Université Abou Bekr Belkaid
Tlemcen Algérie
جامعة أبي بكر بلقايد
تلمسان الجزائر

**République Algérienne Démocratique et Populaire**
**Université Abou Bakr Belkaid– Tlemcen**
**Faculté des Sciences**
**Département d'Informatique**

**Mémoire de fin d'études**
**pour l'obtention du diplôme de Licence en Informatique**

*Thème*

# Développement d'une application à base de l'algorithme de classificationk-means

**Réalisé par :**

- **Belhabib abdelkader**
- **Lagha Omar**

**Encadrer par:**
**Mr F.HADJILA**

*Présenté le 28 Juin 2012 devant la commission d'examination composée de MM.*

- *M.BENAISSA.* *(Examinateur)*
- *N.LABRAOUI* *(Examinateur)*
- *M.MERZOUG* *(Examinateur)*

Année universitaire : 2011-2012

Table des matières

**Table de figure :** ..... **5**

**Résumé :** ..... **6**

**Introduction générale** ..... **7**

**Chapitre 1 : Apprentissage automatique** ..... **8**

1.1 Introduction : ..... 8

1.2 Apprentissage automatique : ..... 8

1.2 .1 Applications : ..... 9

1.2.2 Exemples d'utilisations d'apprentissage automatique : ..... 10

1.2.3 Catégories d'apprentissage automatique : ..... 10

1.2.4 Types d'apprentissage automatique : ..... 10

1.3- Apprentissage supervisé : ..... 11

1.3.1 Définition : (la classification) ..... 12

1.3.2 Définition mathématique : ..... 12

1.3.4 Quelques algorithmes d'apprentissage supervisé: ..... 13

1.4 Apprentissage non supervisé : ..... 13

1.4.1 Définition : ..... 14

1.4.2 Quelques bonnes raisons de s'intéresser à l'apprentissage non supervisé : ..... 14

1.4.3 Les différentes approches à la classification non supervisé: ..... 15

1.4.4 Distance et densité : ..... 16

1.4.5 Difficultés de la classification non supervisée : ..... 17

1.5 Apprentissage semi supervisé: ..... 17

1.5.1 Introduction :.....17
1.5.2 Définition :.....18
1.5.4 Les techniques d'apprentissages semi supervisé :.....18
1.6 Comparaison entre les différentes méthodes d'apprentissage automatique :.....19
1.7 Conclusion :.....20
**Chapitre 2 : k-Means.....21**
2.1 Introduction :.....21
2.2 K-Means :.....22
2.2.1 Définition:.....22
2.2.2 La méthode K-Means :.....23
2.2.3 Les différentes versions de K-Means :.....25
2.2.4 Accélération de k-means :.....27
2 .2.5 Evaluation d'une partition :.....27
2.2.6 Algorithmes de classification à partir de centres :.....28
2.2.7 Les avantages :.....31
2.2.8 Les inconvénients :.....31
2.3 Conclusion :.....31
**Chapitre 3 : Implémentation et conception de prototype.....32**
3.1 Introduction :.....32
3.2 Conception :.....33
3.2.1 Organigramme de l'algorithme de k-means :.....33
3.2.2 Diagramme de cas d'utilisation de k-means :.....34
3.2.3 Diagrammes de séquence de k-means :.....35

3.3 Conclusion : ..... 38

**Conclusion générale ..... 39**

**Référence: ..... 40**

# Table de figure

**FIGURE 1-0-1 : QUELQUES DOMAINES D'APPRENTISSAGE AUTOMATIQUE** ..... 9
**FIGURE 1-0-2 : LES TYPES D'APPRENTISSAGE AUTOMATIQUE** ..... 11
**FIGURE 2-0-1 : CLASSIFICATION A BASE DE K-MEANS(1)** ..... 22
**FIGURE 2-0-2 : CLASSIFICATION A BASE DE K-MEANS(2)** ..... 22
**FIGURE 2-0-3 : EXEMPLE DE CENTROÏDE ET MEDOÏDE** ..... 29
**FIGURE 3-0-1 : ORGANIGRAMME DE L'ALGORITHME DE K-MEANS** ..... 33
**FIGURE 3-0-2: DIAGRAMME DE CAS D'UTILISATION POUR L'ALGORITHME DE K-MEANS** ..... 34
**FIGURE 3-0-3: DIAGRAMME DE SEQUENCE (1)** ..... 35
**FIGURE 3-0-4: DIAGRAMME DE SEQUENCE (2)** ..... 36
**FIGURE 3-0-5 : INTERFACE DE L'APPLICATION K-MEANS (INITIALISATION)** ..... 37
**FIGURE 3-0-6 : AFFICHAGE DE RESULTAT DE CLUSTERING** ..... 37
**FIGURE 3-0-7 : AFFICHAGE DE RESULTAT DE CLUSTERING (DETAILLER)** ..... 38

# Résumé

Le clustering est l'organisation d'un ensemble de données en classes homogènes. Elle a pour but de simplifier la représentation des données initiales. La classification automatique, recouvre l'ensemble des méthodes permettant la construction automatique de tels groupes. Les méthodes de classification non supervisées ont donc un objectif précis : former des classes cohérentes et bien isolées. Les méthodes de classification se sont initialement développées d'un point de vue heuristique autour de méthodes optimisant des critères métriques. Et parmi les algorithmes les plus couramment utilisés on a l'algorithme des centres mobiles (ou k-means). L'objectif de ce mémoire est d'étudier les performances de l'algorithme k means sur un jeu de données construits de façon aléatoire.

# Abstract

The clustering is the organization of a set of data in homogeneous classes. She (it) aims at simplifying the representation of the initial data. The automatic classification, recovers all the methods allowing the automatic construction of such groups. And among the algorithms most usually used we have the algorithm of the mobile centers (or k-means). The objective of this report is to study the performances of the algorithm k means on a game (set, play) of data built in a random way.

# ملخص

التجميع هو تنظيم مجموعة من البيانات إلى فئات متجانسة. انها تهدف الى تبسيط تمثيل البيانات الأولية.فهو يغطي جميع طرق البناء التلقائي لمثل هذه التجميعات .أساليب التصنيف الغير خاضعة للرقابة لها هدف واضح و هو تشكيل مجموعات بشكل متماسك (أو متجانس) ومعزولة بشكل جيد. وقد وضعت في البداية أساليب تصنيف من وجهة نظر ارشادي حول طرق الاستفادة المثلى من معايير مترية. ومن بين الخوارزميات الأكثر شيوعا هي الخوارزمية ذات المراكز المتنقلة. والهدف من هذه الأطروحة هو دراسة أداء هذه الخوارزمية على مجموعة بيانات شيدت عشوائيا.

# Introduction générale

**L**'objectif général de la classification est de pouvoir étiqueter des données en leur associant une classe. L'apprentissage automatique se propose de construire automatiquement une telle procédure de classification en se basant sur des exemples, c'est-à-dire sur un ensemble limité de données disponibles. Si les classes possibles sont connues et si les exemples sont fournis avec l'étiquette de leur classe, on parle **d'*apprentissage supervisé***. Au contraire, **l'*apprentissage non supervisé*** [9] où seuls des exemples sans étiquette sont disponibles et où les classe sont inconnus. L'apprentissage se ramène alors à regrouper les exemples de la manière la plus naturelle possible. Cette volonté de regrouper naturellement est bien sûr ambiguë et le plus souvent formalisée par l'objectif de définir des groupes d'exemples tels que la distance entre exemples d'un même groupe soit minimale et que la distance entre groupes soit maximale (ces deux contraintes vont dans des sens opposés et c'est le meilleur compromis qui doit être trouvé). Cette vision de l'apprentissage non supervisé contraint donc à disposer d'une distance définie sur le langage de description des exemples. On se place ici dans le cas où l'espace de description des exemples est un espace vectoriel numérique (en pratique,$R^n$) dans lequel chaque dimension correspond à un attribut distinct. Chaque exemple est donc décrit par un vecteur d'attributs à valeurs réelles.

Il existe d'autres types d'apprentissage. Citons l'apprentissage semi supervisé, ***l'apprentissage semi-supervisé*** est un bon compromis entre apprentissage supervisé et non-supervisé, car il permet de traiter un grand nombre de données sans avoir besoin de toutes les étiqueter, et, bien utilisé.

À partir de ce paradigme, plusieurs familles de méthodes de *clustering (classification)* ont vu le jour et parmi ces méthodes on a les méthodes basées sur les *K-moyennes ( k-means) [17][12]* .

Dans ce mémoire le travail élaboré s'inscrit dans le cadre de la méthode de k-moyennes (k-means) et notre objectif c'est de présenter une des versions de cette méthode.

# Chapitre 1 : Apprentissage automatique

## *1.1 Introduction*

La faculté d'apprendre est essentielle à l'être humain pour reconnaître une voix, une personne, un objet... On distingue en général deux types d'apprentissage : l'apprentissage «par cœur» qui consiste à mémoriser telles quelles des informations, et l'apprentissage par généralisation où l'on apprend à partir d'exemples un modèle qui nous permettra de reconnaître de nouveaux exemples.

Pour les systèmes informatiques, il est facile de mémoriser un grand nombre de données (textes, images, vidéos...), mais difficile de généraliser. L'apprentissage automatique est une tentative de comprendre et de reproduire cette faculté d'apprentissage. Il nous semble donc approprié d'utiliser des techniques issues pour découvrir et modéliser des connaissances liant texte et image ect..., et pouvoir ainsi réduire le fossé sémantique.

Dans ce chapitre, nous nous intéressons d'abord aux techniques de classification et d'apprentissage.

## *1.2 Apprentissage automatique*

**Définition informelle :**

1. observations d'un phénomène
2. construction d'un modèle de ce phénomène
3. prévisions et analyse du phénomène grâce au modèle le tout automatiquement (sans intervention humaine)

L'apprentissage automatique (machine-learning en anglais) se trouve au carrefour de nombreux domaines : intelligence artificielle, statistiques, sciences cognitives, théorie des probabilités, de l'optimisation, du signal et de l'information... Il est donc bien difficile de donner une taxinomie des techniques d'apprentissages.

