Remerciments

«(Et lorsque votre Seigneur proclama : "Si vous êtes reconnaissants, très certainement J'augmenterai [Mes bienfaits] pour vous)»

[Coran S14.V7]

Avant tout, je remercie Dieu le très haut qui m'a donné le courage et la volonté de réaliser ce modeste travail.

«(CELUI QUI NE REMERCIE PAS LES GENS, NE REMERCIE PAS ALLAH.)»

[Authentique Hadith]

Que Monsieur Mourtada Benazouz, maitre assistant De l'Informatique à l'université « abou bakr belkaid », trouve ici le témoignage de ma profonde reconnaissance. Ses encouragements, et surtout ses critiques, Sa sensibilisation, ont largement contribué à l'accomplissement de mes travaux. Je le remercie infiniment de m'avoir toujours poussé vers l'avant.

Je tiens également à remercier « TOUS » les Messieurs et dames, mes professeurs qui m'ont enseigné durant deux ans de formation master en Informatique, pour leurs précieux conseils et ses orientations,

Mes remerciements vont également aux membres du jury d'avoir accepté d'évaluer mon travail.

Sans oublier de remercier mes amis et mes collègues (de l'université ou dans le monde Virtual « internet ») qui, tous d'une manière différente, ont contribué à ce que je puisse aboutir à la réalisation de ce mémoire.

Enfin, merci à ma famille (ma Chère mère et mes belles sœurs, mon frère) pour le soutien et l'encouragement qu'ils m'ont apporté tout au long de mon travail.

Ce Rapport a été écrit en LYX/LATEX

Dedicace

A LA MÉMOIRE DE MON PÈRE

Koudri Mohammed

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations :

GMM : Gaussian mixture model

EM : expectation maximisation

CA : classification automatique

HMM : hidden markov model

MAP : maximum a posteriori

MV : maximum de vraisamblance

Introduction générale :

Depuis l'aube des temps, l'homme pratique la classification dans sa vie quotidienne, quand il essaie de répondre aux problèmes et questions sur la catégorie des objets , c'est-à-dire d'affectation d'objets a leur classe (en observant leurs formats , couleurs , tailles …etc.), un exemple très simple , le fait qu'il avait distingué entre les plantes, en expérimenta empiriquement les propriétés thérapeutiques, avec les conséquences parfois désastreuses que l'on peut imaginer. Puis les classifications apparurent, liées aux expériences des peuples concernés sur ces plantes et leur utilité. Au cours des siècles, ces classifications s'affinèrent pour nous livrer sous des formes précises et parfois très complexes les différents Herbiers que nous connaissons aujourd'hui (cette plante sert de nourriture, celle-ci pourrait aider efficacement à combattre la maladie et la douleur, ce qui a prit le nom de phytothérapie, …etc.).

Avec l'apparition des écoles de pensées scientifiques la classification a constitué l'un des thèmes majeurs de l'histoire naturelle, et a été utilisée dans plusieurs domaines, commençant par la Taxinomie[1] en laquelle on classe les organismes vivants en les regroupant en entités appelées taxons.**[André Césalpin ,1583 , De Plantis]**, ou encore en biologie, plus précisément l'étude des espaces vivants ce qu'on a appelé : les SYSTEMES DE CLASSIFICATION PHYNOGENETIQUE**[Guillaume** & **Hervé, 2002]**, puis vers la statistique[2] **[Jean Jadot, 2007]** où les premières théories classificatoires ont été fondées, jusqu'à où le grand développement des ordinateurs et des outils d'analyse des grandes données , a basculé , et a permis de rendre la classification un outil primordial de nombreux domaines comme : la reconnaissance des formes, l'apprentissage et la recherche opérationnelle. Récemment, pour des besoins en intelligence artificielle et " fouille de données ", on a eu la naissance de la " classification conceptuelle ".

Dans le présent travail , nous tentons d'adopter une des stratégies de la classification

[1] science des lois de classification des formes vivantes" (Robert).

[2] L'approche classificatoire, élabore les catégories susceptibles de classifier les ensembles et sous-ensembles d'un champ de connaissances.

automatique (clustering= une des grandes familles des méthodes classificatoires) et l'appliquer dans le domaine médicale, en particulier la détection et l'identification des maladies en observant seulement un prélèvement de cellules microscopiques,

en fait, pour que le spécialiste puisse donner un bon diagnostic, il faut qu'il y ait une bonne lisibilité et une bonne reconnaissance des cellules anormales des autres cellules qui composent une image microscopique sanguine.

Autrement dit, notre objectif est d'élaborer un système semi-automatique qui imite la perception humaine de la détection des différents objets, c'est-à-dire, il segmente l'image médicale en plusieurs zones, en identifiant chacune de ses composantes (dans notre cas, les globules rouges, blancs et le fond).

Pour cela on réparti notre travail en quatre chapitres, comme suite :

- Le **premier chapitre** débute par une introduction général citant les définitions et le langage commun des méthodes de la classification, ainsi que nous parlerons d'une de ses grandes approches « le classement, discrimination ».
- Le **deuxieme chapitre** consacré à la deuxième grande approche des méthodes classificatoires, qui est l'approche de notre méthode cible, il s'agit de discussions sur les méthodes non supervisées, en examinant chacune, (ses point forts, ses faiblesses, les domaines qu'elle vise, son principe ….etc.)
- Le **troisieme chapitre** est entièrement dédié à l'etude et l'analyse de notre méthode cible « Expectation maximisation – Gaussian Mixture Model) en examinant ses intérêt , ses entrée , son déroulement , ce qu'elle différencie des autres méthodes ….et.
- Le **chapitre quatre** quand a lui présente les différents outils qui vont servir a l'implémentation de notre projet, ainsi que l'implémentation de notre application.et les discussions sur la réalisation.

Nos applications auront comme nom :

SEM pour (**S**egmenter via **E**xpectation **M**Maximization)

Et nous terminons par une conclusion générale et les quelques perspectives, remarques qu'on a pu constater durant la réalisation de ce modeste travail.

Chapitre 1

Généralités sur la Classification

1.1 Introduction

« Le seul moyen de faire une méthode instructive et naturelle, est de mettre ensemble les choses qui se ressemblent et de séparer celles qui différent les unes des autres. »

M. Georges Buffon, Histoire naturelle, 1749.

Il est claire que le processus générale que la classification dans le domaine informatique essaie de l'appliquer sur des données numériques (points, tableaux, images, sons, . . .etc.), n'échappe pas à la règle imposée par ce célèbre naturaliste et écrivain Georges Buffon , et que le travail général des méthodes de classification , depuis 1749, consiste à imiter et automatiser ce principe en utilisant et inventant des moyens adéquats (matériaux-calculateurs-, et des théories classificatoires. . .etc.)

Allons de ce principe, nous présenterons dans ce chapitre tout d'abord ce que c'est la classification, ses méthodes, techniques, ses grandes approches, domaines d'applications, . . .etc. et on détaillera à la fin une de ses grandes approches en étudiant et analysant deux de ses algorithmes.

1.2 Définitions :

La classification est une discipline relié de près ou de loin a plusieurs domaines, elle est connue aussi sous noms variés (classification, clustering, segmentation,. . .) selon les objets qu'elle traite et les objectifs qu'elle vise à atteindre.

Pour attribuer une définition au terme « classification », il faudrait d'abord définir ses racines, ça vient du verbe "classer" qui désigne plus une action qu'un domaine, ou plutôt une série de méthodes qu'une théorie unifiée.

En mathématique, On appelle classification, la catégorisation algorithmique d'objets. Elle consiste à attribuer une classe ou catégorie à chaque objet (ou individu) à classer, en se basant sur des données statistiques. Elle fait couramment appel aux méthodes d'apprentissage et est largement utilisée en reconnaissance de formes.

Il est important de noter qu'il ne faut pas confondre entre ces deux termes : « classification » et « classement », au fait le mot classification en anglais signifie une chose, alors que le même mot en français ait une autre signification (utilité).

Dans un classement on affecte les objets à des groupes préétablis, c'est le but de l'analyse discriminante que de fixer des règles pour déterminer la classe des objets. La classification est donc, en quelque sorte, le travail préliminaire au classement, savoir la recherche des classes "naturelles" dans le domaine étudié, en anglais « Cluster Analysis ».

Cette collision entre les termes peut se résumer comme suite :

Français	Anglais
Classification	Clustering
Classement	Classification

TAB. 1.1 – les noms attributs à la classification en Français/Englais.

D'une manière général en vertu de ces définitions, la classification se définit alors comme une méthode mathématique d'analyse de données, pour faciliter l'étude d'une population d'effectif important, généralement des bases d'observations caractérisent un domain particulier (animaux, plantes, malades, gènes,. . . etc.), où on les regroupe en plusieurs classes,

1.3 Domaines d'application et points de vocabulaire :

La classification comme dit préalablement joue un rôle dans presque toutes les sciences et techniques qui font appel à la statistique multidimensionnelle. A titre d'exemple les sciences biologiques : botanique, zoologie, écologie, ... qui utilisent le terme "taxinomie" pour désigner l'art de la classification. Ainsi que les sciences de la terre et des eaux : géologie, pédologie, géographie, étude des pollutions, font grand usage de classifications.

Une autre forte utilité des techniques de classification dans les sciences de l'homme : psychologie, sociologie, linguistique, archéologie, histoire, etc ... et sans oublier les techniques dérivées comme les enquêtes d'opinion, le marketing, etc ... Ces dernières emploient parfois les mots de "typologie" et "segmentation" pour désigner la classification, Citons encore la médecine**[Jamoulle,** & **al, 2000]**, l'économie, l'agronomie . . . etc ! Dans toutes ces disciplines la classification peut être employée comme un domaine particulier ; mais elle l'est souvent vue comme une méthode complémentaire à d'autres méthodes statistiques. Elle est très largement utilisée à l'interprétation des graphiques d'analyse factorielle, ou bien déterminer des groupes d'objets homogènes, préalablement à une régression linéaire multiple.
Voilà les quelques exemples de ses utilités :

1.4 Exemples de problèmes de classification :

Ce sont les domaines que la classification pourrait viser et en même temps ils représentent les différents types de données d'entrées des techniques de classification, ce qui est nécessaire d'en présenter Avant d'aborder les méthodes classificatoires.

A/ Prédiction e-mail / Spam :

Comme le fait de différencier un E-mail valide d'un SPAM , d'ailleurs la classification est fort utile en ce qui concerne la catégorisation des documents sur internet quel que soit la nature du document (image , fichier , son . . .etc) **[Michael** & **al, 2007]** et elle peut même être utilisée pour classifier les document selon leur sens (le web sémantique et les moteurs de recherche où on associe des sens pour les termes et pour les classifier il faut développer un langage de traitement/classification sémantique par exemple à base d'ontologie.) ou tout simplement

pour classifier les ouvrages dans le monde des bibliothèques et des archives(le système de Classification de la Bibliothèque du Congrès **LCC [1.1]** .

FIG. 1.1 – E-mail valide / SPAM

B/ Reconnaissance de formes :

Généralement c'est une question qui vise à reconnaitre ou identifier certain motifs à partir de données brutes afin de prendre une décision dépendant de la catégorie attribuée à ce motif **[Peter, 2001]**, ces motifs (formes) peuvent s'agir d'une image (visage, empreinte digitale, rayon X, EEG,...) ou sonore (reconnaissance de parole), et bien d'autres. Comme la reconnaissance des caractères manuscrits :

FIG. 1.2 – reconnaissance des caractère manuscrits

Ou d'empreintes digitales : C'est l'exemple où on cherche à identifier une personne grâce à son empreinte ,

FIG. 1.3 – Reconnaissance d'empreintes digitales

Ou de Reconnaissance vocale : Dans le cas des signaux il ya des méthodes de traitement des langues pour fixer la classe d'appartenance d'un signale, ie : signal(j) appartient à {français, anglais, ... } ?

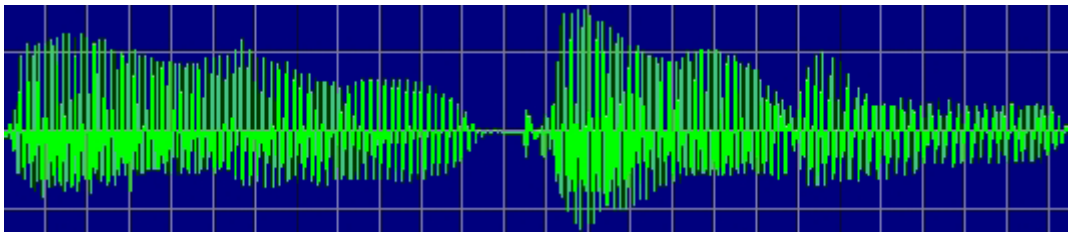

FIG. 1.4 – Reconnaissance vocale

C/ Des Tableaux :

comme les données bancaires : ce type de données est représenté sous forme des tableaux N x M où N le nombres d'exemples (individus , objets) et M l'ensemble des descripteurs ou qualité de ce qui a été décrit (attributs) la case (i,j) de ce tableau contient l'information relative de l'élément j sur i ; le rôle de la classification dans ce cas , consiste à déterminer le comportement d'un individu par rapport a ce qu'on a appris des autres (base d'apprentissage)

Taux de fraude
(Montant de la fraude, en millions d'euros)

Transactions nationales	2004	2005	2006	2007
Paiements	**0,036 % (81,2)**	**0,033 % (82,8)**	**0,035 % (92,3)**	**0,032 % (95,6)**
- dont paiements de proximité et sur automate	0,029 % (63,5)	0,025 % (59,2)	0,024 % (59,1)	0,017 % (45,4)
- dont paiements à distance	0,177 % (17,7)	0,196 % (23,6)	0,199 % (33,2)	0,236 % (50,1)
- dont par courrier / téléphone	nd	nd	0,194 % (19,8)	0,201 % (23,8)
- dont sur Internet	nd	nd	0,208 % (13,4)	0,281 % (26,4)
Retraits	**0,027 % (22,7)**	**0,017 % (15,0)**	**0,019 % (17,4)**	**0,020 % (19,0)**
Total	**0,033 % (103,9)**	**0,029 % (97,8)**	**0,031 % (109,6)**	**0,029 % (114,5)**

Source : Observatoire de la sécurité des cartes de paiement

FIG. 1.5 – données bancaires

1.5 Fondements :

la philosophie des techniques classificatoires sur des objets qu'on souhaite traiter en catégories, se déroule généralement comme suite :

La classification vise à créer ces catégories à partir de traitements ne faisant intervenir que les données et pas la subjectivité de l'utilisateur(expert, simple utilisateur...) mais il est parfois nécessaire d'apprendre a partir d'exemples pour cela on aura besoin de la la subjectivité de l'expérimentateur par le choix des descripteurs qu'il utilise, le classement selon un seul critère nous donnera pas le même classement selon un autre , la classification prends en considération tous les critères disponibles pour les classer en espace multi- dimensionnel.

Bien que les premières bases de l'approche algorithmique de la classification soient relativement anciennes, ce n'est qu'avec le développement de l'informatique qu'il est devenu

possible de les mettre en œuvre sur de grands échantillons de données. Le résultat d'une méthode de classification peut être soit une partition mathématique soit une hiérarchie (mathématiques).

1.6 Les étapes d'une classification :

1. Choix des données.
2. Calcul des similarités entre les n individus à partir des données initiales.
3. Choix d'un algorithme de classification et exécution.
4. L'interprétation des résultats :
   - évaluation de la qualité de la classification,
   - description des classes obtenues.

1.7 Approche Paramétrique versus non-paramétrique :

1- Non paramétrique :

Les approches dites non paramétriques (classification hiérarchique, méthode des centres mobiles) basée sur l'hypothèse : plus deux individus sont proches, plus ils ont de chances de faire partie de la même classe, en plus ce que distingue cette approche est qu'on ne fait aucune hypothèses sur le modèle que suivent les données, C'est le cas des plus proches voisins (k-PPV), donc il suffit de trouver les propriétés de convergence quand le nombre de données est grand.

2- Paramétrique :« Probabilistes »

La seconde grande famille des méthodes de classification, ce sont les approches probabilistes, utilisent une hypothèse sur la distribution des individus à classifier ,c'est-à-dire, on suppose que l'on connaît la forme du modèle qui a généré les données . Par exemple, on peut considérer que les individus de chacune des classes suivent une loi normale. Le problème qui se pose, est de savoir déterminer ou estimer les paramètres des lois (moyenne, variance) et à quelle classe les individus ont le plus de chances d'appartenir à partir de l'ensemble d'apprentissage.

Les paramètres d'une loi peuvent être déterminés de maintes façons, C'est le cas par exemple des classifications bayésiennes ou encore l'algorithme espérance-maximisation. (ce qu'on va l'attribuer tout un chapitre)

1.8 Les types des méthodes de Classifications :

On peut grouper les méthodes classificatoires en deux grandes familles , cette fois-ci , on prends en considération l'intervention ou non d'un « attribut classe » au fur et à mesure du processus de la classification, ces deux types sont : « supervisée (Classement)» et « non supervisée(Classification, Clustering) »,

1. supervisé (classement) : groupes fixés, exemples d'objets de chaque groupe.
2. non supervisé (classification) : on ne connaît pas de groupe.

Cependant, Il existe d'autres types de classification qui s'appuient sur d'autres types de méthodes d'apprentissages comme « l'apprentissage semi-supervisé » et « l'apprentissage par renforcement ». En effet, l'apprentissage semi-supervisé est un bon compromis entre les deux types d'apprentissage « supervisé » et « non-supervisé », car il permet de traiter un grand nombre de données sans avoir besoin de toutes les étiqueter, et il profite des avantages des deux types mentionnés. Alors que L'apprentissage par renforcement est fort utilisé dans le cas d'apprentissage interactif,
Dans le reste de ce chapitre nous ne présenterons que le premier type « supervisé ». Pour le deuxième type (voir chapitre 3).

