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*

# Implémentation d'un algorithme de Clustering à base de k-MEDOIDS

**Réalisé par :**

- ABDELLAH BERREHAIL AMINA
- BOUAFIA NOURIA

*Présenté le 08 Juin 2014 devant la commission d'examination composée de MM.*

- Hadjila F. *(Encadreur)*
- Mouafek B. *(Examinateur)*
- Lahasaini M. *(Examinateur)*
- Berramdan *(Examinateur)*

Année universitaire: 2013-2014

# *Remerciement*

*En préambule à ce mémoire nous remerciant ALLAH qui nous aide et nous donne la patience et le courage durant ces langues années d'étude.*

*Tout d'abord nous tenant à remercier sincèrement notre encadreur Monsieur HADJILA, s'est toujours montrés à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration , pour sa façon à la fois sympathique et compétente, l'aide et le temps qu'il a bien voulu nous consacrer.*

*Nos gratitudes s'adressent également au corps professoral et administratif de la Faculté des Sciences, pour la richesse et la qualité de leur enseignement et qui déploient de grands efforts pour assurer à leurs étudiants une formation actualisée.*

*Nous souhaitant adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire.*

*On n'oublie pas nos parents pour leur contribution, leur soutien et leur patience.*

*Enfin, nous souhaitant adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire.*

**Résumé**

Nous présentons dans ce travail une méthode de **clustering** basée sur les **centroides**.

En particulier nous proposons la conception et l'implémentation de l'algorithme **K-MEDOIDE** Connu aussi sous le nom PAM (Partitioning Around Medoids).

Nous notons que cette approche est plus robuste par rapport à d'autres méthodes en termes d'efficacité, en plus elle moins sensible aux "outliers".

Les résultats obtenus en expérimentation montrent que les **clusters** obtenus ont une meilleure qualité.

✓ **Mots clés :** Clustering, K-medoids, centroide, coût.

**ملخص**

في هذا العمل سنقوم بعرض طريقة **تصنيف** ضمن **مجموعات** تعتمد على **المراكز**.

بصيغة اكتر تدقيقا نقترح إنشاء خوارزمية K-MEDOIDS المعروفة أيضا باسم PAM(**التصنيف** حول **المراكز** ).

كما نشير الى ان هذه الطريقة هي اكثر صلابة و فاعلية بالمقارنة مع الطرق الاخرى بالإضافة إلى أنها أقل حساسية ل"القيم المتطرفة."

ان النتائج المحصل عليها بالتجربة تدل فعلا على ان هده **المصنفات** المحصل عليها ذات جودة عالية .

✓ **الكلمات الرئيسية :**التصنيف,المراكز, المصنف ,مجموعات .

**Abstract**

In this work we present a **clustering** method based on **centroids**. In particular, we propose the design and implementation of the **K-medoid** algorithm also Known as PAM (Partitioning Around Medoids). We note that this approach is more robust compared to other methods in terms of efficiency, in addition to that it is less sensitive to "outliers". The obtained results show that **clusters** have an acceptable quality.

✓ **Keywords:** clustering, K-medoid, centroid, cost.

## Table de matières

# Chapitre I : Introduction générale

## I-1 Contexte

La classification ou le regroupement en classes homogènes consiste à réduire un nuage des points d'un espace quelconque en un ensemble de représentants moins nombreux permettant une représentation simplifiée des données initiales. Ainsi, comme les méthodes d'analyse factorielle.

La classification automatique est un moyen d'étiquetage de données sans l'intervention de l'homme. Il s'agit d'une démarche très courante qui permet de mieux comprendre l'ensemble analysé. Ses applications sont nombreuses, en statistique, traitement d'image, intelligence artificielle, reconnaissance des formes ou encore la compression de données.

L'objectif d'une méthode de classification automatique est la recherche d'une typologie, ou segmentation, c'est-à-dire d'une partition, ou répartition des individus en classes, ou catégories. Ceci est fait en optimisant un critère visant à regrouper les individus dans des classes, chacune le plus homogène possible et, entre elles, les plus distinctes possible. Cet objectif est à distinguer des procédures de discrimination, ou encore de classement (en anglais classification) pour lesquelles une typologie est a priori connue, au moins pour un échantillon d'apprentissage. Ce travaille étudie l'apprentissage non-supervisé, ou en anglais de clustering. Cette méthode a plusieurs possibilités de combinaison avec d'autres méthodes, en pré- ou en post-processing. En effet, elle peut résumer l'information afin de la transmettre à une autre méthode et ainsi mieux analyser les données. Elle peut aussi, suite à un pré-traitement des données, être utilisée pour mieux comprendre la quintessence de l'information contenue dans les corpus.

Le principe de la classification automatique est d'imiter ce mécanisme par une machine. Pour ceci, il faut développer des méthodes qui s'appuient soit sur les données à proprement parlé, sans aucune connaissance autre, soit sur les données et sur un savoir acquis préalablement (automatiquement ou grâce à un expert du domaine).

## I-2 Problématique

On s'intéresse dans notre travail aux méthodes de clustering à base de centroides, et en particulier on cherche des solutions qui tiennent compte de certaines contraintes imposées par les problèmes réels, par exemple le représentant d'un cluster doit appartenir à la base d'exemples, et on ne tolère pas les barycentres comme représentant.

## I-3 Contribution

L'objectif de notre travail est de concevoir et d'implémenter une variante de K-means nommée K MEDOID, qui prenne en compte les exigences citées précédemment.

## I-4 Plan de travail

Notre mémoire est structuré comme suit :

- Chap II : clustering

  Il représente un ensemble de méthodes non supervisées, qui indique pour chacune d'elles, les points forts, et les faiblesses, il montre aussi quelques domaines d'application.

- Chap III : conception et implémentation du prototype

  Ce chapitre représente l'organigramme de l'algorithme de clustering intitulé « K-medoid », il montre aussi l'interface homme machine de notre prototype développé, ainsi que les expériences et les résultats obtenus.

- Conclusion générale : on résume notre travail, en indiquant les avantages de notre algorithme, et on présente aussi les perspectives et les améliorations que l'on peut envisager sur notre prototype.

# Chapitre II : Apprentissage automatique

## II .1 Introduction

De manière générale, nous distinguons deux types d'approches de classification la discrimination (classement) et la classification automatique (clustering), dans ce chapitre nous détaillerons les méthodes du deuxième type « clustering » qui sont des techniques statistiques largement utilisées dans la Fouille de Données. Ce type suit 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 humain 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 certaines caractéristiques sans aucune expertise ou intervention requise. En effet, ce processus requit certains traitements ou combinaison avec d'autres méthodes, en pré- ou en post-processing, surtout pour une grande masse 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.

## II .2 Définitions

1) 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) **[1]**, 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 **[2]**, 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.

2) 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.

