m(j)} = \lambda_k^m \qquad \forall\, j \in Q^{m(j)} = Q^m$

si on remplace maintenant dans l'expression (4.13) on obtient :

$$f(x_i / (\lambda_k, Q)) = \prod_{m=1}^{M} \prod_{j \in Q^m} f(x_i^j / \lambda_k^m) \qquad (4.17)$$

Remarque :

Que l'on soit dans le premier ou le deuxième cas, on se retrouve toujours avec une expression du modèle qui correspond à celle d'un mélange simple (4.13) et (4.15). Pour approfondir l'étude que l'on vient de faire, on se propose de l'appliquer à deux types de fonctions de densités.

5. APPLICATIONS PRATIQUES

Dans ce paragraphe, nous nous proposons de voir ce que deviennent les expressions données par les formules (4.16) et (4.17) en l'appliquant à deux types de familles de fonctions de densité ; on suppose dans tout ce paragraphe qu'on se trouve dans la situation (ii).

5.1. LOIS GAUSSIENNES UNIDIMENSIONNELLES

Premier cas $F(x / \lambda_k) = \prod_{j=1}^{p} f(x^j / \lambda_k^j)$ avec $\lambda_k^j = (\mu_k^j, V_k^j)$

$$F(x / \lambda_k) = \prod_{j=1}^{p} (2\pi V_k^j)^{-1/2}. \exp \left[- \frac{(x^j - \mu_k^j)^2}{2.V_k^j} \right]$$

$$F (x / \lambda_k) = (2\pi)^{-p/2} . | V_k |^{-1/2}. \exp \left[\frac{-{}^t(x-\mu_k).V_k^{-1}.(x-\mu_k)}{2} \right]$$

$\lambda_k = (\mu_k, V_k)$

μ_k: espérance du composant numéro k.

V_k : matrice de variance du composant numéro k.

$F (x / \lambda_k)$ représente donc la densité d'une loi gaussienne multidimensionnelle. On retrouve ainsi le modèle de mélange gaussien proposé par Celeux (1988) pour la classification simple de tableaux décrits par des données quantitatives.

Deuxième cas $f(x / \lambda_k) = \prod_{m=1}^{M} \prod_{j \in Q^m} f(x^j / \lambda_k^m)$

$$= \prod_{m=1}^{M} \prod_{j \in Q^m} (2\pi V_k^m)^{-1/2}. \exp \left[- \frac{(x^j - \mu_k^m)^2}{2.V_k^m} \right]$$

$F(x / \lambda_k)$ représente aussi la densité d'une loi gaussienne multidimensionnelle. Par analogie au premier cas, on peut déduire que ce modèle correspond lui aussi à la classification simple de données quantitatives, dont les éléments $j \in J$ sont répartis en M classes.

5.2. LOIS DE BERNOULLI

La famille analysée est définie par :

$$p(x / \lambda) = \varepsilon^{|x-\lambda|} \cdot (1-\varepsilon)^{1-|x-\lambda|}$$

où $\varepsilon \in]0, \frac{1}{2}[$; $x \in \{0, 1\}$ et $\lambda \in \{0, 1\}$.

$p(x / \lambda)$ désigne la distribution d'une loi de Bernoulli de paramètre $(1-\varepsilon)$ ou de paramètre ε.

Premier cas

$$p(x / \lambda_k) = \prod_{j=1}^{p} p(x^j / \lambda_k^j) = \prod_{j=1}^{p} \left\{ \varepsilon^{|x^j-\lambda_k^j|} \cdot (1-\varepsilon)^{1-|x^j-\lambda_k^j|} \right\}.$$

$p(x / \lambda_k)$ représente alors le produit de p lois de Bernoulli. Ce modèle correspond exactement à celui proposé par Govaert (1988) pour la classification simple de tableaux binaires (cf. chapitre trois).

Deuxième cas

$$p(x / \lambda_k) = \prod_{m=1}^{M} \prod_{j \in Q^m} p(x^j / \lambda_k^m) = \prod_{m=1}^{M} \left\{ \varepsilon^{|x^m-q_m.\lambda_k^m|} \cdot (1-\varepsilon)^{q_m-|x^m-q_m.\lambda_k^m|} \right\}.$$

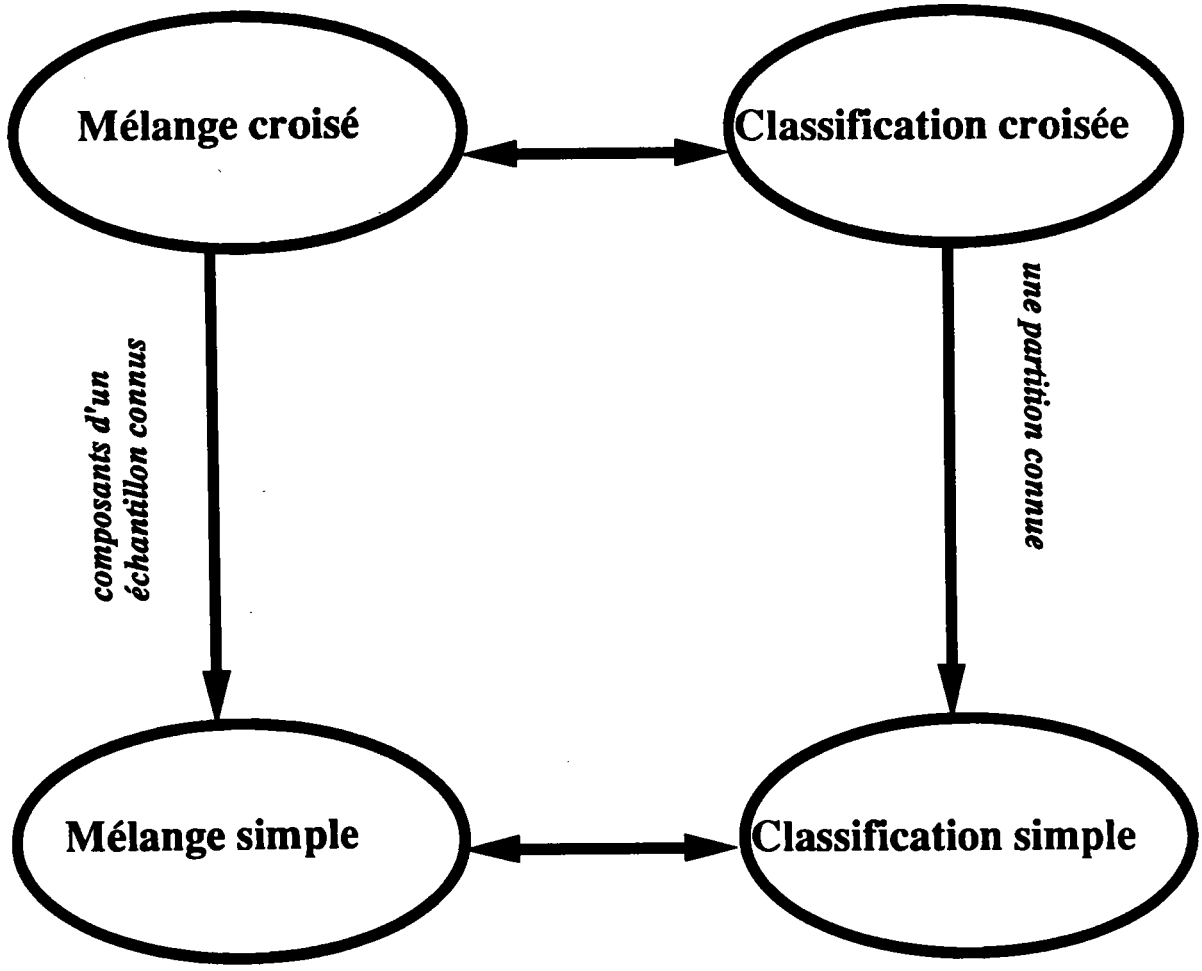
Il est facile de vérifier que la distribution $p(x / \lambda_k)$ ainsi trouvée correspond au modèle de mélange simple associé à la classification croisée de données binaires en supposant connu et fixe la partition Q en colonnes de l'ensemble J (cf chapitre cinq).

Conclusion

L'étude pratique que l'on vient de faire nous conduit à conclure que lorsqu'on se retrouve dans le premier cas, on retrouve exactement le même modèle que celui qui a été proposé pour la classification simple (Celeux (1988) et Govaert (1988)). Si maintenant on se trouve dans le deuxième cas, on montre que le modèle de mélange

trouvé correspond aux méthodes de classifications croisées en supposant fixe et connu une partition.

Ainsi, la connaissance d'une partition d'un ensemble s'interprète en termes de modèle par la connaissance des composants d'un échantillon, et nous venons de montrer par les deux applications faites ci-dessus que le lien qui existe entre les méthodes de classification simple et les modèles de mélanges simples est le même que celui qui existe entre les méthodes de classification croisée en supposant connu une partition et les modèles de mélanges croisés en supposant connu les composants d'un échantillon. (voir figure ci-dessous).



CHAPITRE 5

CLASSIFICATION CROISEE ET MODELES SUR DONNEES BINAIRES

INTRODUCTION

Suivant la nature des données, Govaert (1983) a proposé plusieurs méthodes de classification croisées, telle la méthode CROKI2 destinée à la classification croisée de tableaux de contingence optimisant le critère du Khi2, la méthode CROBIN pour les tableaux binaires, la méthode CROEUC pour les tableaux de mesures, enfin la méthode CROMUL pour la classification croisée d'un questionnaire qui peut être considéré comme un tableau de contingence multiple.

Nous avons montré dans le chapitre précédent que souvent il existe un lien étroit entre les méthodes de classification croisée et les modèles de mélange "croisé". Nous proposons dans ce chapitre d'étudier ces liens dans le cas des tableaux binaires. Dans tout ce chapitre, l'ensemble I et l'ensemble J décrivant le tableau de données binaires seront considérés et traités de manière identique. Dans le cas où un seul des ensembles peut être considéré comme un ensemble où on a des variables binaires, on pourra toujours appliquer les résultats du chapitre précédent, car ces variables binaires sont toutes de même nature (variables qualitatives à deux modalités) et peuvent donc être considérées comme un ensemble d'individus.

Dans le premier paragraphe, nous décrivons la méthode CROBIN (Govaert 1983) qui est une méthode de classification croisée de données binaires et l'algorithme qui lui est associé. Dans le second paragraphe, nous montrons comment la méthode précédente peut être interprétée comme l'approche classification associée à un mélange de distribution de Bernoulli avec le même paramètre pour toutes les classes ; ce modèle correspond au critère de classification croisée binaire utilisant la distance L_1 et des noyaux binaires. Nous étudierons dans le paragraphe trois une extension de ce modèle

au cas où le paramètre varie suivant les classes en ligne, les classes en colonne, puis suivant les classes en lignes et en colonne : c'est le cas le plus général . En outre, en nous appuyons sur des variantes de ce modèle, nous proposons de nouveaux algorithmes de **classification croisée** utilisant des distances **adaptatives binaires.**

Le paragraphe quatre comporte une étude théorique des éventuels problèmes de dégénérescence, et l'on propose des solutions pour les résoudre. Enfin nous terminons par l'interprétation les bons résultats que Govaert (1983) avait obtenu en appliquant la méthode CROBIN sur des données simulées.

1. LA METHODE CROBIN

1.1. LE PROBLEME

Soit **X(I, J)** un tableau croisant un ensemble **I** de n individus et un ensemble **J** de p variables binaires. La méthode CROBIN (Govaert 1983) fournit une solution locale au problème d'optimisation suivant :

Trouver une partition P de I en K classes, une partition Q de J en M classes et un tableau binaire à K lignes et M colonnes, tel que le critère :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} |x_i^j - \lambda_k^m| \qquad (5.1)$$

soit minimal.

L'objectif est de trouver des blocs binaires homogènes, c'est-à-dire des classes remplies soit de 1, soit de 0. On associe à chaque couple (k, m) de classes une valeur binaire (1 ou 0). On obtient ainsi un tableau binaire que l'on appelle noyau. On cherche alors à minimiser le nombre de fois où la valeur associée à un couple (i, j) est différente de la valeur idéale associée au couple de classes auxquelles appartiennent respectivement i et j. Cette quantité, qui représente l'écart entre le tableau initial et le tableau idéal, est le critère W. L'optimisation de ce critère reflète bien la recherche de partitions tendant à donner au tableau la structure représentée par la figure A. L'algorithme CROBIN décrit ci-dessus permet d'optimiser localement ce critère.

La partition obtenue pour l'exemple de la figure A est la suivante :

P = {(A, C, H), (B, F, J), (D, G, I, E)}.

Q= {(1, 4), (3, 5, 7), (2, 6)}.

Carré noir : indique la valeur 1.

Carré blanc : indique la valeur 0.

Réorganisation d'un tableau binaire

figure A

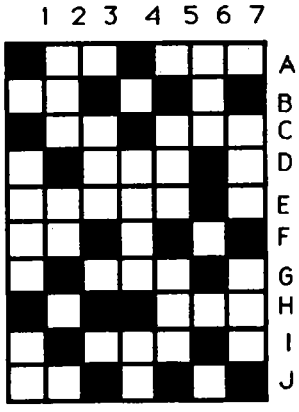


Tableau initial

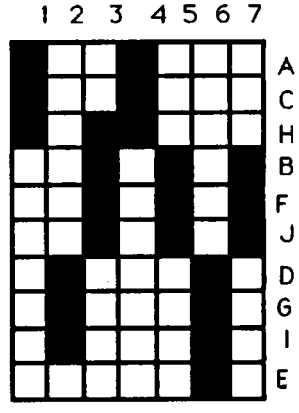


Tableau réorganisé suivant la partition P.

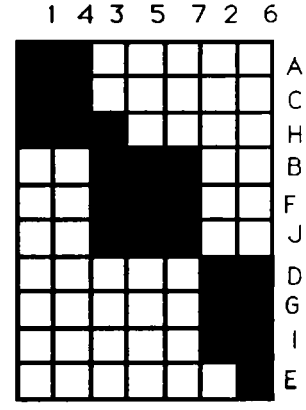


Tableau réorganisé suivant les deux partitions P et Q.

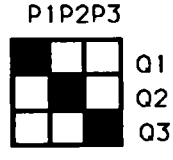


Tableau des noyaux

1.2. L'ALGORITHME

L'algorithme utilisé dans la méthode CROBIN est le suivant : partant d'un élément (P, Q, L) initial, on fixe Q et on cherche à améliorer P et L, puis on fixe P et on cherche à améliorer Q et L on construit ainsi une suite (P^n, Q^n, L^n) qui fait décroître le critère W. L'algorithme est donc construit à partir de deux étapes intermédiaires que nous allons préciser. Remarquons que cet algorithme diffère légèrement de celui qui a été proposé dans le chapitre quatre (paragraphe 1.3) puisque ce dernier se déroule en trois étapes.