1.9 METHODE SUPERVISEE «Classement» ou « DISCRIMINATION» :

1.9.1 Définition :

Le «classement» est une méthode supervisée qui consiste à définir une fonction qui attribue une ou plusieurs classes à chaque donnée. Dans cette approche on suppose qu'un expert fournit auparavant les étiquettes pour chaque donnée, les étiquettes sont des classes d'appartenance. Selon **[Govaert, 2003]** : «(la classification supervisée (appelée aussi classement ou

classification inductive) a pour objectif « d'apprendre » par l'exemple. Elle cherche à expliquer et a prédire l'appartenance de documents à des classes connues a priori. Ainsi c'est l'ensemble des techniques qui visent à deviner l'appartenance d'un individu à une classe en s'aidant uniquement des valeurs qu'il prend)»

1.9.1.1 Principe :

la conception supervisée d'un classifieur à C classe (ensemble fini de classe c_i) est le fait de classifier N objets (x_i) de même nature (des phonèmes, caractères manuscrits,...) sachant que ces N objets sont supposés avoir été préalablement « étiquetés » par un « superviseur » en C ensembles qui forme un ensemble d'apprentissage.
Le superviseur n'est qu'un Classifieur en lequel on a confiance (expert humain , caractère répétitif , le système visuel humain ...) , donc notre système de la classification supervisée va être conçu en basant sur les exemples du superviseur (l'ensemble d'apprentissage où pour tout exemple on connaît à priori sa classe.)
c'est-à-dire, on cherche à prédire si un objet (élément) « x_i» de la base de données, décrit par un ensemble de descripteurs «**d**», appartient ou non à une classe « c_j » parmi N Classes, pour le faire, on a un ensemble d'apprentissage décrit par :

$$A = (x_1, c_2), (x_2, c_4), (x_3, c_2) \ldots (x_i, c_j) / x_i \in R^d, c_j \in C \tag{1.1}$$

Donc pour chaque objet x_i de l'ensemble de données, on peut connaît sa classe a priori c_j. La classification supervisée tente de chercher, à partir des données de A , une fonction de décision Γ qui va associer à tout nouveau élément x_i de test une classe c_j, puis on compare ce que nous a donné cette fonction avec la classe connai a priori de cet élément , de sorte à minimiser les mauvais classements ($\Gamma(x_i) \neq C_j$).
Donc l'objectif est de chercher à prédire la classe de toute nouvelle donnée.

1.9.2 Exemple :

On pourrait donner l'exemple le plus connu : problèmes d'aide ou diagnostic médical, où les superviseurs sont généralement les médecins afin de noter la classe des objets de l'ensemble d'apprentissage à partir des remarques constatées. Ou bien l'exemple d'un tableau où le dernier descripteur (Jouer) représente la classes des exemples

Numéro	Ensoleillement	Température (°F)	Humidité (%)	Vent	Jouer
1	soleil	75	70	oui	oui
2	soleil	80	90	oui	non
3	soleil	85	85	non	non
4	soleil	72	95	non	non
5	soleil	69	70	non	oui
6	couvert	72	90	oui	oui

TAB. 1.2 – Base d'exemples "jouer un match" pour la classification

Ce tableau contient les données de l'apprentissage pour la classification dont chaque instance est labélisée par "Oui" et "Non". Ce qui permet de construire un modèle de classification permettant prédire si ça marche pour jouer un match ou non ?
La déduction se fait par rapport à l'apprentissage sur le jeu de données.

Dans cet exemple on parle de « classification binaire », car on classifie les données en deux classes (|C| =2), et idem pour « n-aire » si on classifie les données en n classes.

Nous allons présenter quelques techniques classiques de classification supervisée : les algorithmes étudiés sont présentés dans un ordre de difficulté croissant : l'algorithme « k plus proches voisins », « la classification bayésienne » .

1.9.3 Les k plus proches voisins (K-PPV) :

La méthode des plus proches voisins (noté parfois k-PPV ou k-NN pour -Nearest-Neighbor) consiste à déterminer pour chaque nouvel individu que l'on veut classer, la liste des plus proches voisins parmi les individus déjà classés. L'individu est affecté à la classe qui contient le plus d'individus parmi ces plus proches voisins. Cette méthode nécessite de choisir une distance, la plus classique est la distance euclidienne (voir chapitre3.section6), et le nombre de voisins à prendre en compte.

Cette méthode supervisée et non-paramétrique est souvent performante. De plus, son apprentissage est assez simple, car il est de type apprentissage par coeur (on garde tous les exemples d'apprentissage). Cependant, le temps de prédiction est très long, car il nécessite le calcul de la distance avec tous les exemples, mais il existe des heuristiques pour réduire le nombre d'exemples à prendre en compte **[Berrani &al, 2002]** .

1.9.3.1 Algorithme :

1. initialisation , choix de :

   - Nombre de classes, Valeur de k, exemples initiaux, mesure de similarité.

2. pour chaque vecteur d'objet à classer :

   - mesurer la distance du vecteur avec tous les autres déjà classés

   - déterminer la liste des k vecteurs les plus proches de lui (k-ppv)

   - déterminer la classe la plus représentée dans la liste des k-ppv et affecter notre vecteur à cette classe.

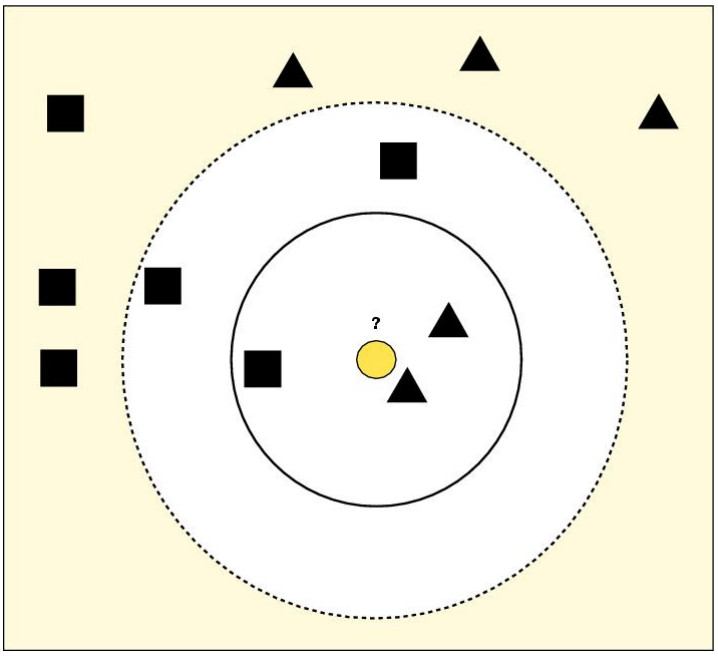

FIG. 1.6 – illustration des k-ppv d'un point

(la décision sera pour l'affecter à la classe majoritairement présente dans les k-ppv).

1.9.3.2 Discussion :

Ce qu'on peut remarquer sur cette méthode, c'est le cout de calcule qu'elle impose au fur et a mesure de ce processus de classification, car ce cout augmente avec chaque vecteur qu'on vient de classifié, plus on ajoute des nouveaux vecteurs déjà classés , plus que ce coute augmente ce qui explique le temps d'exécution qu'elle prend pour classifier. en plus de la sensibilité de cet algorithme à l'initialisation des paramètres d'entrées (le choix de k , la distance utilisée ..) alors

il faut que lors de la sélection des paramètres d'entrées que ces derniers respectent certaines contraintes (comme que k ne soit pas un multiple du nombre de classes pour éviter une surreprésentation d'une classe par apport à une autre .malgré ces points , k-ppv reste une des méthodes les plus utilisées grâce à sa simplicité et robustesse et son caractère de généralisation a partir d'un nombre éminent de données d'apprentissage .

1.9.4 La classification bayésienne :

Un classifieur probabiliste linéaire simple basée sur le théorème de Bayes qui suppose que les descripteurs (attributs) qui décrivent les objets de l'ensemble d'apprentissage sont indépendants.

1.9.4.1 principe

l'ensemble d'apprentissage «A» est connue et chaque objet est étiqueté par sa classe « C_k», l'objectif est de chercher à classer un nouveau objet « X_{new} » non encore étiqueté. Le Classifieur bayésien va choisir la classe « C_k » qui a la plus grande probabilité, on parle de règle **MAP** (maximum a posteriori)**[1.2]** :

$$C_{MAP}=argmax_{C_k\in c}P(C_k|X_{new})=argmax_{C_k\in c}\frac{P(X_{new}|C_k)P(C_k)}{P(X_{new})}=argmax_{C_k\in c}P\left(X_{new}|C_k\right)P(C_k) \tag{1.2}$$

Taille (cm)	Poids (kg)	Pointure (cm)	Sexe
182	81.6	30	masculin
180	86.2	28	masculin
170	77.1	30	masculin
180	74.8	25	masculin
152	45.4	15	féminin
168	68.0	20	féminin

Taille (cm)	Poids (kg)	Pointure (cm)	Sexe
183	59	20	???

FIG. 1.7 – la classification d'un nouvel exemple selon naïf bayes

donc Il nous faut estimer les probabilités P(C_k) et P(X_{new}/C_k) à partir des données d'apprentissage. Les probabilités a priori des classes P(C_k) , peuvent être estimées facilement par :

$p(C_k)$ $\frac{nombre\,d'expression\,d'apprebtissage\,dans\,la\,classe\,C_k}{le\,nombre\,totale\,de\,documents\,dans\,l'ensemble\,d'apprentissage}$

Maintenant pour estimer les valeurs de P(X_{new}/C_k) , puisque les descripteur(attributs) de « X_{new}» sont indépendants , alors on aura grace aux théories d'indépendance bayesienne entre les variables **[1.2]** :

$$P(X_{new} \mid C_k) = P(f_1 \mid C_k)\,P(f_2 \mid C_k)\ldots P(f_{n.\mathscr{F}} \mid C_k) \tag{1.3}$$

Ou les « fi » sont les attributs qui décrivent l'ensemble de données , sachant que :

$$p(C, F_1, \ldots, F_n) = p(C) \prod_{i=1}^{n} p(F_i \mid C) \tag{1.4}$$

Et pour estimer les paramètres d'une loi de probabilité relative à une caractéristique précise, il est nécessaire de présupposer le type de la loi en question (par exemple loi normale).

1.9.4.2 Discussion

Le Classifieur naïf de Bayes est très performant même avec peu de données, car il fait souvent de bonnes hypothèses sur la distribution des données avec un peu de données d'entraînement afin d'estimer les paramètres nécessaires à la classification (moyennes et variances) **[Rennie & al. , 2003]**, mais Lorsque le nombre de descripteurs est grand, il est parfois impossible de construire ce modèle sur des tableaux de probabilités.

1.10 Conclusion

Nous avons vu une généralité sur les conceptions des méthodes de classification et un aperçu superficiel sur les principes de la première grande approche qui infère à partir d'un échantillon d'exemples classés une procédure (fonction de décision) de classification des nouveaux exemples non étiquetés. La Discrimination (ou les méthodes supervisées) peut être basée sur des hypothèses probabilistes (Classifieur naïf de Bayes, méthodes paramétriques) ou sur des notions de proximité (plus proches voisins) ou bien encore sur des recherches dans des espaces d'hypothèses (arbres de décision, réseaux de neurones).

Certes l'approche supervisée est très utilisée pour les raison et les avantages qu'on a mentionné pour chaque méthode , néanmoins il reste qu'il ya un manque de stratégies pour les exemples d'auto-apprentissage (c'est-à-dire , d'apprendre a partir d'une base sans aucune connaissance préalable) que les méthodes supervisée ne peuvent pas traiter , dans ce cadre vient la deuxième approche des méthodes de classification, qui est : l'approche non-supervisée (ou spécifiquement « la classification automatique ») .

Chapitre 2

La classification automatique « Clustering »

2.1 Introduction

Comme nous avons pu le voir dans le premier chapitre qu'il ya deux grandes approches en classification : la discrimination (classement) et la classification automatique (clustering), dans ce chapitre nous détaillerons les méthodes du deuxième type « clustering » qui est une des techniques statistiques largement utilisées dans la Fouille de Données. il est dans un cadre d'apprentissage non supervisé, qui tente d'obtenir des informations sans aucune connaissance préalable, ce qui n'est pas le cas de l'apprentissage supervisé.

la question principale autour de laquelle s'articulera le travail du Clustering est de savoir d'imiter le mécanisme humaine d'apprentissage sans aucune information disponible auparavant, en établant des méthodes qui permettent d'apprendre à partir d'un certain nombre de données et de règles (d'exemples), selon certains caractéristiques sans aucune expertise ou intervention requise. En effet, ce processus requit certains traitements ou combinaison avec d'autres méthodes, en pre- ou en post-processing,surtout pour une grande masses de données, pour bien réaliser entièrement sa tâche de classification, L'ensemble des techniques de traitement est souvent regroupé sous le terme de «fouille de données».

Dans ce chapitre, nous nous intéressons qu'aux techniques de classification automatique **(clustering)** et nous montrons, quels sont leurs avantages et difficultés (voir section : Discussion). En tentant décrire quelques remèdes et de présenter l'avantage de Clustering dans le domaine de traitement de données non-étiquetées (sans connaissance préalable).

2.2 Définition

Le Clustering aussi connu sous nom **(Segmentation)** est un regroupement en classes homogènes consistent à représenter un nuage des points d'un espace quelconque en un ensemble de groupes appelé **Cluster**.

C'est un traitement sur un ensemble d'objets qui n'ont pas été étiquetés par un superviseur. Généralement lié au domaine de l'analyse des données comme ACP (analyse linéaire en composantes principales) **[Saporta 1900] [Lebart et al. 200]**, ce type de méthodes vise à répondre au problème de : diminution de la dimension de l'espace d'entrée, ou pour le groupement des objets en plusieurs catégories (clusters) non définies à l'avance. Parmi les

méthodes qu'on peut trouver dans ce type de classification : les cartes auto-organisatrices de kohonen **[kohonene 1982]**, GMM . . .etc

Un «**Cluster**» est donc une collection d'objets qui sont «similaires» entre eux et qui sont «dissemblables » par rapport aux objets appartenant à d'autres groupes. On peut voir cette définition clairement graphiquement dans l'exemple suivant :

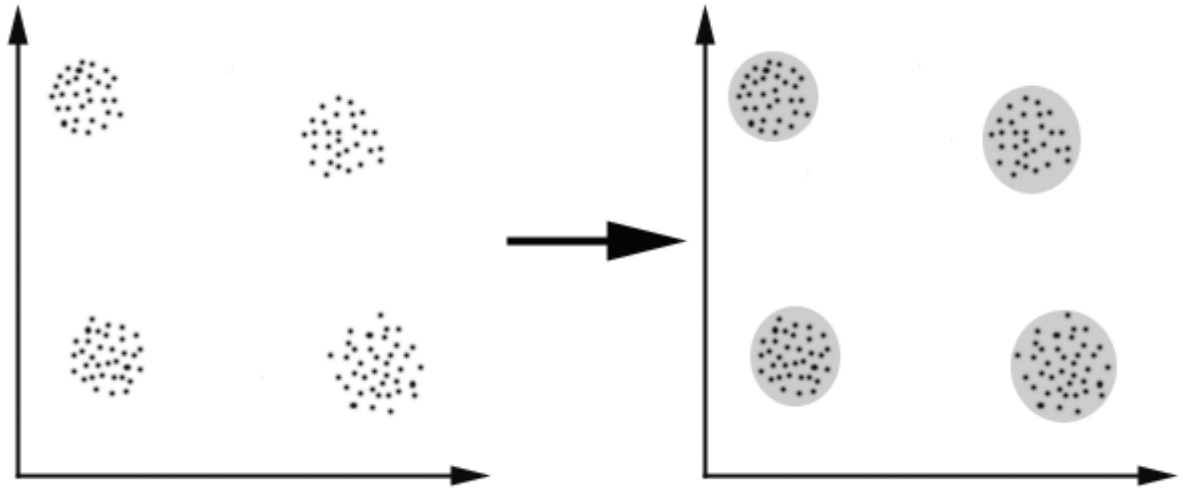

FIG. 2.1 – llustration de regroupement en clusters

Dans ce cas, il est très facile pour une personne d'identifier 4 Clusters dans lesquels les données (nuage des points) peuvent êtres divisées, le critère de similarité est la distance : deux ou plusieurs objets appartiennent au même cluster s'ils sont «proches», bien sur cela dépend d'une distance donnée (dans ce cas la distance géométrique).

Un autre type de regroupement est le clustering conceptuel : deux ou plusieurs objets appartiennent au même cluster si celui-ci définit un concept commun à tous les objets. En d'autres termes, les objets sont regroupés en fonction de leur adéquation aux concepts descriptifs, et non pas en fonction de mesures de similarité simple.

2.3 Principe général

Contrairement à la classification (méthodes supervisées), on ne possède pas des connaissances a priori sur les classes prédéfinies des éléments. Donc La division des objets dans les différents groupes (clusters) se procède en se basant sur le calcul de similarité entre les éléments.

Alors que l'objectif des méthodes du Clustering est de grouper des éléments proches dans un même groupe de manière à ce que deux données d'un même groupe soient le plus similaires possible et que deux éléments de deux groupes différents soient le plus dissemblables possible **[Hartigans, 1975]**.

Mathématiquement, on a un ensemble X de N données décrites chacune par leurs P attributs. Donc Le Clustering consiste à créer une partition ou une décomposition de cet ensemble en sous parties (clusters) telle que :
* Les données appartenant au même groupe se ressemblent,
* Les données appartenant à deux groupes différents soient peu ressemblantes.

2.3.1 Exemple

On utilise souvent ce type de classification en traitement d'images pour fixer les divers objets qu'elles contiennent (segmentation) : routes, villes, rues , des organes humaines (pour les images médicales) . . .

2.4 Les exigences de Clustering

Les principales exigences qu'un algorithme de clustering doit répondre sont les suivantes :

- ✦ Evolutivité des clusters
- ✦ traiter les différents types d'attributs
- ✦ découvrir les clusters de forme arbitraire
- ✦ exigences minimales pour la connaissance du domaine afin de déterminer les paramètres d'entrée.
- ✦ capacité de composer avec le bruit et les valeurs manquantes traiter les dimensionnalités élevées. l'intelligibilité et la convivialité.