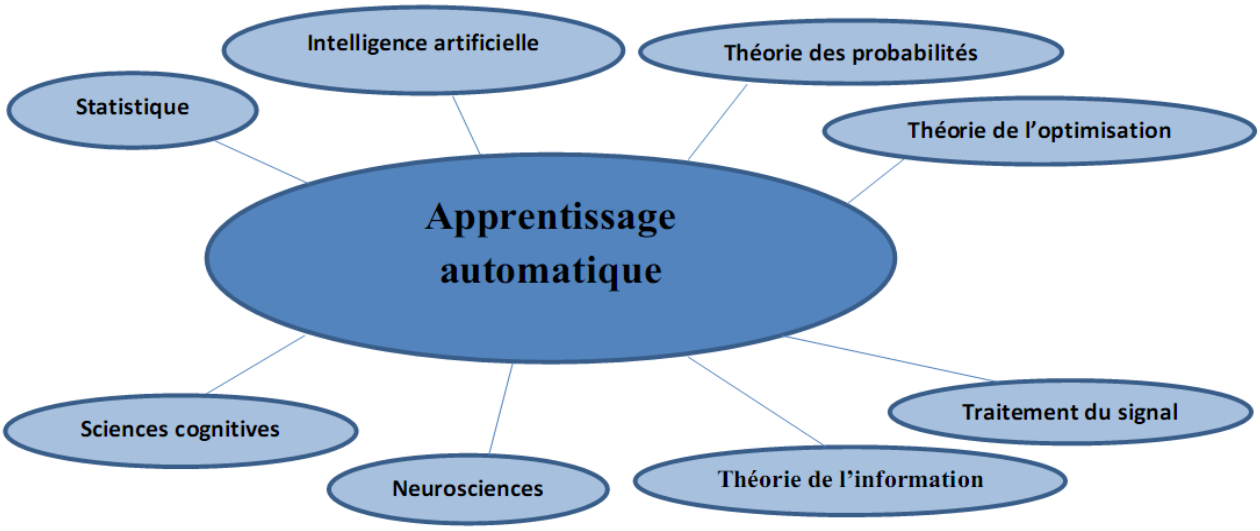


**Figure 1-0-1 : Quelques domaines d'apprentissage automatique**

L'apprentissage automatique consiste à tirer des règles générales à partir d'observations particulières. L'apprentissage automatique a pour objectif d'extraire et d'exploiter automatiquement l'information présente dans un jeu de données. En cela il couvre un vaste champ d'objectifs comme la fouille de données, la classification, la sélection de variables, la discrimination, la régression, la sélection de modèle, la génération et l'inférence de règles, etc. Il s'avère également être fortement pluridisciplinaire puisque selon les données et les objectifs.

**Alors** on peut dire que l'apprentissage automatique c'est la Capacité d'un système à améliorer ses performances via des interactions avec son environnement.

## 1.2 .1 Applications

L'apprentissage automatique est utilisé pour doter des ordinateurs ou des machines de systèmes de : perception de leur environnement : vision, reconnaissance d'objets (visages, schémas, langages naturels, écriture, formes syntaxiques, etc.) ; moteurs de recherche ; aide aux diagnostics, médical notamment, bio-informatique, ; interfaces cerveau-machine ; détection de fraudes à la carte de crédit, analyse financière, dont analyse du marché boursier ; classification des séquences d'ADN ; jeu ; génie logiciel ; sites Web adaptatifs ou mieux adaptés etc.

### 1.2.2 Exemples d'utilisations d'apprentissage automatique

- Un système d'apprentissage automatique peut permettre à un robot ayant la capacité de bouger ses membres mais ne sachant initialement rien de la coordination des mouvements permettant la marche, d'apprendre à marcher. Le robot commencera par effectuer des mouvements aléatoires, puis, en sélectionnant et privilégiant les mouvements lui permettant d'avancer, mettra peu à peu en place une marche de plus en plus efficace.
- La reconnaissance de caractères manuscrits est une tâche complexe car deux caractères similaires ne sont jamais exactement égaux. On peut concevoir un système d'apprentissage automatique qui apprend à reconnaître des caractères en observant des *« exemples »*, c'est-à-dire des caractères connus.

### 1.2.3 Catégories d'apprentissage automatique

• Classification

Classifier le nouvel exemple

• Régression

Faire une prédiction à partir du nouvel exemple

• Estimation de densité

Dire si le nouvel exemple ressemble aux exemples déjà vus.

### 1.2.4 Types d'apprentissage automatique

Les algorithmes d'apprentissage peuvent se catégoriser selon le type d'apprentissage qu'ils emploient, si les classes sont prédéterminées et les exemples étiquetés, on parle alors **d'apprentissage supervisé**. Quand le système ou l'opérateur ne disposent que d'exemples, mais non d'étiquettes, et que le nombre de classes et leur nature n'ont pas été prédéterminés, on parle **d'apprentissage non supervisé [9]**. Et enfin **l'apprentissage semi-supervisé** qui vise à faire apparaître la distribution sous-jacente des exemples dans leur espace de description. Il est mis en œuvre quand des données (ou étiquettes) manquent, le modèle doit utiliser des exemples non-étiquetés pouvant néanmoins renseigner. Plusieurs méthodes d'apprentissage automatique existent. L'utilisation de tel ou tel algorithme dépend fortement de la tâche à

résoudre (classification, estimation de valeurs, etc.), et pour chaque tâche il existe toute une gamme d'algorithmes. Pour un problème d'apprentissage automatique il est souvent difficile de choisir la méthode à utiliser.

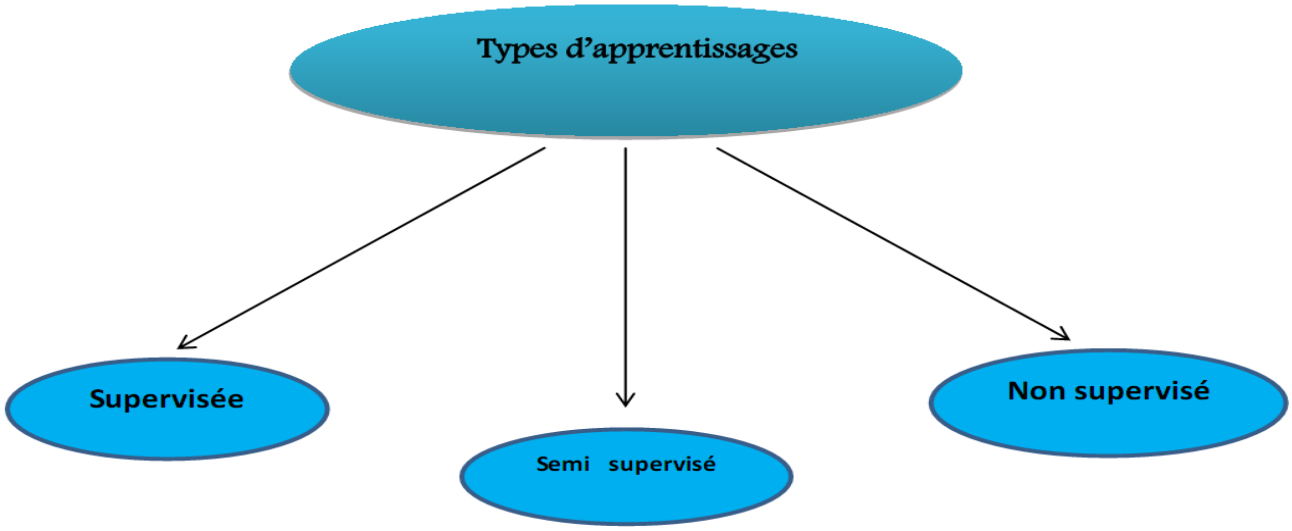


**Figure 1-0-2 : les types d'apprentissage automatique**

## *1.3- Apprentissage supervisé*

L'apprentissage supervisé est une technique d'apprentissage automatique où l'on cherche à produire automatiquement des règles à partir d'une base de données d'apprentissage contenant des *« exemples »* (en général des cas déjà traités et validés). L'apprentissage supervisé consiste à inférer un modèle de prédiction à partir d'un ensemble d'apprentissage, c'est-à-dire plusieurs couples de la forme {observation, étiquette}, où chaque étiquette dépend de l'observation à laquelle elle est associée. L'apprentissage supervisé suppose qu'un oracle fournit les étiquettes de chaque donnée d'apprentissage. On distingue en général trois types de problèmes auxquels l'apprentissage supervisé est appliqué : **la classification supervisée**, la régression, et les séries temporelles. Ces trois types de problèmes se différencient en fonction du type d'étiquettes fournit par l'oracle. Dans le cadre de cette thèse, nous ne nous intéresserons qu'à la classification. Pour ce problème, les étiquettes sont des classes.

### 1.3.1 Définition : (la classification)

La classification supervisée (appelée aussi classement ou classification inductive) a pour objectif « d'apprendre » par l'exemple. Elle cherche à expliquer et à 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. Elle construire un modèle représentatif d'un certain nombre de données organisées en classes (ensemble) que l'on appelle généralement le corpus d'apprentissage - puis d'utiliser ce modèle afin de classer de nouvelles données, c'est à dire de prédire leur classe au vu de leurs caractéristiques (appelées paramètres ou features). La construction du modèle relève de l'apprentissage automatique supervisé, l'ensemble des exemples constituant le corpus d'apprentissage étant annotés, c'est à dire qu'ils portent le label de leur classe donné a priori.

### 1.3.2 Définition mathématique

Une base de données d'apprentissage est un ensemble de couples entrée-sortie $(x_n, y_n)$ $1 \leq n \leq N$ avec $x_n \in X$ et $y_n \in Y$ , que l'on considère être tirées selon une loi sur $X \times Y$ inconnue, **par exemple** $x_n$ suit une loi uniforme et $y_n = f(x_n) + w_n$ où $w_n$ est un bruit centré.

La méthode d'apprentissage supervisé utilise cette base d'apprentissage pour déterminer une représentation compacte de $f$ notée $g$ et appelée *fonction de prédiction*, qui à une nouvelle entrée $x$ associe une sortie $g(x)$.

Le but d'un algorithme d'apprentissage supervisé est donc de généraliser pour des entrées inconnues ce qu'il a pu « apprendre » grâce aux données déjà traitées par des experts, ceci de façon « raisonnable ».