1.2.1 Etapes intermédiaires

Soit P et Q un couple de partitions et L un noyau. Fixons Q et cherchons à améliorer la partition P et le noyau L, c'est à dire cherchons une partition P' et un noyau L' tels que :

$$W(PxQ, L) > W(P'xQ, L')$$

On peut écrire :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{i \in P_k} \left(\sum_{m=1}^{M} \sum_{j \in Q^m} |x_i^j - \lambda_k^m| \right)$$

Posons $x_i^m = \sum_{j \in Q_m} x_i^j$ et $q_m = Card(Q^m)$

et notons B la quantité :

$$B = \begin{cases} x_i^m & \text{si } \lambda_k^m = 0 \\ q_m - x_i^m & \text{si } \lambda_k^m = 1 \end{cases}$$

On peut donc écrire $B = |x_i^m - q_m.\lambda_k^m|$. D'ou l'expression du critère :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{m=1}^{M} |x_i^m - q_m.\lambda_k^m|$$

Considérons l'algorithme des Nuées Dynamiques (Diday 1979) suivant :
- Le tableau de données est le tableau (I, Q) défini par les x_i^m. Les individus sont en lignes et les variables en colonnes. On a donc un ensemble de n éléments et M variables.

- Les noyaux sont de la forme ($q_1.\lambda_k^1$, $q_2.\lambda_k^2$,..., $q_M.\lambda_k^M$) avec $\lambda_k^m \in \{0,1\}$.

Chacune des composantes du noyau ne peut être que le minimum ou le maximum atteint par le regroupement des colonnes du tableau initial, c'est à dire les valeurs 0 ou q_m puisque les valeurs initiales étaient 0 ou 1, et qu'il n'y a que q_m colonnes réunies pour former la classe Q^m.
On notera L la matrice des composantes λ_k^m qui définissent l'ensemble des noyaux précédents.
- La distance est la distance " City - block " ou distance L_1.

Le critère habituel de la méthode des Nuées Dynamiques défini à partir des éléments que l'on vient de donner est alors la fonction W(PxQ, L) que l'on cherche à optimiser. Il suffit de préciser ce que deviennent les fonctions f et g.

Fonction d'affectation (recherche des classes) :

La fonction f d'affectation est toujours la même, chaque élément i est affecté à la classe du noyau de laquelle il est le plus prés.

Fonction de représentation (recherche des noyaux) :

On cherche pour toutes les classes P_k, les éléments $(q_1.\lambda_k^1, q_2.\lambda_k^2, ..., q_M.\lambda_k^M)$ minimisant :

$$\sum_{i \in P_k} \sum_{m=1}^{M} |x_i^m - q_m.\lambda_k^m|$$

Cela revient à chercher les éléments $q_m.\lambda_k^m$ minimisant :

$$\sum_{i \in P_k} |x_i^m - q_m.\lambda_k^m|$$

on obtient :
$$\begin{cases} \sum_{i \in P_k} x_i^m & si\ \lambda_k^m = 0 \\ \sum_{i \in P_k} (q_m - x_i^m) = n_k.q_m - \sum_{i \in P_k} x_i^m & si\ \lambda_k^m = 1 \end{cases}$$

Posons $B' = \sum_{i \in P_k} x_i^m$ $\quad n_k = \text{Card}(P_k)$

Il suffit de prendre comme $m^{ème}$ composante du noyau k, la valeur 0 si B' est plus petit que $(n_k.q_m - B')$ et q_m sinon.
Remarquons que cette fonction correspond aussi à la recherche du noyau λ_k^m associé à un couple de classe $P_k x Q^m$. En effet, les éléments λ_k^m définis suivant la régle précédente seront 0 ou 1 selon que la somme des éléments intervenant dans la classe $P_k x Q^m$ sera proche de 0 ou proche du maximum , c'est à dire $n_k.q_m$. On retrouve ainsi la régle majoritaire. On notera L' (P, Q) le noyau ainsi obtenu.

1.2.2. Convergence de l'algorithme

L'algorithme CROBIN est le suivant :
A partir d'un triplet (P^0,Q^0, L^0) on applique alternativement les deux étapes intermédiaires ; on définit ainsi une suite (P^n, Q^n, L^n) qui vérifie :

$$W(P^0 \times Q^0 , L^0) \geq W(P^1 \times Q^1 , L^1) \geq \ldots \geq W(P^n \times Q^n, L^n) \geq \ldots$$

On peut établir à partir de là les propriétés de convergence habituelles. La suite (P^n, Q^n, L^n) est stationnaire et la valeur du critère associé $W(P^n \times Q^n, L^n)$ décroît strictement jusqu'à la stationnarité. On obtient ainsi une solution localement optimale qui dépend de l'élément de départ choisi. De plus on a montré qu'à chaque étape le noyau L^n est formé en associant à chaque couple de partitions la valeur binaire majoritaire du bloc de 0 et 1 défini par chaque couple de classe. On utilise cette propriété pour simplifier la détermination de l'élément (P^n , Q^n , L^n) : il suffit en effet d'avoir un couple (P^0, Q^0), le noyau L^0 est alors construit à partir de ces partitions.

Pour illustrer le principe de la méthode CROBIN, nous proposons de l'appliquer sur l'exemple suivant.

1.3. EXEMPLE

Soit un ensemble de 10 micro-ordinateurs identifiés par les lettres a à i caractérisés par un ensemble de 10 propriétés identifiées par les nombres 1 à 10. On représente les données initiales (figure 1) sous forme d'un tableau binaire où un 1 indique que la propriété est vérifiée et un 0 qu'elle ne l'est pas.
Si on applique l'algorithme CROBIN en demandant trois classes en ligne et deux classes en colonne, on obtient comme partition de l'ensemble des micro-ordinateurs l'ensemble des classes:

$$P = \{\{a, d, h\}, \{b, e, f, j\}, \{c, g, i\}\}$$

et comme partition de l'ensemble des propriétés l'ensemble des classes:

$$Q = \{\{1, 3, 5, 8, 10\}, \{2, 4, 6, 7, 9\}\}$$

On peut représenter les partitions sur les données initiales (figure 2) en réordonnant simplement les lignes et les colonnes de manière à respecter les partitions. Les noyaux obtenus sont indiqués sur la figure 3 ; la valeur du critère est égale à 16 ce qui indique que sur 100 valeurs initiales du tableau, 16 ne sont pas égales à la valeur idéale représentée par le noyau correspondant.

On peut voir que les micro-ordinateurs de la classe A possède en général les propriétés 1 mais pas 2 ; ceux de la classe B les propriétés 2 mais pas 1 et ceux de la classe C aucune propriété. On a résumé le tableau de 100 valeurs par un tableau de 6 valeurs. Les valeurs du tableau des écarts (fig-4) représentent le nombre de fois où la valeur majoritaire n'a pas été prise pour chaque classe, la somme de ces valeurs représente la valeur du critère.

										0
	1	2	3	4	5	6	7	8	9	1
a	1	0	1	0	1	0	0	1	0	1
b	0	1	0	1	0	1	1	0	1	0
c	1	0	0	0	0	0	0	1	1	0
d	1	0	1	0	0	0	0	1	0	0
e	0	1	0	1	1	1	1	0	1	0
f	0	1	0	0	1	1	1	0	1	0
g	0	1	0	0	0	0	0	1	0	1
h	1	0	1	0	1	1	0	1	1	1
i	1	0	0	1	0	0	0	0	0	1
j	0	1	0	1	0	0	1	0	0	0

fig. 1
Tableau initial

						0				
	1	3	5	8	1	2	4	6	7	9
a	1	1	1	1	1	0	0	0	0	0
d	1	1	0	1	0	0	0	0	0	0
h	1	1	1	1	1	0	0	1	0	1
b	0	0	0	0	0	1	1	1	1	1
e	0	0	0	0	0	1	1	1	1	1
f	0	0	0	0	0	1	0	1	1	1
j	0	0	0	0	0	1	1	0	1	0
c	1	0	0	1	0	0	0	0	0	1
g	0	0	0	1	1	1	0	0	0	0
i	1	0	0	0	1	0	1	0	0	0

fig. 2
Tableau réordonné

	1	2
A	1	0
B	0	1
C	0	0

fig.3
Tableau des valeurs idéales.

	1	2
A	2	2
B	0	3
C	6	3

fig.4
Tableau des écarts.

2. MODELE ASSOCIE AUX DONNEES BINAIRES

En suivant l'approche classification proposée dans le chapitre quatre pour identifier le mélange, on propose alors le modèle suivant.

2.1. LA FORMULE GENERALE

On suppose dans ce modèle que les données initiales forment un échantillon de taille (n, p) d'une variable aléatoire à valeurs dans {0, 1}et dont la distribution de probabilité f est toujours définie par (4.3) et (4.4). Mais ici $p(., \lambda_k^m)$ est une distribution de probabilité sur {0, 1} appartenant à une famille paramétrée de distributions de probabilités.

En suivant l'approche " classification", rappelée dans le paragraphe trois du chapitre quatre pour identifier un mélange, on se ramène à la maximisation d'un critère de vraisemblance classifiante VC(PxQ, L) défini par (4.7). On peut alors utiliser le même algorithme que celui proposé au paragraphe 3.1 du chapitre 4, à partir d'un couple de partition (P^o, Q^o) en K et M classes de l'échantillon en utilisant alternativement les trois étapes de l'algorithme jusqu'à l'obtention d'une partition stable.

2.2. CHOIX DE LA FAMILLE DE DISTRIBUTION :

On suppose que la variable aléatoire Z suit une des deux lois de Bernoulli suivantes:

$$\begin{cases} 1 \text{ avec la probabilité } 1-\varepsilon \text{ et } 0 \text{ avec la probabilité } \varepsilon \\ 1 \text{ avec la probabilité } \varepsilon \text{ et } 0 \text{ avec la probabilité } 1-\varepsilon \end{cases}$$

où $\varepsilon \in]0, \frac{1}{2}[$; c'est à dire la loi de Bernoulli de paramètre (1-ε) et la loi de Bernoulli de paramètre ε.

On peut alors écrire :

$$p(x / \lambda_k^m) = \varepsilon^{|x-\lambda_k^m|} \cdot (1-\varepsilon)^{1-|x-\lambda_k^m|}$$

où λ_k^m indique quelle est la distribution retenue:

$$\begin{cases} \lambda_k^m = 1 \text{ pour la première distribution} \\ \lambda_k^m = 0 \text{ pour la seconde distribution} \end{cases}$$

Les paramètres à estimer sont donc les λ_k^m et la valeur ε.

L'expression du critère devient alors :

$$VC(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \text{Log } p(x_i^j / \lambda_k^m)$$

$$= \left(\text{Log}\frac{\varepsilon}{1-\varepsilon}\right) \sum_{k=1}^{K} \sum_{m=1}^{L} \sum_{i \in P_k} \sum_{j \in Q^m} d(x_i^j, \lambda_k^m) + n.p \text{ Log } (1-\varepsilon) \qquad (5.2)$$

où $d(x_i^j, \lambda_k^m) = |x_i^j - \lambda_k^m|$.

Donc, pour un ε fixé appartenant à $]0, \frac{1}{2}[$, $\text{Log}\frac{\varepsilon}{1-\varepsilon}$ est négatif. La maximisation du critère $VC(PxQ, L, \varepsilon)$ revient donc à la minimisation du critère $W(PxQ, L)$ présenté dans le paragraphe 1, ce qui montre l'équivalence des deux approches.
Il est facile de voir que la valeur ε maximisant le critère $VC(PxQ, L, \varepsilon)$ est $\frac{e}{n.p}$, où e est la valeur du critère obtenu à la convergence.

$$e = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} |x_i^j - \lambda_k^m|.$$

Ce premier mélange (noté M_1) de lois de Bernoulli dépend du paramètre ε qui mesure l'écart d'une classe à son centre, ne dépend ni de la partition en lignes ni de la partition en colonnes ni de la partition en lignes et en colonnes, ce qui, dans certaines situations, peut s'avérer irréaliste. Ainsi nous proposons d'autres critères où le paramètre peut dépendre de la partition en lignes, de la partition en colonnes et de la partition en lignes et en colonnes. On considère alors trois autres mélanges de Bernoulli (notés respectivement M_2, M_3, M_4) que l'on se propose d'étudier dans le paragraphe suivant.

3. EXTENSION DU MODELE BINAIRE

3.2. ETUDE DU MELANGE M_2

Le paramètre varie suivant la partition en ligne. On remplace ici la valeur ε par les valeurs ε_k qui dépendent de chaque classe en lignes mais qui sont toujours les mêmes pour les classes en colonnes.

$$p\{x / (\lambda_k^m, \varepsilon_k)\} = \varepsilon_k^{|x-\lambda_k^m|} \cdot (1-\varepsilon_k)^{1-|x-\lambda_k^m|}$$

Le critère de vraisemblance classifiante va alors s'écrire :

$$VC_3\,(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k}{1-\varepsilon_k} \right) |x_i^j - \lambda_k^m| + \mathrm{Log}\,(1-\varepsilon_k) \right\} \quad (5.3)$$

Fonction de représentation *(recherche des λ_k^m et des ε_k)*

Les λ_k^m sont toujours les valeurs majoritaires de chaque classe $P_k x Q^m$. Les valeurs des ε_k seront définis comme suit :

$$VC_3\,(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k}{1-\varepsilon_k} \right) \left(\sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} |x_i^j - \lambda_k^m| \right) + p.n_k \mathrm{Log}\,(1-\varepsilon_k) \right\}$$

Posons $e_k = \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} |x_i^j - \lambda_k^m|$.

e_k : est le nombre de fois que les valeurs majoritaires n'ont pas été prises pour la classe P_k.

$$VC_3\,(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k}{1-\varepsilon_k} \right) e_k + p.n_k \mathrm{Log}\,(1-\varepsilon_k) \right\}$$

$$= \sum_{k=1}^{K} \left\{ (p.n_k - e_k)\,\mathrm{Log}\,(1-\varepsilon_k) + e_k.\mathrm{Log}\,\varepsilon_k \right\} = \sum_{k=1}^{K} \psi(\varepsilon_k)$$

Il faut donc maximiser la fonction ψ.

$$\psi'(\varepsilon_k) = \frac{-(p.n_k - e)}{1-\varepsilon_k} + \frac{e_k}{\varepsilon_k} = \frac{e_k - p.n_k.\varepsilon_k}{(1-\varepsilon_k).\varepsilon_k} = 0$$

Le maximum est atteint pour $\varepsilon_k = \frac{e_k}{n_k.p}$ qui appartient bien à l'intervalle $]0, \frac{1}{2}[$, sauf dans le cas très particulier où $e_k = 0$.