2.5 Les types de Clustering

Il existe deux grands types du clustering :

A/ le clustering hiérarchique : **d'agglomération («bottom-up»)**

B/ le clustering non-hiérarchique : **de division («top-down»)**

Dans le premier cas, on décompose l'ensemble d'individus en une arborescence de groupes.

Dans le 2ème, on décompose l'ensemble d'individus en K groupes , les algorithmes de ce type peuvent aussi être utilisés comme algorithmes de division dans le clustering hiérarchique.

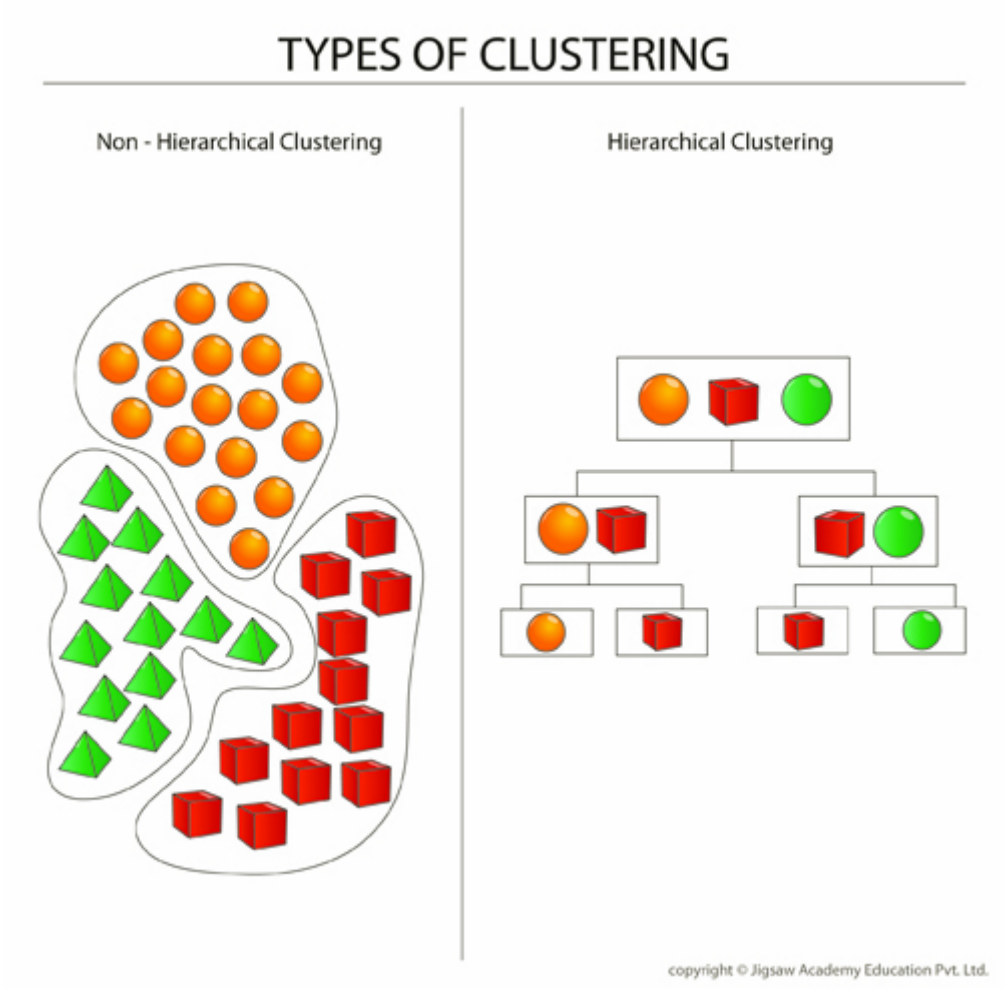

FIG. 2.2 – les deux types de clustering non-hiérarchique/hiérachique

Cependant dans certains ouvrages on classifie les types des Algorithmes de clustering en 4 groupes à cause des méthodes qui ne respectent plus les normes du premier classement comme

le cas de la règle « Chaque objet doit appartenir à un seul groupe. » alors que les versions floues la tempèrent et permettent à un objet d'appartenir à plusieurs classes selon un certain degré.

Les 4 types sont :

1. Clustering exclusif
2. Overlapping Clustering (fuzzy clustering)
3. Clustering Hiérarchique
4. Clustering probabiliste

Dans le premier cas, les données sont regroupées d'une manière exclusive, de sorte que si une donnée certaine appartient à un amas définie alors il ne pourrait pas être inclus dans un autre cluster. Un simple exemple de cela est montré dans la figure ci-dessous, où la séparation des points est définie par une ligne droite sur un plan bidimensionnel.

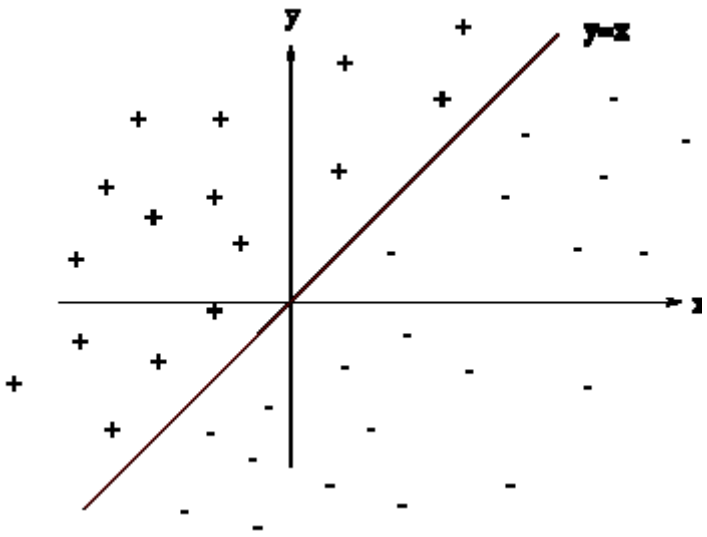

FIG. 2.3 – Exemple d'un problème de discrimination à deuxclasses, avec une séparatrice linéaire : la droite d'équation y=x. Le problème est linéairement séparable.

Au contraire le second type, le regroupement overlapping, utilise des ensembles flous aux données de cluster, de sorte que chaque point peut appartenir à deux ou plusieurs groupes avec différents degrés d'appartenance. Dans ce cas, les données seront associées à une valeur d'une composition appropriée.

Comme nous l'avons dit, un algorithme de clustering hiérarchique est fondé sur l'union entre les deux plus proches clusters cad : consiste à trouver des clusters successifs utilisant des clusters précédemment établis. La première condition est de mettre, au début, chaque objet

dans un cluster distinct et les fusionner en clusters successivement plus grand. Après quelques itérations on atteint le final Cluster voulu qui regroupe tous les sous-clusters (sous-partitions).

Enfin, le dernier type de regroupement utilise une approche complètement probabiliste basant sur la probabilité d'appartenance aux clusters.

2.6 Les algorithmes de Clustering :

Dans ce qu'il suit nous présentons quelques algorithmes de Clustering, voilà quelques exemples :

1. K-means
2. Fuzzy C-means
3. Hierarchical clustering
4. Mixture of Gaussians (Expectation maximisation)

Chacun de ces algorithmes appartient à l'un des types de clustering énumérés ci-dessus. Par exemple , K-means est un algorithme de clustering exclusif ,pendant que Fuzzy C-means est un algorithme de Overlapping Clustering, alors que clustering hiérarchique il est claire qu'il s'agit de troisième type de clustering, et enfin Mélange de Gaussien est un algorithme de clustering probabiliste. Nous allons discuter et définir les principes de ces méthode de clustering dans quelques lignes.

Dans ce qui suit, nous analysons le cadrage théorique de chaque méthode, concernant la 4ème méthode (qui est la méthode cible de ce travail) nous l'allouons spécialement tout un chapitre **(voir chapitre 3)**.

2.6.1 K-means

L'algorithme k-means mis au point par McQueen en 1967**[MacQueen, 1967]**, un des plus simples algorithmes d'apprentissage non supervisé , appelée **algorithme des centres mobiles [Benzécri, 1973] [Celeux et al, 1989]** , il attribue chaque point dans un cluster dont le centre (centroïde) est le plus proche. Le centre est la moyenne de tous les points dans le cluster , ses coordonnées sont la moyenne arithmétique pour chaque dimension séparément de tous les points dans le cluster cad chaque cluster est représentée par son centre de gravité.

2.6.1.1 Principe

L'idée principale est de définir les k centroïdes arbitraires c_1, c_2,..., c_k (k le nombre de clusters fixé a priori, chaque c_i représente le centre d'une classe), Ces centroïdes doivent être placés dans des emplacements différents. Donc, le meilleur choix est de les placer le plus possible éloignés les uns des autres. La prochaine étape est de prendre chaque point appartenant à l'ensemble de données et l'associer au plus proche centroïde. Càd Chaque classe Si sera représentée par un ensemble d'individus les plus proches de son c_i , Les nuées dynamiques sont une généralisation de ce principe, où chaque cluster est représenté par un noyau mais plus complexe qu'une moyenne.

Lorsqu'aucun point n'est en attente, la première étape est terminée et un groupage précoce est fait. À ce point nous avons besoin de recalculer les k nouveaux centroïdes mi des groupes issus de l'étape précédente qui vont remplacer les c_i (m_j est le centre de gravité de la classe S_j, calculé en utilisant les nouvelles classes obtenues). Après, on réitère Le processus jusqu'à atteindre un état de stabilité où aucune amélioration n'est possible, nous pouvons constater que les k centroïdes changent leur localisation par étape jusqu'à plus de changements sont effectués. En d'autres termes les centroïdes ne bougent plus.

2.6.1.2 Algorithme

Choisir k moyennes c_1, c_2,..., c_k initiales (par exp au hasard)

1. Répéter :

   affectation de chaque point à son cluster le plus proche :

$$S_i^{(t)} = \left\{ x_i : \| x_j - m_i^{(t)} \| \leq \| x_j - m_{i*}^{(t)} \| \ for all \ i^* = 1, ..., k \right\} \tag{2.1}$$

   mettre à jour la moyenne de chaque cluster

$$m_i^{(t+1)} = \frac{1}{| S_i^{(t)} |} \sum_{x_j \in s_i^{(t)}} x_j \tag{2.2}$$

2. Jusqu'à : atteindre la convergence quand il n'y a plus de changement.

Fin.

2.6.1.3 Discussion

Cette méthode est la plus populaire des méthodes de clustering , malgré ça , un de ses problèmes majors est qu'il tend à trouver des classes sphériques de même taille. en plus K-means est connu par sa complexité de « NP-difficile ». Il est donc fréquemment faire appeler une heuristique en pratique, ce qui explique qu'elle est convergente et surtout avantageuse du point de vue calcul mais elle dépend essentiellement de la partition initiale (Des initialisations différentes peuvent mener à des clusters différents «problèmes de minima locaux ») cela risque d'obtenir une partition qui ne soit pas optimale pourtant qu'elle donne surement une partition meilleure que la partition initiale. De plus, la définition de la classe se fait à partir de son centre, qui pourrait ne pas être un individu de l'ensemble à classer, d'où le risque d'obtenir des classes vides.

2.6.2 méthode Fuzzy C-means :

2.6.2.1 Principe

Fuzzy C-means (FCM) est une méthode de clustering qui permet à un objet de données d'appartenir à deux ou plusieurs clusters. Cette méthode dérivée de l'algorithme c-means **[Ball et Hall,1967]**, identique à l'algorithme k-means décrit précédemment , elle a été développée par Dunn **[Dunn, 1973]**en 1973 et améliorée par Bezdek **[Bezdek, 1981]** en 1981, est fréquemment utilisée dans la reconnaissance des formes. Il est basé sur la minimisation de la fonction objective suivante :

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} \parallel x_i - c_j \parallel^2 \qquad 1 \leq m < \infty \tag{2.3}$$

où m est un nombre réel (> 1), U_{ij} est le degré d'appartenance de x_i dans le j ème Cluster, x_i est le ième élément des données mesurées, c_j est le centre d'un cluster et $\|*\|$ est toute norme exprimant la similarité entre les données mesurées et le centre. Ce Partitionnement logique floue (fuzzy) est réalisé grâce à une optimisation itérative de la fonction objectif indiqué ci-dessus, avec la mise à jour de l'appartenance u_{ij} et les centres des clusters c_j.

On peut résumer la différence entre fuzzy C-means et k-means dans la fonction d'appartenance d'un nuage de points dans deux clusters dans l'exemple suivant :

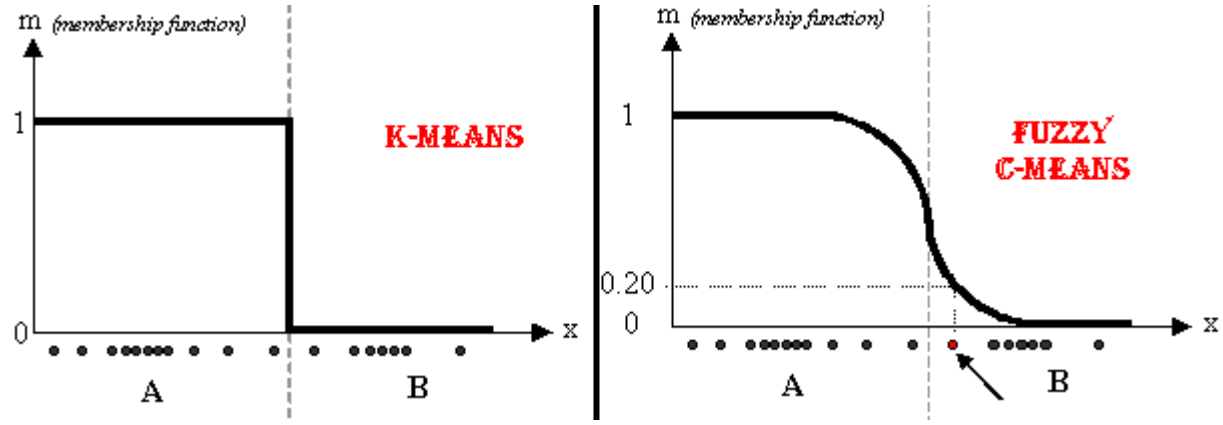

FIG. 2.4 – Fonction d'appartenance dans kmeans/Fuzzy C-means

Dans le cas de k-means un objet ne peut pas appartenir dans deux clusters Simultanément, ce qui explique la Discrimination binaire entre les clusters mais en FCM il est possible qu'un objet appartient à deux ou plusieurs clusters selon différents pourcentages cad que les données sont liés à chaque groupe par le biais d'une fonction d'appartenance, ce qui représente le comportement floue de cet algorithme. Pour le faire, nous devons simplement construire une matrice appropriée nommée U dont les facteurs sont des nombres entre 0 et 1, et représentent le degré d'appartenance entre les centres de données et des clusters.

$$U_{MC} = \begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ .. & .. \\ 0.9 & 0.1 \end{bmatrix}$$

Il est également important de noter que les initialisations différentes causent différentes évolutions de l'algorithme. En fait, il pourrait converger vers le même résultat, mais probablement avec un nombre différent d'itérations.

2.6.2.2 Algorithme

(il ya des parties des equations cathé il faut les récriture

1. Initialiser U = $[u_{ij}]$ matrice $U_{(0)}$.
2. A la k-étape : calculer les centres $C_{(k)} = [c_j]$ avec $U_{(k)}$

   $c_j = \sum_{i=1}^{N} u_{ij}^m . x_i \sum_{i=1}^{N} u_{ij}^m$

3. Mise à jour de $U_{(k)}$, $U_{(k+1)}$
   $u_i^{j=1} C_{k=1} \| x_i - c_j \| \| x_i - c_k \| 2m - 1$
4. Si $\|U_{(k+1)} - U_{(k)}\| < \varepsilon$ $(0<\varepsilon<1)$, alors STOP, sinon le retour à l'étape 2.

2.6.2.3 Discussion

Une méthode que son caractère hybride (la notion de centre de gravité et la notion Floue) le rend simple, rapide . La FCM exige des paramètres d'entrées, et que la matrice de partition floue, doit être initialisée d'une manière appropriée . Ces paramètres sont choisis d'une façon arbitraire, ces paramètres ont une grande influence sur le résultat attendu. Ce qu'il nous oblige de faire une étude approprié sur les données en entrée et le regroupement que l'on souhaite obtenir.

Ce type d'algorithme est fort utilisé en traitement d'images**[Gesu, 1988] [Hadi &benmhammed, 2005] [oppner &al., 2000]** afin d'identifier des zones similaires (contours, coins, région homogènes…).

2.6.3 Méthodes hiérarchiques

Le processus basique des méthodes hiérarchiques a été donné par **[Johnson, 1967] [Lance & Williams, 1967]**, Ce type de clustering consiste à effectuer une suite de regroupements en Clusters de moins en moins fines en agrégeant à chaque étape les objets (simple élément) ou les groupes d'objets (un Cluster-partition-) les plus proches. Ce qui nous donne une arborescence de clusters**[Celeux &al., 1989]**. Cette approche utilise la mesure de similarité pour refléter l'homogénéité ou l'hétérogénéité des classes.

2.6.3.1 Principe

Son principe est simple, initialement chaque individu forme une classe, soit **n** classes , donc on cherche à réduire ce nombre de classe **newnbrclss < n** itérativement de sorte que dans chaque étape on fusionne deux classes ensemble (Les deux classes choisies pour être fusionnées sont celles qui sont les plus "proches" en fonction de leur dissimilarité) ou ajouter un nouveau élément à une classe (un élément appartient à une classe s'il est plus proche de cette classe que de toutes les autres) La valeur de dissimilarité est appelée **indice** d'agrégation. Qui commence dans la première itération faible, et croîtra d'itération en itération.

Parmi les algorithmes plus connus de ce type : La classification ascendante hiérarchique (CHA) où le mot ascendante est utilisé pour désigner qu'elle part d'une situation dont tous les individus représentent des clusters à part entière, puis on cherche les rassembler en classes de plus en plus grandes. Ainsi Le qualificatif "hiérarchique" désigne le fait qu'elle produit une hiérarchie, (une amélioration a été proposée en 2002 par P. Bertrand , appelée Classification Ascendante 2-3 Hiérarchique).