On peut voir cette définition clairement graphiquement dans l'exemple suivant :

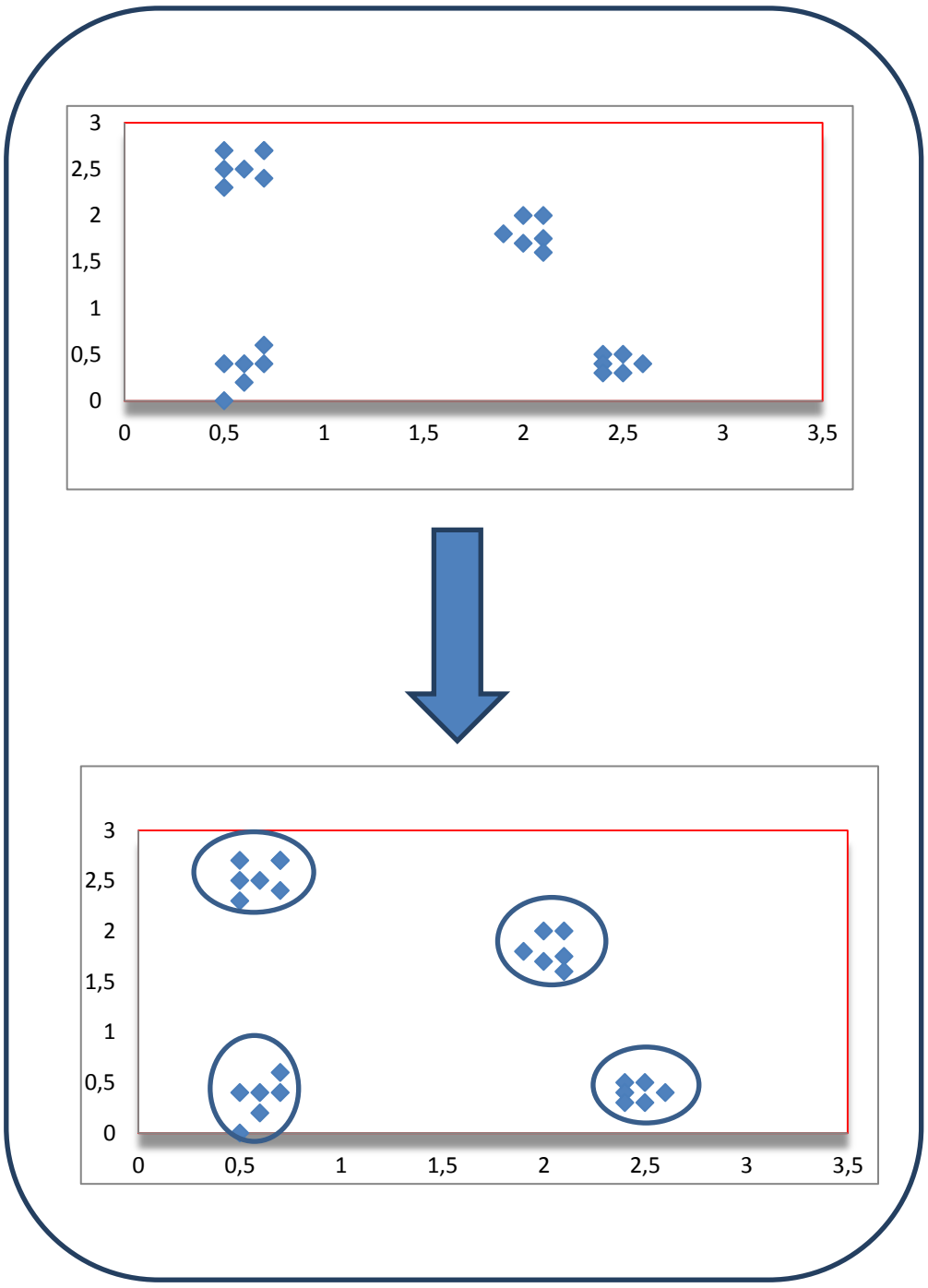



**Figure II.1** : Illustration de regroupement en clusters

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 sûr 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.

## II .3 Domaines d'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.

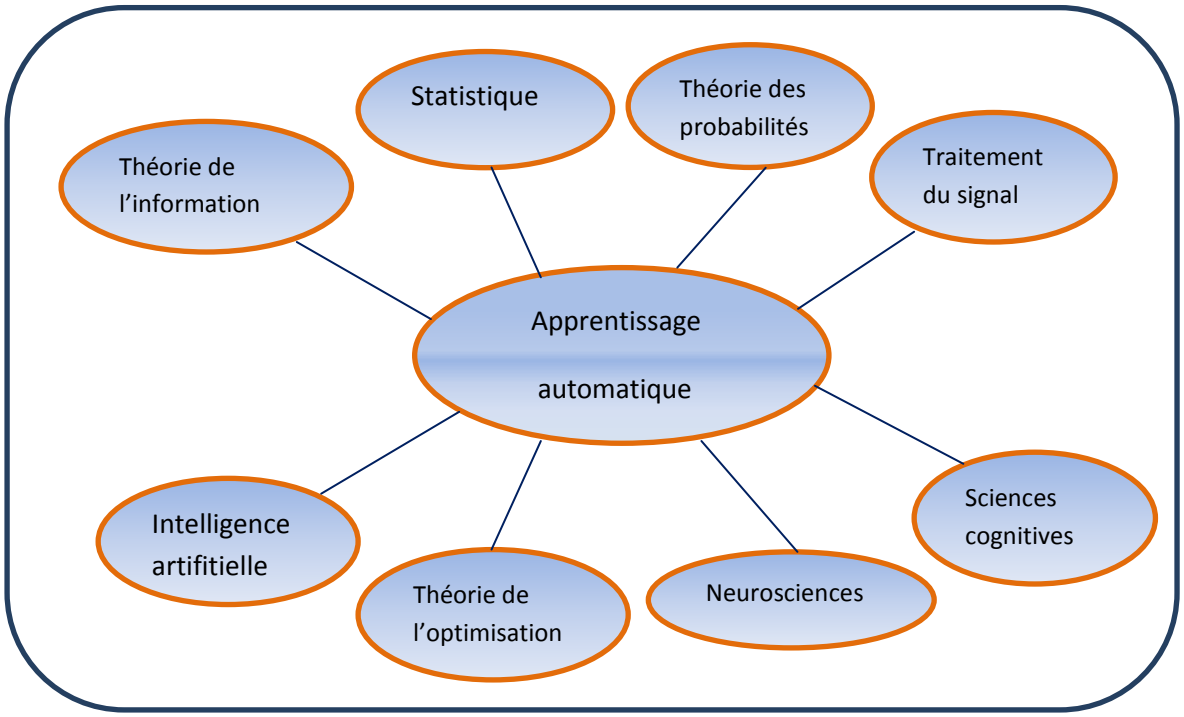



**Figure II.2:** Quelques domaines d'apprentissage automatique

- 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 traitées les dimensionnalités élevées. L'intelligibilité et la convivialité.

## II .4 Les types de Clustering

Il existe plusieurs modèles de classification des approches de clustering, selon le critère d'hiérarchie, on distingue plusieurs classes :