Fonction d'affectation (recherche des classes)

- <u>On fixe la partition Q :</u>

$$VC_3\,(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{m=1}^{M} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k}{1-\varepsilon_k} \right) \left(\sum_{j \in Q^m} |x_i^j - \lambda_k^m| \right) + q_m.\mathrm{Log}\,(1-\varepsilon_k) \right\}$$

Posons $x_i^m = \sum_{j \in Q_m} x_i^j$ et $\mu_k^m = q_m.\lambda_k^m$.

$$VC_3 (PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{i \in P_k} \left\{ \left(\text{Log} \frac{\varepsilon_k}{1-\varepsilon_k} \right) \left(\sum_{m=1}^{M} |x_i^m - \mu_k^m| \right) + p.\text{Log}(1-\varepsilon_k) \right\}$$

$$= - \sum_{k=1}^{K} \sum_{i \in P_k} \left\{ \left(\text{Log} \frac{1-\varepsilon_k}{\varepsilon_k} \right) .d(x_i, \mu_k) - A_k \right\}$$

d : est une distance de type L_1 et A_k est la quantité $p.\text{Log}(1-\varepsilon_k)$ qui dépend du point x_i. On affectera donc l'élément x_i à la classe P_k qui minimise la quantité :

$$\left(\text{Log} \frac{1-\varepsilon_k}{\varepsilon_k} \right) d(x_i, \mu_k) - p.\text{Log}(1-\varepsilon_k)$$

- <u>On fixe la partition P :</u>

$$VC_3 (PxQ, L, \varepsilon) = \sum_{m=1}^{M} \sum_{j \in Q^m} \sum_{k=1}^{K} \left\{ \left(\text{Log} \frac{\varepsilon_k}{1-\varepsilon_k} \right) \left(\sum_{i \in P_k} |x_i^j - \lambda_k^m| \right) + n_k.\text{Log}(1-\varepsilon_k) \right\}$$

$$= \sum_{m=1}^{M} \sum_{j \in Q^m} \left\{ - \sum_{k=1}^{K} \left(\text{Log} \frac{1-\varepsilon_k}{\varepsilon_k} \right) |x_k^j - \gamma_k^m| + \sum_{k=1}^{K} n_k.\text{Log}(1-\varepsilon_k) \right\}$$

$$= - \sum_{m=1}^{M} \sum_{j \in Q^m} d_\varepsilon(x^j, \gamma^m) + A'$$

où $x_k^j = \sum_{i \in P_k} x_i^j$ et $\gamma_k^m = n_k.\lambda_k^m$.

d_ε est une distance de type L_1 pondérée par les quantités $\text{Log} \frac{1-\varepsilon_k}{\varepsilon_k}$ qui dépendent du vecteur ε et A' est la quantité $p. \sum_{k=1}^{K} n_k.\text{Log}(1-\varepsilon_k)$ qui ne dépend pas du point x^j.

3.2. ETUDE DU MELANGE M_3

Dans ce mélange le paramètre varie suivant la partition en colonnes. On se place exactement dans les mêmes conditions que dans le mélange M_2. On obtient des résultats symétriques.

3.3. ETUDE DU MELANGE M_4

Le paramètre varie cette fois-ci suivant la partition en lignes et en colonnes ; c'est le cas le plus général .

$$p\{x / (\lambda_k^m,\ \varepsilon_k^m)\} = \varepsilon_k^{m\ |x-\lambda_k^m|} \cdot (1-\varepsilon_k^m)^{1-|x-\lambda_k^m|}$$

Le critère de vraisemblance classifiante devient :

$$VC_4(PxQ,\ L,\ \varepsilon) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i\in P_k} \sum_{j\in Q^m} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k^m}{1-\varepsilon_k^m} \right) |x_i^j-\lambda_k^m| + \mathrm{Log}\,(1-\varepsilon_k^m) \right\} \qquad (5.4)$$

Fonction de représentation *(recherche des* λ_k^m *et des* ε_k^m *)*

Les λ_k^m sont toujours les valeurs majoritaires pour les classes $P_k x Q^m$. Il reste donc à maximiser le critère $VC_4(PxQ, L, \varepsilon)$ pour trouver les valeurs des ε_k^m .

$$VC_4(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{m=1}^{M} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k^m}{1-\varepsilon_k^m} \right) \left(\sum_{i\in P_k} \sum_{j\in Q^m} |x_i^j-\lambda_k^m| \right) + n_k.q_m.\mathrm{Log}\,(1-\varepsilon_k^m) \right\}$$

Posons $e_k^m = \sum_{i\in P_k} \sum_{j\in Q^m} |x_i^j-\lambda_k^m|$

où e_k^m est le nombre de fois que la valeur majoritaire n'a pas été prise dans une classe $P_k x Q^m$.

$$VC_4(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{m=1}^{M} \left\{ \left(\mathrm{Log} \frac{\varepsilon_k^m}{1-\varepsilon_k^m} \right) e_k^m + n_k.q_m.\,\mathrm{Log}\,(1-\varepsilon_k^m) \right\}$$

$$= \sum_{k=1}^{K} \sum_{m=1}^{M} \left\{ (n_k.q_m - e_k^m)\,\mathrm{Log}\,(1-\varepsilon_k^m) + e_k^m.\mathrm{Log}\,\varepsilon_k^m \right\} = \sum_{k=1}^{K} \sum_{m=1}^{M} \varphi(\varepsilon_k^m)$$

Il faut donc maximiser la fonction φ :

$$\varphi(\varepsilon_k^m) = (n_k.q_m - e_k^m).\mathrm{Log}\,(1-\varepsilon_k^m) + e_k^m.\mathrm{Log}\,\varepsilon_k^m$$

Le maximum est atteint pour $\varepsilon_k^m = \frac{e_k^m}{n_k.q_m}$ qui appartient bien à l'intervalle $]0, \frac{1}{2}[$, sauf dans le cas très particulier où $e_k^m = 0$.

Fonction d'affectation (recherche des classes) :

- On fixe la partition Q :

$$VC_4(PxQ, L, \varepsilon) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{m=1}^{M} \left\{ \left(\text{Log}\frac{\varepsilon_k^m}{1-\varepsilon_k^m}\right) \left(\sum_{j \in Q^m} |x_i^j - \lambda_k^m| \right) + q_m.\text{Log}\,(1-\varepsilon_k^m) \right\}$$

$$= \sum_{k=1}^{K} \sum_{i \in P_k} \left\{ \left(\sum_{m=1}^{M} \left(\text{Log}\frac{\varepsilon_k^m}{1-\varepsilon_k^m}\right) |x_i^m - \mu_k^m| \right) + \sum_{m=1}^{M} q_m.\text{Log}\,(1-\varepsilon_k^m) \right\}$$

$$= - \sum_{k=1}^{K} \sum_{i \in P_k} \left\{ d_{\varepsilon k}(x_i, \mu_k) - \sum_{m=1}^{M} q_m.\text{Log}\,(1-\varepsilon_k^m) \right\}$$

Cette fois-ci, le second terme dépend de k et aura donc une influence. On affectera x_i à la classe P_k qui minimise :

$$d_{\varepsilon k}(x_i, \mu_k) - \sum_{m=1}^{M} q_m.\text{Log}\,(1-\varepsilon_k^m)$$

- On fixe maintenant la partition P :

$$VC_4(PxQ, L, \varepsilon) = \sum_{m=1}^{M} \sum_{j \in Q_m} \left\{ \sum_{k=1}^{K} \left(\text{Log}\frac{\varepsilon_k^m}{1-\varepsilon_k^m}\right) |x_k^j - \gamma_k^m| + \sum_{k=1}^{K} n_k.\text{Log}\,(1-\varepsilon_k^m) \right\}$$

$$= - \sum_{m=1}^{M} \sum_{j \in Q^m} \left\{ d_{\varepsilon^m}(x^j, \gamma^m) - \sum_{k=1}^{K} n_k.\text{Log}\,(1-\varepsilon_k^m) \right\}$$

On se retrouve là aussi dans la même situation que dans le cas précédent. Le second terme dépend de m et aura donc une influence. On affectera x^j à la classe Q^m qui minimise :

$$d_{\varepsilon^m}(x^j, \gamma^m) - \sum_{k=1}^{K} n_k.\text{Log}\,(1-\varepsilon_k^m)$$

où $d_{\varepsilon^m}(x^j, \gamma^m) = \sum_{k=1}^{K} \left(\text{Log}\frac{1-\varepsilon_k^m}{\varepsilon_k^m}\right) |x_k^j - \gamma_k^m|$.

Comme on a pu le constater, l'extension du modèle proposé pour la méthode CROBIN pose un problème de dégénérescence au niveau du calcul des coefficients de pondération ; nous proposons dans le paragraphe quatre d'étudier ces problèmes et d'essayer de les résoudre.

4. PROBLEMES DE DEGENERESCENCE

On remarque, dans l'étude que l'on vient de faire, que les coefficients de pondération obtenus sont tous exprimés en fonction de e_k (pour le mélange M_2), e^m (pour le mélange M_3) et e_k^m (pour le mélange M_4) ; on a remarqué que certains d'entre eux ne sont pas définis pour les e_k nuls, ou e^m nuls ou encore pour les e_k^m nuls. On peut alors s'interroger sur la validité des critères $VC_2(PxQ, L, \varepsilon)$, $VC_3(PxQ, L, \varepsilon)$ et $VC_4(PxQ, L, \varepsilon)$ dans ce cas de figure.

Posons :

$$\alpha_k = \text{Log}\frac{1-\varepsilon_k}{\varepsilon_k} = \text{Log}\frac{p.n_k - e_k}{e_k} \qquad (5.5)$$

$$\alpha^m = \text{Log}\frac{1-\varepsilon^m}{\varepsilon^m} = \text{Log}\frac{n.q_m - e^m}{e^m} \qquad (5.6)$$

$$\alpha_k^m = \text{Log}\frac{1-\varepsilon_k^m}{\varepsilon_k^m} = \text{Log}\frac{n_k.q_m - e_k^m}{e_k^m} \qquad (5.7)$$

i) Supposons que pour k_o (respectivement m_o), tous les éléments de chaque classe $P_{k_o}xQ^m$ (m = 1, ..., M) ((respectivement $P_kxQ^{m_o}$ (k = 1, ..., K)) prennent la même valeur (0 ou 1), alors $e_{k_o}^m$ (respectivement $e_k^{m_o}$) est nul pour tout m (respectivement tout k) ce qui entraîne que e_{k_o} (respectivement e^{m_o}) est nul, et par conséquent les pondérations α_{k_o} et α^{m_o} ne sont plus définies par les formules (5.5) et (5.6).

ii) Supposons maintenant qu'il existe un k_o et un m_o tel que tous les éléments de la classe $P_{k_o}xQ^{m_o}$ prennent la même valeur (0 ou 1), alors $e_{k_o}^{m_o}$ est nul et par suite la pondération $\alpha_{k_o}^{m_o}$ n'est plus définie par la formule (5.7).

Pour éviter ces problèmes de dégénérescence, on propose comme solution, à chaque fois qu'on se retrouve dans la situation (i) où (ii) de garder les anciennes valeurs des coefficients de pondération, c'est-à-dire les valeurs de α_k, α^m et α_k^m qui correspond à l'itération précédent le point de dégénérescence. On montre alors qu'on procédant ainsi on rend le calcul du critère possible en l'augmentant mais on ne l'optimise pas. Comme nous allons le voir ces valeurs n'auront aucune influence sur la valeur du critère.

i) Si $e_{ko} = 0$ alors α_{ko} non définie.

$$VC_3(PxQ, L, \alpha(\varepsilon), \varepsilon) = \sum_{k=1}^{K} \{\alpha_k \cdot e_k - n_k.p \, Log(1-\varepsilon_k)\}$$

$$= \sum_{k \neq k_o} \{\alpha_k \cdot e_k - n_k.p \, Log(1-\varepsilon_k)\} + \{\alpha_{ko} \cdot e_{ko} - n_k.p \, Log(1- \frac{e_{ko}}{n_{ko}.p})\} \qquad (5.8)$$

quelle que soit la valeur que l'on donne à la pondération α_{ko}, le second terme est toujours nul, le calcul du critère est possible.

ii) si $e_{k_o}^{mo} = 0$, alors $\alpha_{k_o}^{mo}$ n'est pas définie.

$$VC_4(PxQ, L, \alpha(\varepsilon), \varepsilon) = \sum_{k=1}^{K} \sum_{m=1}^{M} \{ \alpha_k^m.e_k^m - n_k.q_m.Log(1-\varepsilon_k^m)\}$$

$$= \sum_{k \neq k_o} \sum_{m \neq mo} \{\alpha_k^m.e_k^m - n_k.q_m.Log(1\varepsilon_k^m)\} + \{(\alpha_{k_o}^{mo}.e_{k_o}^{mo} - n_{ko}.q_{mo}.Log(1- \frac{e_{ko}^{mo}}{n_{ko}.q_{mo}})\} \qquad (5.9)$$

quelque soit la valeur que l'on donne à la pondération α_{k0}^{j0}, la deuxième partie de l'expression (5.9) est toujours nulle, le calcul du critère devient possible.

5. INTERPRETATION DES BONS RESULTATS OBTENUS PAR LA METHODE CROBIN SUR DES DONNEES SIMULEES

Afin de comparer les résultats obtenus par les deux algorithmes CROBIN et CROKI2 sur des données binaires, Govaert (1983) a réalisé un programme de simulation de données. Ce programme construit un tableau de données plus ou moins proche d'un tableau idéal structuré en blocs de 1 et de 0.

Pour cela, il faut fournir les nombres de classes des partitions en lignes et en colonnes, le nombre d'éléments de chacune des classes des deux partitions, le tableau des valeurs idéales de chaque bloc, et enfin la probabilité avec laquelle le tableau ainsi défini va s'approcher du tableau idéal. Chaque valeur est obtenue en faisant un tirage de

Bernoulli des deux nombres 0 et 1 avec les probabilités p et (1-p) si la valeur idéale est 0, (1-p) et p si la valeur idéale est 1. A la fin du programme, le tableau est permuté de manière aléatoire en ligne et en colonne.

Dans ce paragraphe, nous ne nous intéressons pas à la comparaison des résultats obtenus par CROBIN et CROKI2, nous nous limitons aux résultats obtenus uniquement par la méthode CROBIN. On reprend alors les deux exemples traités par Govaert (1983) :

Exemple 1 : données simulées 1

A l'aide du programme précédent, Govaert a simulé un tableau binaire de dimension 100x30. Il correspond à une probabilité de tirage de chaque valeur idéale de 0.9 et au tableau des valeurs idéales suivant :

1 0 1 1
0 1 0 0
1 1 1 0
0 1 0 1
1 0 0 0
0 0 1 1

Les deux partitions correspondant à la simulation sont les suivantes :
P = {(6, 9, 23, 33, 44, 58, 69, 79, 80, 81, 87, 90, 91, 98, 100), (3, 8, 11, 12, 18, 20, 27, 28, 29, 34, 39, 47, 50, 67, 70, 71, 77, 82, 88, 97), (4, 5, 25, 30, 32, 35, 37, 40, 46, 51, 56, 59, 63, 74, 84, 89, 93), (2, 7, 13, 16, 17, 21, 22, 24, 31, 41, 43, 48, 54, 55, 64, 68,76, 95), (1, 14, 15, 26, 38, 42, 45, 52, 53, 57, 65, 66, 75, 83, 85, 86, 96, 99), (10, 19, 36, 49, 60, 61, 62, 72, 73, 78, 92, 94)}.
Q = {(2, 5, 9, 12, 17, 20, 22, 23), (1, 4, 7, 8, 27, 28, 30), (3, 10, 13, 14, 15, 18, 19, 24, 29), (6, 11, 16, 21, 25, 26)}.