On distingue deux types de problèmes solvables avec une méthode d'apprentissage automatique supervisée :

- $X \subset \mathbb{R}$ : lorsque la sortie que l'on cherche à estimer est une valeur dans un ensemble continu de réels, on parle d'un problème de régression.
- $Y = \{1, \ldots, I\}$ : lorsque l'ensemble des valeurs de sortie est fini, on parle d'un problème de classification, qui revient à attribuer une *étiquette* à chaque entrée.

### 1.3.4 Quelques algorithmes d'apprentissage supervisé

La plupart des algorithmes d'apprentissage supervisés tentent de trouver un **modèle** (une fonction mathématique) qui explique le lien entre des données d'entrée et les classes de sortie. Ces jeux d'exemples sont donc utilisés par l'algorithme.

Il existe de nombreuses méthodes d'apprentissage supervisé :

- **Méthode des k plus proches voisins**.
- **Machine à vecteurs de support (SVM) [13].**
- **Mélanges de lois**.
- **Réseau de neurones**.
- **Arbre de décision**.
- **Classification naïve bayesienne**.
- **Inférence grammaticale**.

## *1.4 Apprentissage non supervisé*

Si seuls des exemples sans étiquette sont disponibles, et si les classes et leur nombre sont inconnus, on parle d'apprentissage non supervisé, ou clustering. Dans ce cas, l'apprentissage se ramène alors à cibler les groupes homogène d'exemples existant dans les données, c'est-à-dire à identifier des groupes, et que les exemples les plus différents soient séparés dans différents groupes, la notion de similarité étant le plus souvent ramenée à une fonction de distance entre paires d'exemples.

L'**apprentissage non supervisé** (parfois dénommé **clustering** ) s'agit pour un logiciel de diviser un groupe hétérogène de données, en sous-groupes de manière à ce que les données considérées comme les plus similaires soient associées au sein d'un groupe homogène et qu'au contraire les données considérées comme différentes se retrouvent dans d'autres groupes distincts ; l'objectif étant de permettre une extraction de connaissance organisée à partir de ces données.

Le clustering est une problématique de recherche étudiée depuis de nombreuses années dans différentes communautés**:** machine learning, data mining, pattern recognition, statistiques, …etc. Son objectif, très général, consiste à séparer un ensemble d'objets en différents groupes (ou clusters) en fonction d'une certaine notion de similarité. Les objets qui sont considérés comme similaires sont ainsi associés au même cluster alors que ceux qui sont considérés comme différents sont associés à des clusters distincts.

### 1.4.1 Définition

**L'**apprentissage non supervisé consiste à inférer des connaissances sur des classes sur la seule base des échantillons d'apprentissage, et sans savoir *a priori* à quelles classes ils appartiennent. Contrairement à l'apprentissage supervisé, on ne dispose que d'une base d'entrées et c'est le système qui doit déterminer ses sorties en fonction des similarités détectées entres les différentes entrées (règle d'auto organisation). On pourrait imaginer que l'algorithme d'apprentissage décide lui-même des classes qui existent et de la classification de chaque exemple.

Contrairement à l'apprentissage supervisé, dans l'apprentissage non-supervisé il n'y a pas d'oracle qui explicite les étiquettes. L'utilisation de ce type d'algorithme permet de trouver des structures, des dépendances entre descripteurs... qui nous sont inconnues (on dit aussi latentes ou cachées).

### 1.4.2 Quelques bonnes raisons de s'intéresser à l'apprentissage non supervisé

- Profusion d'enregistrements et de variables.
- Constituer des échantillons d'apprentissage étiquetés peut être très couteux.
- Découvertes sur la structure et la nature des données à travers l'analyse exploratoire.
- Utile pour l'étude de caractéristiques pertinentes.
- Prétraitement avant l'application d'une autre technique de fouille de données.

- il peut être intéressant de découvrir de l'information sur un grand nombre de données non annotées, et d'utiliser ensuite les méthodes supervisées seulement sur les *clusters* trouvés,
- de meilleurs résultats peuvent être obtenus à l'aide d'une méthode non-supervisée dans le cas où les motifs changent doucement avec le temps,
- les méthodes non-supervisées permettent de découvrir des informations de nature et de structure des données utiles.

### 1.4.3 Les différentes approches à la classification non supervisé

Devant un problème défini de façon aussi imparfaite, il était naturel de voir apparaitre un grand nombre de technique, souvent à fort parfum heuristique. On peut aujourd'hui les regrouper en deux grandes familles : la classification par partition, et la classification hiérarchique.

#### 1.4.3.1 Classification par partition

**° Partitionnement ''dur''**

L'idée générale est de découper l'espace des observations en un certain nombre de régions disjointes, définies par des frontières, et de décréter que toutes les observations situées dans une même région de l'espace appartiennent à une même classe. Chaque classe est représentée par un ''prototype'', observation virtuelle sensée être la plus représentative de la population de la classe. Le prototype d'une classe sera le plus souvent le barycentre des observations de la classe.

Ces prototypes sont positionnés de façon itérative dans les zones à forte densité, et les observations sont affectées aux classes sur la base d'un critère de proximité aux différents prototypes.

**° Partitionnement "doux"**

L'idée selon laquelle chacune des classes réelles, sous-jacentes, occupe une région limitée de l'espace peut paraître irréaliste. En particulier, l'Analyse

Discriminante nous a habitués à penser en termes de classes ayant des distributions multi-normales, et donc se chevauchant nécessairement. Il est donc naturel de considérer la possibilité que les classes empiètent les unes sur les autres.

### 1.4.3.2 Classifications hiérarchiques

La " classification hiérarchique" est une famille de techniques qui génèrent des suites de partitions emboîtées les unes dans les autres, et allant depuis la partition triviale à une seule classe (contenant toutes les observations) jusqu'à la partition triviale où chaque observation est une classe. Entre ces deux extrêmes figurent de nombreuses partitions plus réalistes entre lesquelles l'analyste devra choisir.

a. Méthodes descendantes -*divisive*-

b. Méthodes ascendantes -*agglomerative*-

### 1.4.4 Distance et densité

Les classes rencontrées dans les applications ont souvent des distributions uni-modales : à partir d'un noyau central, la densité des observations décroît de façon monotone dans toutes les directions de l'espace. Beaucoup de techniques de classification non supervisée s'appuient sur cette image et portent une attention particulière aux ensembles d'observations ayant entre elles de faibles distances (régions de forte densité). Elles le font de diverses façons :

***** En utilisant les distances entre observations pour construire les classes (p. ex. méthodes hiérarchiques).

***** En reconnaissant que les zones peuplées mais de faible inertie autour de leurs barycentres sont des zones de forte densité (K-means).

***** En faisant de l'estimation de densité de façon paramétrique (modèles de mélanges) ou non paramétriques (méthodes basées sur l'estimation de densité par K-Premiers Voisins).

### 1.4.5 Difficultés de la classification non supervisée

L'absence d'étiquette de classe est un lourd handicap qui n'est que très partiellement surmontable. Seule l'analyse de la répartition des observations peut permettre de "deviner" où sont les véritables classes. Les deux difficultés essentielles que rencontre la classification non supervisée sont les suivantes :

- ✓ S'il est naturel de reconnaître comme "appartenant à une même classe" des observations regroupées dans une même zone de forte densité, il n'en est pas de même dans des zones de faible densité. En particulier, on peut s'attendre à ce que la définition de frontières entre les classes (image inférieure de l'illustration ci-dessus) soit sujette à caution, et pour le moins hasardeuse.
- ✓ L'œil humain est un extraordinaire outil de classification non supervisée. Malheureusement, il n'est opérationnel que pour des données bidimensionnelles, alors que les données que rencontre l'analyste sont couramment décrites par des dizaines de variables ou plus. Il s'avère que reproduire les performances de l'œil humain dans des espaces de grande dimension est un exploit aujourd'hui hors d'atteinte des machines.

## *1.5 Apprentissage semi supervisé*

### 1.5.1 Introduction

Les algorithmes semi supervisés ont connu un regain d'intérêt ces dernières années dans la communauté d'apprentissage. A la différence de l'apprentissage supervisé, l'apprentissage semi supervisé vise les problèmes avec relativement peu de données étiquetées et une grande quantité de données non étiquetées. Ce type de situation peut se produire quand l'étiquetage des données est coûteux, comme dans le cas de la classification de pages internet. La question qui se pose est alors de savoir si la seule connaissance des points avec labels est suffisante pour construire une fonction de décision capable de prédire correctement les étiquettes des points non étiquetés. Différentes approches proposent de déduire des points non étiquetés des informations supplémentaires et de les inclure dans le problème d'apprentissage.

### 1.5.2 Définition

L'apprentissage semi-supervisé est une classe de techniques d'apprentissage automatique qui utilise un ensemble de données étiquetées et non-étiquetés. Il se situe ainsi entre l'apprentissage supervisé qui n'utilise que des données étiquetées et l'apprentissage non-supervisé qui n'utilise que des données non-étiquetées [4]. Il a été démontré que l'utilisation de données non-étiquetées, en combinaison avec des données étiquetées, permet d'améliorer significativement la qualité de l'apprentissage. Un autre intérêt provient du fait que l'étiquetage de données nécessite l'intervention d'un utilisateur humain. Lorsque les jeux de données deviennent très grands, cette opération peut s'avérer fastidieuse. Dans ce cas, l'apprentissage semi-supervisé, qui ne nécessite que quelques étiquettes, revêt un intérêt pratique évident.

### 1.5.4 Les techniques d'apprentissages semi supervisé

Plusieurs sortes de techniques ont été développées pour réaliser la tâche de l'apprentissage semi-supervisé. Dans cette section, nous allons décrire en bref certaines catégories de ces techniques :

#### 1.5.4.1 Auto-apprentissage (Self-Training)

C'est une technique très répandue pour faire de l'apprentissage semi-supervisé. Le classifieur est d'abord entraîné par les données étiquetées et on utilise le résultat pour classer les données non étiquetées. Les données non étiquetées qui sont classées avec moins d'ambiguïté (par exemple pour un classifieur à sortie probabiliste, on sélectionne des données qui ont une forte probabilité de classement), sont ajoutées aux données étiquetées pour former l'ensemble d'apprentissage, puis on répète la procédure. Nous notons que cette technique utilise ses propres prédictions pour s'améliorer à chaque itération.