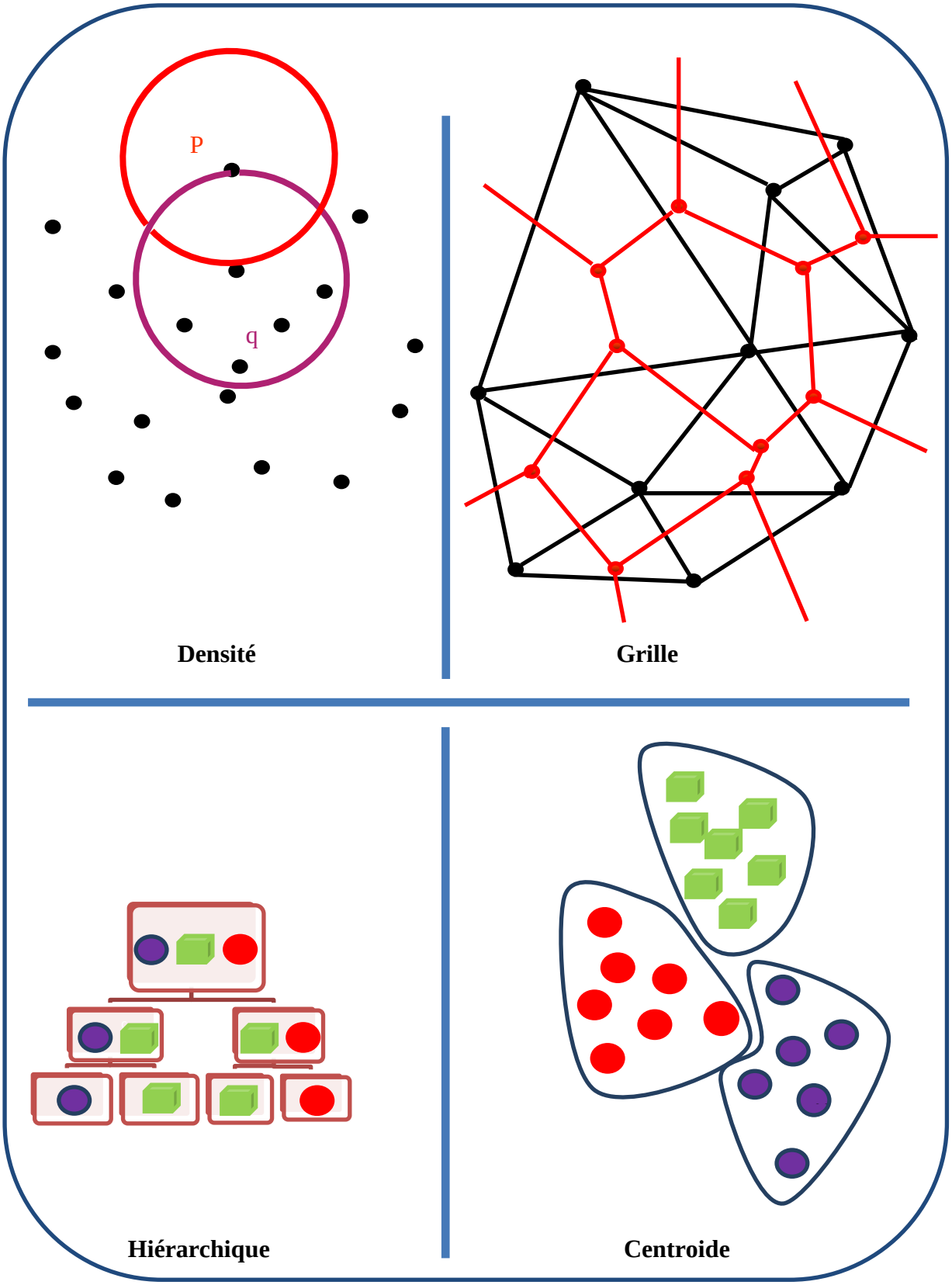



**Figure II.3 :** Quelques types de clustering

Selon le principe de groupement utilisé par l'algorithme on distingue les classes suivantes :

- Méthodes par partitionnement (centroides) :
  - Construire k partitions et les corriger jusqu'à obtenir une similarité satisfaisante.
  - k-means, k-medoids, CLARANS
- Méthodes hiérarchiques :
  - Créer une décomposition hiérarchique par agglomération ou division de groupes similaires ou dissimilaires.
  - AGNES, DIANA, BIRCH, CURE, ROCK, …
- Méthodes par densité :
  - Grouper les objets tant que la densité de voisinage excède une certaine limite
  - DBSCAN, OPTICS, DENCLUE
- Méthodes par grille :
  - Diviser l'espace en cellules formant une grille multi-niveaux et grouper les cellules voisines en termes de distance.
  - STING, WaveCluster, CLIQUE
- Méthodes basées sur les distributions :
  - COBWEB, SOM, EM

## II.5 Les caractéristiques des différentes méthodes

Quel que soit le type de la classification il y a 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 (c'est à dire : que la distance utilisée est prise en considération afin d'améliorer les résultats).

## II .6 Etat de l'art

Dans ce qu'il suit nous présentons quelques algorithmes de Clustering, et en l'occurrence nous étudions

**a**- K-means

**b**- Fuzzy C-means

**c**- Hierarchical clustering

**d**- Mixture of Gaussians (Expectation maximization)

**e-** K-MEDOID (voir chapitreII)

….

**a- K-means**

L'algorithme k-means mis au point par McQueen en 1967**[3]**, un des plus simples algorithmes d'apprentissage non supervisé, appelée algorithme des centres mobiles **[4][5]**, 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 c'est-à-dire chaque cluster est représentée par son centre de gravité.

### ❖ Principe

L'idée principale est de définir les k centroïdes arbitraires c1, c2,..., ck (k le nombre de clusters fixé a priori, chaque ci 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. C'est à dire Chaque classe Si sera représentée par un ensemble d'individus les plus proches de son ci, 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 ci (mj est le centre de gravité de la classe Sj, 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.

### ❖ Discussion

Cette méthode est la plus populaire des méthodes de clustering, malgré cela, un de ses problèmes majeurs est qu'il tend à trouver des classes sphériques de même taille. En plus K-means est connu par sa complexité de « NP-difficile ». Cette approche 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.