Exemple 2 : données simulées 2.

Les données sont simulées à partir des mêmes paramètres, à l'exception de la probabilité de tirage qui passe de 0.9 à 0.8. Les classes seront donc un peu plus floues. Le couple de partition ayant servi à la simulation est :

P = {(6, 9, 23, 33, 44, 58, 69, 79, 80, 81, 87, 90, 98, 100), (3, 8, 11, 12, 18, 20, 27, 28, 29, 34, 39, 47, 50, 67, 70, 71, 77, 82, 88, 97), (4, 5, 25, 30, 32, 35, 37, 40, 46, 51, 56, 59, 63, 74, 84, 89, 93), (2, 7, 13, 16, 17, 21, 22, 24, 31, 41, 43, 48, 54, 55, 64, 68, 76, 95), (1, 14, 15, 26, 38, 42, 45, 52, 53, 57, 65, 66, 75, 83, 85, 86, 96, 99), (10, 19, 36, 49, 60, 61, 62, 72, 73, 78, 92, 94)}.
Q = {(2, 5, 9, 12, 17, 20, 22, 23), (1, 4, 7, 8, 27, 28, 30), (3, 10, 13, 14, 15, 18, 19, 24, 29), (6, 11, 16, 21, 25, 26)}.

En appliquant le programme CROBIN sur les mêmes données (données simulées 1 et données simulées 2) avec le même nombre de classes, Govaert a obtenu les résultats ci-dessous, les tableaux situés à gauche correspondent à l'exemple 1, ceux de droite à l'exemple 2.

Tableau (P, Q)

	1	2	3	4
1	97	66	9	4
2	18	14	138	17
3	22	10	11	125
4	140	13	107	120
5	128	79	8	112
6	18	93	109	9

Tableau (P, Q)

	1	2	3	4
1	23	113	20	42
2	72	99	26	112
3	100	27	110	41
4	58	18	17	81
5	22	109	98	126
6	18	25	105	39

Tableau des valeurs idéales

	1	2	3	4
1	1	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	1	1
5	1	1	0	1
6	0	1	1	0

Tableau des valeurs idéales

	1	2	3	4
1	0	1	0	0
2	1	1	0	1
3	1	0	1	0
4	1	0	0	1
5	0	1	1	1
6	0	0	1	0

homogénéité par classe

	1	2	3	4
1	90	92	89	88
2	90	89	94	90
3	86	91	91	87
4	92	87	90	88
5	95	88	92	93
6	88	91	92	93

homogénéité par classe

	1	2	3	4
1	79	78	84	74
2	80	83	75	83
3	83	83	79	77
4	88	80	78	82
5	78	80	82	82
6	84	84	79	77

Le couple de partition (P, Q) obtenu par l'algorithme CROBIN sur l'exemple 1 est strictement identique à celui ayant servi à la simulation. Pour l'exemple 2 les résultats sont légèrement différents : en colonnes on obtient la même partition que celle de la simulation en lignes, nous remarquons qu'il y a uniquement deux éléments qui ont été classés de manière différente.

Govaert à donc supposé, à partir de ces bons résultats, que l'algorithme CROBIN est bien adapté au modèle qui lui a servi à simuler les données. Cette supposition devient une affirmation si on remarque que ce modèle n'est autre que celui qui nous a permi d'interpréter la méthode CROBIN au paragraphe 2 de ce chapitre ; nous justifions ainsi la bonne qualité de ces résultats.

6. CONCLUSION

Après avoir vu que la méthode CROBIN peut s'interpréter comme l'approche classification associé à un mélange de distribution de Bernoulli avec le même paramètre pour toutes les classes. La généralisation de ce modèle en considérant différents paramètre permet de proposer un nouvel algorithme de classification croisée binaire utilisant des distances adaptatives binaires appelé algorithme **CROBIN adaptatif** qui n'est autre que l'ancien algorithme auquel s'ajoute trois variantes pour la distance qui correspondent respectivement aux mélanges étudiés précédemment.

Le lien qui existe entre la méthode de classification croisée et le modèle probabiliste permet aussi d'expliquer les bons résultats que Govaert (1983) avait obtenu à l'aide de cet algorithme sur des données simulées qui justement suivaient ce modèle. Il resterait à tester les nouveaux algorithmes utilisant ces distances adaptatives sur des données simulées qui elles aussi suivent à chaque fois une des variantes du modèle.

CHAPITRE 6

CLASSIFICATION CROISEE ET MODELES SUR DONNEES QUANTITATIVES

INTRODUCTION

Contrairement aux tableaux binaires sur lesquels portait le chapitre précédent, les tableaux de mesures présentent une **dissymétrie**. Les lignes sont formées "d'unités ayant un caractère répétitif" : ce sont les individus. Les colonnes peuvent être plus hétérogènes : ce sont les variables. Ces variables correspondent à des mesures. On les appelle en général variables quantitatives.

La méthode de reconnaissance des composants d'un mélange "croisé" développée dans le chapitre quatre, nous permet d'interpréter des méthodes de classification croisée qui traitent les deux ensembles **I** et **J** de manière identique. Nous avons alors remarquer que cette méthode s'appliquait parfaitement aux tableaux binaires car les deux ensembles décrivant ces tableaux sont considérés comme étant de même nature (cf chapitre cinq).

Les tableaux de mesures posent un problème, car comme nous venons de le signaler les ensembles décrivant ces tableaux sont de nature différente ; pour pouvoir appliquer la méthode de reconnaissance des composants d'un mélange "croisé" développée dans le chapitre quatre à des tableaux de mesures, nous nous sommes efforcés dans tout ce chapitre de considérer avec un peu " **d'abus**" que l'ensemble **J** des p variables quantitatives constitue un échantillon de taille p d'une certaine population Ω' (où Ω' est l'ensemble de toutes les variables possibles). Nous pourrons alors appliquer tous les résultats obtenus au chapitre quatre sur les tableaux de mesures.

La méthode CROEUC (Govaert 1983) a été développée pour ce type de tableaux. Nous nous intéressons dans ce chapitre à l'interprétation de cette méthode en termes de modèle.

La méthode CROEUC correspond à la classification croisée de tableaux de mesure optimisant un critère défini à l'aide d'une distance Euclidienne ; l'étude détaillée de cette méthode fera l'objet du premier paragraphe de ce chapitre où l'on précisera le critère d'inertie intraclasse optimisé par cette méthode. Nous montrons dans le deuxième paragraphe que la méthode précédente peut s'interpréter comme l'approche classification associée à un mélange de lois gaussiennes unidimensionnelles ; suivant différentes hypothèses sur les variances des différents composants du mélange, on retrouve à chaque fois le critère d'inertie utilisé pour la classification de données quantitatives.

1. LA METHODE CROEUC

L'objectif de la méthode de classification croisée sur données quantitatives est la recherche d'un couple de partitions, tels que la " perte d'information " dûe au regroupement soit minimale ; c'est-à-dire telle que la différence entre l'information apportée par le tableau initial et celle apportée par le tableau obtenu après regroupement soit minimale.

L'algorithme proposé par G.Govaert (1983) pour la classification simultanée adapté à ce type de données utilise deux variantes voisines l'une de l'autre et basées toutes les deux sur la méthode des Nuées Dynamiques. Elles sont encore appelées " algorithme des Centres Mobiles " (Benzécri 1973) qui représente l'un des algorithmes de partitionnement le plus utilisé ; sa grande popularité tient à sa simplicité et à sa rapidité. Ainsi, il est particulièrement efficace pour classer de très grand tableaux de données (Lebart, Morineau, Tabard, 1977). Ce type d'algorithme, ou une classe est représentée par son centre de gravité, à été étudié par différents auteurs (Thorndike 1953, Bonner 1964, Forgy 1965, Ball et Hall 1965, Mac Queen 1967).

1.1. NOTATIONS

Soit I un ensemble de n individus décrits par p variables ou caractères quantitatifs. Les données sont rangées dans un tableau de description X à n lignes et p colonnes.

$$\mathbf{X} = (x_i^j) \qquad i \in I \quad \text{et} \quad j \in J$$

où x_i^j est la valeur de la variable j pour l'individu i .

D'autre part on supposera que les individus sont munis de poids p_i tels que :

$$\sum_{i \in I} p_i = 1$$

Associons à chaque individu i le vecteur $x_i = (x_i^1, ..., x_i^p)$ de $\mathbf{R}^p$ correspondant à la ligne i du tableau **X**. L'ensemble des x_i munis des pondérations p_i forme un nuage N(I) contenu dans $\mathbf{R}^p$.
De la même façon, à chaque variable j est associé le vecteur $x^j = (x_1^j, ..., x_n^j)$ correspondant à la colonne j du tableau **X**. L'ensemble des x^j munis de pondérations q_j forment un nuage N(J) contenu dans $\mathbf{R}^n$.

Pour mesurer la proximité entre individus, on munit l'espace des individus de la métrique quadratique définie par la matrice diagonale de terme général q_j correspondant à l'importance donnée à la variable j.
On a donc :

$$d^2 (x_i, x_{i'}) = \sum_{j=1}^{p} q_j (x_i^j - x_{i'}^j)^2$$

de même pour mesurer la proximité entre variables, on munit l'espace des variables de la métrique des poids Dp.

$$\text{On a :} \qquad d^2 (x^j, x^{j'}) = \sum_{i=1}^{n} p_i (x_i^j - x_i^{j'})^2$$

1.2. LE PROBLEME

La méthode CROEUC fournit une solution locale au problème d'optimisation suivant :

Il s'agit de trouver une partition P de I en K classes, une partition Q de J en M classes tel que le critère d'inertie intraclasse suivant :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} p_i q_j (x_i^j - g_k^m)^2 \qquad (6.1)$$

soit minimal.

g_k^m : étant le centre de gravité de la classe $P_k x Q^m$.

1.3. L'ALGORITHME

On utilise le même algorithme que celui proposé dans le chapitre quatre au paragraphe 1.3. On peut préciser cet algorithme dans le cas de la méthode CROEUC en associant à chacune des trois étapes les fonctions f et h d'affectation et la fonction g de représentation.

La fonction de représentation g :

Cette fonction permet de déterminer les K.M noyaux minimisant le critère W(PxQ), g(PxQ)). On peut facilement voir que ces noyaux sont les centres de gravité des classes. Si nous notons $\{(g_k^m) ; k = 1, ..., K \text{ et } m = 1, ..., M \}$ l'ensemble de ces centres, on a :

$$g_k^m = \frac{1}{\left(\sum_{i \in P_k} \sum_{j \in Q^m} p_i . q_j \right)} \sum_{i \in P_k} \sum_{j \in Q^m} p_i . q_j \, x_i^j$$

La fonction d'affectation f :

Cette fonction minimise le critère W(f(PxQ), L) en affectant chaque individu à la classe P_k du noyau g_k de laquelle il est le plus proche (au sens de la distance euclidienne) en supposant que la partition Q et le noyau L sont fixés. Considérons le nuage des individus obtenus après regroupement des variables selon la partition Q ; il est définie par :

$$\{ x_i = (x_i^1 , ..., x_i^M), i \in I \} \quad \text{où} \quad x_i^m = \frac{1}{\sum_{j \in Q^m} q_j} \sum_{j \in Q^m} q_j \, x_i^j$$

Considérons l'algorithme des Nuées Dynamiques (Diday 1979) suivant :
Le tableau de données est le tableau X(I, Q) défini par les x_i^m, les individus sont en lignes et les variables en colonnes. On a donc un ensemble de n éléments et M variables.

-Les vecteurs sont les lignes du tableau X(I, Q).

-Les pondérations des individus sont toujours les p_i.

- Les noyaux sont de la forme $(g_k^1, g_k^2, ..., g_k^M)$ qui représentent le centre de gravité de la classe P_k.

-La métrique associée à ce nouveau tableau X(I, Q) est définie par la matrice diagonale de terme générale :

$$q'_m \quad \text{où} \quad q'_m = \sum_{j \in Q^m} q_j \ .$$

La fonction d'affectation f range alors chaque élément i à la classe P_k du noyau de laquelle il est le plus prés.

La fonction d'affectation h :

Cette fonction minimise le critère W(h(PxQ), L) en supposant que la partition P et le noyau L sont fixés. On applique la méthode des Nuées Dynamiques sur le tableau X(P, J) défini par les x_k^j. Les individus sont en colonnes et les variables en lignes. On obtient un ensemble de p éléments et K variables. La fonction d'affectation h range alors chaque élément $j \in J$ à la classe Q^m du noyau $g^m = (g_1^m, g_2^m, ..., g_K^m)$ de laquelle il est le plus prés au sens de la métrique diagonale dont la diagonale à pour terme général $p'_k = \sum_{i \in P_k} p_i$.

1.4. CAS PARTICULIER

On reprend l'expression du critère optimisé par la méthode CROEUC :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} p_i q_j (x_i^j - g_k^m)^2$$

où g_k^m est le centre de gravité de la classe $P_k x Q^m$.

posons $\quad p_i = \frac{1}{n} \quad q_j = \frac{1}{p} \quad \forall i \in I \text{ et } \forall j \in J$

Le critère (6.1) s'écrit :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \frac{1}{n.p} (x_i^j - g_k^m)^2 \qquad (6.2)$$

avec $\quad g_k^m = \frac{1}{n_k . q_m} \sum_{i \in P_k} \sum_{j \in Q^m} x_i^j \quad$ où $\quad n_k = \text{Card}(P_k) \quad$ et $\quad q_m = \text{Card}(Q^m)$

Comme le terme $\frac{1}{n.p} > 0$, la minimisation du critère (6.2) revient à la minimisation du critère (6.3) suivant :

$$W(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - g_k^m)^2 \qquad (6.3)$$

Le critère (6.3) correspond à la version la plus simple et la plus utilisé de la méthode CROEUC.