### 1.5.4.2 Co-apprentissage (Co-training)

La technique de Co-apprentissage est basée sur l'idée selon laquelle, l'espace des caractéristiques peut être divisé en deux sous-espaces procurant chacun un bon cadre d'apprentissage.

Ainsi, initialement deux classifieurs sont entraînés avec les données étiquetées sur deux sous-espaces différents. Puis chaque classifieur obtenu pour chaque sous-espace, est utilisé pour déterminer la classe probable des données non étiquetées qui seront utilisées pour re-entraîner l'autre classifieur.

### 1.5.4.3 Autres méthodes

Plusieurs autres méthodes existent pour faire de l'apprentissage semi-supervisée. On peut citer la transduction proposée par Vapnik (Transductive SVM), qui est une technique spécifique pour entraîner de façon semi-supervisée les machines à vecteurs de support. Nous avons aussi des méthodes basées sur la régularisation de l'information, sur les arbres, sur la minimisation de l'entropie, etc.

## *1.6 Comparaison entre les différentes méthodes d'apprentissage automatique*

D'une façon générale, plus on a d'exemples, plus il semble intéressant de travailler dans un contexte de classification. Il existe de nombreuses méthodes de classification. Il n'y pas de méthodes globalement meilleures que les autres. Une bonne connaissance du problème est nécessaire pour choisir la bonne méthode à utiliser. Le choix de la méthode dépend notamment du problème posé, de la nature des données, des propriétés de la fonction à estimer... De plus, la difficulté intrinsèque du problème dépend de la qualité des données. En effet, dans la pratique, les données peuvent être fausses, incomplètes, manquantes, non-exhaustives, les résultats sont donc souvent imprécis. Avec des algorithmes exacts sur des données réelles, les résultats fournis sont justes par rapport aux données, mais pas nécessairement par rapport à la réalité.

## *1.7 Conclusion*

Dans ce chapitre nous avons présenté brièvement les différentes méthodes (types) d'apprentissage automatique. Parmi les méthodes proposées dans la littérature, nous nous sommes plus particulièrement intéressés par la méthode d'apprentissage non supervisé. On peut distinguer deux grandes familles de cette méthode d'apprentissage: les méthodes de partitionnement simple et les méthodes hiérarchiques. Les deux méthodes et leurs performances sont fortement dépendantes de la distance utilisée. En pratique, il peut s'agir de la distance euclidienne ou, au mieux, d'une distance fournie par un expert du domaine. Celui-ci affecte alors un poids à chaque attribut, poids qui traduit l'importance de cet attribut pour le problème considéré.

# Chapitre 2 : k-Means

## *2.1 Introduction*

Le partitionnement des données est une tâche importante en analyse de données, elle divise un ensemble de données en plusieurs sous-ensembles, ces sous-ensembles appelés groupes ou clusters. Ces groupes sont caractérisés idéalement par une forte similarité à l'intérieur et une forte dissimilarité entre les membres de différents groupes [5].

L'usage de cette technique vise à identifier un résumé de la structure interne de ces données, sans aucune connaissance a priori sur les caractéristiques des données [2]. Cela touche plusieurs domaines dont la reconnaissance des formes, l'imagerie, la bioinformatique et l'indexation des bases d'images. Dans ce cadre plusieurs méthodes ont été développées, la plus populaire est celle des k moyennes (K-means), elle doit sa popularité à sa simplicité et sa capacité de traiter de larges ensembles de données [7].

Cependant, la principale limite de cette méthode est la dépendance des résultats des valeurs de départ (centres initiaux). À chaque initialisation correspond une solution différente (optimum local) qui peut dans certain cas être très loin de la solution optimale (optimum global). Une solution naïve à ce problème consiste à lancer l'algorithme plusieurs fois avec différentes initialisations et retenir le meilleur regroupement trouvé. L'usage de cette solution reste limité du fait de son coût et que l'on peut trouver une meilleure partition en une seule exécution.

## *2.2 K-Means*

### 2.2.1 Définition

K-means est un algorithme de quantification vectorielle (clustering en anglais). K-means est un algorithme de minimisation alternée qui étant donné un entier K, va chercher à séparer un ensemble de points en K clusters .

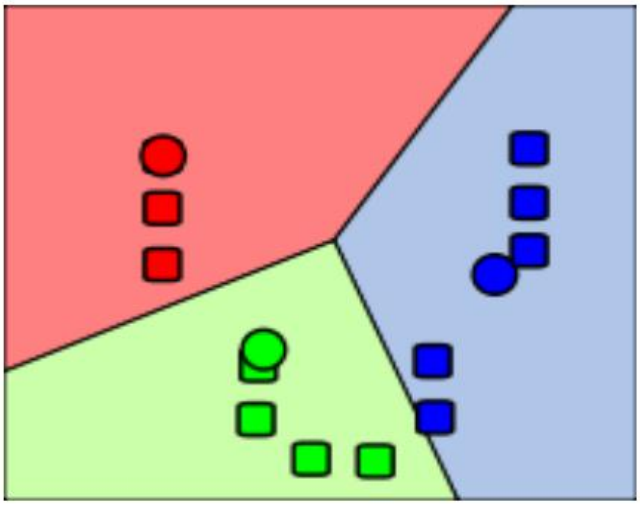

**Figure 20-1 : Classification à base de K-Means(1)**

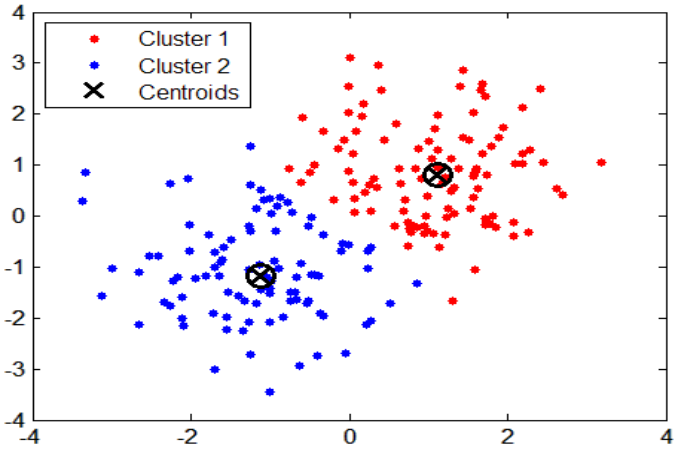


**Figure 2-0-2 : Classification à base de K-Means(2)**

K-means définit par McQueen [3] est un des plus simples algorithmes de classification

automatique des données. L'idée principale et de choisir aléatoirement un ensemble de centres fixé a priori et de chercher itérativement la partition optimale.

Chaque individu (également appelé centroïde ou centroid en anglais) est affecté au centre le plus proche, après l'affectation de toutes les données la moyenne de chaque groupe est calculé, elle constitue les nouveaux représentants des groupes, lorsqu'ont abouti à un état stationnaire (aucune donnée ne change de groupe) l'algorithme est arrêté.

#### 2.2.1.1 Exemples d'applications

Marketing : segmentation du marché en découvrant des groupes de clients distincts à partir de bases de données d'achats.

Environnement : identification des zones terrestres similaires (en termes d'utilisation) dans une base de données d'observation de la terre.

Assurance : identification de groupes d'assurés distincts associés à un nombre important de déclarations.

Planification de villes : identification de groupes d'habitations suivant le type d'habitation, valeur, localisation géographique, …

Médecine : Localisation de tumeurs dans le cerveau

Nuage de points du cerveau fournis par le neurologue.

Identification des points définissant une tumeur.

### 2.2.2 La méthode K-Means

L'algorithme de k-means, est de loin l'outil le plus populaire utilisé dans les applications scientifiques et industrielles de clustering. Le nom dérive du fait que, pour représenter chacune des *K* classes *Ck*, on utilise la moyenne (ou la moyenne pondérée) $\pi k$ de ses points, appelée centroide (ou centre de masse).

Chacune des *C* composantes du vecteur $\pi k$ est calculee par:

$$\pi kj = \frac{1}{|Ck|} \sum_{Oi \epsilon Ci} Pij$$

Dans le cas de données numériques, cela donne un sens géométrique et statistique à la

méthode. L'inertie intra-classe constitue le critère à optimiser. Elle est définie comme la moyenne des carrés des distances des objets de la classe au centre de gravité de celle-ci. On cherche ainsi à construire des classes compactes.

L'inertie intra-classe associée à la classe *Ck* s'écrit formellement

$$Ik=\frac{1}{|Ck|}\sum_{Oi\epsilon Ci} d^2\,(O\mathrm{i}, \pi\mathrm{kj})$$

L'objectif est alors de minimiser la somme de l'inertie intra-classe sur l'ensemble des classes. L'algorithme procède en deux étapes : dans la première phase, on réassigne tous les objets au centroïde le plus proche, et dans la deuxième phase, on recalcule les centroïdes des classes qui ont été modifiées. Pour mesurer la proximité entre un centroïde et un objet, on calculera une distance entre ce deux vecteurs. On pourra utiliser, par exemple, la distance euclidienne, calculée de la manière suivante : $\mathrm{d}(O\mathrm{i}, \pi\mathrm{k})=\sqrt{\sum_{j=1}^{C}(P\mathrm{ij}-\pi\mathrm{kj})^2}$

Les deux phases sont itérativement répétées jusqu' à ce qu'un critère d'arrêt soit atteint (par exemple, si aucune modification n'a eu lieu, ou si le nombre maximum d'itérations a été atteint).

Les principaux problèmes de l'approche des *k-means* comme des autres approches partitionnelles, sont l'influence de la partition initiale (qui est souvent choisie de façon aléatoire), et le choix du paramètre *K* qui n'est pas toujours évident.