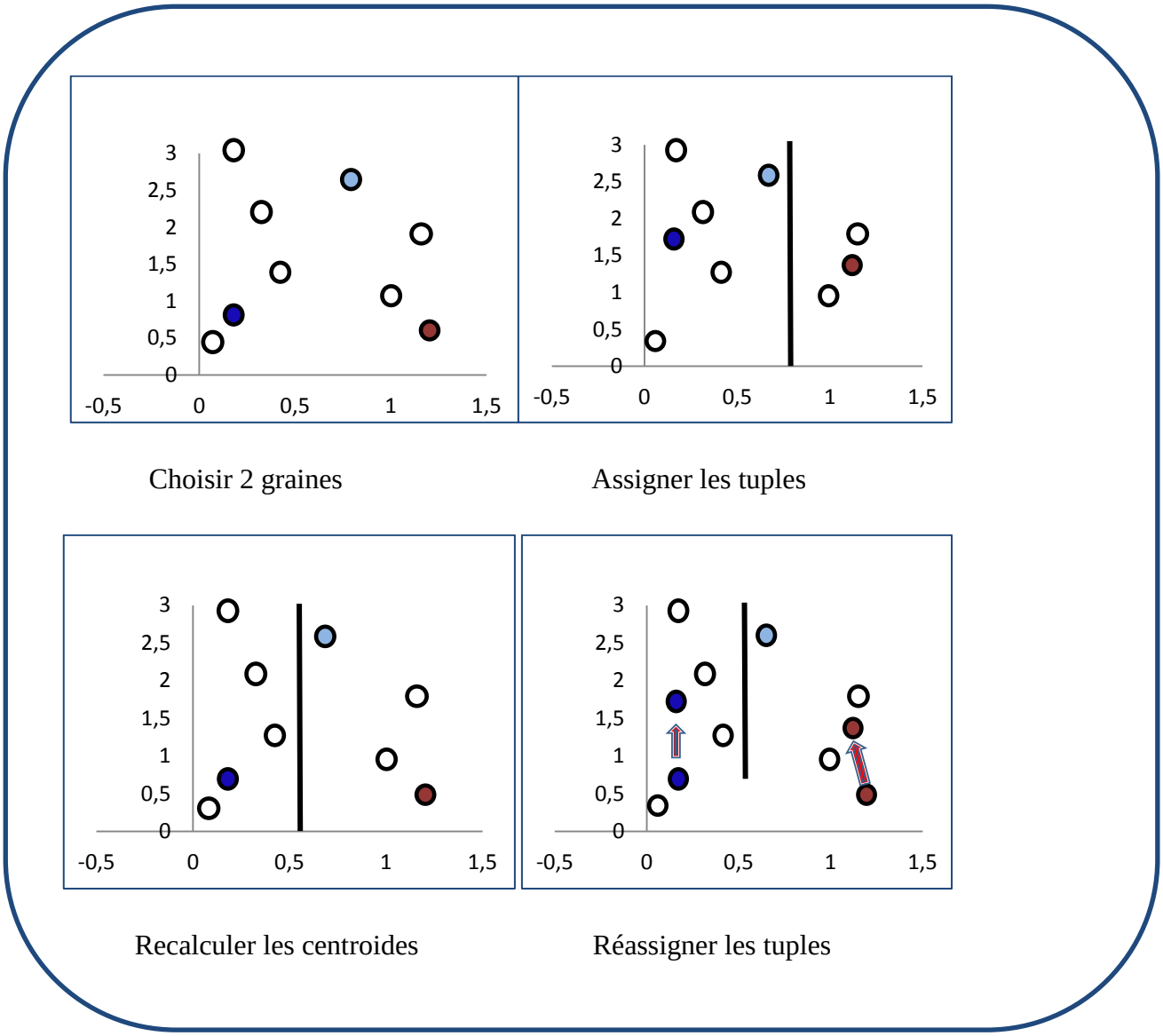



**Figure II.4 :** Exemple de K-Means (k=2)

**b- Fuzzy C-means**

❖ **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 **[6]**, identique à l'algorithme k-means décrit précédemment, elle a été développée par Dunn **[7]** en 1973 et améliorée par Bezdek **[8]** en 1981, est fréquemment utilisée dans la reconnaissance des formes. Il est basé sur la minimisation de la fonction objective suivante :

$$Jm = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} \left\| x_i - c_j \right\|^2 \quad 1 \leq m < \infty$$

Où m est un nombre réel (> 1), Ui j est le degré d'appartenance de xi dans le j ème Cluster, xi est le ième élément des données mesurées, c j est le centre d'un cluster et k*k 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 ui j 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 :

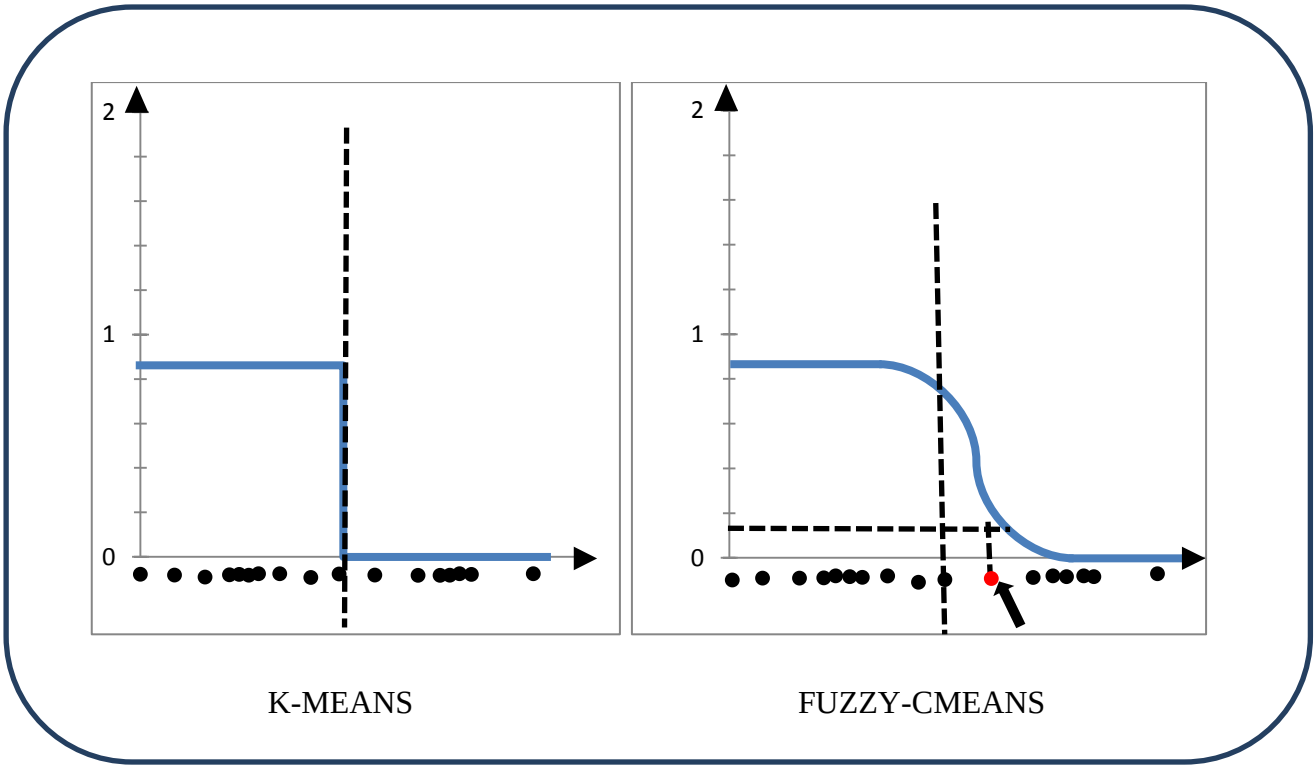



**Figure II.5** : Fonction d'appartenance dans K-MEANS/FUZZY C-MEANS

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 c'est-à-dire que les données sont liées 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.

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

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.

❖ **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 **[9] [10]** afin d'identifier des zones similaires (contours, coins, région homogènes. . .).

### c- Méthodes hiérarchiques

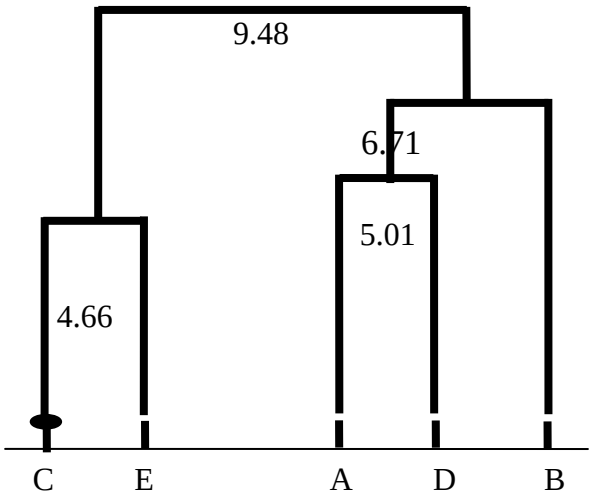
Le processus basique des méthodes hiérarchiques a été donné par Johnson et Lance **[11] [12]**, Ce type de clustering consiste à effectuer une suite de regroupements en Clusters de moins en moins fines en agrégeant à chaque étape les objets (simple élément) ou les groupes d'objets (un Cluster-partition-) les plus proches. Ce qui nous donne une arborescence de clusters **[13]**. Cette approche utilise la mesure de similarité pour refléter l'homogénéité ou l'hétérogénéité des classes.

#### ❖ 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 Hiérarchique).

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




– 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.

❖ **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 y a plusieurs techniques permet de le faire : Saut minimal (single linkage) ; Saut maximal (complète linkage) ; Saut moyen ; Barycentre. . . ,une autre faiblesse est : la complexité de temps d'au moins O(n2), 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 **[14]**, 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 **[15] [16]**.