2.6.3.2 Algorithme de CHA

1. Initialisation :
   Chaque individu est placé dans son propre cluster, Calcul de la matrice de ressemblance M entre chaque couple de clusters (ici les points)
2. Répéter :
   * Sélection dans M des deux clusters les plus proches C_i et C_j
   * Fusion de C_i et C_j par un cluster C_G plus général
   * Mise à jour de M en calculant la ressemblance entre C_G et les clusters existants
   Jusqu'à fusionner les 2 derniers clusters.

Dans la figure suivante, on représente une illustration du principe de CHA et la hiérarchie finale obtenue où Les liens hiérarchiques apparaissent clairement.

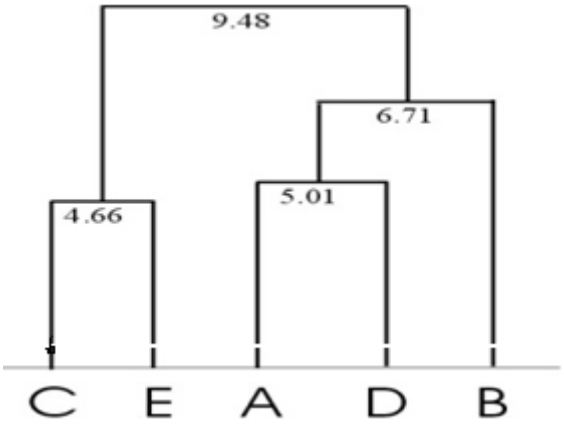

FIG. 2.5 – le dendrogramme de la hiérarchie H de la suite de partitions d'un ensemble {a, b, c, d, e} :

note : Un dendrogramme = la représentation graphique d'une classification ascendante hiérarchique sous forme d'un arbre binaire

2.6.3.3 Discussion

la CAH ne nécessite pas de connaître le nombre de clusters a priori. De plus, il n'y a pas de fonction d'initialisation, ainsi une seule construction d'un cluster (équivalent à une itération pour les méthodes de partitionnement).

En ce qui concerne généralement les méthodes hiérarchiques le problème qu'on peut rencontrer réside dans la sélection d'une ultra-métrique (distance pour calculer la similarité entre clusters) soit la plus proche de la métrique utilisée pour les individus, car ces méthodes sont heuristiques, pour cela ya plusieurs techniques permet de le faire : Saut minimal (single linkage) ; Saut maximal (complete linkage) ; Saut moyen ; Barycentre...
une autre faiblesse est : la complexité de temps d'au moins $O(n^2)$, où n est le nombre d'objets au total, ainsi qu'on pourrait jamais défaire ce qui a été fait précédemment.
Il est difficile parfois d'apporter une justification aux méthodes hiérarchique (CAH, CDH..), Cependant, dans **[Kamvar &al., 2002]** , une interprétation probabiliste de la CAH, basée sur une estimation par maximum de vraisemblance des modèles de mélange, est proposée comme solution pour mieux interpréter les résultats.

Un autre inconvénient de ce type de méthodes est qu'une action effectuée (fusion ou décomposition), elle ne peut être annulée. Celà permet de réduire le champ d'exploration, mais une telle astuce ne peut corriger une décision erronée.

afin améliorer la qualité d'une classification hiérarchique, on peut profiter de deux techniques :

- ✦ analyser attentivement les liens entre objets à chaque étape **[Guha et al., 1998] et [Karypis et al., 1999]**.
- ✦ améliorer la partition obtenue avec une méthode de deuxième type de clustering (partitionnement) **[Zhang et al., 1996]**.

2.7 Mesure de similarité

Pour comparer homogénéité ou le ressemblance, la similarité entre deux objets (points,images , classes , phonème ..), il faut pouvoir mesurer la similarité (ou la dissimilarité) entre eux. Nous allons décrire maintenant des mesures de similarité pour prouver la similarité entre les objets, selon **[Bisson, 2000], «tout systeme ayant pour but d'analyser ou d'organiser automatiquement un ensemble de données ou de connaissances doit utiliser, sous une forme ou une autre, un opérateur de similarité dont le but est d'établir les ressemblances ou les relations qui existent entre les informations manipulées»**.

Donc la similarité est une partie importante de la définition d'une méthode de clustering, elle consiste en effet à définir et formaliser une mesure de similarité adaptée aux caractéristiques des données. Si les composantes des vecteurs de données d'instance sont toutes dans les mêmes unités physiques alors il est possible que la distance euclidienne est suffisante pour réussir à grouper les données similaires. Cependant, même dans ce cas, la distance euclidienne peut parfois être trompeuse. La Figure ci-dessous illustre ceci avec un exemple vu selon la largeur et la hauteur d'un objet. Malgré que les deux mesures aient été prises dans les mêmes unités physiques,

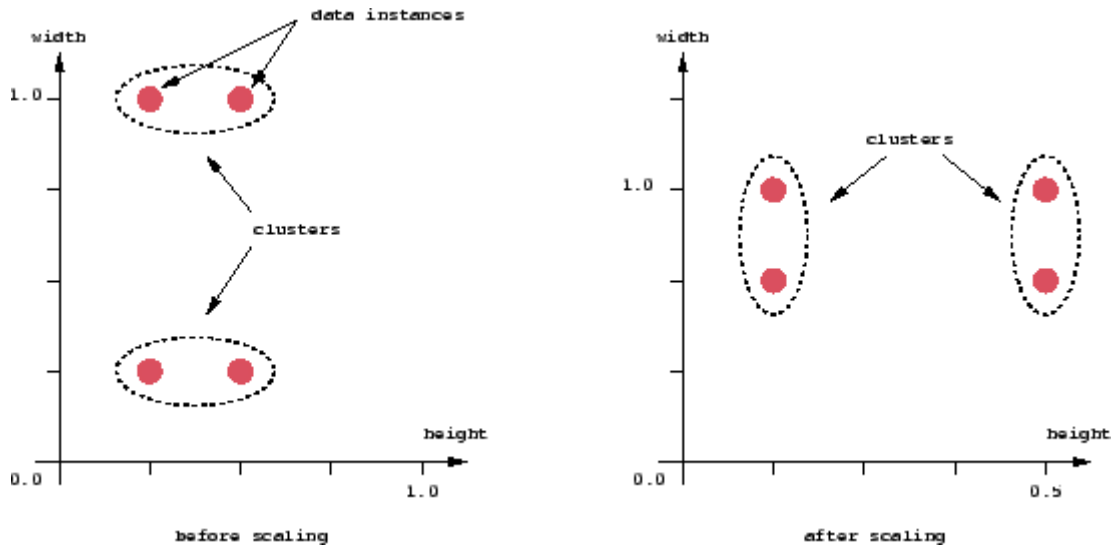

FIG. 2.6 – différents écaillages peuvent conduire à différents clustering

Donc une décision éclairée doit être faite quant à la mise à l'échelle relative. Comme le montre la figure, différents écaillages peuvent conduire à différents clustering.

2.7.1 Vocabulaire

Il est à noter qu'il ya deux concepts pour exprimer la notion de proximité entre les objets à classifier :

1. Mesure de dissimilarité DM : plus la mesure est faible plus les points sont similaires (distance).
2. Mesure de similarité SM : plus la mesure est grande, plus les points sont similaires.
3. On parle souvent de « distances » en désignant une mesure de similarité, lorsque ces mesures ont les propriétés de non-négativité, réflexivité, symétrie (la distance entre l'objet A à B est la même que la distance de B à A) et qui respectent l'inégalité triangulaire.

Il existe un grand nombre de mesures de similarité, dans ce qui suit, nous présentons quelques unes des fonctions entre deux objets **d(x1 ; x2)**.

2.7.2 Fonctions de similarité

1. **La distance euclidienne :** (aussi appelée la distance à vol d'oiseau) Un rapport de clusters analysis en psychologie de la santé a conclu que la mesure de la distance la plus courante dans les études publiées dans ce domaine de recherche est la distance euclidienne ou la distance au carré euclidienne.

$$d^2(x_1,x_2) = \sum_i (x_{1i} - x_{2i})^2 = (x_1 - x_2)(x_1 - x_2)' \tag{2.4}$$

2. **La distance de Manhattan :** (appelée aussi taxi-distance)

$$d^2(x_1,x_2) = \sum_i | x_{1i} - x_{2i} | \tag{2.5}$$

3. **La distance de Mahalanobis :** corrige les données pour les différentes échelles et des corrélations dans les variables, L'angle entre deux vecteurs peuvent être utilisés comme mesure de distance quand le regroupement des données de haute dimension. Voir l'espace produit scalaire.

$$\begin{gathered} d^2(x_1,x_2) = (x_1 - x_2) C^{-1} (x_1 - x_2)' \\ (C = covariance) \end{gathered} \tag{2.6}$$

4. **la distance de Sebestyen :**

$$\begin{gathered} d^2(x_1,x_2) = (x_1 - x_2) W (x_1 \vdash x_2)' \\ (W = matrice\, diagonale\, de\, pondération) \end{gathered} \tag{2.7}$$

5. **La distance de Hamming :** mesure le nombre minimum de substitutions nécessaires pour changer un membre dans un autre. Elle permet ainsi , de quantifier la différence entre deux séquences de symboles, généralement utilisée dans le cas des valeurs discrètes (vecteurs)

$$d(a,b) \sum_{i=0}^{n-1} (a_i \oplus b_i) \tag{2.8}$$

Exemple : Considérons les suites binaires suivantes :

$$a = (\ 0\quad 0\quad 0\quad 1\quad 1\quad 1\quad 1)\ et\, b = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} alors\, d = 1+1+0+0+1+0+0 \tag{2.9}$$

La distance entre a et b est égale à 3 car 3 bits diffèrent.

6. **Distances entre distributions :** La similarité entre distributions consiste à déterminer si deux distributions peuvent être issues de la même distribution de probabilités.
   Le test statistique du X^2 chi-square) permet de décider si deux vecteurs $\overrightarrow{x}$ et $\overrightarrow{y}$ sont engendrés par la même distribution. La version symétrique du test est :

$$x^2\left(\overrightarrow{x},\overrightarrow{y}\right)=\sum_{i=1}^{n}\frac{\left(x_i-y_i\right)^2}{x_i+y_i} \tag{2.10}$$

   Cependant que pour les données de grandes dimensions, il ya une distance spécifique très utilisée :

7. **La métrique Minkowski :** Pour les données dimensionnelles, c'est la mesure populaire

$$d_p\left(x_i,x_j\right)=\left(\sum_{k=1}^{d}\mid x_{ik}-x_{jk}\mid^p\right)^{\frac{1}{p}} \tag{2.11}$$

   où d est la dimensionnalité des données. La distance euclidienne est un cas particulier où p = 2, alors que Manhattan p = 1. Néanmoins, il n'existe pas de directives générales théoriques pour la sélection d'une mesure à une application donnée. Une autre question, est de savoir comment mesurer la distance entre 2 classes D(C_1 ; C_2) ? Pour cela il ya certaines fonctions permettent de mesurer cette distance comme :
   plus proche voisin :

$$\min\left(d\left(i,j\right),i\in C_1,,j\in C_2\right) \tag{2.12}$$

   diamètre maximum :

$$\max\left(d\left(i,j\right),i\in C_1,,j\in C_2\right) \tag{2.13}$$

   distance moyenne :

$$\frac{\sum_{i,j}d\left(i,j\right)}{n_1n_2} \tag{2.14}$$

   distance des centres de gravité :

$$d\left(\mu_1,\mu_2\right) \tag{2.15}$$

   distance de Ward :

$$\sqrt{\frac{n_1n_2}{n_1+n_2}}d\left(\mu_1,\mu_2\right) \tag{2.16}$$

2.7.3 Discussion

Une note importante est de savoir si le clustering utilise une distance symétrique ou asymétrique. Bon nombre des fonctions énumérées ci-dessus ont la propriété que les distances sont symétriques. Dans d'autres applications (par exemple, la séquence-alignement des méthodes, voir **Prinzie** & **Van den Poel (2006)**), ce n'est pas le cas.

Certaines mesures sont spécifiques aux domaines particuliers comme histogrammes ou aux distributions. Dans **[Puzicha et al., 1999]**, on trouvera une étude comparative de quelques de ces fonctions .

En plus, ces mesures rencontrent certaines difficultés lorsque'on change le jeu de données comme le fait de travailler sur des espaces de couleurs où quelque distance ne sont pas recommandées. L'inconvénient major de la plupart de ces fonctions, c'est qu'elles sont coûteuses en temps de calcul et sont de plus sensibles à la dimension des données. Pour remédier le problème de dimensions, il ya des techniques ont été proposées pour la réductions de dimensions, qui permettent d'appréhender cette difficulté **[Berrani** & **al., 2002]**.

2.8 Les limites de Clustering

Il ya un certain nombre de problèmes avec le clustering. Parmi eux :

- ✦ les techniques de clustering actuelles ne traitent pas tous les besoins de façon adéquate (et simultanément), comme le fait que si nous n'avons pas des variables continuées (la longueur), mais les catégories nominales, comme les jours de la semaine. Dans ces cas encore, la connaissance du domaine doit être faite pour formuler le clustering appropriée.
- ✦ traitement d'un grand nombre de dimensions et de grand nombre de données, question peut être problématique en raison de la complexité du temps de calcule.
- ✦ l'efficacité de la méthode dépend de la définition de «distance» utilisée.
- ✦ si la mesure de la distance n'existe pas, nous devons la «définir», ce qui n'est pas toujours facile, surtout dans des espaces multidimensionnels.
- ✦ le résultat de l'algorithme de clustering peut être interprété de différentes manières.
- ✦ Beaucoup d'algorithmes de clustering exigent la spécification du nombre de clusters à produire en entrée de l'ensemble de données, avant l'exécution de l'algorithme. ie : connaissance de la valeur correcte à l'avance, la valeur appropriée doit être déterminée,

un problème pour lequel un certain nombre de techniques ont été développées.

2.9 Les caractéristiques des différentes méthodes

Quel que soit le type de la classification il ya Trois éléments permettent de caractériser les différentes méthodes :

1. La classification se déroule séquentiellement en regroupant les observations les plus 'semblables' (méthodes hiérarchiques) ou elle regroupe en k groupes toutes les observations simultanément (méthodes non-hiérarchiques).
2. Le critère de 'ressemblance' entre deux observations.
3. Le critère de 'ressemblance' entre deux groupes ou entre une observation et un groupe.

Ces trois éléments permettent de définir le déroulement ainsi que le type de la méthode, le deuxième et le troisième caractère ont un point primordial dans la performance et la qualité du résultat attendu d'une méthode, car il y aura certainement une différence de calcule (précision) entre le fait d'utiliser la distance euclidien au lieu de la distance de Hamming (cad que la distance utilisée est prise en considération afin d'amélioré les résultats) **[Gower &legendre, 1986] [Dalirsefat& al., 2009]**

2.10 Conclusion :

Les méthodes de clustering comme toutes les autres méthodes de classification , ont leurs avantages , faiblesses (voir section : discussion) , cependant , il n'y a pas que le type statistique , il y'en a d'autre type qui s'appuie sur la théorie de probabilité . Dans le chapitre suivant nous nous intéresserons à une nouvelle méthode de conception totalement différente de ce que nous l'avons vu jusqu'à maintenant, basée sur la conception du modèle de mélange, une méthode qui a été classé la 5ème parmi les méthodes de classification les plus utilisées/populaires de DATA MINING ces dernières années **[Xindong &Vipin, 2009]** , un classement prouve le succès qu'il a commencé à rencontrer très rapidement ce type de méthodes.

Chapitre 3

Expectation Maximization (GMM)

3.1 introduction

Il ya une autre façon de traiter les problèmes de clustering : une approche basée entièrement sur les modèles, qui consiste à utiliser certains modèles pour les clusters et de tenter d'optimiser l'adéquation entre les données et le modèle.

En pratique, chaque groupe peut être représenté mathématiquement, par une distribution paramétrique, comme une gaussienne, ou une loi de Poisson (discrète). L'ensemble des données est donc modélisé par un mélange de ces distributions. Une distribution individuelle utilisée pour modéliser un cluster spécifique est souvent désignée comme une distribution de composantes. La méthode de classification la plus largement utilisée de ce genre est celle basée sur l'apprentissage d'un mélange de gaussiennes : on peut effectivement considérer les clusters comme des distributions gaussiennes centrées sur leurs barycentres« centre de gravité », comme on peut le voir sur la figure suivante, où le cercle gris représente la première variance de la distribution

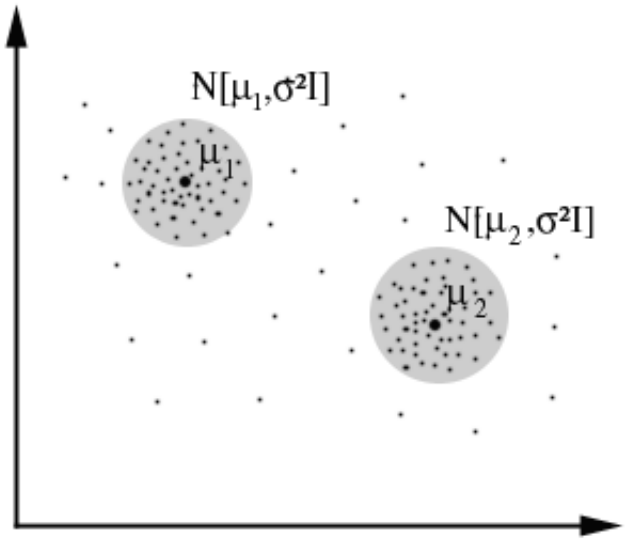

FIG. 3.1 – deux distributions caractérisées par leur μ,σ

3.2 Définition

3.2.1 Modèle de mélange

Cette téchnique suppose que l'ensemble de la population (l'ensemble de données) est représenté par une distribution de probabilité qui est un mélange de « s » distributions de probabilités associées aux classes. L'objectif final de cette méthode est définie les « s » distributions en estimant leurs paramètres(ça dépend du loi de probabilité utilisée). on trouve une multitude d'études qui ont traité Ce problème du mélange depuis longtemps **[Day, 1969][Wolfe, 1970]**. Pour expliquer ce que veut dire un modèle de mélange, prenons l'exemple d'une variable continue dans la figure suivante, où deux classes sont présentées

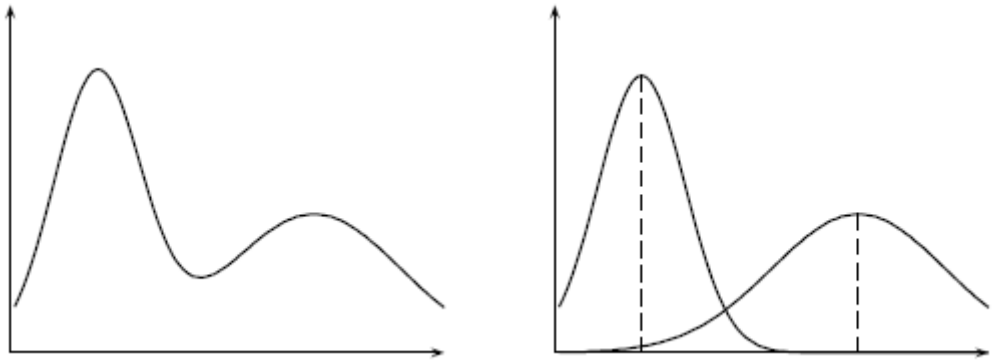

FIG. 3.2 – Une variable Aléatoire décomposée en deux Variables distinctes (deux composantes)

A la gauche on a la distribution de la variable globale. En effet, on peut remarquer qu'il y a deux maximums locaux ce qui veut dire que nous pouvons décomposer dès lors cette variable globale en deux variables distinctes, ayant chacune une moyenne et une variance propre, la figure de droite nous montre bien l'intérêt du modèle de mélange pour décomposer une population en plusieurs classes.