2. MODELE ASSOCIE AUX DONNEES QUANTITATIVES

En suivant l'approche classification proposée dans le chapitre quatre pour identifier un mélange dans le cas de la classification croisée, on propose le modèle suivant :

2.1. LA FORMULE GENERALE

On considère dans ce modèle que les données initiales forment un échantillon de taille (n, p) d'une variable aléatoire à valeurs dans **R** dont la distribution de probabilité f est toujours définie par (4.3) et (4.4).
En suivant l'approche " classification", rappelée dans le paragraphe 3 du chapitre 4 pour identifier un mélange, on se ramène à la maximisation du critère de vraisemblance classifiante VC(PxQ, L) défini par (4.7). On peut alors utiliser le même algorithme que celui proposé dans le paragraphe 1.3 du chapitre 4, à partir d'un couple de partition (P^0, Q^0) en K et M classes de l'échantillon en utilisant alternativement les trois étapes de l'algorithme jusqu'à l'obtention d'une partition stable.

2.2. CHOIX DE LA FAMILLE DE DISTRIBUTION

On considère dans ce modèle que les données du tableau proviennent d'un mélange de lois gaussiennes unidimensionnelles, les paramètres λ_k^m pour $k = 1, ..., K$ et $m = 1,..., M$ s'écrivent $\lambda_k^m = (\mu_k^m , \sigma_k^m)$ où :

μ_k^m : espérance du composant (k, m)

σ_k^m : écart-type du composant (k, m).

Soit $V_k^m = (\sigma_k^m)^2$ qui représente la variance du composant (k, m) .

La fonction de densité associée à la variable aléatoire **Z** est définie par :

$$f(x / \lambda_k^m) = (2\pi)^{-1/2}. (\sigma_k^m)^{-1}. \exp \; -\frac{1}{2} \left[\frac{(x - \mu_k^m)}{\sigma_k^m}\right]^2$$

$$= (2\pi \, v_k^m)^{-1/2}. \; \exp \; - \frac{(x - \mu_k^m)^2}{2.V_k^m}$$

Les paramètres à estimer sont donc les μ_k^m et les V_k^m .

Le critère de **vraisemblance classifiante** s'écrit :

$$VC\,(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \text{Log}\left\{ (2\pi V_k^m)^{-1/2} . \exp\left[- \frac{(x_i^j - \mu_k^m)^2}{2.V_k^m} \right] \right\}$$

$$= -\frac{n.p}{2} \text{Log}\, 2\pi - \frac{1}{2} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \left\{ \frac{(x_i^j - \mu_k^m)^2}{V_k^m} + \text{Log}\, V_k^m \right\}$$

On remarque que le premier terme de cette expression est constant ; la maximisation du critère VC (PxQ, L) revient donc à la minimisation de l'expression suivante :

$$C\,(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \left\{ \frac{(x_i^j - \mu_k^m)^2}{V_k^m} + \text{Log}\, V_k^m \right\} \qquad (6.4)$$

Pour optimiser ce critère, on va devoir étudier trois cas différents en faisant plusieurs hypothèses sur les variances V_k^m des composants du mélange.

Premier cas

On suppose dans ce premier cas que les variances de tout les composants du mélange sont les mêmes et supposées connues .

$V_k^m = V \quad \forall\, k = 1, ..., K \text{ et } m = 1, ..., M$

Le critère à minimiser devient alors :

$$C_1\,(PxQ, L) = \frac{1}{V} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - \mu_k^m)^2 + n.p\, \text{Log}\, V \qquad (6.5)$$

Comme V> 0, le critère à minimiser se résume à :

$$C_1'\,(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - \mu_k^m)^2 \qquad (6.6)$$

Les partitions étant fixées, l'estimateur du maximum de vraisemblance classifiante de μ_k^m est la moyenne de la classe $P_k x Q^m$ (son centre de gravité).

$$\mu_k^m = g_k^m = \frac{1}{p_k.q_m} \sum_{i \in P_k} \sum_{j \in Q^m} x_i^j$$

Le critère prend alors la forme :

$$W\,(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - g_k^m)^2$$

Ce critère correspond bien au critère minimisé par la méthode CROEUC dans le cas le plus simple (expression 6.3).

Deuxième cas

Cette fois-ci les variances de tous les composants du mélange sont les mêmes et supposées inconnues .

$$C_2\,(PxQ, L) = \frac{1}{V} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - \mu_k^m)^2 + n.p\ Log\ V \qquad (6.7)$$

Notons par W l'expression $\sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - \mu_k^m)^2$

Le critère à minimiser s'écrit alors sous la forme :

$$C_2\,(PxQ, L) = \frac{1}{V}\,.W + n.p\ LogV$$

$$\frac{\partial\, C(PxQ, L)}{\partial\, V} = \frac{\partial}{\partial\, V}\left(\frac{1}{V}\,.W + n.p\ Log\ V\right) = 0$$

d'où $V = \frac{W}{n.p}$. Celui-ci représente l'estimateur du maximum de vraisemblance de V ; si on remplace maintenant la valeur de V dans l'expression du critère à minimiser on obtient :

$$C_2\,(PxQ, L) = n.p + n.p\ Log\ W - n.p\ Log\ n.p$$

Minimiser $C_2\,(PxQ, L)$ revient aussi à minimiser l'expression :

$$C_2'\,(PxQ, L) = Log\ W \qquad (6.8)$$

Il est facile de vérifier que la minimisation de l'expression (6.8) est équivalente à la minimisation de l'expression suivante :

$$W(PxQ, L) = W = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - \mu_k^m)^2$$

Là aussi l'estimateur de $\mu_k^m = g_k^m$ n'est autre que le centre de gravité de la classe $P_k x Q^m$; on se retrouve exactement dans la même situation que dans le cas précédent.

Troisième cas

On suppose cette fois que pour chaque composante du mélange, les variances sont différentes entre elles et inconnues.

$$C_3(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i \in P_k} \sum_{j \in Q^m} \left\{ \frac{(x_i^j - \mu_k^m)^2}{V_k^m} + \text{Log } V_k^m \right\} \qquad (6.9)$$

$$\text{Posons } C(PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} L_k^m$$

$$\text{où } L_k^m = \sum_{i \in P_k} \sum_{j \in Q^m} \left\{ \frac{(x_i^j - \mu_k^m)^2}{V_k^m} + \text{Log } V_k^m \right\}$$

A partition PxQ fixée la recherche des paramètres V_k^m et μ_k^m se fait par la maximisation de L_k^m .

$$\text{Posons } W_k^m = \sum_{i \in P_k} \sum_{j \in Q^m} (x_i^j - \mu_k^m)^2 \text{ d'ou } L_k^m = \frac{W_k^m}{V_k^m} + n_k.q_m \text{ Log } V_k^m.$$

$$\frac{\partial L_k^m}{\partial V_k^m} = \frac{\partial}{\partial V_k^m}\left(\frac{W_k^m}{V_k^m} + n_k.q_m \text{ Log } V_k^m \right) = 0 \quad \text{d'ou } V_k^m = \frac{W_k^m}{n_k.q_m}$$

$$\frac{\partial L_k^m}{\partial \mu_k^m} = \frac{\partial}{\partial \mu_k^m}\left(n_k.q_m + n_k.q_m \text{ Log } \frac{W_k^m}{n_k.q_m} \right) = 0$$

on obtient :

$$\mu_k^m = g_k^m = \frac{1}{p_k.q_m} \sum_{i \in P_k} \sum_{j \in Q^m} x_i^j$$

Le critère se met alors sous la forme :

$$C_3 (PxQ, L) = n.p + \sum_{k=1}^{K} \sum_{m=1}^{M} n_k.q_m \, Log \, V_k^m$$

Minimiser C_3 (PxQ, L) revient à minimiser l'expression :

$$C_3' (PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} n_k.q_m \, Log \, V_k^m \qquad (6.10)$$

Remarque

On sait que le critère utilisé dans l'algorithme des Nuées Dynamiques avec k distances adaptatives (Govaert 1975) dans le cas de la classification simple élaboré pour permettre de reconnaître des classes de formes différentes s'écrit :

$$W (P, L) = \sum_{k=1}^{K} |W_k|^{1/p} = \sum_{k=1}^{K} n_k \, |V_k|^{1/p} \qquad (6.11)$$

V_k : Matrice de variances associée à la classe P_k

Ce critère est très proche du critère : $W(P, L) = \sum_{k=1}^{K} n_k \, Log \, |V_k|$ (6.12)

En pratique, il s'avère que les deux méthodes (6.11) et (6.12) donnent des résultats quasi-identiques (Govaert 1975).

Il est facile de vérifier que l'expression du critère donnée par la formule (6.3), où les métriques des poids sont toutes les deux de la forme $\gamma.I_d$, où I_d est la matrice identité et γ est un réel, peut s'écrire sous la forme suivante :

$$W (PxQ, L) = \sum_{k=1}^{K} \sum_{m=1}^{M} W_k^m = \sum_{k=1}^{K} \sum_{m=1}^{M} n_k.q_m \, V_k^m \qquad (6.13)$$

Nous remarquons que lorsqu'on fixe l'une ou l'autre des deux partitions P et Q, les deux critères (6.10) et (6.13) se comportent respectivement comme les critères (6.12) et (6.11). Nous pouvons alors supposer que les critères (6.10) et (6.13) peuvent à leur tour donner des résultats quasi-identique. Il reste tout de même à tester ces deux méthodes.

3. CONCLUSION

Nous venons de voir dans ce chapitre comment les algorithmes de partitionnements simultanés utilisant un critère d'inertie peuvent se présenter comme des méthodes pour identifier un mélange gaussien à l'aide d'une classification. Suivant différentes hypothèses sur les variances des différents composants, on retrouve à chaque fois le critère d'inertie utilisé pour la classification de données quantitatives. Nous avons aussi remarquer que dans les deux premiers cas étudiés où l'on suppose que les variances sont toutes les mêmes pour tout les composants, conduisent au même critère qui correspond à la version la plus simple et la plus utilisée de la méthode CROEUC.

CONCLUSION

CONCLUSION

Les liens existant entre la notion de mélange et la classification automatique nous ont amenés à conclure que si la théorie des mélanges livre à l'analyste un ensemble de modèles mathématiques imitant les mécanismes de fonctionnement de phénomènes réels hypothétiques ou de systèmes de nature stochastique, l'une des principales vocations de la classification automatique est le choix fondé, au sein d'un ensemble de modèles admissibles, du modèle qui répond le mieux (dans un certain sens) aux données initiales caractérisant le comportement réel du système analysé.

Donc, la meilleure méthode de classification de données est un problème dont la résolution dépend en premier lieu de la connaissance de modèles convenables et de l'habileté de l'analyste à "les ajuster" à la réalité étudiée, et au besoin, à construire un nouveau modèle reflétant les traits spécifiques du problème étudié. Ces traits se résument dans la qualité de la partition obtenue qui est mesurée par une fonction qu'on appelle souvent critère. Ce critère ne fait aucune référence à la notion de modèle, n'a donc jamais été justifié en termes de modèle probabiliste.

La méthode de reconnaissance des compopsants d'un mélange Shroeder (1974) à permi de voir que, souvent le passage d'un critère métrique à un critère probabiliste peut apporter une argumentation concernant le choix du critère et de la métrique utilisée. Cette méthode a été utile pour beaucoup de chercheurs du même domaine qui se sont servi de cet algorithme pour apporter un éclairage nouveau de quelques critères de classification. C'est en utilisant cette méthode que l'on à pu nous aussi dans ce travail apporter un éclairage nouveau sur les deux méthodes **MNDQAL** et **MNDDIJ** destinées toutes deux à la classification de tableaux décrits par des variables qualitatives. Nous avons alors montré que ces méthodes s'interprètent en termes de modèles probabilistes et l'on à justifier du choix des critères optimisés par ces méthodes.

En restant toujours dans le cadre de la notion de modèle, nous avons repris le modèle proposé par Govaert pour les données binaires pour faire une étude comparative entre les algorithmes adaptatifs et les algorithmes non adaptatifs. Ce modèle à permis de proposer de nouveaux algorithmes utilisant des distances adaptatives de type L_1 (**MNDBIN adaptatif**). Comme nous l'avons déjà vu au chapitre trois, ces algorithmes permettent d'améliorer la qualité de la partition et prennent l'ancien algorithme non adaptatif en

défaut, ce dernier ne tient pas compte des pondérations des variables, ce qui statistiquement peut être vu comme un défaut.

Nous avons ensuite **généralisé** les liens qui existent entre les modèles probabilistes et les méthodes de classification automatique au cas où les données mettent en jeu **deux ensembles**, dans le but d'interpréter les méthodes de classification croisée et d'établir un lien entre ces méthodes et la notion de modèle. La méthode de reconnaissance des composants d'un mélange que nous avons proposé dans le chapitre quatre, peut être considérée comme une extension de celle proposée par Schroeder (1974) et de la représentation de Celeux (1988).
Comme le principe de la classification croisée consiste à subdiviser la population des individus et celle des variables en un petit nombre de groupes ou classes homogènes sans faire aucune distinction entre les individus et les variables à classer, le problème des mélanges "croisé" à été résolu de façon analogue, c'est à dire que l'on à traité les deux ensembles décrivant le tableau de la même façon. Ainsi nous avons pu estimer les paramètres d'un modèle de mélange "croisé" par la méthode d'estimation du maximum de vraisemblance classifiante.

La méthode proposée nous à permis d'interpréter deux méthodes de classification croisée très utilisées actuellement et intégrées au logiciel Sicla, la méthode **CROEUC** (Classification croisée d'un tableau de mesure utilisant la métrique Euclidienne) et la méthode **CROBIN** (Classification croisée d'un tableau binaire utilisant la distance L_1).
Nous avons remarqué que le modèle que l'on a proposé pour la méthode CROBIN coincide parfaitement avec celui qui a servi la simulation des données binaires sur lesquelles Govaert (1983) avait obtenu de très bons résultats en utilisant la méthode CROBIN. Nous justifions ainsi la bonne qualité de ces résultats. De plus ce modèle par son extension nous à permis de proposer de nouveaux algorithmes de classification croisée utilisant des distances adaptatives de type L_1 (**CROBIN adaptatif**). Il serait donc intéressant de tester ces nouveaux algorithmes sur des données simulées qui elles aussi suivent à chaque fois une des variantes du modèle. Ce qui semble également important est d'essayer d'établir les liens qui existent entre les critères métriques et les critères probabilistes dans le cas de la classification croisée. Cette étude nous permettra peut être d'interpréter d'autres méthodes de classification croisée telle que la méthode CROKI2 (classification croisée d'un tableau de contigence) et la méthode CROMUL (classification croisée d'un questionnaire).