Améliorer la partition obtenue avec une méthode de deuxième type de clustering (partitionnement) **[17]** .

**d- Expectation-Maximisation(EM)**

Il s'agit d'une méthode itérative qui tente de maximiser la vraisemblance de la probabilité cible en deux étapes. La première expectation consiste en l'évaluation de la valeur moyenne sur les exemples complets. Puis dans l'étape maximisation la valeur manquante est remplacée par la valeur maximisant la vraisemblance (Dempster et al, 1977).

❖ **Principe**

On suppose que un échantillon X1,…Xn suit une distribution.

Exp: normale de paramètre $\Theta = (\mu, \Sigma)$

Pas de données manquantes :

$p(X|\Theta)$ la densité $\longrightarrow L(X1|\Theta)=\log(p(X|\Theta))$

Avec des données manquantes :

Variable modifiée $Z=(X,Y)$ où Y représente l'ensemble des données manquantes

$p(Z|\Theta)= p(Y|X,\Theta)\, p(X|\Theta) \longrightarrow L(X,Y|\Theta)=\log(p(Z|\Theta))$

❖ **Discussion**

EM est intéressant et très puissant (application multiple) mais pas de bonne qualité pour des séries avec une grande variance-covariance.

Alternative : ex: EMBootstrapping.

Pour ce qui est des paramètres du modèle, il est possible d'obtenir de bons résultats avec une bonne précision.

Les caractéristiques globales sont donc bien reproduites par l'algorithme.

Remarque : la censure informative peut être traitée par l'algorithme EM.

## II .6 Mesure de similarité

Pour comparer homogénéité ou la 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 **[18]**, «tout système 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 soit 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,

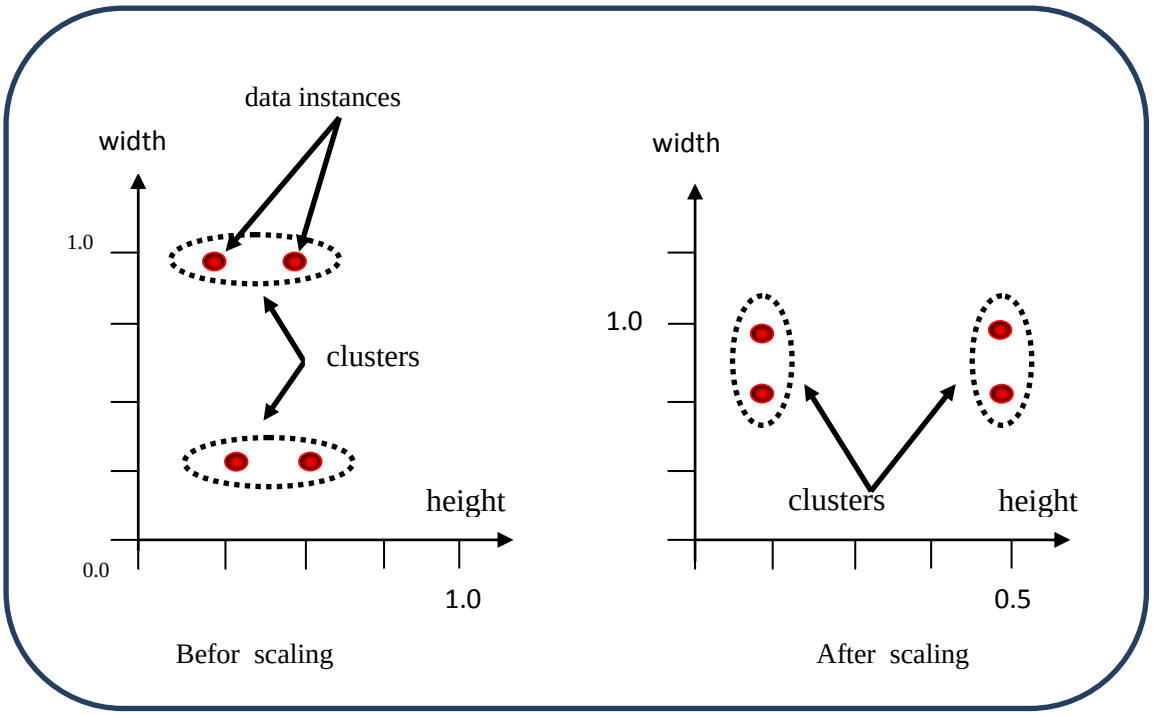



**Figure II.6:** Différents écaillages peuvent conduire à différents clustering

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.

**Remarque :**
Maximiser la similarité = minimiser une distance,
en général on veut sim dans [0,1]
et dist dans [0, +∞] on peut passer de l'un à l'autre
par exemple :
avec sim(x,y) =1/(1+d(x,y)) ou dist(x,y) = -ln(sim(x,y))

## II .7 Quelques distances usuelles

**a- 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.

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

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

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

**c- 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.

$d(x_1, x_2) = (x_1- x_2)\ C^{-1}\ (x_1- x_2)$

(C = covariance)

**d- La distance de Sebestyen** :

$d(x_1, x_2) = (x_1-x_2)\ W\ (x_1 + x_2)$

(W = matrice diagonale de pondération)

**e- 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)

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

Exemple : Considérons les suites binaires suivantes :

a= (0 0 0 1 1 1 1) et b= (1 1 0 1 0 1 1) alors d =1+1+0+0+1+0+0

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

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

$d_p(x_i,x_j) = (\sum_{k=1}^{d} |x_{ik} - x_{jk}|^p)^{1/p}$

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.

❖ **Discussion**

Une note importante est de savoir si le clustering utilise une distance symétrique ou asymétrique. Le 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) ce n'est pas le cas.

En plus, ces mesures rencontrent certaines difficultés lorsqu'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 y a des techniques ont été proposées pour les réductions de dimensions, qui permettent d'appréhender cette difficulté.

## II .8 Les limites de Clustering

Il y a 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 continues (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. i.e. : 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.

## II .9 Conclusion

Dans ce chapitre nous avons présenté brièvement les différentes méthodes (types) d'apprentissage automatique avec quelques exemples en indiquant leurs avantages et faiblaisses.

On peut distinguer deux grandes familles de clustering: les méthodes de partitionnement simple et les méthodes hiérarchiques. Les deux méthodes et leurs performances sont fortement dépendantes de la distance utilisée.

Parmi les méthodes proposées, nous nous intéresserons dans le chapitre suivant par la méthode d'apprentissage automatique K-MEDOID qui se base sur des centroides pour créer des partitions clusters.

# Chapitre III: Implémentation et conception de prototype

## III.1 Introduction

Le choix de l'algorithme de clustering dépend à la fois du type de données disponibles et sur le but et l'application particulière.

Dans de nombreuses approches de clustering, on s'intéresse à l'algorithme K-MEDOIDES(PAM).

Dans cet algorithme une classe est représentée par un de ses individus (medoide). C'est une méthode itérative combinant la réaffectation des individus dans des classes avec une intervention des medoids et des autres individus. C'est une méthode simple parce qu'elle couvre n'importe quel type de variables. Quand des medoids sont choisis, des classes sont définies comme sous-ensembles des individus près des medoids les plus proches par rapport à une mesure de distance fixée à l'avance.