Principaux avantages du clustering basé sur modèle de mélange :

- ✦ bien étudier les techniques d'inférence statistique disponible ;
- ✦ souplesse dans le choix de la distribution de composantes ;
- ✦ le modèle de mélange "couvre" bien les données (on peut même avoir une distribution à l'intérieur d'une autre « distribution dominantes »)
- ✦ les données dans les distributions de composantes (cluster) sont serrés .

- ✦ il est facile d'obtenir une estimation de la densité pour chaque cluster ;
- ✦ . . .etc.

3.2.2 Distribution Gaussienne

En théorie de probabilité, on a plusieurs types de distributions où chaque distribution [Annexe 1] suit une loi spécifiée, par exemple la distribution gaussienne suit la loi normal (ou loi normale gaussienne ou loi de Laplace-Gauss, introduite par le mathématicien Abraham de Moivre 1733, et a prise le nom de celui qui la met en évidence : Gauss au XIXe), Ou encore la distribution de Poisson (loi de poisson), Distribution de Dirac, Distribution de Bose-Einstein, Distribution Zeta . . .etc. Dans ce chapitre nous nous intéressons qu'à la distribution gaussienne.

En effet, en probabilité on dit qu'une variable aléatoire réelle X est une variable gaussienne, si elle suit une loi normale gaussienne d'espérance «μ» et d'écart type «σ» strictement positif , si seulement cette variable aléatoire réelle X admet comme densité de probabilité la fonction p(x) définie comme suit : pour tout nombre réel x :

$$p(x) = \frac{1}{\sqrt[\sigma]{2\Pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{3.1}$$

on la note par :

$$X \sim N\left(\mu, \sigma^2\right) \tag{3.2}$$

qui veut dire la variable : Aléatoire X suit la loi normale des paramètres «μ» et «σ^2»

Cette distribution est une des plus importantes, qui est à la base du modèle de mélange gaussien, connue aussi sous le nom de distribution de courbe en cloche car sa densité de probabilité (3.1), dessine une courbe dite la courbe en cloche ou courbe de Gauss. Comme la montre la figure suivante :

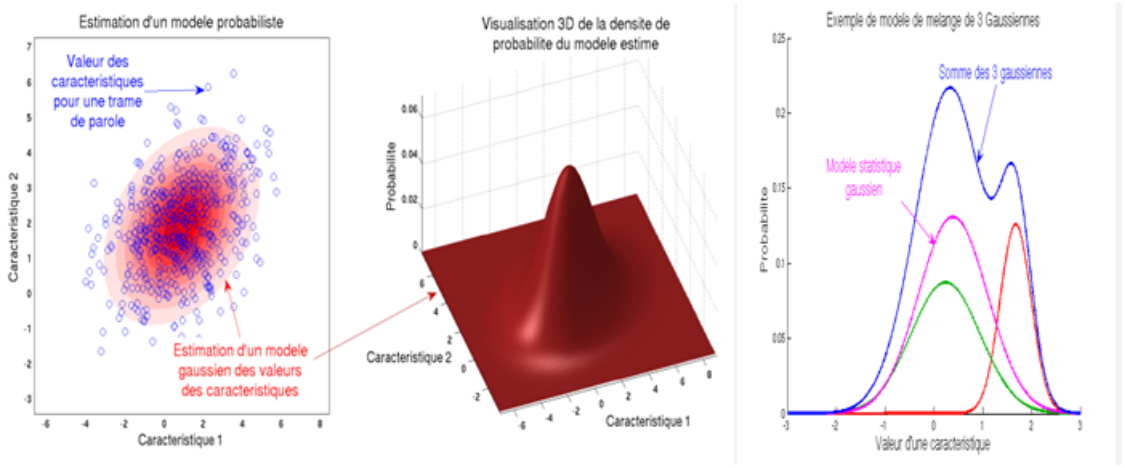

FIG. 3.3 – représentations des distributions gaussiennes (Source : [3.1])

Cette distribution a l'avantage qu'on peut connait grâce aux paramètres (L'écart-type, moyenne) la dispersion d'un ensemble de données, en fait, le savoir de la moyenne et l'écart-type permet de déterminer l'intervalle dans lequel on trouve 95% de la population. D'ailleurs on sait qu'on trouve 95% de la population dans l'intervalle $[\mu - 2\sigma\,;\mu + 2\sigma]$ et on trouve 68% de la population dans l'intervalle $[\mu - \sigma\,;\,\mu + \sigma]$.

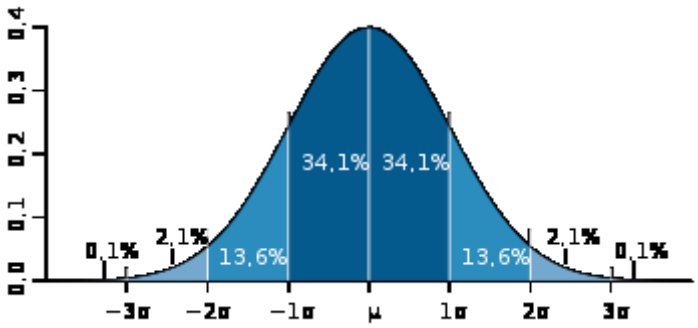

FIG. 3.4 – la dispersion des données dans une distribution gaussienne

3.2.3 Modèle de mélanges gaussiens

Un modèle statistique (souvent abrégé par GMM pour Gaussian Mixture Model en anglais) exprimé par une densité mélange[1] qui sert à estimer paramétriquement la distribution de variables aléatoires, ces dernières sont usuellement modéliser comme une somme de plusieurs

[1]densité mélange = loi mélange ou une fonction de densité issue d'une combinaison convexe de plusieurs fonctions de densité.

gaussiennes. Il s'agit donc de déterminer les paramètres de chaque gaussienne (la variance, la moyenne [Annexe 2]).

Pour le faire , on optimise ces paramètres selon un critère de maximum de vraisemblance [Annexe 3] afin approcher le plus possible la distribution recherchée. cette procédure se fait le plus souvent itérativement en utilisant l'algorithme espérance-maximisation (EM). Ce modèle de mélanges, est fort utilisée en classification automatique (clustering), on suppose qu'un échantillon de données suit une loi de probabilité gaussienne dont la fonction de densité est une gaussienne (qui est la plus courante), ce qui explique le nom « mélange gaussien ».

3.2.3.1 Principe

L'hypothèse est donc consiste à considérer que les objets (données) suivent les lois normales, alors on se place dans le cadre des modèles de mélanges gaussiens.

Prenons l'exemple d'un échantillon de données composé de n individus

$(x_1, ..., x_2)$appartenant à$\mathbb{R}^p$ (i.e. décrits par p variables continues). Ce modèle de mélanges, considère que ces objets (individus) appartiennent chacun à un des g groupes (g étant fixé a priori par l'utilisateur ou une expertise)G_1,...,$G_g$ces groupes sont caractérisés chacun par les paramètres de la loi normale qui suivent, a savoir : la moyenne μ_k(k=1,..,g) et de matrice de variance-covariance $\sum_k$ On peut donc décrire La forme de la densité de probabilité de ce mélange (la loi globale du mélange) que suit l'échantillon est donnée par :

$$g(x, \Phi) \sum \Pi_k f(x, \theta_k) \tag{3.3}$$

Où :

- θ_k=(μ_k, Σ_k) : est le paramètre de chaque loi normale.
- Π_1,...,Π_g : ce sont la probabilité qu'un élément de l'échantillon.

suive la loi g, qui sont les proportions des différents groupes (ou le poids du mélange) tel que :

$\sum_{k=1}^{g} \Pi_k$=1.

- Φ=$(\Pi_1, ..., \Pi_g, \theta_1, ..., \theta_g)$: représente le paramètre du mélange, qui est inconnu.
- $f(x, \theta_k)$: est la loi normale multidimensionnelle(ou loi de Gauss à plusieurs variables) paramétrée par θ_kc'est à dire (μ_k et Σ_k), cette fonction de densité gaussienne peut être

calculée par la formule :

$$f_k(x,\theta_k) = \frac{1}{(2\Pi)^{\frac{N}{2}} \mid \sum_k \mid^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)_k^{\top}\Sigma^{-1}(x-\mu_k)} \tag{3.4}$$

Donc l'objectif derrière cette approche consiste à déterminer le meilleur paramètre Φ c'est-à-dire estimer les proportions des composantes $\Pi_1,...,\Pi_g$et les paramètres$\theta_1,...,\theta_2$Pour cela, on a « Maximum de vraisemblance MV» qui offre une approche générale à l'estimation de ces paramètres à l'aide des données de l'échantillon, en cherchant le paramètre qui maximise la vraisemblance**[Redner & Walker, 1984]**, (cependant qu'il ya d'autres techniques pour les estimer, come celle des moments **[Pearson, 1894]** la fonction de vraisemblance notée V peut s'écrire as :

$$V(x_1,...,x_n;\Phi) = \prod_{i=1}^{n} g(x_i,\Phi) = \prod_{i=1}^{n}\sum_{k=1}^{g} \Pi_k f(x_i,\theta_k) \tag{3.5}$$

On cherche alors , à trouver le maximum de cette vraisemblance pour que les probabilités des réalisations observées soient aussi maximums, Pour des raisons de simplification de traitement, il est souvent plus simple de maximiser le logarithme népérien de cette fonction (car la vraisemblance est positive et le logarithme népérien est une fonction croissante) de vraisemblance (le produit donc se transforme en somme, ce qui est plus simple à dériver) , que l'on nommera la **log-vraisemblance « L »** et qui s'écrit :

$$L(x;\Phi) = \sum_{i=1}^{n} \log\left(\sum_{k=1}^{g} \Pi_k f(x_i,\theta_k)\right) \tag{3.6}$$

Donc L'estimateur au sens de MV se traduit par :

$$\hat{\Phi} = arg\max_{\Phi}.L(x,\Phi) \tag{3.7}$$

Il suffit maintenant de mettre la dérivé à zéro et la résoudre directement car le maximum de (3.5) doit annuler sa dérivée(on pourra s'assurer qu'il s'agit bien d'un maximum en vérifiant que la dérivée seconde est négative.).

Maintenant, L'estimation des paramètres peut être effectuée grâce à l'algorithme EM (Expectation-Maximisation) qui est une solution itérative pour la résolution de ce problème. Elle consiste à résoudre itérativement les équations de vraisemblance.

3.3 Algorithme d'Expectation-Maximisation

En français « L'algorithme d'espérance-maximisation », souvent abrégé par EM, Un des premiers articles sur EM a été écrit en 1958 **[Hartley,1958.]** mais la référence pratique qui a formalisé EM et a fourni une preuve de convergence est le document de Dempster, Laird et Rubin en 1977**[Dempster and al, 1977]**, tandis que le livre de **[Tanner, 1996]** est une autre référence populaire et très utile. Son objectif est de trouver le maximum de vraisemblance des paramètres de modèles probabilistes (comme modèle gaussien).

On compte une multitude de domaines d'applications de cet Algorithme, à titre d'exemple : il est usuellement utilisé dans la vision artificielle ou encore dans le traitement d'images, plus spécifique en ce qui concerne la segmentation (image médicale, satellitaire…etc.), ou tout simplement en clustering pour regrouper les données homogènes dans un groupe,…etc. Un livre récent consacré entièrement à EM et ses extensions, en plus des applications est : **[McLachlan et Krishnan, 1997]** Donc L'estimation des paramètres du modèle passe par la maximisation de $L(x;\Phi)$, pour cela cet algorithme comprend deux étapes essentielles :

1. E-steps (E) : une étape d'évaluation de l'espérance, c'est dans cette étape qu'on calcule l'espérance de la vraisemblance en tenant compte des dernières variables observées,
2. M-steps (M) : une étape de maximisation de vraisemblance qu'on a trouvée à l'étape(E), en tentant d'estimer le maximum de vraisemblance des paramètres.

Et c'est ainsi qu'on itère l'algorithme en utilisant les paramètres trouvés à l'étape(M) pour évaluer à nouveau l'espérance.

3.3.1 Principe

Voyons maintenant comment fonctionne l'algorithme EM pour un mélange de gaussiennes : Soit X (X_1,..,X_n) un échantillon d'observations issues d'un mélange de gaussiennes et soit Z (_pZ_1,._,Z_n) la donnée cachée où Z_i détermine la distribution dont est issue x_i (nous utilisons

une représentation marginale de la vraisemblance selon les « données cachées Z »), par exemple si on a un mélange de deux gaussiennes bidimensionnelles alors :

$\mathscr{L}(X_i\{Z_i=1\})$=$N_2(\mu_1,\Sigma_1)$; $\mathscr{L}(X_i\{Z_i=2\})=N_2(\mu_2,\Sigma_2)$

avec $\mathrm{P}\{Z_i=1\}=\lambda_1$ et $\mathrm{P}\{Z_i=2\}=\lambda_2=1-\lambda_1$.

Alors que à ce stade , on ne connait que : l'échantillon d'observations (exemple : le nuage de points **X** tel celui en figure (3.1)), on cherche donc , à estimer les paramètres inconnus à savoir :

$\Phi=(\Pi_1,...,\Pi_g,\theta_1,...,\theta_g)$

La vraisemblance des données complètes est :

$$L(X,Z\mid\Phi)=\prod_{i=1}^{n}\sum_{k=1}^{g}1\{z_{i=j}\}\Pi_k f_k(x_i,\theta_k)\,. \tag{3.8}$$

ou :

f_k : est une densité gaussienne bidimensionnelle de paramètres μ_j et Σ_j (3.4) ce qui se traduit en log-vraisemblance des données complétés par :

$$\log(L(X,Z|\Phi))=\sum_{i=1}^{n}\left[\sum_{k=1}^{g}1\{z_{i=j}\}\left(\log(\Pi_k)-\log\left(2\Pi\right)-\tfrac{1}{2}\log(\det(\Sigma_k))-\tfrac{1}{2}\left(x-\mu_k\right)^T\Sigma_k^{-1}(x-\mu_k)\right)\right] \tag{3.9}$$

maintenant à chaque itération, l'étape E nécessite de définir la distribution a posteriori de Z_j connaissant X_i et Φ. On définit :

$$P\left(Z_{i=j}\mid X_k=x_i,\Phi^{EM}\right)=\frac{\Pi_k^{EM}f_k(x_1,\theta_k^{EM})}{\Sigma_{1=1}^{g}\Pi_1^{EM}f_1(x_1,\theta_1^{EM})} \tag{3.10}$$

Cette formule donne donc la possibilité de calculer la probabilité conditionnelle pour que l'objet x_i appartienne à la gaussienne qui correspond au groupement(distribution) numéro j donnant les paramètres Φ^{EM} ainsi estimés. Autrement dit c'est la probabilité a posteriori pour que le point X_i soit issu de la distribution $f_k\equiv \mathrm{N}(\mathrm{p}\mu_k\ ,\Sigma_k)$connaissant Φ^{EM}. La maximisation de log-vraisemblande en Φ (3.9), ne présente aucune difficulté majeure, et conduit aux estimateurs suivants :

$$\begin{aligned} \Pi_k^{new} &= \frac{1}{N}\Sigma_{i=1}^{N} P(Z_k|X_i,\Phi^{old}) \\ \mu_k^{new} &= \frac{\Sigma_{i=1}^{N} X_1\, P(Z_k|X_1,\Phi^{old})}{\Sigma_{i=1}^{N} P(Z_k|X_i,\Phi^{old})} \\ \Sigma_k^{new} &= \frac{\Sigma_{i=1}^{N} P(Z_k|X_1,\Phi^{old})\,(X_1-\mu_k^{new})\,(X_1-\mu_k^{new})^T}{\Sigma_{i=1}^{N} P(Z_k|X_1,\Phi^{old})} \end{aligned} \tag{3.11}$$

En Grosso modo, on peut résumer ces étapes par : on démarre l'algorithme avec une ignorance absolue des données cachées Z et en initialisant θ (pour chaque composante «gaussienne») à une valeur θ_0 d'une manière totalement arbitraire, potentiellement très loin de la réalité.

L'algorithme se sert donc de θ_0 pour estimer Z, puis se sert de meilleure estimation de Z (lors de E-step) pour réestimer les paramètres en une valeur θ_1 plus pertinente.