Soit $P_0 = \{C_1,\ldots\ldots\ldots\ldots\ldots,C_k\}$

**Répéter**

***Affectation* : générer une nouvelle partition en assignant chaque objet groupe dont le centre de gravité est le plus proche.**

***Représentation* : calculer les centres de gravité associés à la nouvelle Partition.**

**Jusqu'à *convergence de l'algorithme vers une partition stable*.**

### 2.2.2.1 Algorithme de K-means

Cette méthode est supervisée dans le sens où le nombre de classes doit être donné, mais pas nécessairement leurs paramètres. Son algorithme est le suivant :

- Choisir *k* moyennes $\mathbf{m}_1^{(1)},\ldots,\mathbf{m}_k^{(1)}$ initiales (au hasard par exemple)
- Répéter jusqu'à convergence:

- assigner chaque observation à la moyenne la plus proche (*i.e* effectuer une partition de Voronoï selon les moyennes).

$$S_i^{(t)} = \left\{ \mathbf{x}_j : \left\| \mathbf{x}_j - \mathbf{m}_i^{(t)} \right\| \leq \left\| \mathbf{x}_j - \mathbf{m}_{i^*}^{(t)} \right\| \text{ for all } i^* = 1, \ldots, k \right\}$$

- mettre à jour la moyenne de chaque cluster

$$\mathbf{m}_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{\mathbf{x}_j \in S_i^{(t)}} \mathbf{x}_j$$

La convergence est atteinte quand il n'y a plus de changement.

### 2.2.3 Les différentes versions de K-Means

Il y a plusieurs versions de k-means. On peut les distinguer selon deux critères : la différence dans la mise à jour des classes et le critère pour faire cette mise à jour.

Pour le premier critère, les algorithmes de ce type diffèrent dans le détail de la génération et de l'ajustement des classes. Il y a 3 algorithmes de base de ce type :
Standard K-means, l'algorithme de Lloyd, et continuous K-means qui a été proposé par McQueen en 1967 [15].

### 2.2.3.1 L'algorithme de Lloyd

L'initialisation de l'algorithme est similaire à la description ci-dessus. Les ajustements sont réalisés en calculant le centroïde pour chaque classe et en utilisant ces centroids comme les points de référence dans l'itération suivante pour tous les points de données. La mise à jour des centroïdes n'est faite qu'**après** une itération.

### 2.2.3.2 Standard k-means

Cet algorithme est meilleur que celui de Lloyd en terme de l'utilisation plus efficace de l'information à chaque pas d'itération. C'est à dire la mise à jour des centroïdes est faite **pendant** et **après** une itération. Si un point appartient à une classe et que pour lui, le centroïde de cette classe est le point de référence le plus proche, alors il n'y aura aucun ajustement. Mais si après avoir affecté un point x à une classe A, on trouve qu'il y a une autre classe B dont le centroïde est le point de référence plus proche de x que celui de A, alors il faut réaffecter x à la classe B et recalculer les centroïdes de toutes les deux classes, et les points de référence de ces deux classes se déplacent aux nouveaux centroïde.

### 2.2.3.3 Continuous k-means

Cet algorithme diffère au standard k-means par le choix des points de référence initiaux et la sélection des points pour la mise à jour des classes. Dans la première différence, pas comme dans Lloyd ou standard k-means où les points de référence initiaux sont arbitrairement choisis, dans cet algorithme, ces points sont choisis comme un échantillon aléatoire de la population entière des points. Si l'échantillon est assez gros, alors la distribution des points de référence initiaux pourrait refléter celle des points de la population. La deuxième différence, contrairement au standard k-means où tous les points sont séquentiellement examinés, cet algorithme n'examine qu'un échantillon aléatoire des points. Si le jeu de données est gros et l'échantillon est représentatif du jeu de données, alors l'algorithme peut converger plus vite qu'un algorithme qui doit examiner séquentiellement tous les points.

Pour le deuxième, il y a deux versions de l'optimisation itérative de k-means [12].

### 2.2.3.4 L'algorithme de Forgy

Est similaire à l'algorithme EM et ses itérations disposent de deux pas : réaffecter tous les points à leur centroïde le plus proche et recalculer les centroïdes des nouveaux groupes créés. Les itérations continuent jusqu'à ce qu'on atteigne un critère de terminaison (par exemple, il n'y a plus de réaffections).

Les avantages sont la capacité de travailler sur toutes les normes Lp, la facilité de paralléliser, l'insensibilité à l'ordre des données.

### 2.2.3.5 L'algorithme d'optimisation itérative

Réaffecte les points en se basant sur une analyse plus détaillée des effets sur la fonction objective quand un point est déplacé de sa classe à une classe potentielle. Si l'effet est positif, ce point sera réaffecté et deux centroïde seront recalculés. L'expérimentation prouve que la version classique est souvent meilleure que celle de Forgy [13].

### 2.2.4 Accélération de k-means

En utilisant l'inégalité triangulaire, « Elkan » a proposé une amélioration qui permet de diminuer le coût de calcul des distances dans k-means. L'idée est basée sur le fait que la plupart des calculs de distance sont redondants. Si un point se trouve très loin d'un centre, alors ce n'est pas nécessaire de calculer la distance exacte entre lui et ce centre afin de savoir si ce point peut être affecté à ce centre. En plus, si un point est vraiment plus proche d'un centre que tous les autres centres, on n'a pas besoin de calculer les distances exactes pour décider si le point appartient à ce centre.

### 2 .2.5 Evaluation d'une partition

Un critère général pour évaluer les résultats d'un clustering consiste à comparer la partition calculée avec une partition" correcte". Cela signifie que les instances des données sont déjà associées à des étiquettes jugées correctes, et que l'on va pouvoir quantifier la conformité entre étiquettes calculées et étiquettes correctes. Une mesure classique est l'indice de Rand pour évaluer la conformité entre deux partitions de $L$ éléments.

Si C = {$C1$ ……..$Cs$}est la structure issue de la classification et que

$P$={$P1$ …….. $Pt$} est une partition prédéfinie, chaque paire de points peut être affectée au même cluster ou à deux clusters différents. Soit $a$ le nombre de paires appartenant au même cluster de C et au même cluster de P. Soit $b$ le nombre de paires dont les points appartiennent à deux clusters différents de C et à deux clusters différents de P. La conformité entre C et P peut être estimée au moyen de la formule : $\text{Rand}(C,P)=\frac{a+b}{L.(L-1)/2}$

Cet indice prend des valeurs entre 0 et 1 et il est maximisé lorsque $s = t$.

Nous utilisons l'indice de Rand pour calculer la précision dans nos expériences.

### 2.2.6 Algorithmes de classification à partir de centres

Dans cette section, nous présentons un ensemble d'algorithmes pour lesquels les classes sont représentées par un centre. Ce dernier est soit une combinaison d'attributs des éléments de la classe, soit un sous-ensemble de la classe.

Ces algorithmes sont, pour la plupart, de type k-means et proposent généralement une partition **dure** comme résultat.

#### 2.2.6.1 Représentation d'une classe

La représentation d'une classe est une description de l'ensemble des individus constituant la classe. Cette description, caractérisée par un ensemble de paramètres, permet de définir la forme et la taille de la classe dans un espace de données.

Dans les algorithmes de classification, la description d'une classe est simplement caractérisée par une combinaison linéaire des individus ou un axe médian ne donnant qu'une forme implicite, celle du centre de la classe. Une telle description permet d'assigner chaque individu à un centroïde ou à un axe médian suivant une règle, qui est usuellement l'affectation d'un individu au centre qui lui est le plus proche.

##### *2.2.6.1.1 Représentation avec les centroïdes*

La représentation d'une classe C est définie comme étant la moyenne des éléments présents dans C. Plus précisément, un centroïde est un vecteur de termes pondérés pour lequel chaque composante correspond à la moyenne arithmétique des composantes correspondantes, de tous les vecteurs d'individus présents dans C. Pour une classe C donnée, le vecteur du centroïde V(C) est défini par l'équation :

$$V(C)=\frac{1}{C}\sum_{i=1}^{|C|} d_i$$

### *2.2.6.1.2 Représentation avec les médoïdes*

Un deuxième mode de représentation des classes consiste à prendre k individus parmi tous les individus d'une classe. Ces k éléments sont centraux vis à vis de la classe, c'est-à-dire qu'ils sont proches du centre géométrique de la classe.

Cette représentation permet de représenter la classe non pas par un point unique mais par un noyau censé définir au mieux la classe. Le choix de la valeur de k est empirique (dans [6],K=3 ). Le choix des éléments du noyau est fondé sur un calcul de distances entre tous les individus de la classe. Pour une classe Cj donnée, le noyau N(Cj) correspond aux individus qui ont la plus petite somme de distances par rapport aux autres individus de la classe.

Pour une classe Cj, X est un élément du noyau s'il minimise l'inertie Ij définie par l'équation :

$$Ij=\sum_{c\in C_j} \mu_j \, d^2(X,c)$$ Avec $\mu_j$ un poids

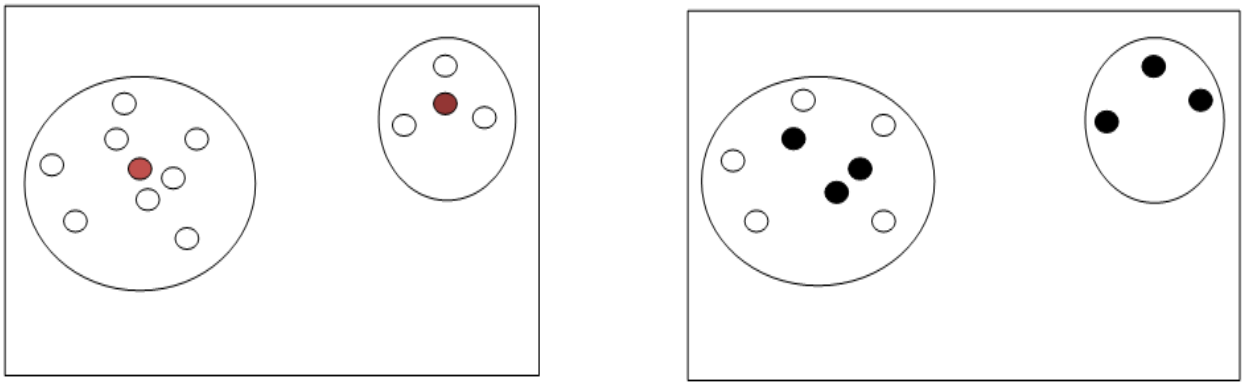

**Figure 2-0-3 : Exemple de centroïde et médoïde**

Dans les sections suivantes, nous présentons quelques méthodes fondées sur les centroïdes, dont la plus connue est k-means, ainsi que celles fondées sur les médoïdes.