### III.2 Outil utilisé

Dans notre travaille nous avons utilisé la machine TOSHIBA qui a les performances suivantes :

- Processeur : i3
- RAM : 4,00 Go
- Type du système d'exploitation : windows7 64bit

## III.3 Conception

L'algorithme PAM, proposé par Kaufman et Rousseuw [1990] **[19],** est fondé sur le principe suivant :

1. Choisir les medoids initiaux ($M_1$ … $M_K$ ) de manière aléatoire.
2. Evaluer la qualité de cette partition
3. Mise à jours de la partition quasi_optimale (partition quasi optimale = partition initiale et le coût (partition quasi optimale) = coût de la partition initiale).
4. Tant que (it <= itmax))
    a. Echanger $M_1$ et $O_j$ appartient à data base.
    b. Recalculer la partition courante (affecter des objets en centres).
    c. Calculer le coût de la partition courante.
    d. Mise à jour de la partition quasi optimale.

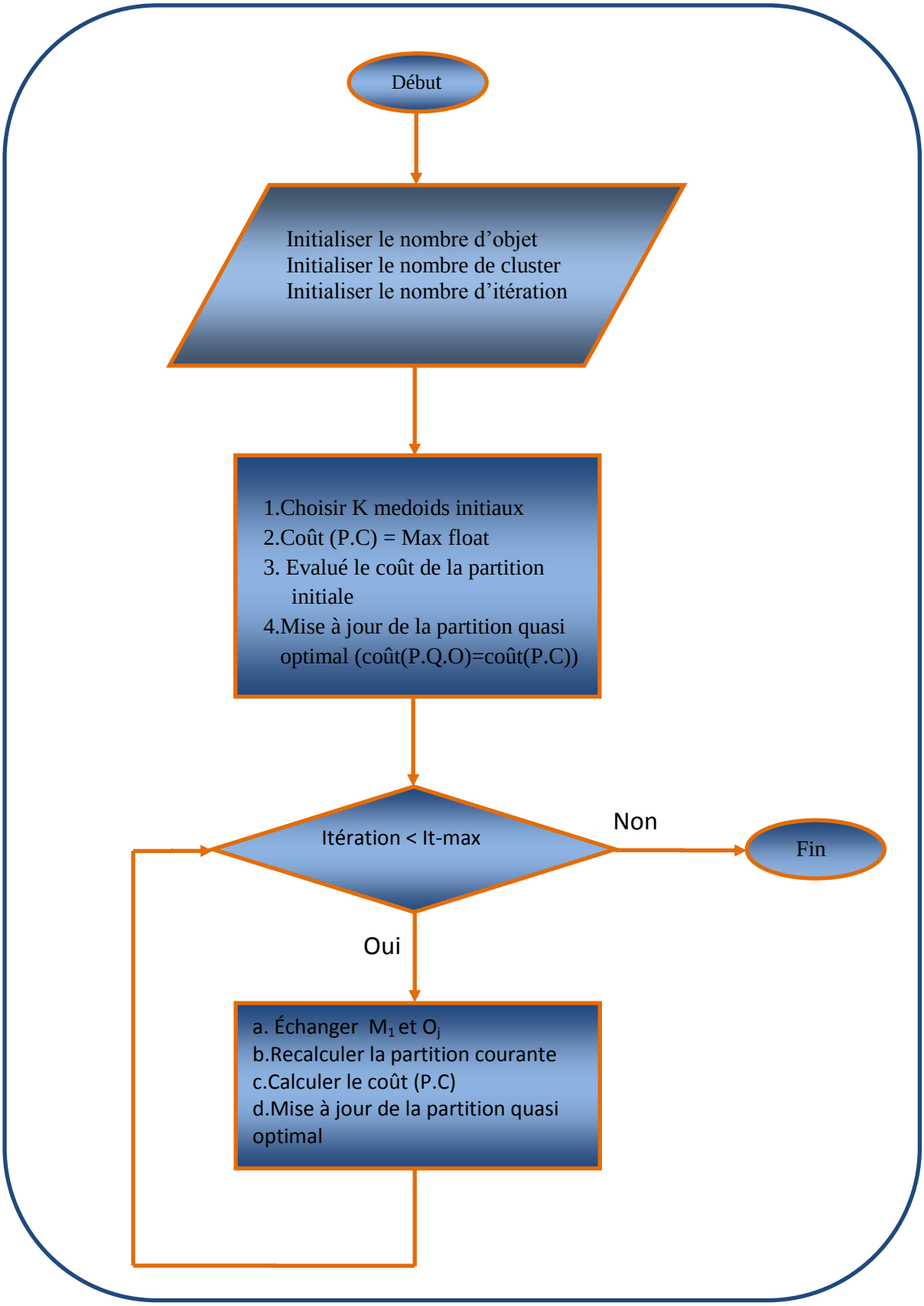



**Figure III.1:** Organigramme de l'algorithme k-MEDOIDS(PAM)

## III.4 Prototype

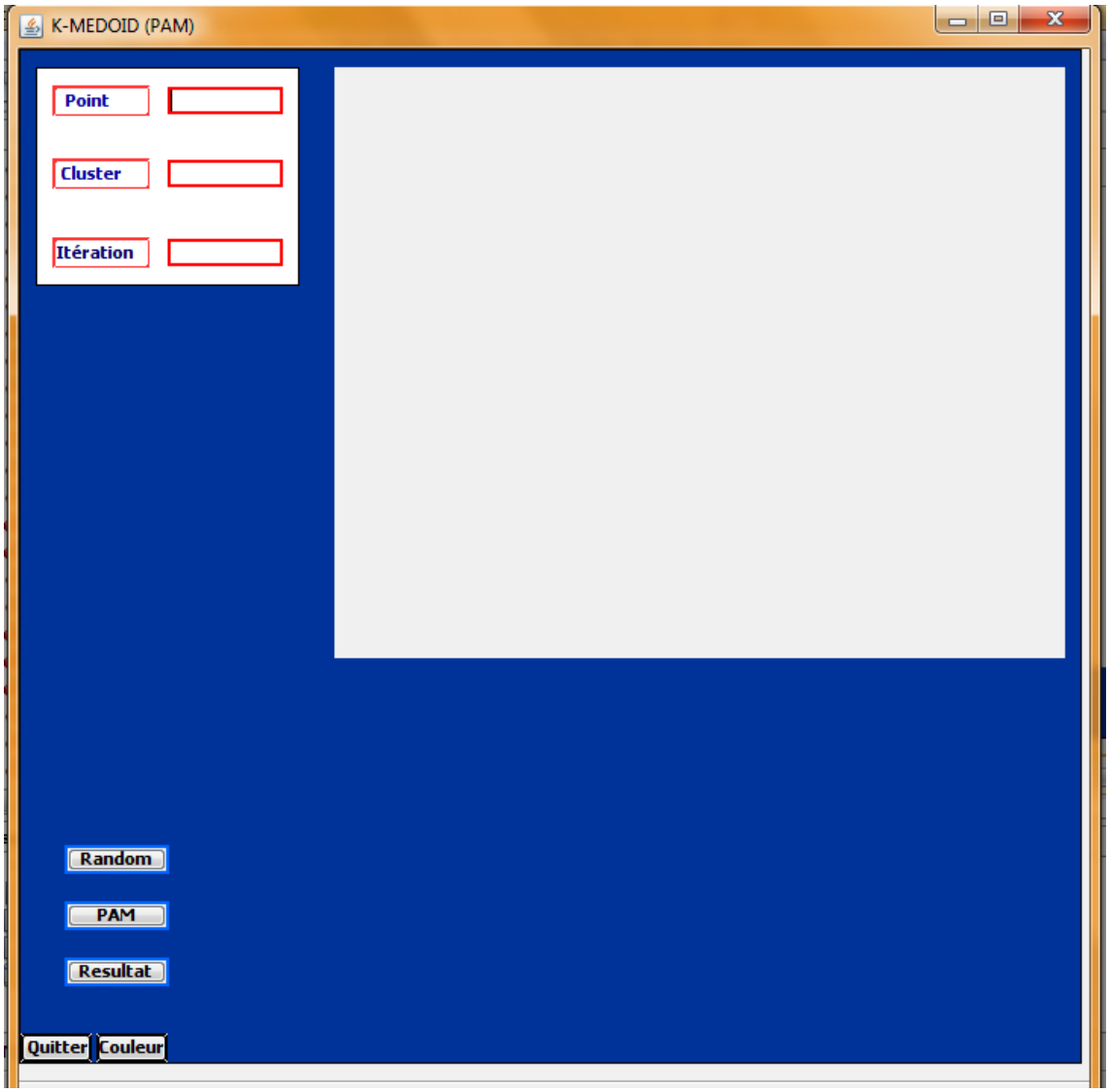



**Figure III.2 : Fenêtre d'interface principale**

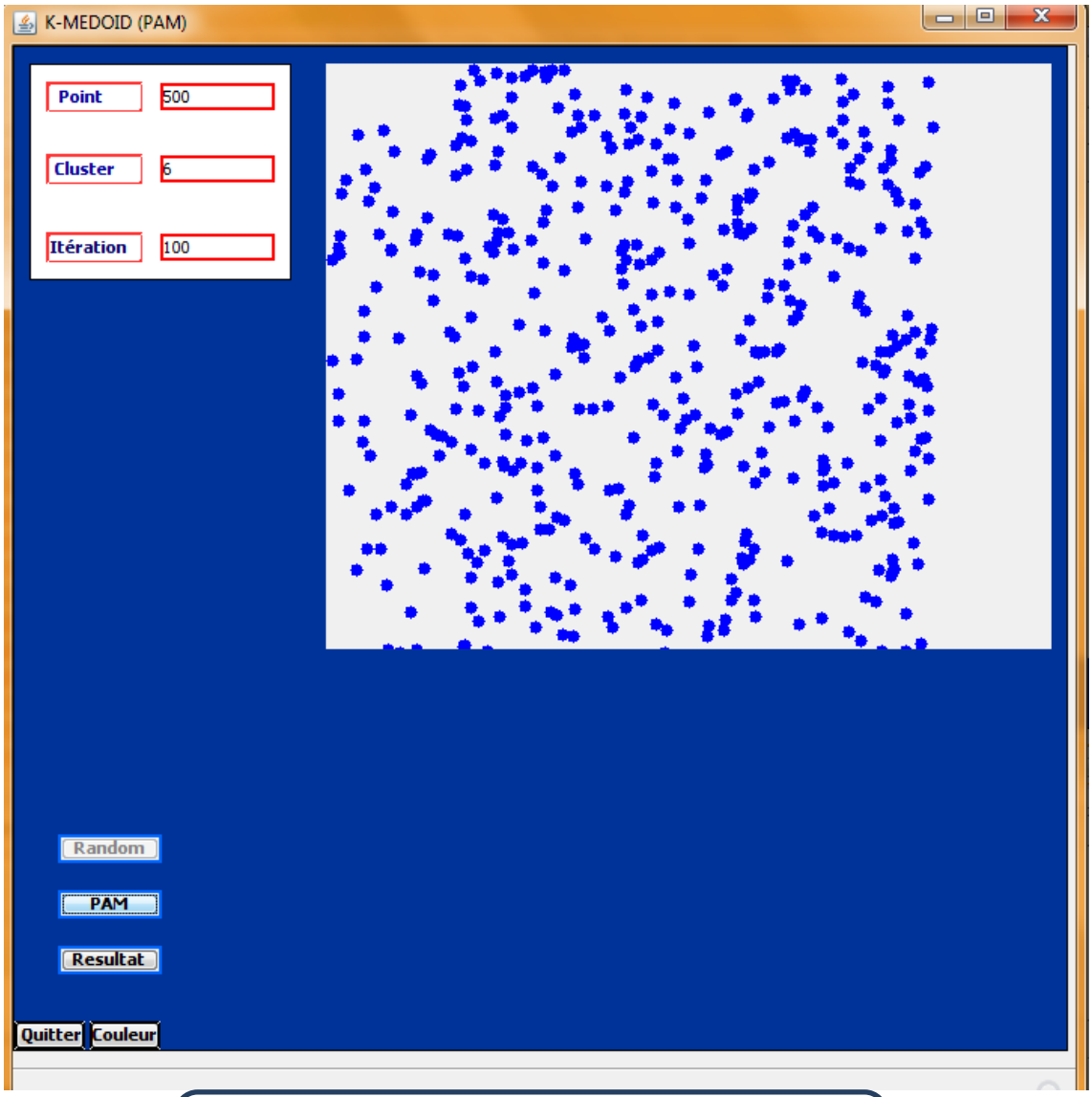



**Figure III.3 : Génération des points aléatoires**

❖ Dans cette figure nous avons exécuté le bouton « Random » pour la génération des points aléatoirement dont nous avons saisi :

* Le nombre de point = 50.

* Le nombre de cluster = 3.

* Le nombre d'itération = 100.