À l'itération suivante, on évalue donc l'estimation des données cachées Z puisque cette nouvelle estimation se base cette fois sur θ_1. Et cette meilleure estimation sur Z conduit à son tour à une meilleure précision sur θ_2, et ainsi de suit jusqu'à atteindre la convergence.

Et au final en plus que l'algorithme nous fournit une meilleur estimation de θ , il estime aussi les variables cachées (latentes) Z qui montre bien une autre utilisation de l'algorithme EM, à savoir : la complétion des données manquantes .

3.3.2 La Convergence

L'objectif de convergence est de trouver (mettre à jour) une « meilleure » valeur qui augmente la vraisemblance d'une itération à l'autre, donc il suffit de trouver une fonction comme :

$$\triangle(\theta,\theta_m) := \log P(X \mid \theta) - \log P(X \mid \theta_m) \geq 0 \tag{3.12}$$

On souhaite bien sûr que cette différence soit la plus grande possible , en fait dans **[Sean Borman , 2009][Xindong &vipin, 2009]** on a prouvé qu'une telle fonction est possible appelée **Q-function** , et ce qu'il prouve que l'algorithme EM admet «la croissance de la vraisemblance d'une itération à l'autre ».

Il faut noter que, dans certains cas, l'algorithme peut ne pas bien converger ou converger que vers un maximum local de la vraisemblance...., cela revient aux conditions initiales choisies θ_0 arbitrairement, cependant pour certaines mauvaises valeurs, l'algorithme peut rester gelé en un point selle, alors qu'il convergera vers le maximum global pour d'autres valeurs initiales plus

pertinentes. Donc il est recommandé de relancer plusieurs fois l'algorithme avec différentes initialisations pour appréhender ce problème de convergence. En fait, il ya aucun théorème général de convergence pour l'algo EM **[Boyles ,1983]**, la convergence de la sequence { $\theta^{(m)}$ à l'itération m} dépends entièrement aux choix des caractéristiques de **L(θ)** ,du modèle , ainsi que des points initiaux de start $\theta^{(0)}$.

Il ya une multitude d'études qui ont été fait pour appréhender ce problème chacune traite un domaine particulier , par exemple dans **[Xu** & **Jordan,1996]**, on trouve tous les aspects concernant la convergence dans un model de mélange gaussien, ou encore dans **[Meng** & **Rubin, 1994]**, on trouvera des discussions sur les méthodes pour le calcule de convergence des différents algorithmes EM.

3.3.3 Algorithme

On peut donc définir toutes les étapes de cet Algorithme par :

1. initialisation au hasard de $\Phi^{(0)}$
2. m=0
3. Tanque EM n'a pas convergé, faire :
   * Evaluation de l'espérance (E-steps) : Q(Φ ; $\Phi^{(m)}$)
   (3.9)
   *maximisation (M-steps) : $\Phi^{(m+1)}$ =argmax$_\Phi\Phi$ (Q(Φ ; $\Phi^{(m)}$))
   (3.11)
   *m= m+1
4. Fin.

3.3.4 L'aspect classificatoire

Une fois l'estimation terminée, il suffit d'attribuer à chaque individu la classe à laquelle il appartient le plus probablement. Pour cela, on utilise la règle d'inversion de Bayes. D'après celle-ci, on a :

$$P(x \in G_k \mid x) = \frac{P(x \mid x \in G_k) \, . \, P(x \in G_k)}{P(x)}, \tag{3.13}$$

ce qui se traduit, dans notre cas, par :

$$p(x_i \in G_k) = \frac{\Pi_k f(x_i, \theta_k)}{\sum_{\ell=1}^{g} \Pi_\ell f(x_i, \theta_\ell)} \tag{3.14}$$

Il suffit alors d'attribuer chaque individu X_i à la classe pour laquelle la probabilité a posteriori $P(x_i \in G_k)$est la plus grande.

Exemple :

Voilà un exemple qui montre la progression de la construction des gaussien, en classifiant les points au fur et à mesure de l'estimation des paramètres de chaque gaussien jusqu'à atteint la convergence à « l'itération 20 » qui présente un état stable où aucun changement n'est plus possible.

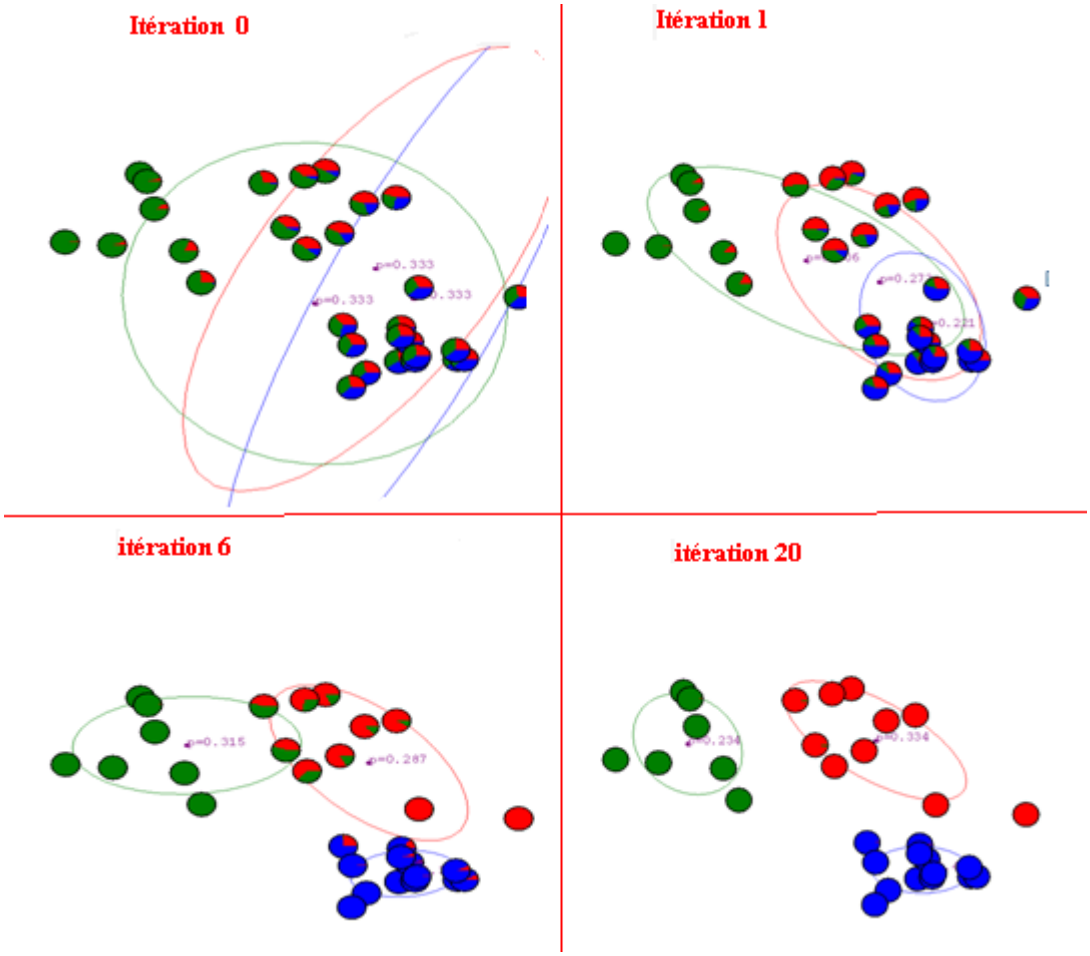

FIG. 3.5 – Construction des gaussiens par la méthode EM-GMM (source : [3.2])

3.3.5 Discussion

En effet L'algorithme EM est une méthode de clustering générale, ce qui veut dire qu'il ya des variétés de détails dans le cadre d'implémentation pour une situation donnée, en fait on compte plusieurs extensions (variantes) de cet algorithme chacune traite un domaine particulier **[McLachlan et Krishnan, 2008]** à fin de répondre aux problématiques qu'il rencontre EM, à titre d'exemple : GEM (Generalized EM) qui permet de simplifier le problème de l'étape maximisation ; l'algorithme CEM (Classification EM) permettant de prendre en compte l'aspect classification lors de l'estimation, ainsi que l'algorithme SEM (Stochastic EM) dont l'objectif est de réduire le risque de tomber dans un optimum local de vraisemblance.

Il est également à noter que dans notre travail nous n'avons traité que le cas continue gaussien, mais dans beaucoup de situations pratiques, l'étape E de l'algorithme nécessite de faire appel aux autres méthodes pour avoir une valeur approchée de l'espérance conditionnelle.

Contrairement aux approches traditionnelles de clustering cette méthode est adaptée pour traiter des grandes masses des données de grandes dimensions , pourtant elle marque quelques faiblesse comme nous l'avons remarqué en ce qui concerne l'influence des paramètres de départ (initiaux) sur les résultats attendus , ainsi que le temps de calcule …etc. alors que on peut remédier ces inconvénients en l'intégrant avec d'autres méthodes de classification (hiérarchique , k-means …), en plus, Le côté itératif de l'algorithme pourra peut-être paraître un peu mystérieux, mais comme nous l'avons vu, l'algorithme garantit que la vraisemblance augmente à chaque itération, ce qui conduit donc à des estimateurs de plus en plus corrects.

3.4 Conclusion

Dans ce chapitre nous avons abordé tous ce qui concerne la méthode EM basé sur les modèles de mélange (ses point fort , sa puissance , ses utilités , … ses faiblesses) , comme nous l'avons vu , La reconnaissance de mélanges gaussiens est une des applications fondamentales de l'algorithme EM, qui a été formalisée comme une approche pour résoudre les problèmes arbitraires du maximum de vraisemblance. Cependant, cet algorithme est très riche d'un point de vu d'implémentation car il a été adapté selon plusieurs problèmes (HMM « L'algorithme EM est également très utilisé pour déterminer les paramètres d'un modèle» **[Rabiner, 1989]**, GMM…ect) et cela grâce à ses extensions et à sa légèreté à la modification **[McLachlan** & **Krishnan, 2008]**.

Aujourd'hui, EM et ses variantes sont régulièrement utilisées pour résoudre un large éventail de problèmes d'estimation, de : l'EM pour les motifs élicitassions (MEME) , à : algorithme EM pour motif d'enquête dans les séquences ADN**[Bailey** & **Elkan, 1995]**, ou encore des modèles de mélange de montage pour lever l'ambiguïté des objectifs de fouillis dans le radar**[Wang** &**al, 2006]**.

En effet , EM touche un grand nombre de problèmes d'estimation dans divers domaines : traitement du signal, traitement de l'image, notamment en imagerie médicale **[McLachlan** & **chang, 2004]**…etc.

Chapitre 4

Application

4.1 Préliminaire

Dans ce dernier chapitre et après l'aperçu théorique des chapitres précedents, nous présentons le côté pratique de notre application .Notre but est la réalisation d'un système flexible et fiable qui segmente les images sanguines d'une manière automatique sans intervention ou connaissances préalables, pour l'aide au diagnostic médical (surtout dans le domaine de l'anatomie pathologique .Nous commençons par la description de la base utilisée, le choix de l'environnement de travail ainsi que les étapes fondamentaux de la conception de notre application. Notre application porte le nom « **SEM** » pour (Segmenter via Expectation Maximization)

4.2 L'environnement de travail

Pour que notre travail atteint l'objectif qu'on visait, on a pris l'initiative d'exploiter et d'implémenter notre algorithme sur la version : Windows XP P3 Sweet 6.2Final.

Ce choix se traduit par l'efficacité de cet L'environnement en ce qui concerne la structure d'interaction événementielle qu'elle dispose pour communiquer avec des applications actives , ainsi que les ressources de la machines qu'il offre aux différentes applications , enfin , son système d'allocation de mémoire qui est un des meilleurs présents dans ce domaine.

4.3 Le langage de codage

A :

Cette application (SEM) a été codé en sa globalité par le language C++ a travers la plateforme Qt (prononcez "Quioute" vennu du mot anglais « Cute » ce qui signifie "Mignonne") ce choix repose sur le fait que Qt est une bibliothèque Framework, Qt est donc pour créer des GUI (programmes de fenêtres). Elle est écrite en C++, à la base elle est faite pour être utilisée qu'en C++, néanmoins il est possible de l'utiliser dans d'autres langages comme Java, Python, etc.[4.1]

FIG. 4.1 – le logo de Qt

Comme nous l'avons dit , Qt n'est pas une simple bibliothèque pour développer des application graphique , mais elle est un Framework qui veut dire qu'elle regroupe plusieurs bibliothèques (modules) de domaines différents , comme :

- ✦ Module GUI : c'est la partie dédie à la création des fenêtres.
- ✦ Module OpenGL : le module qui a la tache de 3D et pour géré par OpenGL.
- ✦ Module réseau : Qt toute une boite d'outils pour accéder au réseau, que ce soit pour créer un logiciel de Chat, un client FTP, un client Bittorent, un lecteur de flux RSS...
- ✦ Module de script : la partie Javascript (ou ECMAScript), qu'on peut l'utiliser dans des applications pour ajouter des fonctionnalités, sous forme de plugins.
- ✦ Module XML : une partie puissante pour le XML, à fin d'échanger des données avec des fichiers formés à l'aide de balises, presque comme le XHTML.
- ✦ Module SQL : un module qui permet un accès aux bases de données (MySQL, Oracle, PostgreSQL...).
- ✦ …etc.

En plus , un autre critère nous a poussé vers ce choix , c'est que Qt est multi-platforme , c'est-à-dire , que vous n'avez pas besoin de coder 3 fois le même code pour 3 différents OS (système d'exploitation), d'ailleurs la principale idée auquel Qt s'appui , c'est « codez une seule fois (sous n'importe quel OS) , et vous pouvez le compiler dans tous les OS pour générer l'exécutable correspondant » ce qui explique la rapidité de ses applications par rapport aux autres langages comme java.

Qt dispose d'un nombre important d'outils de développement d'applications, comme :

- ✦ **QtDesigner :** qui est un outil pour le dessin d'applications graphiques visuellement en glissant les composantes (Widgets) en question sur la fenêtre, c'est-à-dire vous n'aurez pas besoin de coder tout le code qui génère la fenêtre.

- **QtLinguist :** un très bon moyen pour distribuer vos applications en plusieurs langues, ce que vous devrez le faire c'est de respecter certaines normes de nommages et à la fin de votre développement vous passez votre code dans cet outil pour générer un petit fichier linguistique , qui va servir comme moyen afin de traduire votre application dans la langue que vous choisissez.
- **QtAssistant :** un autre performant style pour accompagner les développeurs pendant et durant toute la période de la réalisation des applications , grâce à QtAssistant vous pouvez trouvez toute la documentation de Qt , bien structurée d'une manière qui ne laisse pas le doute sur une fonctionnalité d'une telle fonction ou autre, ainsi que une grande liste des exemple bien commentés pour donner un aperçu sur la fonction en question.
- **..etc**.

En fait , à proprement parler , Qt n'utilise pas que le langage C++ , mais elle a son propre langage , ses propres structures , conventions de nommage , ses propres Classes . . .etc. donc elle regroupe entre la rapidité et la simplicité de C/C++ ainsi que ses avantages d'interface graphique pour développer des bonnes applications.

B :

pour ce qui concerne la partie de traitement d'images nous avons utilisée une bibliothèque appelée OPENCV (Open Source Computer Vision), comme son nom le suggère , OPENCV est une bibliothèque dédié totalement à la vision par ordinateur ce qui signifie traitements des images ,matrices ,vidéos. . .. données visuelles.

FIG. 4.2 – Logo d'opencv

OpenCV est très fortement influencée par les avancées de la recherche dans ce domaine , car

il s'agit de la bibliothèque d'INTEL c'est-à-dire : une bibliothèque créée « par des chercheurs, pour les chercheurs » en fin d'année 2010 , elle a dépassé 3 millions de téléchargements . Le choix de cette bibliothèque c'est qu'elle est « gratuite », « libre » et « multi-plateformes » et elle est utilisé en C, C++ et Python, ce qui va nous permettre de l'utiliser aussi bien sous Windows, sous GNU/Linux, ou sous Mac OS.[4.2].

Donc en grosso modo, dans cette applications nous avons travaillé par 3 langages différents (le C/C++, Qt et OPENCV) .

4.4 Description de la base de Données

La base qu'on a adopté pour nos testes de segmentations afin d'établir notre algorithme , ce n'est qu'un ensemble d'images des cellules sanguines (27 images) prises du laboratoire hémobiologie du C.H.U Tlemcen en utilisant l'appareil LEICA qui est un microscope avec une caméra permet de capter, en bonne qualité ,des images de ces cellules sur des lames , colorées après , par la méthode May Grunwald Giemsa.

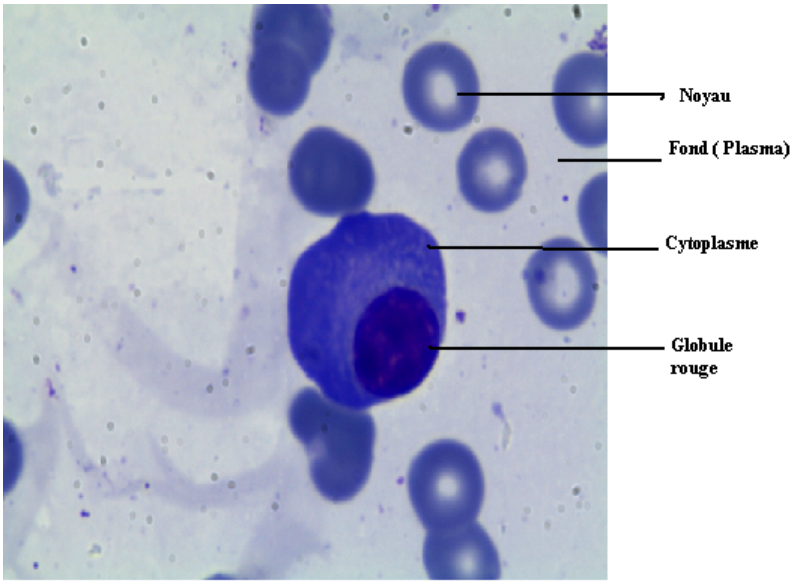

FIG. 4.3 – Exemple d'image microscopique sanguine

Il est clair qu'on dispose de 4 classes d'objets dans ce type d'image, à savoir : Le noyau, le Fond, Cytoplasme et les globules rouges, auxquels chacun prend un format spécifié ainsi que une couleur déférente que les autres, ce qui va nous aider dans leur reconnaissance.