ANNEXE 1

COMMANDE : MNDBIN <> nuees dynamiques sur variables binaires

variables selectionnees :

VA 1 VA 2 VA 3 VA 4

Valeurs des Parametres

nombre de classes	2
mode d'initialisation	1
nombre de tirages au hasard	20
classe vide possible	oui
sortie uniquement du meilleur resultat	oui
impression tableau initial reordonne	non
choix de la distance	1

Valeur du critere obtenu a chaque tirage :

375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
627.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
491.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
627.0000000	(convergence atteinte)
491.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
375.0000000	(convergence atteinte)
627.0000000	(convergence atteinte)

ANALYSE DU MEILLEUR TIRAGE

valeur du critere obtenu 375.0000000

Partition

classe 1 : 276 elements

i001 i002 i003 i004 i005 i006 i007 i008 i009 i010 i011 i012 i013 i014 i015 i016
i017 i018 i019 i020 i021 i022 i023 i024 i025 i026 i027 i028 i029 i030 i031 i032
i033 i034 i035 i036 i037 i038 i039 i040 i041 i042 i043 i044 i045 i046 i047 i048
i049 i050 i051 i052 i053 i054 i055 i056 i057 i058 i059 i060 i061 i062 i101 i102
i103 i104 i105 i155 i156 i157 i158 i159 i160 i161 i162 i163 i164 i165 i166 i167
i168 i169 i170 i171 i172 i173 i174 i175 i176 i177 i178 i179 i180 i181 i182 i183
i184 i185 i186 i187 i188 i189 i190 i191 i192 i193 i194 i195 i196 i197 i198 i199
i200 i201 i202 i203 i204 i205 i206 i207 i208 i209 i210 i211 i212 i213 i214 i215
i216 i217 i218 i219 i220 i221 i222 i223 i224 i225 i226 i227 i228 i229 i230 i231
i232 i233 i234 i235 i236 i237 i238 i239 i240 i241 i242 i243 i244 i245 i246 i247
i248 i249 i250 i251 i252 i253 i254 i255 i256 i257 i258 i259 i260 i261 i262 i263
i264 i265 i266 i267 i268 i269 i270 i271 i272 i273 i274 i275 i276 i277 i278 i279
i280 i281 i282 i283 i284 i285 i286 i287 i288 i289 i290 i291 i292 i293 i294 i295
i296 i297 i298 i299 i300 i301 i302 i303 i304 i305 i306 i307 i308 i309 i310 i311
i312 i313 i314 i315 i316 i317 i318 i319 i320 i321 i322 i323 i324 i325 i326 i327
i328 i329 i330 i331 i332 i333 i334 i335 i336 i337 i338 i339 i340 i341 i342 i343
i344 i345 i346 i347 i348 i349 i385 i386 i387 i388 i389 i390 i391 i392 i393 i394
i395 i396 i397 i398

classe 2 : 136 elements

i063 i064 i065 i066 i067 i068 i069 i070 i071 i072 i073 i074 i075 i076 i077 i078
i079 i080 i081 i082 i083 i084 i085 i086 i087 i088 i089 i090 i091 i092 i093 i094
i095 i096 i097 i098 i099 i100 i106 i107 i108 i109 i110 i111 i112 i113 i114 i115
i116 i117 i118 i119 i120 i121 i122 i123 i124 i125 i126 i127 i128 i129 i130 i131
i132 i133 i134 i135 i136 i137 i138 i139 i140 i141 i142 i143 i144 i145 i146 i147
i148 i149 i150 i151 i152 i153 i154 i350 i351 i352 i353 i354 i355 i356 i357 i358
i359 i360 i361 i362 i363 i364 i365 i366 i367 i368 i369 i370 i371 i372 i373 i374
i375 i376 i377 i378 i379 i380 i381 i382 i383 i384 i399 i400 i401 i402 i403 i404
i405 i406 i407 i408 i409 i410 i411 i412

TABLEAU (P,J)

	V A 1	V A 2	V A 3	V A 4
1	196	229	208	199
2	0	73	0	96

TABLEAU DES VALEURS IDEALES

	VVVV AAAA 1234
1	1111
2	1 1

HOMOGENEITE PAR CLASSE

	V A 1	V A 2	V A 3	V A 4
1	71	82	75	72
2	100	53	100	70

TABLEAU DES ECARTS

	V A 1	V A 2	V A 3	V A 4
1	80	47	68	77
2	0	63	0	40

TABLEAU DES PARAMETRES DE LA LOI DE BERNOUILLI

	V A 1	V A 2	V A 3	V A 4
1	0.27	0.27	0.27	0.27
2	0.27	0.27	0.27	0.27

TABLEAU DES COEFFICIENTS DE PONDERATION

	V A 1	V A 2	V A 3	V A 4
1	1.0	1.0	1.0	1.0
2	1.0	1.0	1.0	1.0

Ecriture du constituant No 1 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

```
COMMANDE :  MNDBIN <> nuees dynamiques sur variables binaires


variables selectionnees :

VA 1 VA 2 VA 3 VA 4


Valeurs des Parametres


nombre de classes                          2
mode d'initialisation                      2
nombre de tirages au hasard                1
classe vide possible                       oui
sortie uniquement du meilleur resultat     oui
impression tableau initial reordonne       non
choix de la distance                       2


Lecture du constituant No     1 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN


Valeur du critere obtenu a chaque tirage :

828.2286000    (convergence atteinte)


ANALYSE DU MEILLEUR TIRAGE

valeur du critere obtenu     828.2286000


Partition

classe  1 :    208 elements
-----------
i001 i002 i003 i004 i005 i006 i007 i027 i028 i029 i030 i031 i032 i041 i042 i043
i044 i045 i046 i047 i048 i049 i050 i051 i052 i053 i054 i055 i056 i057 i058 i059
i060 i061 i062 i101 i102 i103 i104 i105 i155 i156 i157 i158 i159 i160 i161 i162
i163 i164 i165 i166 i167 i168 i169 i170 i171 i172 i173 i174 i175 i176 i177 i178
i179 i180 i181 i182 i183 i184 i185 i186 i187 i188 i189 i190 i191 i192 i193 i194
i195 i196 i197 i198 i199 i200 i201 i202 i203 i204 i205 i206 i207 i208 i209 i210
i211 i212 i213 i214 i215 i216 i217 i218 i219 i220 i221 i222 i223 i224 i225 i226
i227 i228 i229 i230 i231 i232 i233 i234 i235 i236 i237 i238 i239 i240 i241 i242
i243 i244 i245 i246 i247 i248 i249 i250 i251 i252 i253 i254 i255 i256 i257 i258
i259 i260 i261 i262 i297 i298 i299 i300 i301 i302 i303 i311 i312 i313 i314 i315
i316 i317 i318 i319 i320 i321 i322 i323 i324 i325 i326 i327 i328 i329 i330 i331
i332 i333 i334 i335 i336 i337 i338 i339 i340 i341 i342 i343 i344 i345 i346 i347
i348 i349 i385 i386 i387 i388 i389 i390 i391 i392 i393 i394 i395 i396 i397 i398

classe  2 :    204 elements
-----------
i008 i009 i010 i011 i012 i013 i014 i015 i016 i017 i018 i019 i020 i021 i022 i023
i024 i025 i026 i033 i034 i035 i036 i037 i038 i039 i040 i063 i064 i065 i066 i067
i068 i069 i070 i071 i072 i073 i074 i075 i076 i077 i078 i079 i080 i081 i082 i083
i084 i085 i086 i087 i088 i089 i090 i091 i092 i093 i094 i095 i096 i097 i098 i099
i100 i106 i107 i108 i109 i110 i111 i112 i113 i114 i115 i116 i117 i118 i119 i120
i121 i122 i123 i124 i125 i126 i127 i128 i129 i130 i131 i132 i133 i134 i135 i136
i137 i138 i139 i140 i141 i142 i143 i144 i145 i146 i147 i148 i149 i150 i151 i152
i153 i154 i263 i264 i265 i266 i267 i268 i269 i270 i271 i272 i273 i274 i275 i276
i277 i278 i279 i280 i281 i282 i283 i284 i285 i286 i287 i288 i289 i290 i291 i292
i293 i294 i295 i296 i304 i305 i306 i307 i308 i309 i310 i350 i351 i352 i353 i354
i355 i356 i357 i358 i359 i360 i361 i362 i363 i364 i365 i366 i367 i368 i369 i370
i371 i372 i373 i374 i375 i376 i377 i378 i379 i380 i381 i382 i383 i384 i399 i400
i401 i402 i403 i404 i405 i406 i407 i408 i409 i410 i411 i412
```

TABLEAU (P,J)

	V A 1	V A 2	V A 3	V A 4
1	128	176	208	154
2	68	126	0	141

TABLEAU DES VALEURS IDEALES

	VVVV AAAA 1234
1	1111
2	1 1

HOMOGENEITE PAR CLASSE

	V A 1	V A 2	V A 3	V A 4
1	61	84	100	74
2	66	61	100	69

TABLEAU DES ECARTS

	V A 1	V A 2	V A 3	V A 4
1	80	32	0	54
2	68	78	0	63

TABLEAU DES PARAMETRES DE LA LOI DE BERNOUILLI

	V A 1	V A 2	V A 3	V A 4
1	0.36	0.27	0.17	0.28
2	0.36	0.27	0.17	0.28

TABLEAU DES COEFFICIENTS DE PONDERATION

	V A 1	V A 2	V A 3	V A 4
1	0.6	1.0	1.6	0.9
2	0.6	1.0	1.6	0.9

Ecriture du constituant No 2 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

COMMANDE : MNDBIN <> nuees dynamiques sur variables binaires

variables selectionnees :

VA 1 VA 2 VA 3 VA 4

Valeurs des Parametres

nombre de classes	2
mode d'initialisation	2
nombre de tirages au hasard	1
classe vide possible	oui
sortie uniquement du meilleur resultat	oui
impression tableau initial reordonne	non
choix de la distance	3

Lecture du constituant No 1 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

Valeur du critere obtenu a chaque tirage :

784.8664000 (convergence atteinte)

ANALYSE DU MEILLEUR TIRAGE

valeur du critere obtenu 784.8664000

Partition

classe 1 : 242 elements

i001 i002 i003 i004 i005 i006 i007 i008 i009 i010 i011 i012 i013 i014 i015 i016
i017 i018 i019 i020 i021 i022 i023 i024 i025 i026 i027 i028 i029 i030 i031 i032
i041 i042 i043 i044 i045 i046 i047 i048 i049 i050 i051 i052 i053 i054 i055 i056
i057 i058 i059 i060 i061 i062 i155 i156 i157 i158 i159 i160 i161 i162 i163 i164
i165 i166 i167 i168 i169 i170 i171 i172 i173 i174 i175 i176 i177 i178 i179 i180
i181 i182 i183 i184 i185 i186 i187 i188 i189 i190 i191 i192 i193 i194 i195 i196
i197 i198 i199 i200 i201 i202 i203 i204 i205 i206 i207 i208 i209 i210 i211 i212
i213 i214 i215 i216 i217 i218 i219 i220 i221 i222 i223 i224 i225 i226 i227 i228
i229 i230 i231 i232 i233 i234 i235 i236 i237 i238 i239 i240 i241 i242 i243 i244
i245 i246 i247 i248 i249 i250 i251 i252 i253 i254 i255 i256 i257 i258 i259 i260
i261 i262 i263 i264 i265 i266 i267 i268 i269 i270 i271 i272 i273 i274 i275 i276
i277 i278 i279 i280 i281 i282 i283 i284 i285 i286 i287 i288 i289 i290 i291 i292
i293 i294 i295 i296 i297 i298 i299 i300 i301 i302 i303 i311 i312 i313 i314 i315
i316 i317 i318 i319 i320 i321 i322 i323 i324 i325 i326 i327 i328 i329 i330 i331
i332 i333 i334 i335 i336 i337 i338 i339 i340 i341 i342 i343 i344 i345 i346 i347
i348 i349

classe 2 : 170 elements

i033 i034 i035 i036 i037 i038 i039 i040 i063 i064 i065 i066 i067 i068 i069 i070
i071 i072 i073 i074 i075 i076 i077 i078 i079 i080 i081 i082 i083 i084 i085 i086
i087 i088 i089 i090 i091 i092 i093 i094 i095 i096 i097 i098 i099 i100 i101 i102
i103 i104 i105 i106 i107 i108 i109 i110 i111 i112 i113 i114 i115 i116 i117 i118
i119 i120 i121 i122 i123 i124 i125 i126 i127 i128 i129 i130 i131 i132 i133 i134
i135 i136 i137 i138 i139 i140 i141 i142 i143 i144 i145 i146 i147 i148 i149 i150
i151 i152 i153 i154 i304 i305 i306 i307 i308 i309 i310 i350 i351 i352 i353 i354
i355 i356 i357 i358 i359 i360 i361 i362 i363 i364 i365 i366 i367 i368 i369 i370
i371 i372 i373 i374 i375 i376 i377 i378 i379 i380 i381 i382 i383 i384 i385 i386
i387 i388 i389 i390 i391 i392 i393 i394 i395 i396 i397 i398 i399 i400 i401 i402
i403 i404 i405 i406 i407 i408 i409 i410 i411 i412

TABLEAU (P,J)

	V A 1	V A 2	V A 3	V A 4
1	181	229	189	182
2	15	73	19	113

TABLEAU DES VALEURS IDEALES

	VVVV AAAA 1234
1	1111
2	1

HOMOGENEITE PAR CLASSE

	V A 1	V A 2	V A 3	V A 4
1	74	94	78	75
2	91	57	88	66

TABLEAU DES ECARTS

	V A 1	V A 2	V A 3	V A 4
1	61	13	53	60
2	15	73	19	57

TABLEAU DES PARAMETRES DE LA LOI DE BERNOUILLI

	V A 1	V A 2	V A 3	V A 4
1	0.25	0.05	0.22	0.25
2	0.09	0.43	0.11	0.34

TABLEAU DES COEFFICIENTS DE PONDERATION

	V A 1	V A 2	V A 3	V A 4
1	1.1	2.9	1.3	1.1
2	2.3	0.3	2.1	0.7

Ecriture du constituant No 3 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

ANNEXE 2

COMMANDE : MNDBIN <> nuees dynamiques sur variables binaires

variabl es selectionnees :

VA 1 VA 2 VA 3 VA 4

Valeurs des Parametres

nombre de classes	2
mode d'initialisation	1
nombre de tirages au hasard	20
classe vide possible	oui
sortie uniquement du meilleur resultat	oui
impression tableau initial reordonne	non
choix de la distance	2

Valeur du critere obtenu a chaque tirage :

785.0630000 (convergence atteinte)
788.3127000 (convergence atteinte)
785.0630000 (convergence atteinte)
788.3127000 (convergence atteinte)
788.3127000 (convergence atteinte)
788.3127000 (convergence atteinte)
788.3127000 (convergence atteinte)
788.3127000 (convergence atteinte)
785.0630000 (convergence atteinte)
785.0630000 (convergence atteinte)
785.0630000 (convergence atteinte)
788.3127000 (convergence atteinte)
785.0630000 (convergence atteinte)
785.0630000 (convergence atteinte)
785.0630000 (convergence atteinte)
788.3127000 (convergence atteinte)
785.0630000 (convergence atteinte)
834.2594000 (convergence atteinte)
785.0630000 (convergence atteinte)
788.3127000 (convergence atteinte)