### 2.2.6.2 Algorithme fuzzy k-means

L'algorithme fuzzy k-means (FKM) [1] est proche de l'algorithme k-means, à la différence près que FKM utilise une fonction d'appartenance graduelle au lieu d'une fonction de partitionnement.

FKM(X,C)=$\sum_{i=1}^{nr}\sum_{j=1}^{k} u_i^r \|x_i - c_i\|^2$

avec $\sum_{j=1}^{k} u_{ij}$=1,$\forall i$ et $u_{ij}$≥0 et avec $r$≥1 , où $r$ est le degré de regroupement entre classes et C l'ensemble des centres. Le paramètre $u_{ij}$ représente le degré d'appartenance du document $x_i$ au centre Cj.

### 2.2.6.3 Algorithme K-Harmonic means

L'algorithme « K-Harmonic means » (KHM) [16] est similaire à k-means uniquement dans le sens où cette approche est une méthode itérative basée sur les centres. Le but de cette approche est de pallier le problème de l'initialisation des centres, que l'on rencontre pour les méthodes k-means ou EM, par exemple. Cet algorithme diffère notamment de k-means par le critère d'optimisation (ou fonction d'objectivité). En effet, ce critère est fondé sur la moyenne harmonique de la distance de chaque document avec tous les centres :

$$KHM(X,C)=\sum_{i=1}^{n} \frac{k}{\sum_{j=1}^{k} \frac{1}{\|x_i - c_j\|^p}}$$

Cette fonction d'objectivité donne un bon score (c'est-à-dire un poids faible) pour tout document qui se trouve proche d'au moins un centre. Cette fonction se comporte comme la fonction min de k-means. Ceci est la propriété voulue de cette fonction pour mesurer la qualité des classes retrouvées. A noter que p est un paramètre de l'algorithme.

### 2.2.7 Les avantages

- L'algorithme de k-means est très populaire du fait qu'il est très facile à comprendre et à mettre en œuvre.
- La méthode résolve une tâche non supervisée, donc elle ne nécessite aucune information sur les données.
- Sa simplicité conceptuelle.
- Sa rapidité et ses faibles exigences en taille mémoire.
- La méthode est applicable à tout type de données (mêmes textuelles), en choisissant une bonne notion de distance.

### 2.2.8 Les inconvénients

- La partition finale dépend de la partition initiale. Le calcul des centroïdes, après chaque affectation d'un individu, influence le résultat de la partition finale. En effet, ce résultat dépend de l'ordre d'affectation des documents.
- Le nombre de classes est un paramètre de l'algorithme. Un bon choix du nombre k est nécessaire, car un mauvais choix de k produit de mauvais résultats.

## *2.3 Conclusion*

Finalement, l'algorithme k-means est très populaire du fait qu'il est très facile à comprendre et à mettre en œuvre. Le degré d'appartenance d'un document à une classe étant binaire et la pondération de chaque document étant constante, cela facilite son utilisation dans différents domaines de recherche tels que la génomique [10], bio-informatique, chémoinformatique, analyse financière, classification des séquences d'ADN, génie logiciel, locomotion de robots, etc…

# Chapitre 3 : Implémentation  et conception de prototype

## *3.1 Introduction*

Le clustering K-means est une méthode bien connue d'attribution de l'appartenance au cluster qui repose sur la minimisation des différences entre les éléments d'un cluster et la maximisation de la distance entre les clusters. Le terme « means » dans K-means fait référence au *centroïde* du cluster, c'est-à-dire un point de données choisi arbitrairement puis affiné de manière itérative jusqu'à ce qu'il représente la moyenne vraie de tous les points de données dans le cluster. La lettre « k » fait référence au nombre arbitraire de points qui sont utilisés pour ensemencer le processus de clustering. L'algorithme K-means calcule les distances euclidiennes au carré entre les enregistrements de données dans un cluster et le vecteur qui représente la moyenne du cluster, puis converge vers un jeu final de k clusters lorsque cette somme atteint sa valeur minimale.

L'algorithme K-means attribue chaque point de données à un seul cluster et n'accepte pas l'incertitude de l'appartenance. L'appartenance dans un cluster est exprimée sous forme d'une distance par rapport au centroïde.

L'algorithme K-means est généralement utilisé pour créer des clusters d'attributs continus du fait de la simplicité du calcul de la distance à une moyenne.

## *3.2 Conception*

### 3.2.1 Organigramme de l'algorithme de k-means

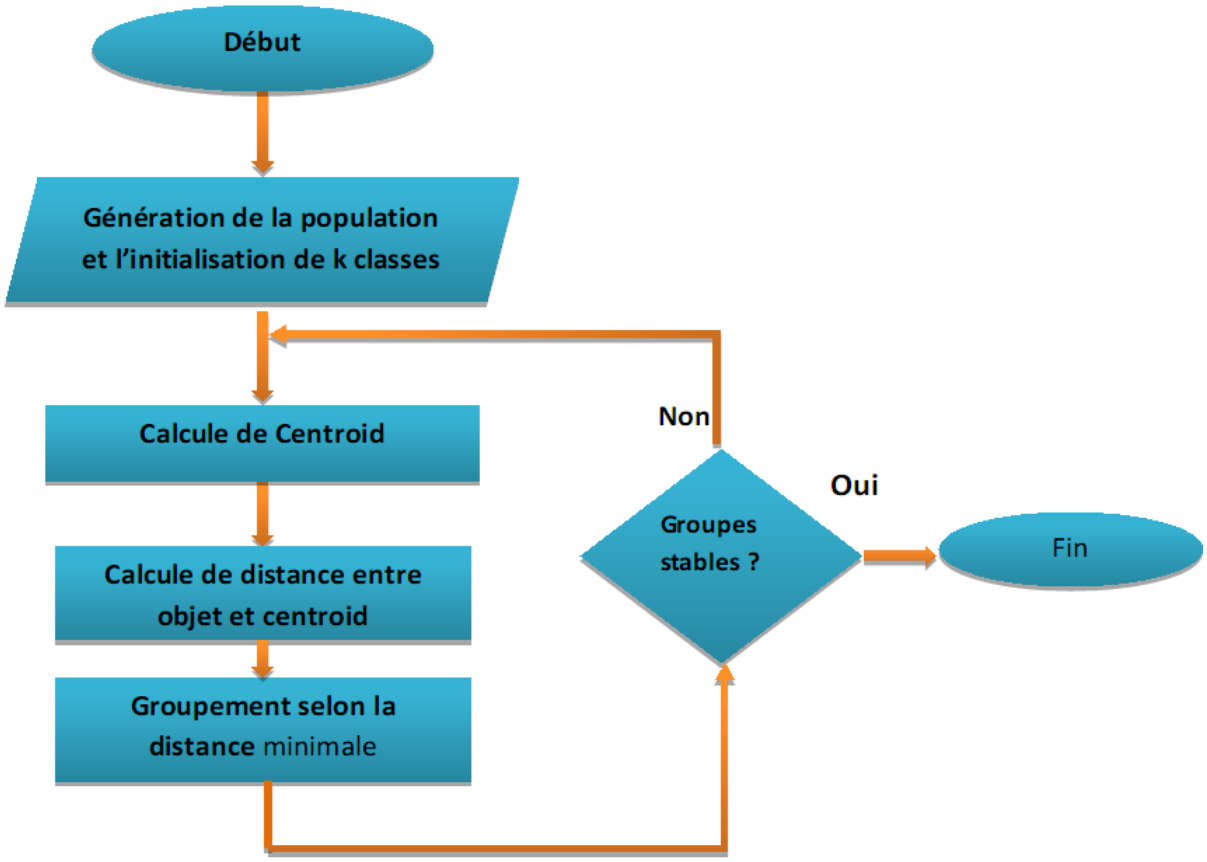


**Figure 3-0-1 : Organigramme de l'algorithme de k-means**

**Principe général**

L'algorithme consiste à grouper les points selon un critère bien déterminé.

L'entrée de l'algorithme est le nombre k de groupes (cluster). Une fois le nombre de groupes saisi, l'algorithme choisit arbitrairement k points comme centres «initiaux » des k groupes.

L'étape suivante consiste à calculer la distance entre chaque individu (point) et les k centres, la plus petite distance est retenue pour inclure cet individu dans le groupe ayant le centre le plus proche.

Une fois tous les individus groupés, on aura k sous-nuages (cluster) disjoints du nuage total. Pour chaque groupe, l'algorithme calcule le nouveau centre de gravité.

L'algorithme s'arrête lorsque les groupes construits deviennent stables.

### 3.2.2 Diagramme de cas d'utilisation de k-means

L'utilisateur étant l'acteur principal. Les cas d'utilisation de base qui vont être mis en évidence pour réaliser l'ensemble des groupes seront :

- Configuration de k-means
    - Saisir le nombre de points.
    - Saisir le nombre cluster.
    - Saisir le nombre d'itération maximale.
- Clutering