4.5 Description de l'application :

Interface et composants :

En cliquant sur « Fichier » un menu déroulant apparaît vous permettant d'effectuer les fonctions suivantes :

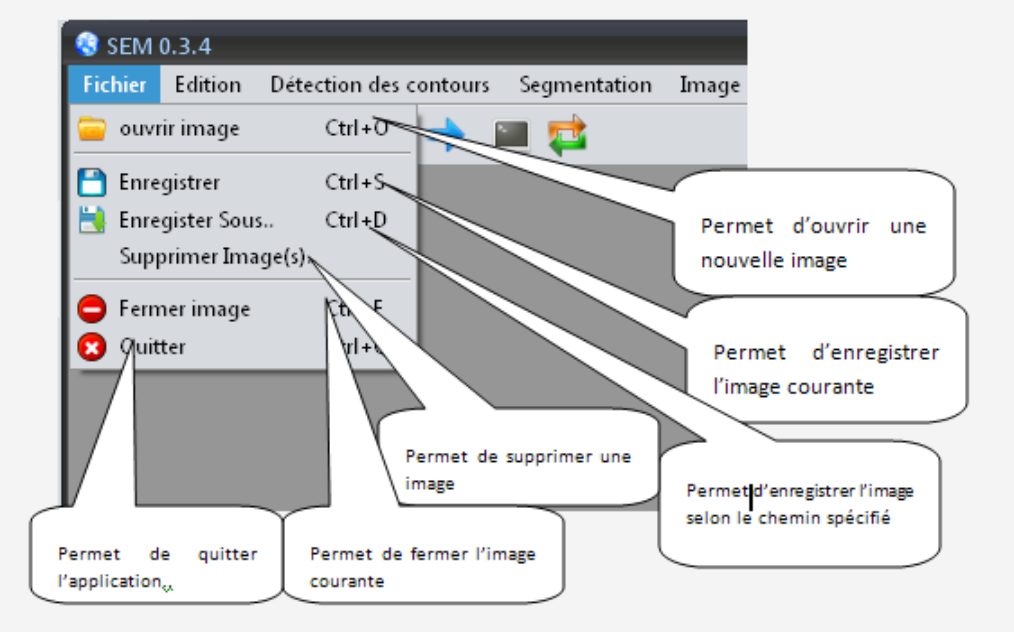

FIG. 4.4 – Menu Fichier.

En cliquant sur « Edition » un menu déroulant apparaît vous permettant d'éffectuer les fonctions suivantes :

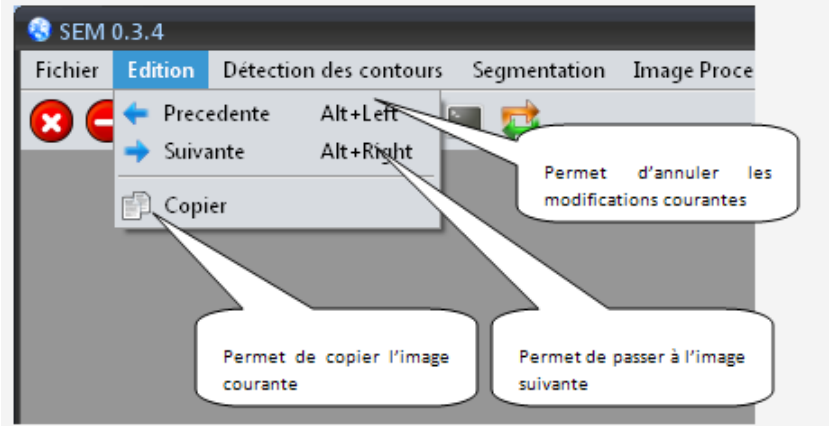

FIG. 4.5 – Menu Edition.

En cliquant sur « Détection des contours » un menu déroulant apparaît vous permettant d'effectuer les fonctions suivantes :

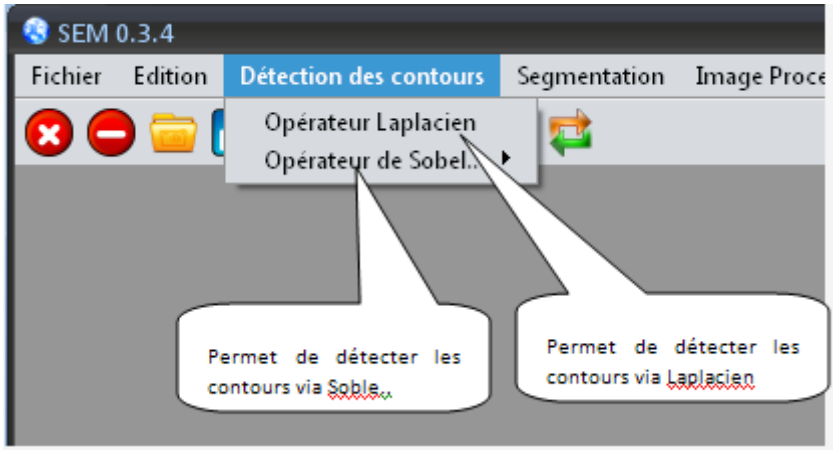

FIG. 4.6 – Menu Détection des contours.

En cliquant sur « Segmentation » un menu déroulant apparaît vous permettant d'éffectuer les fonctions suivantes :

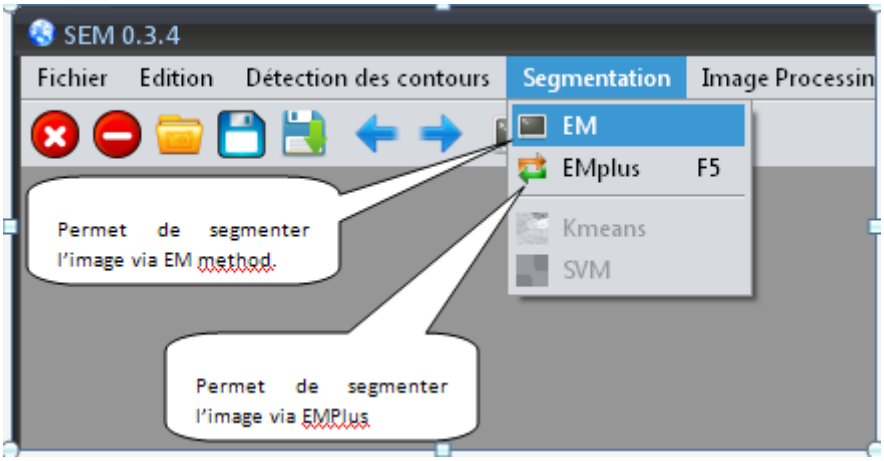

FIG. 4.7 – Menu Segmentation.

En cliquant sur « Image processing » un menu déroulant apparaît vous permettant d'éffectuer les fonctions suivantes :

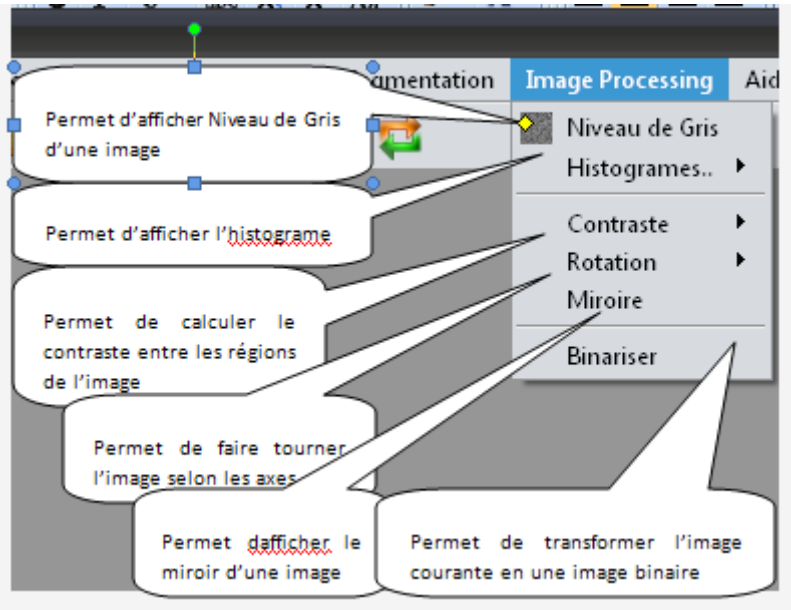

FIG. 4.8 – Menu Image processing.

4.6 Conception :

Rappelons tout d'abord que ce travail est focalisé principalement sur la segmentation des images couleurs. La méthode utilisée est non supervisée, elle est basée sur un modèle de mélanges de gaussiennes GMM. Les paramètres du modèle sont estimés par l'algorithme EM cette méthode utilise le principe de L'approche de maximum vraisemblance (ML) pour ajuster le 'GMM' le plus convenable aux données qu'on désire modéliser.

Cette méthode mène à estimer les paramètres du mélange par (3.6) qui donne les paramètres estimés :$\hat{\Phi} = arg\,max_{\Phi}.L(x,\Phi)$calculé itérativement et remit à jours par les formules (3.11), en procédant en deux étapes successives (E-step , M-step).

Pour une image couleur il suffit de prendre les N pixels d'une image comme x_j , où Tous les échantillons (pixels) de l'image forment un ensemble de données. Donc on Suppose que x_j est généré par un mélange de distributions des gaussiennes, et que le nombre de composantes gaussiennes K est connu (k = le nombre des clusters qu'on souhaite en avoir), il nous reste juste qu'appliquer l'algorithme EM sur ces valeurs en respectant le modèle du mélange, ce qui va estimer les paramètres du modèle , à ce stade , En utilisant ces paramètres fournis par l'algorithme EM pour un mélange ayant le nombre de gaussiennes égal à K, on peut donc calculer la probabilité conditionnelle pour que le pixel x_j appartienne à la gaussienne qui

correspond au cluster numéro i , ce qu'on peut le calculer par la formule bayesienne (3.14) , mais le problème, c'est que , pour chaque pixel on a k probabilités conditionnelles qui le lie avec les Clusters , et pour pouvoir l'assigner à un Seul cluster (gaussienne) si : sa probabilité correspondante au cluster i*, est la probabilité maximale pour i=1,2,..,K.

Pour les initialisations comme tous les algorithmes itératifs, l'algorithme EM nécessite l'initialisation des paramètres du modèle de mélange des gaussiennes. Les matrices de covariances sont initialisées par des matrices identités, et pour les K vecteurs moyennes il ya plusieurs manières, en fait on peut les initialiser d'une façon aléatoire en invoquant les Random() sur les pixels de l'image en choisissant k emplacements aléatoirement afin que l'algorithme puisse commencer , cependant il ya une autre technique plus pratique et performante, en utilisant une méthode de pré-clustering afin de nous approcher des réels centres des différentes gaussiennes du mélange , cela est fait par l'algorithme de K-moyennes qui nous renvoie les centroides les plus proches possibles des réels centroides des différentes zones présentes dans l'image.

En ce qui concerne la répartition des classes de notre application on l'a réparti en 2 grandes Classes actives :

- « Classe FenPrincipale » qui contient toutes les méthodes d'interaction et communication avec l'utilisateur (coté interface graphique), ainsi que, elle fait appel aux fonctions de traitement d'images.
- « Classe Image » : contient toutes les fonctions qui manipulent les images, en plus des propriétés d'une image.
- Au plus de ça, nous avons écrit des fichiers de collection de méthodes qui regroupent les méthodes des traitements intermédiaires entre les bibliothèques utilisées .
- (Qt – OpenCV) comme IplImageTwoQimage qui est un fichier regroupe les méthodes de conversion entre les différentes configurations d'images dans les deux bibliothèques mentionnées .

4.7 Expérimentations

Un exemple d'image d'origine à segmenter est représenté dans la figure (4.3) , notre objectif est de segmenter les différentes régions qu'elle contient notre image en colorant à nouveau les parties identifies de chaque cluster au fur et à mesure d'estimation de ses paramètres en

attribuant une couleur distincte (vert pour le noyau, jaune pour le cytoplasme, rouge pour globule rouge et noir pour le fond) .

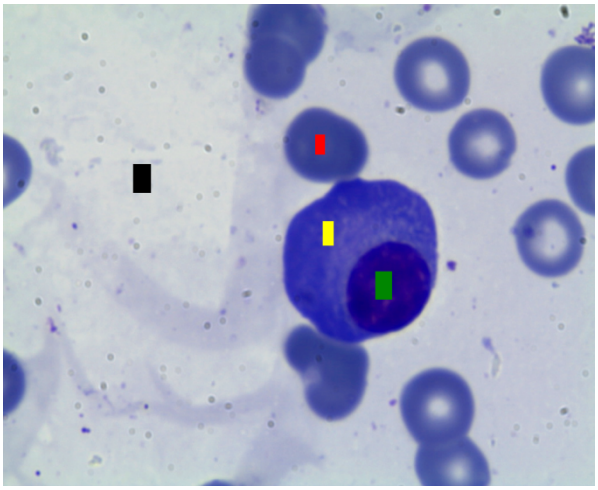

FIG. 4.9 – Définition des différentes régions de l'image

Résultat final de l'appel à la méthode EM (qui fait appel à la méthode k-means d'une manière implicite à fin d'approcher aux réels centres pour éviter les longues convergences) pour estimer les paramètres des 4 Clusters de cette image est montré dans la figure 4.10

FIG. 4.10 – Résultat de Segmentation

Cependant, elle ne reconnu pas correctement certaines régions (comme : cytoplasme) de certaines images, a cause de la différence dans la configuration colorimétrique (zone bruitée, confusion) comme l'exemple de l'image suivante :

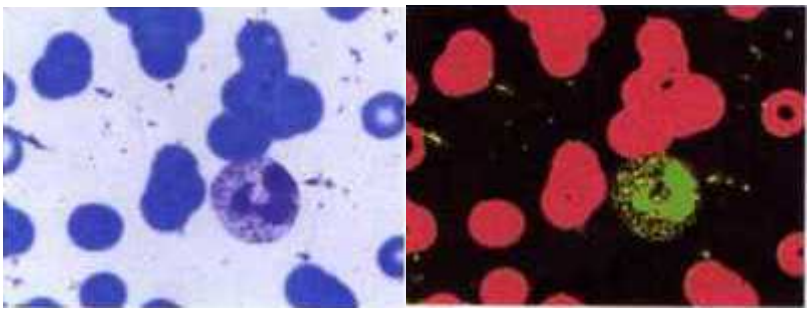

FIG. 4.11 – Exemple de segmentation.

Où on constate que , une grande partie du cytoplasme est mal classée (classé en noir = fond) . Après avoir fait plusieurs testes de segmentation on peut calculer le taux de reconnaissance et les précisions pour chaque zones de l'image de teste en la comparant avec une image segmentée manuellement par un expert de domaine, qui nous a donné les chiffres :

Précision :

Noyau	Cytoplasme	Rouge	Fond
96.39%	79.28%	78.28%	93.12%

Taux :

Noyau	Cytoplasme	Rouge	Fond
94.16%	28.80%	89.56%	57.52%

Nous constatons que la classe Cytoplasme est la plus faible classe reconnue , cela revient à son caractère colorimétrique par rapport aux autres classes (la qualité de l'image joue un rôle important sinon il faut faire appel aux méthodes pré-traitment pour l'améliorer), tandis que les classes comme Le fond et le noyau , ont été bien reconnues , pour les globules rouges , c'est leur noyau qui cause le problème , car il est toujours reconnu comme un fond et ne pas une partie d'elles .

EMPLus :

EMPlus n'est pas une nouvelle méthode, mais plutôt une proposition au problème de cytoplasme (des zones mal reconnues).

Comme nous l'avons remarqué , il ya des zones qui ont connue une faible reconnaissance par l'algorithme EM , parce que leur configuration colorimétrique est d'une modeste énergie ou il ya une confusion de couleurs dans la même zone (le cas des cytoplasmes) , dans ce cadre , j'ai pensé d'utiliser en plus de mélange du gaussien GMM, une autre notion afin de pouvoir aider le spécialiste qui cherche à identifier d'une façon correcte ces zones de conflits , cette notions est très utilisée dans la détection des visages , elle consiste à créer un MASK (une petite image qui a presque les mêmes caractéristiques de format de la zone cible qui na pas était reconnue en utilisant EM) pour cela le spécialiste doit créer manuellement ces MASK, après il suffit de donner le MASK avec l'image qui cause le problème en spécifiant la zone de conflit (cytoplasme) et le programme commence à chercher les zones similaires (en fonction de leur format) dans l'images par rapport au MASK , quand il trouve la moindre ressemblance il commence à estimer les paramètres de notre gaussien de mask et de l'image source, il peut même compléter les données manquantes (en cas ou le MASK n'est pas correctement conçu) , quand cela sera fait , il inverse les couleurs des gaussiens en prenant la couleur de l'image de teste.