ANALYSE DU MEILLEUR TIRAGE

valeur du critere obtenu 785.0630000

Partition

classe 1 : 208 elements

i001 i002 i003 i004 i005 i006 i007 i027 i028 i029 i030 i031 i032 i041 i042 i043
i044 i045 i046 i047 i048 i049 i050 i051 i052 i053 i054 i055 i056 i057 i058 i059
i060 i061 i062 i101 i102 i103 i104 i105 i155 i156 i157 i158 i159 i160 i161 i162
i163 i164 i165 i166 i167 i168 i169 i170 i171 i172 i173 i174 i175 i176 i177 i178
i179 i180 i181 i182 i183 i184 i185 i186 i187 i188 i189 i190 i191 i192 i193 i194
i195 i196 i197 i198 i199 i200 i201 i202 i203 i204 i205 i206 i207 i208 i209 i210
i211 i212 i213 i214 i215 i216 i217 i218 i219 i220 i221 i222 i223 i224 i225 i226
i227 i228 i229 i230 i231 i232 i233 i234 i235 i236 i237 i238 i239 i240 i241 i242
i243 i244 i245 i246 i247 i248 i249 i250 i251 i252 i253 i254 i255 i256 i257 i258
i259 i260 i261 i262 i297 i298 i299 i300 i301 i302 i303 i311 i312 i313 i314 i315
i316 i317 i318 i319 i320 i321 i322 i323 i324 i325 i326 i327 i328 i329 i330 i331
i332 i333 i334 i335 i336 i337 i338 i339 i340 i341 i342 i343 i344 i345 i346 i347
i348 i349 i385 i386 i387 i388 i389 i390 i391 i392 i393 i394 i395 i396 i397 i398

classe 2 : 204 elements

i008 i009 i010 i011 i012 i013 i014 i015 i016 i017 i018 i019 i020 i021 i022 i023
i024 i025 i026 i033 i034 i035 i036 i037 i038 i039 i040 i063 i064 i065 i066 i067
i068 i069 i070 i071 i072 i073 i074 i075 i076 i077 i078 i079 i080 i081 i082 i083
i084 i085 i086 i087 i088 i089 i090 i091 i092 i093 i094 i095 i096 i097 i098 i099
i100 i106 i107 i108 i109 i110 i111 i112 i113 i114 i115 i116 i117 i118 i119 i120
i121 i122 i123 i124 i125 i126 i127 i128 i129 i130 i131 i132 i133 i134 i135 i136
i137 i138 i139 i140 i141 i142 i143 i144 i145 i146 i147 i148 i149 i150 i151 i152
i153 i154 i263 i264 i265 i266 i267 i268 i269 i270 i271 i272 i273 i274 i275 i276
i277 i278 i279 i280 i281 i282 i283 i284 i285 i286 i287 i288 i289 i290 i291 i292
i293 i294 i295 i296 i304 i305 i306 i307 i308 i309 i310 i350 i351 i352 i353 i354
i355 i356 i357 i358 i359 i360 i361 i362 i363 i364 i365 i366 i367 i368 i369 i370
i371 i372 i373 i374 i375 i376 i377 i378 i379 i380 i381 i382 i383 i384 i399 i400
i401 i402 i403 i404 i405 i406 i407 i408 i409 i410 i411 i412

TABLEAU (P,J)

	V A 1	V A 2	V A 3	V A 4
1	128	176	208	154
2	68	126	0	141

TABLEAU DES VALEURS IDEALES

	VVVV AAAA 1234
1	1111
2	1 1

HOMOGENEITE PAR CLASSE

	V A 1	V A 2	V A 3	V A 4
1	61	84	100	74
2	66	61	100	69

TABLEAU DES ECARTS

	V A 1	V A 2	V A 3	V A 4
1	80	32	0	54
2	68	78	0	63

TABLEAU DES PARAMETRES DE LA LOI DE BERNOUILLI

	V A 1	V A 2	V A 3	V A 4
1	0.36	0.27	0.08	0.28
2	0.36	0.27	0.08	0.28

TABLEAU DES COEFFICIENTS DE PONDERATION

	V A 1	V A 2	V A 3	V A 4
1	0.6	1.0	2.4	0.9
2	0.6	1.0	2.4	0.9

Ecriture du constituant No 4 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

COMMANDE : MNDBIN <> nuees dynamiques sur variables binaires

variables selectionnees :

VA 1 VA 2 VA 3 VA 4

Valeurs des Parametres

nombre de classes	2
mode d'initialisation	2
nombre de tirages au hasard	1
classe vide possible	oui
sortie uniquement du meilleur resultat	oui
impression tableau initial reordonne	non
choix de la distance	3

Lecture du constituant No 4 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

Valeur du critere obtenu a chaque tirage :

792.3874000 (convergence atteinte)

ANALYSE DU MEILLEUR TIRAGE

valeur du critere obtenu 792.3874000

Partition

classe 1 : 227 elements

i001 i002 i003 i004 i005 i006 i007 i008 i009 i010 i011 i012 i013 i014 i015 i016
i017 i018 i019 i020 i021 i022 i027 i028 i029 i030 i031 i032 i041 i042 i043 i044
i045 i046 i047 i048 i049 i050 i051 i052 i053 i054 i055 i056 i057 i058 i059 i060
i061 i062 i155 i156 i157 i158 i159 i160 i161 i162 i163 i164 i165 i166 i167 i168
i169 i170 i171 i172 i173 i174 i175 i176 i177 i178 i179 i180 i181 i182 i183 i184
i185 i186 i187 i188 i189 i190 i191 i192 i193 i194 i195 i196 i197 i198 i199 i200
i201 i202 i203 i204 i205 i206 i207 i208 i209 i210 i211 i212 i213 i214 i215 i216
i217 i218 i219 i220 i221 i222 i223 i224 i225 i226 i227 i228 i229 i230 i231 i232
i233 i234 i235 i236 i237 i238 i239 i240 i241 i242 i243 i244 i245 i246 i247 i248
i249 i250 i251 i252 i253 i254 i255 i256 i257 i258 i259 i260 i261 i262 i263 i264
i265 i266 i267 i268 i269 i270 i271 i272 i273 i274 i275 i276 i277 i278 i279 i280
i281 i282 i283 i284 i285 i297 i298 i299 i300 i301 i302 i303 i311 i312 i313 i314
i315 i316 i317 i318 i319 i320 i321 i322 i323 i324 i325 i326 i327 i328 i329 i330
i331 i332 i333 i334 i335 i336 i337 i338 i339 i340 i341 i342 i343 i344 i345 i346
i347 i348 i349

classe 2 : 185 elements

i023 i024 i025 i026 i033 i034 i035 i036 i037 i038 i039 i040 i063 i064 i065 i066
i067 i068 i069 i070 i071 i072 i073 i074 i075 i076 i077 i078 i079 i080 i081 i082
i083 i084 i085 i086 i087 i088 i089 i090 i091 i092 i093 i094 i095 i096 i097 i098
i099 i100 i101 i102 i103 i104 i105 i106 i107 i108 i109 i110 i111 i112 i113 i114
i115 i116 i117 i118 i119 i120 i121 i122 i123 i124 i125 i126 i127 i128 i129 i130
i131 i132 i133 i134 i135 i136 i137 i138 i139 i140 i141 i142 i143 i144 i145 i146
i147 i148 i149 i150 i151 i152 i153 i154 i286 i287 i288 i289 i290 i291 i292 i293
i294 i295 i296 i304 i305 i306 i307 i308 i309 i310 i350 i351 i352 i353 i354 i355
i356 i357 i358 i359 i360 i361 i362 i363 i364 i365 i366 i367 i368 i369 i370 i371
i372 i373 i374 i375 i376 i377 i378 i379 i380 i381 i382 i383 i384 i385 i386 i387
i388 i389 i390 i391 i392 i393 i394 i395 i396 i397 i398 i399 i400 i401 i402 i403
i404 i405 i406 i407 i408 i409 i410 i411 i412

TABLEAU (P,J)

	V A 1	V A 2	V A 3	V A 4
1	166	214	189	182
2	30	88	19	113

TABLEAU DES VALEURS IDEALES

	VVVV AAAA 1234
1	1111
2	1

HOMOGENEITE PAR CLASSE

	V A 1	V A 2	V A 3	V A 4
1	73	94	83	80
2	83	52	89	61

TABLEAU DES ECARTS

	V A 1	V A 2	V A 3	V A 4
1	61	13	38	45
2	30	88	19	72

TABLEAU DES PARAMETRES DE LA LOI DE BERNOUILLI

	V A 1	V A 2	V A 3	V A 4
1	0.27	0.06	0.17	0.20
2	0.16	0.48	0.10	0.39

TABLEAU DES COEFFICIENTS DE PONDERATION

	V A 1	V A 2	V A 3	V A 4
1	1.0	2.8	1.6	1.4
2	1.6	0.1	2.2	0.5

Ecriture du constituant No 5 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

ANNEXE 3

COMMANDE : MNDBIN <> nuees dynamiques sur variables binaires

variables selectionnees :

VA 1 VA 2 VA 3 VA 4

Valeurs des Parametres

nombre de classes	2
mode d'initialisation	1
nombre de tirages au hasard	20
classe vide possible	oui
sortie uniquement du meilleur resultat	oui
impression tableau initial reordonne	non
choix de la distance	3

Valeur du critere obtenu a chaque tirage :

800.6928000 (convergence atteinte)
761.3879000 (convergence atteinte)
792.6068000 (convergence atteinte)
764.3589000 (convergence atteinte)
823.0933000 (convergence atteinte)
784.8664000 (convergence atteinte)
822.3993000 (convergence atteinte)
823.0755000 (convergence atteinte)
795.2903000 (convergence atteinte)
793.3649000 (convergence atteinte)
814.4106000 (convergence atteinte)
792.3874000 (convergence atteinte)
802.0881000 (convergence atteinte)
761.3879000 (convergence atteinte)
814.4106000 (convergence atteinte)
793.6940000 (convergence atteinte)
795.1987000 (convergence atteinte)
761.6578000 (convergence atteinte)
802.6893000 (convergence atteinte)
784.8664000 (convergence atteinte)

ANALYSE DU MEILLEUR TIRAGE

valeur du critere obtenu 761.3879000

Partition

classe 1 : 216 elements

i041 i042 i043 i044 i045 i046 i047 i048 i049 i050 i051 i052 i053 i054 i055 i056
i057 i058 i059 i060 i061 i062 i063 i064 i065 i066 i067 i068 i069 i070 i071 i072
i073 i074 i075 i076 i077 i078 i079 i080 i081 i082 i083 i084 i085 i086 i087 i088
i089 i090 i091 i092 i093 i094 i095 i096 i097 i098 i099 i100 i101 i102 i103 i104
i105 i106 i107 i108 i109 i110 i111 i112 i113 i114 i115 i116 i117 i118 i119 i120
i121 i122 i123 i124 i125 i126 i127 i128 i129 i130 i131 i132 i133 i134 i135 i136
i137 i138 i139 i140 i141 i142 i143 i144 i145 i146 i147 i148 i149 i150 i151 i152
i153 i154 i311 i312 i313 i314 i315 i316 i317 i318 i319 i320 i321 i322 i323 i324
i325 i326 i327 i328 i329 i330 i331 i332 i333 i334 i335 i336 i337 i338 i339 i340
i341 i342 i343 i344 i345 i346 i347 i348 i349 i350 i351 i352 i353 i354 i355 i356
i357 i358 i359 i360 i361 i362 i363 i364 i365 i366 i367 i368 i369 i370 i371 i372
i373 i374 i375 i376 i377 i378 i379 i380 i381 i382 i383 i384 i385 i386 i387 i388
i389 i390 i391 i392 i393 i394 i395 i396 i397 i398 i399 i400 i401 i402 i403 i404
i405 i406 i407 i408 i409 i410 i411 i412

classe 2 : 196 elements

i001 i002 i003 i004 i005 i006 i007 i008 i009 i010 i011 i012 i013 i014 i015 i016
i017 i018 i019 i020 i021 i022 i023 i024 i025 i026 i027 i028 i029 i030 i031 i032
i033 i034 i035 i036 i037 i038 i039 i040 i155 i156 i157 i158 i159 i160 i161 i162
i163 i164 i165 i166 i167 i168 i169 i170 i171 i172 i173 i174 i175 i176 i177 i178
i179 i180 i181 i182 i183 i184 i185 i186 i187 i188 i189 i190 i191 i192 i193 i194
i195 i196 i197 i198 i199 i200 i201 i202 i203 i204 i205 i206 i207 i208 i209 i210
i211 i212 i213 i214 i215 i216 i217 i218 i219 i220 i221 i222 i223 i224 i225 i226
i227 i228 i229 i230 i231 i232 i233 i234 i235 i236 i237 i238 i239 i240 i241 i242
i243 i244 i245 i246 i247 i248 i249 i250 i251 i252 i253 i254 i255 i256 i257 i258
i259 i260 i261 i262 i263 i264 i265 i266 i267 i268 i269 i270 i271 i272 i273 i274
i275 i276 i277 i278 i279 i280 i281 i282 i283 i284 i285 i286 i287 i288 i289 i290
i291 i292 i293 i294 i295 i296 i297 i298 i299 i300 i301 i302 i303 i304 i305 i306
i307 i308 i309 i310

TABLEAU (P,J)

	V A 1	V A 2	V A 3	V A 4
1	0	134	80	153
2	196	168	128	142

TABLEAU DES VALEURS IDEALES

	VVVV AAAA 1234
1	1 1
2	1111

HOMOGENEITE PAR CLASSE

	V A 1	V A 2	V A 3	V A 4
1	100	62	62	70
2	100	85	65	72

TABLEAU DES ECARTS

	V A 1	V A 2	V A 3	V A 4
1	0	82	80	63
2	0	28	68	54

TABLEAU DES PARAMETRES DE LA LOI DE BERNOUILLI

	V A 1	V A 2	V A 3	V A 4
1	0.09	0.38	0.37	0.29
2	0.25	0.14	0.35	0.28

TABLEAU DES COEFFICIENTS DE PONDERATION

	V A 1	V A 2	V A 3	V A 4
1	2.3	0.5	0.5	0.9
2	1.1	1.8	0.6	1.0

Ecriture du constituant No 6 sur 1 archive :store.sar
Nom du constituant : CRBIClasses obtenues avec MNDBIN

ANNEXE 4

COMMANDE : HASBIN <> creation de tableaux binaires

VALEURS DES PARAMETRES

nombre d'individus	100
nombre de variables	10
nombre de classes	3
parametre de la loi binomiale	1

EFFECTIF DES CLASSES

classe 1 : 33
classe 2 : 33
classe 3 : 34

TABLEAU DES VALEURS IDEALES

	V A 1	V A 2	V A 3	V A 4	V A 5	V A 6	V A 7	V A 8	V A 9	V A 10
1	1	1	1	0	1	1	0	1	1	1
2	0	0	1	1	0	0	0	0	0	0
3	1	0	1	0	1	0	1	1	0	0