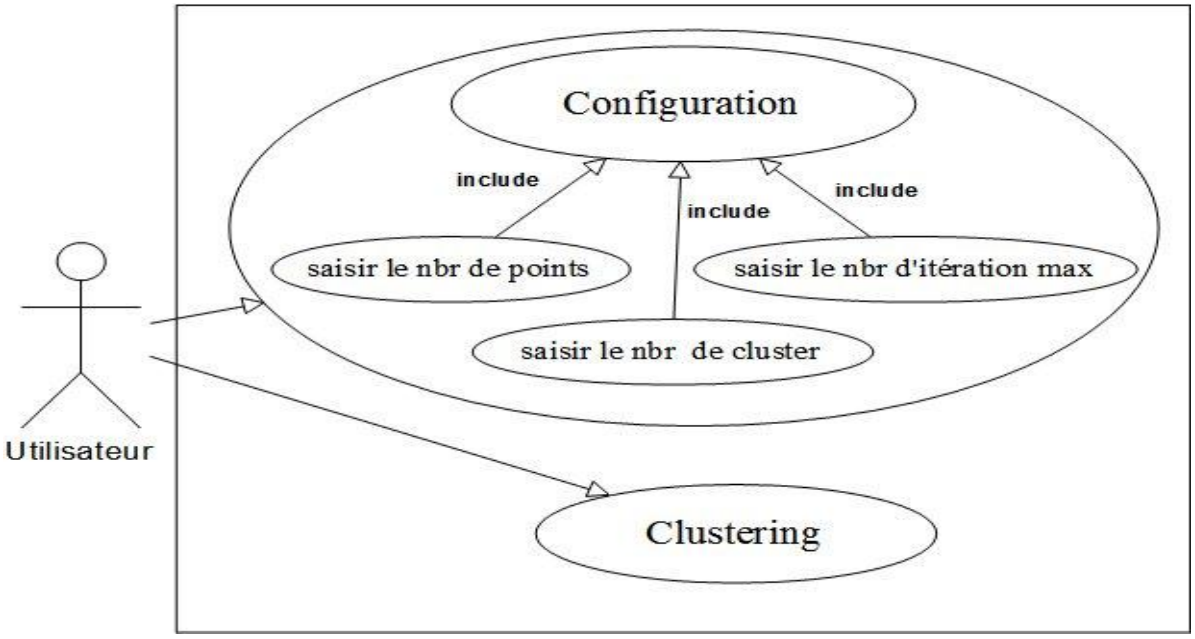


**Figure 3-0-2: Diagramme de cas d'utilisation pour l'algorithme de k-means**

### 3.2.3 Diagrammes de séquence de k-means

Dans cette phase, et après identification de cas d'utilisation, nous représentons la méthode de k-means à l'aide des diagrammes de séquence :

1) « Initialisation de processus du clustering avec des paramètres correctes »
2) « Initialisation de processus du clustering avec des paramètres erronés »