Exemple : voilà une image qu'on cherche d'elle de reconnaitre le cytoplasme

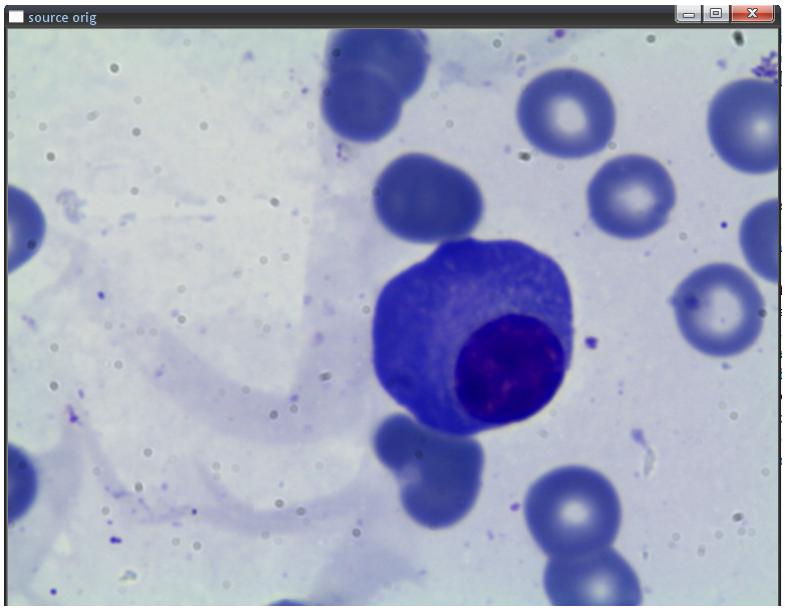

FIG. 4.12 – image de teste

Et voilà le MASK qu'on a crée spécialement pour cette image

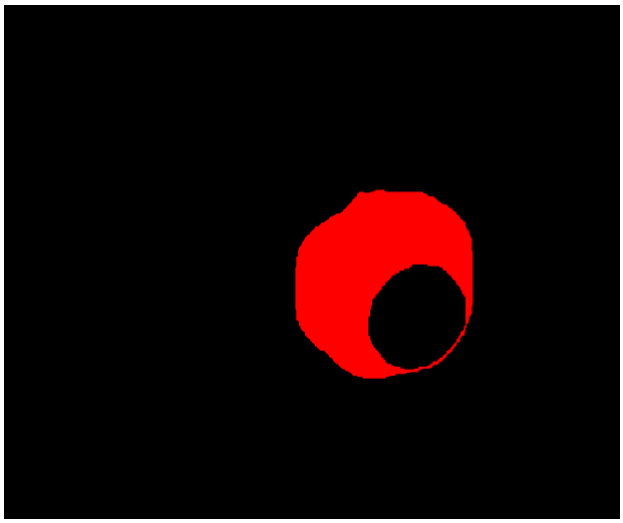

FIG. 4.13 – Mask pour cytoplasme

En passant ces deux images à notre méthode EMPLus on aura le resultat suivant :

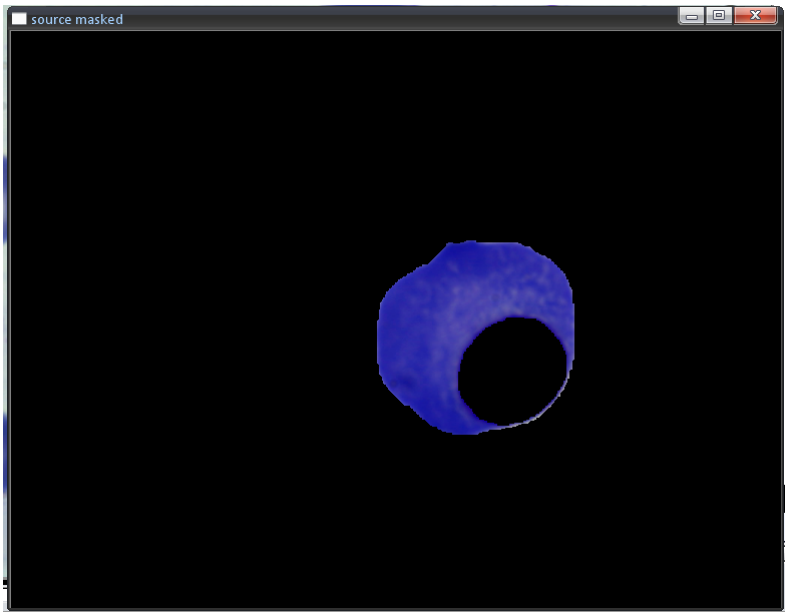

FIG. 4.14 – Le cytoplasme bien identifié

Ce que veut dire, que notre algorithme à identifié à 100% le cytoplasme de l'image de teste

Conclusion générale

La méthode EM et a travers ses diverses variantes[McLachlan et Krishnan, 1997] reste une des méthodes qui ont connu un succès d'usage ces dix dernières années , dans l'univers des méthodes classificatoires de DATA MINING [Xindong & vipin, 2009] , ce n'est pas de hasard qu'elle a marqué son succès, mais grâce à son adaptabilité aux données (manquantes[Nadif, 1991 , grande masse de donnée-images- ….) ,à sa simplicité de modification, à sa robustesse en ce qui concerne le travail complexe qu'elle fasse simultanément(estimation des paramètres et détermination des classes en même temps) ,a sa force d'estimation pour les grandes dimensions, … tous ces critères font d'elle une méthodes digne d'être améliorée et moins d'être à bout de ces fonctionnalités,

pour cela et d'après ce qu'on a pu remarqué dans ce travail on pourrait envisager quelques perspectives a fin de noter ce qu'on peut améliorer dans le travail du clustering automatique via EM basée sur le modèle de mélange .

Comme nous l'avons constaté que EM n'échape pas à la règle de la majorité des méthodes de Clustering , en ce qui concerne la recherche du nombre de classes, en fait , EM requit en avance la détermination de ce nombre « k » par un expert , alors pour automatiser cette détermination , on peut par exemple adopter une stratégie hybride où on commence par une méthode, comme méthode hiérarchique, qui nous renvoi une hiérarchie bien représentante de la répartition des données, et en suit nous lançons EM avec « k » approprié . [Jollois, 2003]

Une autre idée pour trouver une remède à ce problème serait de commencer avec un nombre de classes k , puis au fur et à mesure de l'exécution , on l'évolue en imposant des contraintes de stabilité a fin de trouver le bon « k » qui représente bien le jeu de données.

En termes de Convergence, EM prends de temps pour atteindre sa convergence , alors une des techniques qui ont été proposées dans ce cadre , consiste à faire la classification au même moment de l'etape E-steps de EM , cette technique est en fait , présente actuellement sous

nom d'une nouvelle amélioration de EM , c'est la méthode CEM qui a prouvé qu'il faut en moyenne deux fois plus d'itérations à EM pour converger.(il ya d'autres méthodes basées sur EM : « IEM, SpEM,eM et LEM » [McLachlan et Krishnan, 1997]) .

Voilà en grosso modo pour quoi l'algorithme EM est très présenté dans les études de clustering(cluster analysis) et de ce qu'elle est capable d'offrir aux domaines tels que traitement des grandes données (images,...) et que son caractère de complexité et d'utilisation des modèles n'est qu'un avantage de sa souplesse et adaptabilité .

Bibliographie

[1] [Ball et Hall, 1967] : Ball, G. H. et Hall, D. J. ISODATA, an Iterative Method of Multivariate Analysis and Pattern Recognition. Behavior Science, 153, 1967.

[2] [Bailey & Elkan, 1995] : T. L. Bailey and C. Elkan , “Unsupervised learning of multiple motifs in biopolymers using expectation maximization,” Machine Learning, vol. 21, pp. 51–80, 1995.

[3] [Benzécri, 1973] : Benzécri J.P. L’analyse des données. Dunod, Paris, 197.

[4] [Berrani & al., 2002] : Berrani, S.-A., Amsaleg, L., & Gros, P. Recherche par similarités dans les bases de données multidimensionnelles : panorama des techniques d’indexation. Ingénierie des systèmes d’information (RSTI série ISI-NIS), 7(5-6), pp 65-90.2002.

[5] [Bezdek, 1981] : J. C. Bezdek (1981) : "Pattern Recognition with Fuzzy Objective Function Algoritms", Plenum Press, New York. 1981.

[6] [Bisson, 2000] Bisson, G. , La similarité : une notion symbolique/numérique. Chap. XX of : Apprentissage symbolique-numérique (tome 2). Editions CEPADUES.2002.

[7] [Celeux & al., 1989] : Celeux, G., Diday, E., Govaert, G., Lechevallier, Y., and Ralambondrainy, H. Classification automatique des données, environnement statistique et informatique. DUNOD informatique. 1989.

[8] [Dalirsefat & al., 2009] : Dalirsefat S, Meyer A, Mirhoseini S. Comparison of similarity coefficients used for cluster analysis with amplified fragment length polymorphism markers in the silkworm, Bombyx mori. Journal of Insect Science 9 :71, available online : insectscience.org/9.71 .2009.

[9] [Day, 1969] : N.E. Day : Estimating the Components of a Mixture of Normal Distributions. Biometrika, no 56, pp 464-474, 1969.

[10] [Dempster and al, 1977] : A.P. Dempster, N.M. Laird , and D.B. Rubin : Maximum-vraisemblance from incomplete data via the EM algorithm", J.Royal Statist. Society, B39, pp1-38, 1977

[11] [Dunn, 1973] : J. C. Dunn (1973) : "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", Journal of Cybernetics ,no 3, pp 32-57. 1973.

[12] [Gesu, 1988] V. di Gesu. « Mathematical Morphology and Image Analysis : A Fuzzy Approach ». Workshop on Knowledge-Based Systems and Models of Logical Reasoning, Reasoning, 1988.

[13] [Gower & legendre, 1986] : Gower, J. C. & P. Legendre : Metric and Euclidean properties of dissimilarity coefficients, Journal of Classification, vol 3, pp. 5-48 .1986.

[14] [Govaert, 2003] : Gérard. Govaert, Analyse des données. IC2(série Traitement du signal et de l'image), Lavoisier.2003

[15] [Guillaume & Hervé, 2002] : LECOINTRE Guillaume et LE GUYADER Hervé, Classification phylogénétique du vivant, 2e édition, 2002, Belin, Paris.)

[16] [Guha & al., 1998] : Guha, S., Rastogi, R., et Shim, K. CURE : an e_cient clustering algorithm for large databases. Dans Proceedings of ACM SIGMOD International Conference on Management of Data, pp 73-84, 1998.

[17] [Hadi & benmhammed, 2005] : Fairouz Hadi , Khier Benmahammed , Etude comparative entre la morphologie mathématique floue et le regroupement flou , Faculté des Sciences de l'Ingénieur, Université Ferhat Abbas-Sétif, Algérie.,, 3rd International Conference : SETIT 2005

[18] [Hartley,1958.] : H. Hartley, Maximum likelihood estimation from incomplete data. Biometrics, no 14, pp174–194. 1958.

[19] [Hartigans, 1975] : J. Hartigans. clustering algorithms. John Wiley and Sons, Inc., 1975.

[20] [Jamoulle, & al, 2000] : Marc Jamoulle, Michel Roland , Jacques Humbert, Jean-François Brûlet. Traitement de l'information médicale par la Classification internationale des soins primaires, deuxième version : CISP-2. Care Edition, Bruxelles, 2000

[21] [Jean Jadot, 2007] : Taxonomie et troubles anxiodépressifs : anxiété, dépression, démoralisation l'Information Psychiatrique. Volume 83, No 6,pp 459-66, Juin-Juillet 2007.

[22] [Johnson, 1967] : S. C. Johnson : "Hierarchical Clustering Schemes" Psychometrika, no 2, pp 241-254, 1967.

[23] [Jollois, 2003] : François-Xavier Jollois, Contribution de la classification automatique à la Fouille de Données, thèse de doctorat, Université de Metz, 2003

[24] [Kamvar & al. 2002] : Kamvar, S. D., Klein, D., & Manning, C. D., Interpreting and Extending Classical Agglomerative Clustering Algorithms using a Model-Based approach. Pp 283-290 of : International Conference on Machine Learning (ICML).2002.

[25] [Karypis & al., 1999] : Karypis, G., Eui-Hong, H., et Kumar, V. Chameleon : Hierarchical Clustering Using Dynamic Modeling. Computer, no 32(8) :68-75, 1999.

[26] [kohonen, 1982] : Kohonen T. self-organized formation of topologically correct feature maps. Biological cybernetics no 43, pp59-69, reprinted in Anderson & Rosenfeld , Eds, Neurocomuting : foundations of research, MIT press, Cambridge Ma, 1988.

[27] [Lebart & al. 200] : Lebart L., Morineau A. & piron M.statistique exploratoire multidimensionnelle. Dunod, 3ème édition, paris, 2000.

[28] [Lance & Williams, 1967] : Lance, G.N., & Williams, W.T. : A general theory of classificatory sorting strategies : I. Hierarchical systems. Computer Journal, no 9,pp 373-380, 1967.

[29] [Michael & al, 2007] : W.B. Michael and Malu Castellanos, survey of text mining clustering, classification and retrieval Survey of Text Mining : Clustering, Classification, and Retrieval , Second Edition, Springer, pp 3-22, 2007.

[30] [MacQueen, 1967] : J. B. MacQueen (1967) : "Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability", Berkeley, University of California Press,no 1, pp281-297.1967.

[31] [McLachlan et Krishnan, 1997] : G. McLachlan, and T. Krishnan, The EM algorithm and extensions. Wiley series in probability and statistics. John Wiley & Sons.1997.

[32] [McLachlan & chang, 2004] : G.J. McLachlan and S.U. Chang. Mixture modelling for cluster analysis. Statistical Methods in Medical Research, no 13, pp 347–361,2004.

[33] [McLachlan et Krishnan, 2008] : Geoffrey. McLachlan and Thriyambakam. Krishnan. The EM Algorithm and Extensions (2nd edition). Wiley, New Jersey,2008.»

[34] [Nadif, 1991] : M.Nadif , Classification automatique et données manquantes. Thèse de Doctorat,Université de Metz, 1991.

[35] [oppner & al., 2000] : F. H¨oppner, F. Klawonn, R. Kruse, T. Runkler. Fuzzy Cluster Analysis, Methods for classification, data analysis and image recognition. Wiley, 2000.

[36] [Peter, 2001] : Richard O. Duda, Peter E. Hart, David G. Stork, Pattern classification, Wiley-interscience, 2001.

[37] [Pearson, 1894] : K. Pearson, Contribution to the Mathematic Theory of Evolution. Philo. Trans. Soc., 185, 1894

[38] [Rabiner, 1989.] : L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, NO 77, pp : 257–286, 1989.

[39] [Rennie & al. , 2003] : Rennie J, Shih L, Teevan J, and Karger D. Tackling The Poor Assumptions of Naive Bayes Classifiers. In Proceedings of the Twentieth International Conference on Machine Learning (ICML). 2003.

[40] [Redner and Walker, 1984] : R.A. Redner, H.F. Walker : Mixture densities, maximum vraisemblance and the EM algorithm, SIAM Review, 26, pp195-239, 1984.

[41] [Saporta, 1990] : Saporta G. , probabilities, analyse des données et statistiques. Technip , paris, 1990.

[42] [Sean Borman , 2009] : Sean borman , The Expectation Maximization Algorithm A short tutorial , 2009

[43] [Tanner, 1996] : M. Tanner, Tools for Statistical Inference. Springer Verlag, New York. Third Edition.1996.

[44] [Wang & al, 2006] : J. Wang, A. Dogandzic, and A. Nehorai "Maximum likelihood estimation of compound-Gaussian clutter and target parameters," IEEE Transactions on Signal Processing, vol. 54, no. 10, pp.3884–3898,October 2006.

[45] [Wolfe, 1970] : J.H. Wolfe, Pattern Clustering by Multivariate Mixture Analysis. Multivar. Behavior. Res., no 5, pp 329-350, 1970.

[46] [Xindong & vipin, 2009] : Xindong Wu , vipin Kumar , the top ten Algorithms in Data mining ,chapman & hall/CRC, pp :93-116, 2009.

[47] [Zhang & al., 1996] : Zhang, T., Ramakrishnan, R., et Livny, M. BIRCH : an efficient data clustering method for very large databases. Dans Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pp 103-114, 1996.

Nethographie

[1.1] http ://www.loc.gov/catdir/cpso/lcco/

[1.2] http ://learning.cis.upenn.edu/cis520_fall2009/index.php ?n=Lectures.NaiveBayes

[1.2] http ://fr.wikipedia.org/wiki/Classification_naïve_bayesienne

[3.1] http ://metiss-demo.irisa.fr/descriptions/reco_locuteur/notions_asr.php

[3.2] http ://www.cs.cmu.edu/afs/cs/Web/People/awm/tutorials/gmm14.pdf

[4.1] http ://doc.trolltech.com/

[4.2] http ://opencv.willowgarage.com/wiki/

[A.1] http ://www.statsoft.com/textbook/distribution-fitting/ ?button=1

Annexe

Nous présentons ici quelques rappels sur les notions utiles afin de faciliter la compréhension de ce travail .

1. **Distribution :**

   Une distribution (appelée aussi fonction généralisée) est un objet qui généralise la notion de fonction et de mesure, en probabilité, c'est une loi (loi normal , loi de poisson..) qui décrit les probabilités de chaque valeur d'une variable aléatoire discrète, Il existe deux types de distributions[A.1], à savoir : les « distributions continues » (uniformes, gaussiennes, multi-gaussiennes, de Dirichlet...) et les «distributions discrètes » (binomiales, multinomiales...). La plupart des distributions continues sont également définies dans le cas discret.

2. **Ecart type** &**variance** :

   écart type (noté par «σ»**sigma**) est une mesure réelle positive, (parfois infinie),sert à caractériser la la dispersion d'une variable aléatoire réelle autour de sa moyenne «μ». En effet, ces deux mesures (la moyenne, l'écart type) caractérisent entièrement les lois gaussiennes à un paramètre réel, de sorte qu'elles sont utilisées pour les paramétrer. En peut généraliser la notion de l'écart type, à travers son carré, appelé variance«σ^2», qui permet de caractériser des lois gaussiennes en dimension supérieure.

   Exemple :

   la répartition des notes d'une classe. On peut constater que, plus l'écart-type est faible, plus la classe est homogène.si = 0 cela veut dire que tous les élèves ont eu la même note.

3. **Fonction de vraisemblance** :

   La fonction de vraisemblance, notée $L(x_1, \ldots, x_n \mid \theta_1, \ldots, \theta_k)$ est une fonction de probabilités conditionnelles qui décrit les paramètres θ_j d'une loi statistique en fonction des valeurs x_i supposées connues. Elle s'exprime à partir de la fonction de densité $f(x \mid \theta)$ par : $L(x_1, ..., x_n \mid \theta) = \prod_{i=1}^{n} f(x_i; \theta)$.