TABLEAU DES PARAMETRES DE LA LOI BINOMIALE

	V A 1	V A 2	V A 3	V A 4	V A 5	V A 6	V A 7	V A 8	V A 9	V A 10
1	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10
2	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10
3	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10

PARTITION SERVI A LA SIMULATION

classe numero : 1

1 2 3 4 7 8 10 13 15 16 18 20 23 25 26 27 40 45 47 58
68 71 74 77 78 79 81 83 85 89 92 93 94

classe numero : 2

5 6 9 11 12 17 21 29 31 35 37 39 43 44 46 48 49 50 51 54
55 57 59 60 61 62 63 64 69 72 73 87 95

classe numero : 3

14 19 22 24 28 30 32 33 34 36 38 41 42 52 53 56 65 66 67 70
75 76 80 82 84 86 88 90 91 96 97 98 99 100

COMMANDE : HASBIN <> creation de tableaux binaires

VALEURS DES PARAMETRES

nombre d'individus	100
nombre de variables	10
nombre de classes	3
parametre de la loi binomiale	2

EFFECTIF DES CLASSES

classe 1 : 33
classe 2 : 33
classe 3 : 34

TABLEAU DES VALEURS IDEALES

	VA 1	VA 2	VA 3	VA 4	VA 5	VA 6	VA 7	VA 8	VA 9	VA 10
1	1	1	1	0	1	1	0	1	1	1
2	0	0	1	1	0	0	0	0	0	0
3	1	0	1	0	1	0	1	1	0	0

TABLEAU DES PARAMETRES DE LA LOI BINOMIALE

	VA 1	VA 2	VA 3	VA 4	VA 5	VA 6	VA 7	VA 8	VA 9	VA 10
1	0.01	0.20	0.01	0.48	0.01	0.30	0.40	0.20	0.01	0.20
2	0.01	0.20	0.01	0.48	0.01	0.30	0.40	0.20	0.01	0.20
3	0.01	0.20	0.01	0.48	0.01	0.30	0.40	0.20	0.01	0.20

PARTITION SERVI A LA SIMULATION

classe numero : 1

2	3	4	5	6	7	10	11	12	15	17	20	22	23	24	26	27	28	31	33
37	51	59	67	70	73	75	78	83	91	92	93	97							

classe numero : 2

1	8	13	14	21	30	32	34	35	38	40	41	43	44	45	46	47	48	49	52
53	54	55	56	57	58	62	65	68	71	77	82	87							

classe numero : 3

9	16	18	19	25	29	36	39	42	50	60	61	63	64	66	69	72	74	76	79
80	81	84	85	86	88	89	90	94	95	96	98	99	100						

COMMANDE : HASBIN <> creation de tableaux binaires

VALEURS DES PARAMETRES

nombre d'individus	100
nombre de variables	10
nombre de classes	3
parametre de la loi binomiale	3

EFFECTIF DES CLASSES

classe 1 : 33
classe 2 : 33
classe 3 : 34

TABLEAU DES VALEURS IDEALES

	V A 1	V A 2	V A 3	V A 4	V A 5	V A 6	V A 7	V A 8	V A 9	V A 10
1	1	0	1	0	1	0	1	1	1	0
2	0	0	0	0	0	1	1	1	1	1
3	0	1	0	0	0	1	0	0	0	0

TABLEAU DES PARAMETRES DE LA LOI BINOMIALE

	V A 1	V A 2	V A 3	V A 4	V A 5	V A 6	V A 7	V A 8	V A 9	V A 10
1	0.00	0.10	0.20	0.01	0.20	0.01	0.50	0.30	0.01	0.20
2	0.10	0.10	0.01	0.30	0.10	0.10	0.10	0.50	0.35	0.20
3	0.50	0.01	0.01	0.30	0.01	0.20	0.50	0.01	0.01	0.01

PARTITION SERVI A LA SIMULATION

classe numero : 1

3 5 7 10 11 12 16 18 20 21 25 28 30 31 33 37 40 42 43 47
54 56 64 65 66 67 73 80 82 85 87 90 95

classe numero : 2

1 4 8 9 17 19 29 35 36 38 39 41 44 45 46 48 49 50 52 53
57 58 60 61 62 71 74 77 79 89 93 99 100

classe numero : 3

2 6 13 14 15 22 23 24 26 27 32 51 55 59 63 68 69 70 72 75
76 78 81 83 84 86 88 91 92 94 96 97 98

BIBLIOGRAPHIE

BIBLIOGRAPHIE

ANDERBERG M.R. (1973), " Cluster Analysis for Application ". Academic Press, New-York.

AGRAWALA A.K. (1970), " Learning with a probabilistic teachear " IEEE Trans. On Information theory, vol..IT. 16, n°4.

BALL G.H et HALL D.J. (1965), " Isodata a novel method of data analysis and pattern classification". Technical report, S.R.I. project 5533, stanford research institute, menbo park calif. U.S.A.

BALL G.H. et HALL D.J. (1967), " A Clustering Technique for summarizing Multivariate Data ". Behavorial Science 12, n°2, 153 - 155.

BENZECRI J.P. (1972), " La régression (Regr)". Polycopié du Laboratoire de Statistique Mathématique. Université de PARIS 6.

BENZECRI J.P. (1973), "L'Analyse des Données (1. la Taxinomie, 2. l'Analyse des Correspondances)". Dunod.

BOCK H.H. (1986), "Loglinear Models and Entropy Clustering Methods for Qualitative Data". Classification as a Tool of Research, W. Gaul and M. Schader (editors).

BONNER R.H. (1964), " On some clustering techniques ". I.B.M journal vol 22.

BRYANT P. et Williamson J. (1978), " Asymptotic behaviour of Classification ML estimates ". Biometrika vol. 65.

CELEUX G. (1988), "Classification et Modèle". RSA vol 36, nº 3. Rapport n°810 INRIA.

CELEUX G., DIDAY E. , GOVAERT G. , LECHEVALIER Y. , RALAMBONDRAINY H. (1989), " Classification Automatique des données ", Dunod

CELEUX G. et GOVAERT G. (1989), "Clustering Criteria for Discrete Data and Latent Class Models". Rapport nº 1122 INRIA.

CHERNOFF H (1970), " Metric considerations in the K-means method of cluster analysis ". Classification society meetings. Ohio University.

COOPER D.B et COOPER P.W (1964). " Non supervised adaptative signal detection and pattern recognition". Information and Control. 7, pp 416.

DAY N.E. (1969) " Estimating the components of a mixture of normal distributions". Biometrica 56, 3 pp 463.

DIDAY E. (1972). " Nouvelles méthodes et nouveaux concepts en classification automatique et reconnaissance des formes ". Thèse d'Etat Université PARIS 6.

DIDAY E. , SCHROEDER A. et OK Y (1974). " The Dynamic Clusters method in Pattern Recognition ". Procceding of IFIP Congress Stockholm.

DIDAY E. (1975). " Classification automatique séquentielle pour grands tableaux". RAIRO Intelligence Artificielle et Reconnaissance des Formes.

DIDAY E. SCHROEDER A. (1976). " A new approach in mixed distributions detection " . RAIRO. Recherche Opérationnelle. vol. 10, n°6.

DIDAY E. GOVAERT G. (1977), " Classification avec distances adaptatives ". RAIRO, V-11, n°4, pp. 329 - 349.

DIDAY E. et COLLABORATEURS. (1980), "Optimisation en classification automatique". INRIA, Rocquencourt.

DUDA R. O. et HART. R.E. (1973) " Pattern classification and scene analysis ". Wiley, N.Y.

FORGY E. W. (1965). "Cluster analysis of multivariante data : efficiency versus interpretability of classification", Biométrics vol 21.

FRIEDMAN H. et RUDIN J. (1967), " On some invariant criterion for grouping data " . JASA 62.

GOLDSTEIN M. DILLON W. R. (1978). " Discrete discriminant analysis ". Sons, New-York.

GOVAERT G. (1975) , " Classification avec distance adaptative ". Thèse de Doctorat de 3ème cycle , PARIS 6.

GOVAERT G. (1983), " Classification Croisée ". Thèse de Doctorat d'Etat, Université Pierre et Marie Curie, Paris VI.

GOVAERT G. (1988), " Classification Binaire et Modèle". Rapport de Recherche INRIA, nº 949.

GOVAERT G. (1989), "Modèle de classification et distance dans le cas continu". Rapport de Recherche INRIA, nº 988.

GOVAERT G. (1989). Clustering model and metric with continuous data. In " Data analysis, learning symbolic and numeric knowledge". Diday E., ed.New-York : Nova, science publishers, pp 95-102.

GOVAERT G (1990), "Modèle de classification et distance dans le cas discret". Rapport de Recherche INRIA. (à paraître).

JAMBU M., (1971). " Classification automatique hierarchique ". Thèse doctorat de 3ème cycle Université de Paris.

JARDINE N. et SIBSON R. (1968), " THE construction of hierarchic and non-hierarchical classification ". Computer J. 11, p 177.

JENSEN R.E. , (1969), " A dynamical programming algorithm for cluster data analysis". Université of Maine. Crono Haine.

JOHNSON S.C. (1967), " Hierarchical Clustering Schemes ". Psychometrika . Vol 32. n°3.

LANCE G.N. , WILLIAMS W.T. (1967), " A general theory of classification sorting strategies, 1 : hierarchical systems ". Computer journal vol. 9, n°4.

LEBART L., MORINEAU A. et TABARD N. (1977), "Techniques de la Description Statistique". Dunod.

LEBART L. , MORINEAU A , JP FENELON (1979). " Traitement des données statistiques " , méthodes et programmes Dunod.

LERMAN I.C. (1981), " Classification et analyse ordinale des données ". Dunod.

MAC QUEEN J.B. (1967), " Some méthods for classification and analysis of multivariante observations " , proc of the 5th Berkeley symposium on math. Statistics and probability. vol 1 p 281.

MARCHETTI F. (1989) , " Contribution à la classification de données binaires et qualitatives ". Thèse de Doctorat de l'Université de Metz.

MARRIOTT F. (1975), " Separating mixtures of normal distributions ". Biometrika, vol 31.

PATRICK E.A. et HANCOCK J.C. (1966), " Non supervised séquential classification and recognition of patterns ". IEEE Trans. On Information theory, vol I T 16, n°5.

PATRICK E.A. CASTELLO J.P (1970), " On un supervised estimation algorithms IEEE TRANS. On Information theory, vol IT 16, n°5.

PATRICK E.A. (1972), " Fundamentals of Pattern Recongnition ". Prentice Hall inc. New Jersey.

PEARSON K. (1894), " Contributions to the mathematic theory of evolution ". Philos. Trans. Soc. , n°185.

RAO C.R. (1948). " Utilization of multiple measurement in problems of biological classification ". Journal of the Royal Statistical Society.

REGNIER S. (1965), "Sur quelques aspects mathématiques des problèmes de classification automatique". ICC bulletin. Vol.4, p 175.

ROUX M. (1968), "Un algorithme pour construire une hierarchie particulière" . Thèse de Doctorat de 3ème cycle. Faculté des sciences de Paris.

RUSPINI E.M. (1969), "A new approach to clustering". Information and control 15, p 22.

RALAMBONDRAINY H. (1988), "Etude des données qualitatives par les méthodes typologoques". Actes au congrés de l'association française de marketing. Montpelier.

SCHROEDER A. (1974), " Reconnaissance des composants d'un mélange". Thèse de Doctorat de 3ème cycle . Université de Paris 6.

SCHROEDER A. (1976), "Analyse d'un Mélange de Distribution de Probabilité de même Type". R.S.A vol 24, nº 1. 39-62.

SCOTT A. et SYMONS M. (1971), "Clustering Methods Based on Likelihood Ratio Criteria". Biometrics 27. 387-397.

SICLA : Système intéractif de classification automatique, manuel de l'utilisateur, CISIA, 1990.

SOKAL R.R. MICHNER C.D. (1958), " A statistical method for evaluating systematic relationships ". Université Kansas Sci. Bull., 38.

SOKAL R.R. et SNEATH P.H. (1963), " Principales of numerical taxonomy ". San Francisco : Freeman.

THORNDIKE R.L. (1953), " Who belongs in the family ? " , Psychométrika vol 18.

WOLF J.H. (1970), " Pattern clustering by multivariate mixture analysis ". Multivariate Behavioral Res. (July 1970).

Résume :

Jusqu'à présent deux tendances parallèles se sont dégagées dans le développement et la pratique du traitement statistique des données ; La première met en jeu des méthodes qui envisagent la possibilité d'une interprétation probabiliste, la deuxième fait intervenir une classe assez vaste de méthodes de classification automatique conçues dans un cadre purement géométrique. Notre travail se situe à mi-chemin entre ces deux approches ; en effet les liens qui existent entre l'approche probabiliste et l'approche géométrique nous ont permis d'interpréter des méthodes de classification automatique en termes probabilistes, de justifier à posteriori certaines contraintes imposées souvent pour des raisons techniques d'optimisation et de proposer de nouveaux critères pouvant améliorer la qualité de la partition ; nous généralisons ensuite l'étude de ces liens aux cas où les données mettent en jeu deux ensembles. Nous montrons comment la classification croisée peut être vu comme une solution à un problème d'estimation de paramètres d'un modèle de mélanges, nous développons une méthode de reconnaissance des composants d'un mélange "croisé" qui nous permettra d'interpréter des méthodes de classification croisées et de proposer de nouveaux algorithmes de classification croisée utilisant des distances adaptatives. Certaines méthodes proposées dans ce travail ont été programmées et intégrées au logiciel d'Analyse de Données SICLA (Système Interactif de Classification Automatique, INRIA).

Mots clés

Données binaires, données qualitatives, données quantitatives, distance L_1, classification automatique, mélange de lois de probabilité, mélange croisé.

Abstract

Up to now, two parallel trends have emerged in the developement and practice of statistical data processing. The first one involves methods that consider the possibility of a probabilistic interpretation ; the second one uses a rather large group of automatic clustering methods applied within a purely geometrical framework. Our study is set halfway betwen those two approaches ; Indeed the links that exist betwen the probabilistic approach and the geometrical approach have enabled us to interpret automatic clustering methods in probabilistic terms, to justify a posteriori some constraints often used for technical optimization reasons and to propose new criteria that can improve the quality of the partition ; we then extend the study of these links to cases were the data involve two sets ; we show how the cross clustering can be seen as a solution to a problem for the estimation of the parameters of a model with crossed mixture , we develop a method of identification of crossed mixture ; this method will enable us to interpret cross clustering methods and to propose new cross clustering algorithms using adaptative distances. Some methods proposed in this study have been programmed and integrated into the data analysis software SICLA (Interactif Systeme of Automatic Clustering, INRIA)

Key words

Binary data, qualitative data , quantitative data, L_1 distance, automatic clustering, mixture, cross mixture.