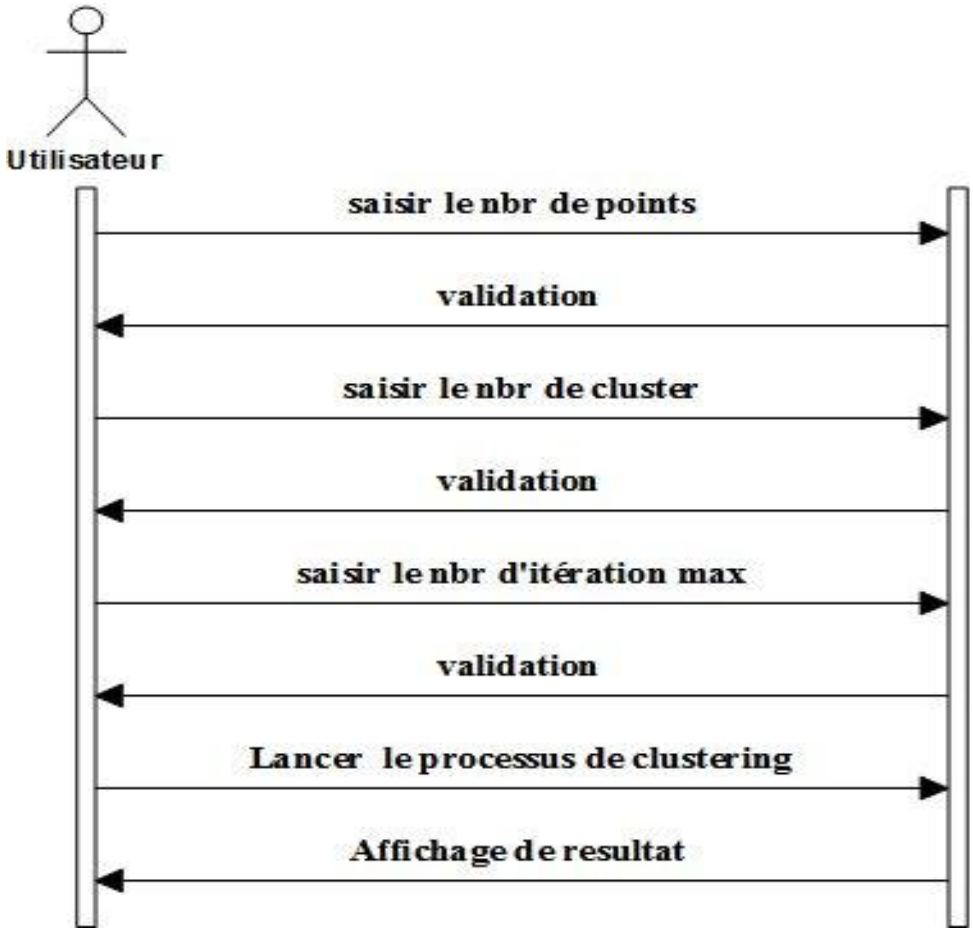


**Figure 3-0-3: Diagramme de séquence (1)**

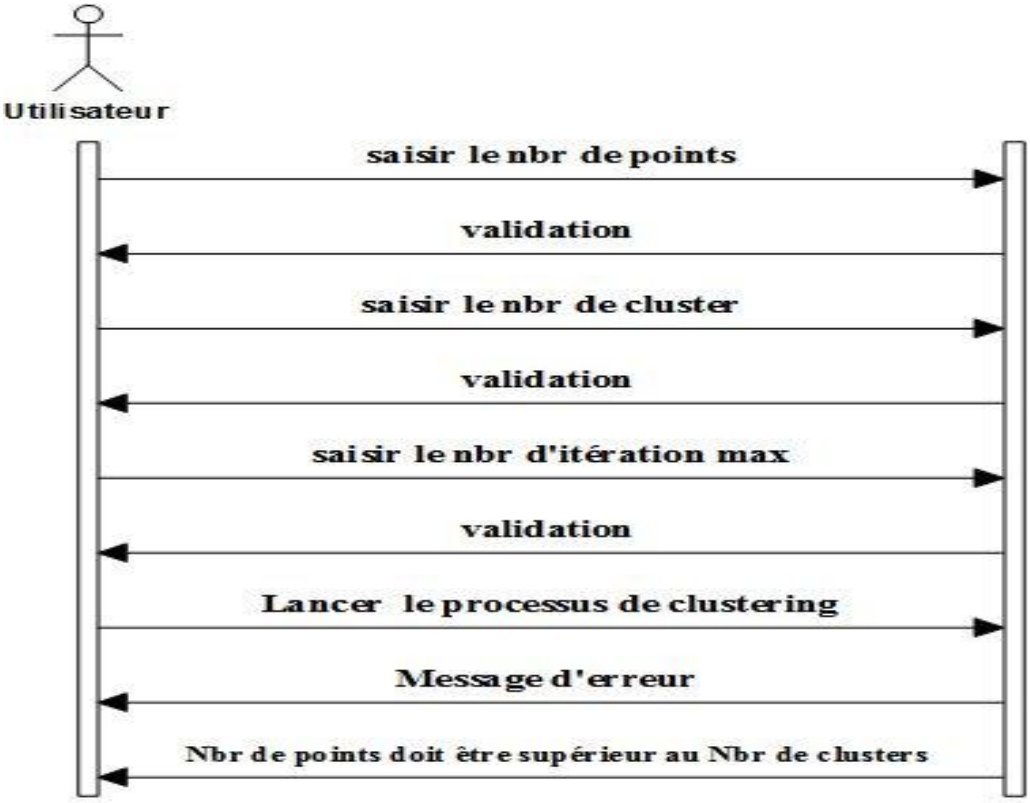


**Figure 3-0-4: Diagramme de séquence (2)**

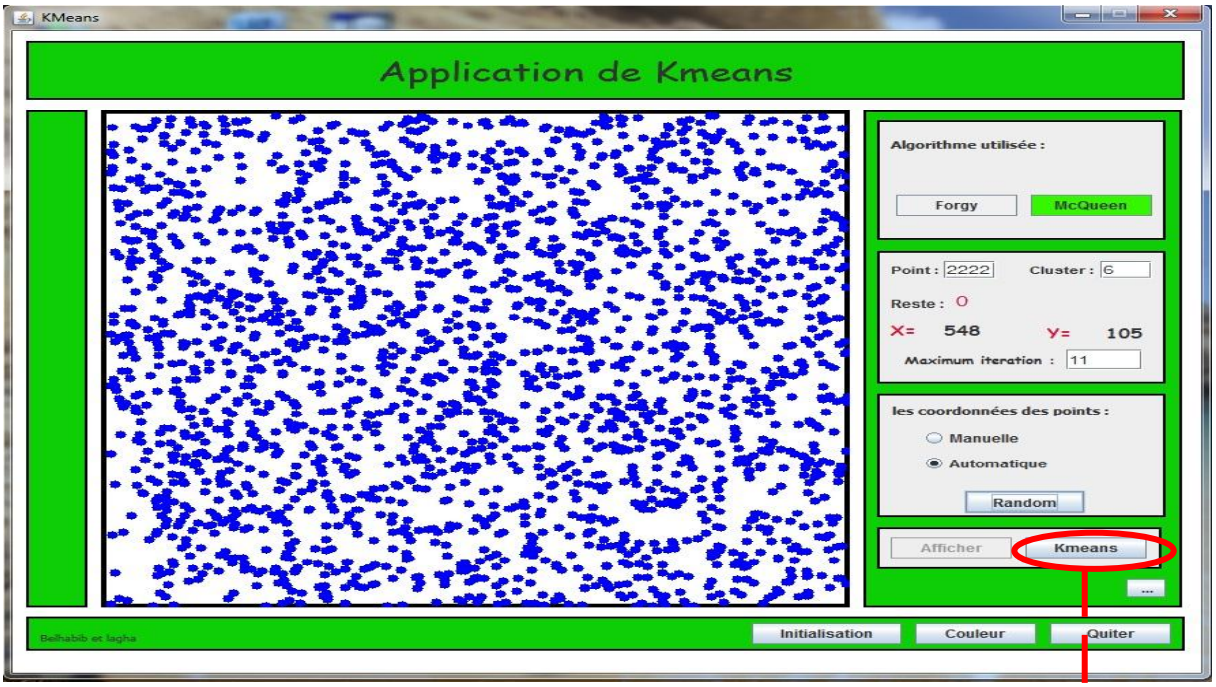


**Figure 3-0-5 : Interface de l'application k-means (initialisation)**

**Exécuter**

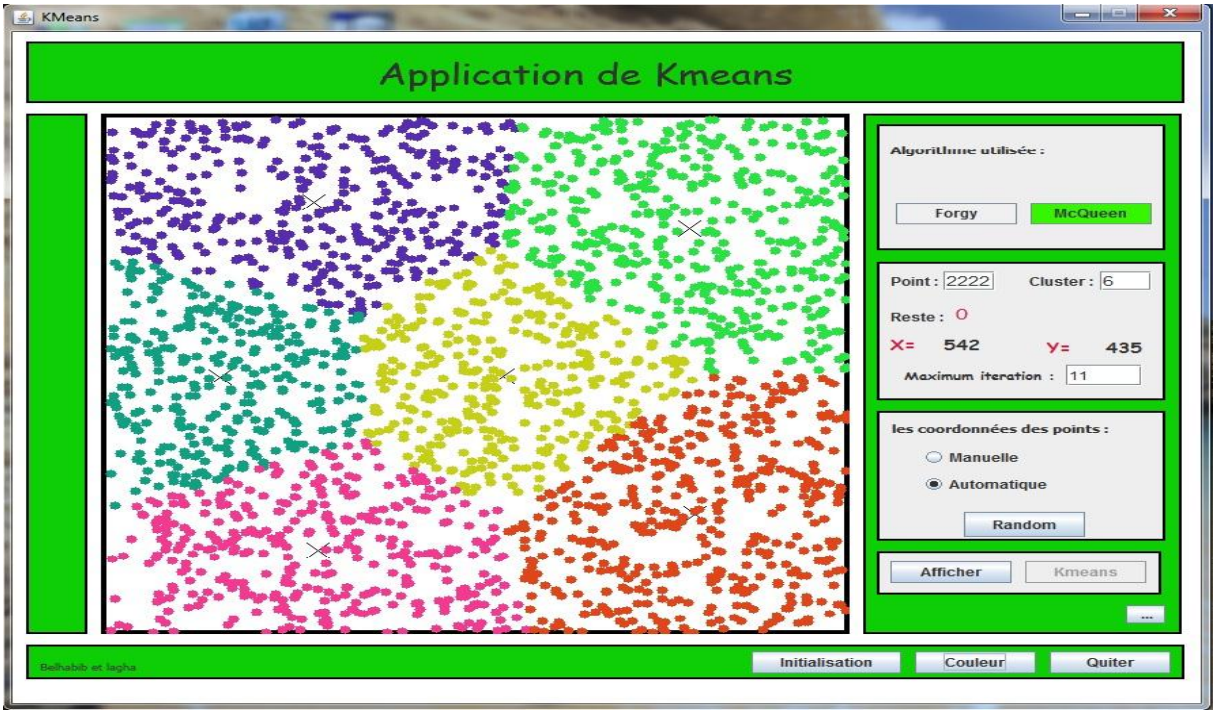


**Figure 3-0-6 : affichage de résultat de clustering**

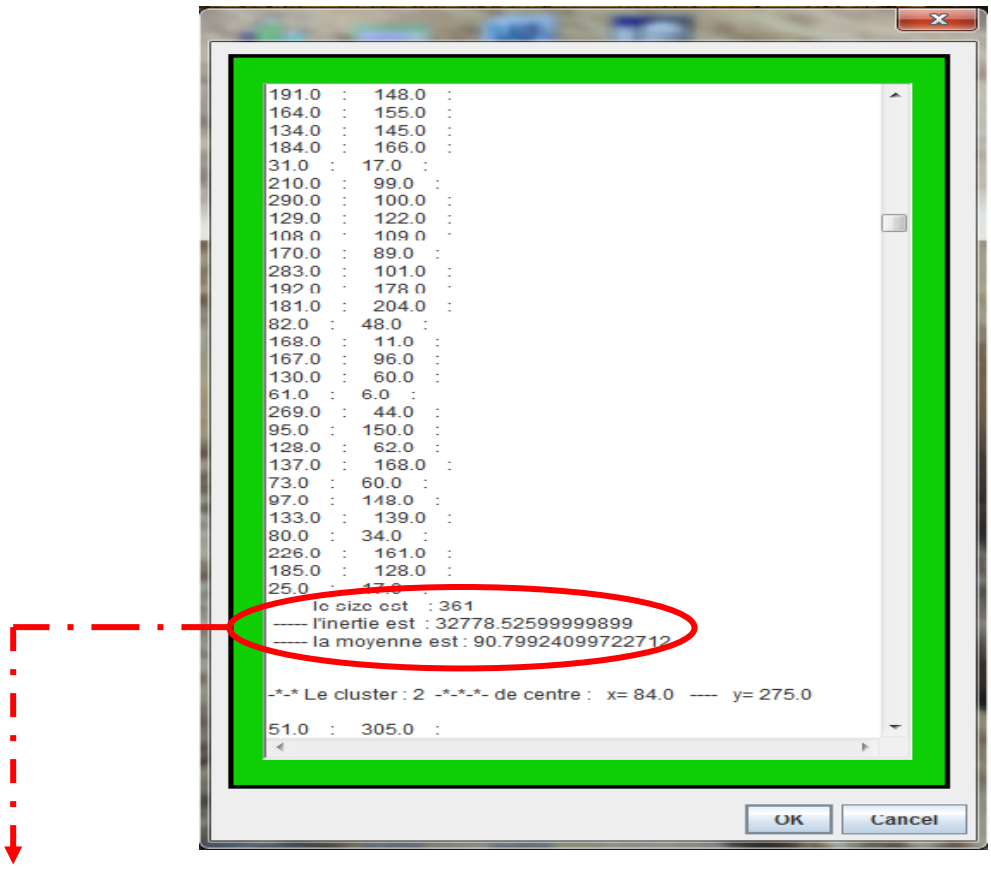


**Figure 3-0-7 : Affichage de résultat de clustering (détailler)**

**Remarque**

Dans notre travaille l'inertie représente la sommes des distances euclidiennes de chaque point (individu) par rapport au centre dans le même cluster, et la moyenne représente la division de cette somme sur la taille de cluster (nbr de points dans un cluster).

Alors le critère d'inertie vaut zéro si chaque document du corpus est présent dans une classe singleton.

## *3.3 Conclusion*

Dans ce chapitre on a présenté une version de l'algorithme de k-means qui permet de regrouper les individus dans des clusters homogènes, Le plus souvent, à l'issue d'une analyse par les *k*-moyennes, nous examinons les inerties de chaque cluster. Plus la moyenne d'inertie du cluster est faible plus la classification est bonne et vice versa.

# Conclusion générale

Dans ce mémoire on a présenté une version de l'algorithme de k-means qui permet de regrouper les individus dans un ensemble des clusters homogènes, On observe que dans la majorité des cas, les k classes trouvées par cette méthode sont de meilleure qualité, Malgré que les algorithme de classification par partitionnement souffrent du problème de représentant unique (en effet ils n'utilisent dans qu'un seul point comme représentant d'une classe ).

Comme perspectives, On peut comparer les performances des autres algorithmes par rapport à k-means qui est une méthode de type *hard clustering*. Cela signifie qu'un point de données peut appartenir à un seul cluster et qu'une probabilité unique est calculée pour l'appartenance de chaque point de données à ce cluster, contrairement à cette approche, l'algorithme d'EM (Expectation Maximization) est une méthode de type soft clustering. Cela signifie qu'un point de données appartient toujours à plusieurs clusters et qu'une probabilité est calculée pour chaque combinaison point de données/cluster. Il utile de noter que l'algorithme k-means est très performant en termes de temps d'exécution, mais il souffre du problème de dépendance des résultats aux choix effectués lors de l'initialisation.

# Référence

[1].Bezdek, 1981 -J. C. Bezdek. Pattern Recognition with Fuzzy Objective Function Algorithms. New York: Plenum Press, 1981.

[2].Boris Mirkin. Clustering for Data Mining: A Data Recovery Approach, Chapman & Hall/CRC, Taylor & Francis Group, Boca Raton, 2005.

[3].Celeux G,.Diday E., Govaert G., Lechevallier Y., Ralam-Bondrainy H. Classification Automatique des Données. Bordas, Paris, 1989.

[4].Chapelle, O., Schölkopf, B., & Zien, A. (eds). 2006. Semi-Supervised Learning. Cambridge, MA : MIT Press.

[5]. Daniel ]T. Larose. Discovering Knowledge in Data: An Introduction to Data Mining, John Wiley & Sons, Inc., Hoboken, New Jersey. 2005

[6].Diday et al., 1982 -E. Diday, J. Lemaire, J. Pouget et F. Testu. Eléments d'analyse des données, Dunod Informatique, 1982.

[7]. Jacob Kogan, Introduction to Clustering Large and High-Dimensional Data, Cambridge University Press, Cambridge, 2007.

[8].J.R.Quinlan. *C4.5 : programs for machine learning* . 1993.

[9]. J.R.Quinlan. *Induction of Decision Trees.* 1985, Machine Learning, Vol. 1, pp. 81-106.

[10].Nédellec et al., 2001-C. Nédellec, M. Ould Abdel Vetah et P. Bessières. Sentence Filtering for Information Extraction in Genomics, a Classification Problem. In Proceedings of the Conference on Practical Knowledge Discovery in Databases, PKDD'2001, p. 326-338, Freiburg, septembre 2001

[11]. MacQueen, J. Some Methods for Classification and Analysis of Multivariate Observations. Dans Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability, pages 281_297, 1967.

[12]. P. Berkshire: *Survey of Clustering Data Mining Techniques*, 2002

[13] Steinbach, Karypis, Kumar: *A comparison of document clustering techniques*, ACM SIGKDD, 6th World Text Mining Conference, 2000

[14]. VAPNIK V., *The Nature of statistical Learning Theory (second ed.)*, Springer, 1995.

[15]. V. Faber : *Clustering and Continuous K-means*, Los Alamos Science, 1994

[16].Zhang, 2000- B. Zhang. Generalized k-harmonic Means – Boosting in Unsupervised Learning. Technical Report HPL-2000-137, Hewlett-Packard Labs, 2000.

[17].ZHANG T., RAMAKRISHNAN R. & LIVNY M. (1997). Birch : A new data clustering algorithm and its applications. *Data Mining and Knowledge Discovery*, **1**(2), p141–182.