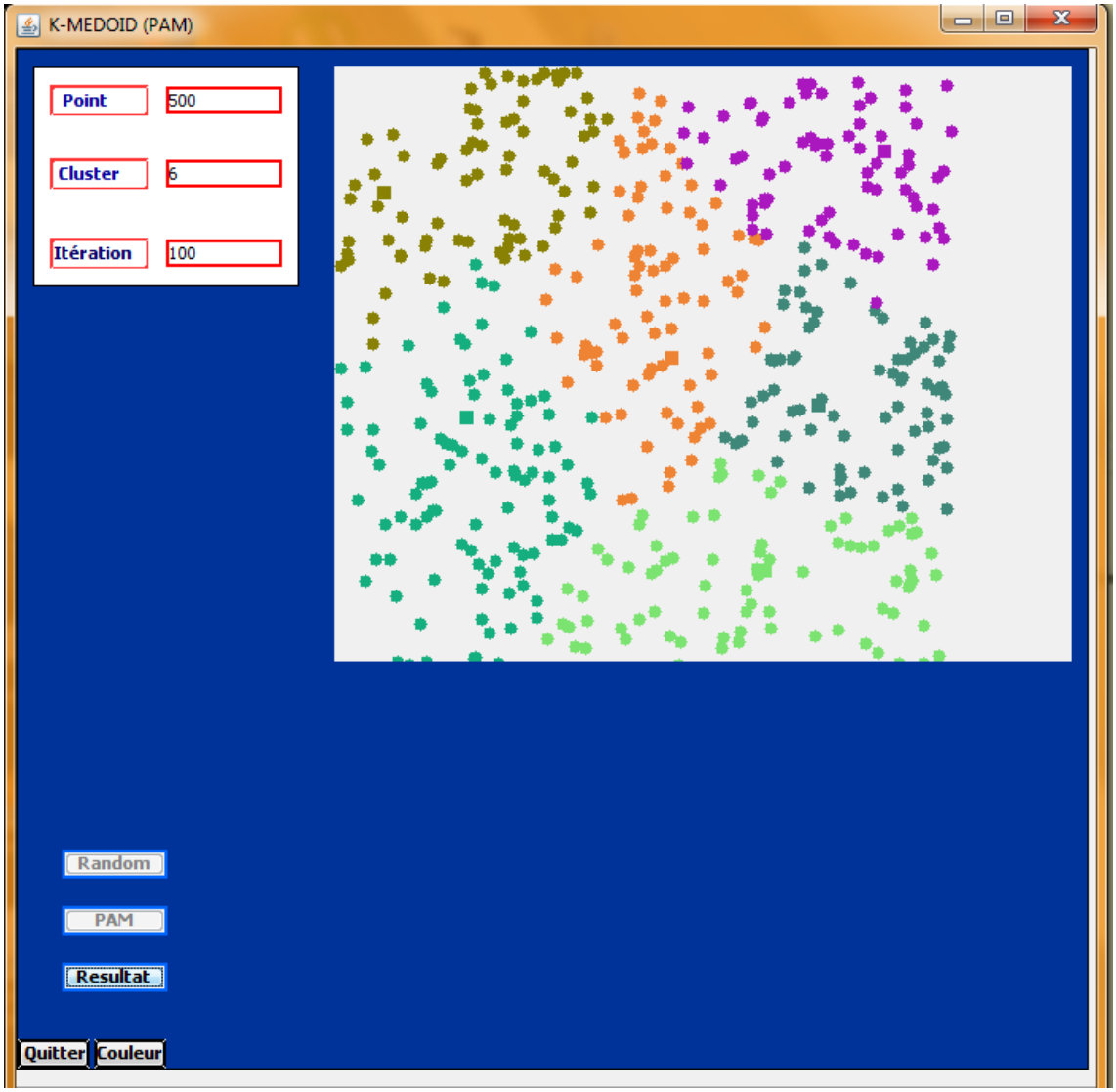



**Figure III.4 : Exécution de l'algorithme PAM**

- Comme 2eme étape, nous avons exécuté le bouton « PAM » pour que les 50 points soit regroupés en 3 partition (Partition quasi-optimale).
  Nous notons que le centre de chaque cluster est représenté en petit carré coloré.

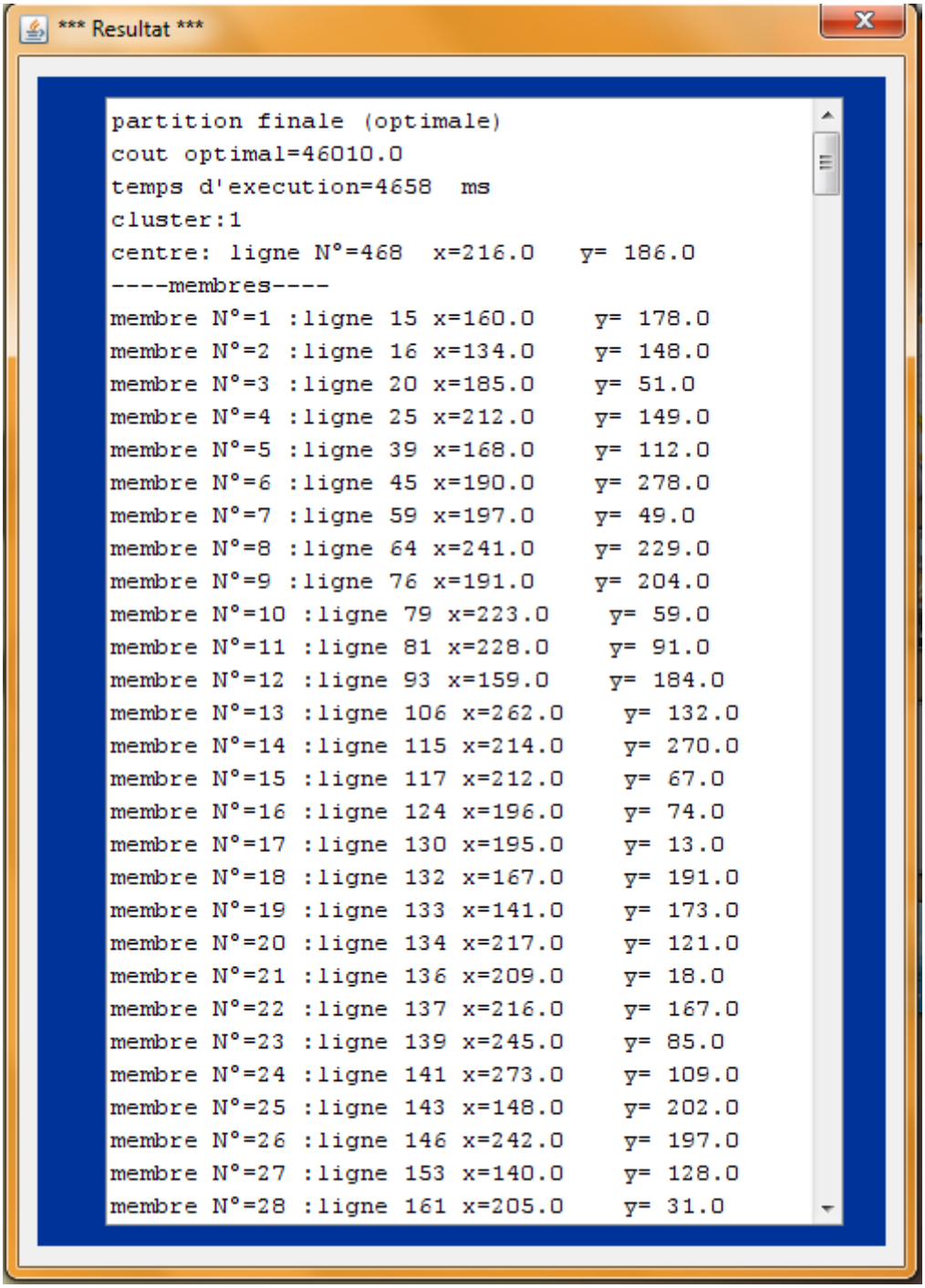



**Figure III.5 : Fenêtre d'affichage du résultat**

Dans cette fenêtre, nous avons affiché les résultats de la partition finale (quasi-optimale) avec leur cout optimal et le temps d'exécution et aussi nous avons calculé l'inertie de chaque cluster.

## II.5 Expérimentation

- Nombre de point : 50 (Base d'exemple fixée)

-Nombre de cluster : 4

| | Nombre d'exemples | Inertie | Temps d'exécution | Cout optimal |
|---|---|---|---|---|
| Cluster 1<br>X=98.0<br>Y=145.0 | 14 | 1381.0 | 108 ms | 5006.0 |
| Cluster 2<br>X=246.0<br>Y=361.0 | 9 | 918.0 | | |
| Cluster 3<br>X=260.0<br>Y=35.0 | 15 | 1615 .0 | | |
| Cluster 4<br>X=46.0<br>Y=264.0 | 12 | 1092.0 | | |

**Table III.1 :** Nombre d'itération=20

| | Nombre d'exemples | Inertie | Temps d'exécution | Cout optimal |
|---|---|---|---|---|
| Cluster 1<br>X=30.0<br>Y=202.0 | 11 | 1380.0 | 394 ms | 5046.0 |
| Cluster 2<br>X=312.0<br>Y=341.0 | 8 | 688.0 | | |
| Cluster 3<br>X=315.0<br>Y=53.0 | 15 | 1402.0 | | |
| Cluster 4<br>X=147.0<br>Y=210.0 | 16 | 1576.0 | | |

**Table III.2 :** Nombre d'itération=100

- Nous remarquons que le temps d'exécution et le coût optimal ont augmentés pour le nombre d'itération=100, par rapport au nombre d'itération=20.

  Notons que l'augmentation du coût optimal du à l'initialisation des centres.

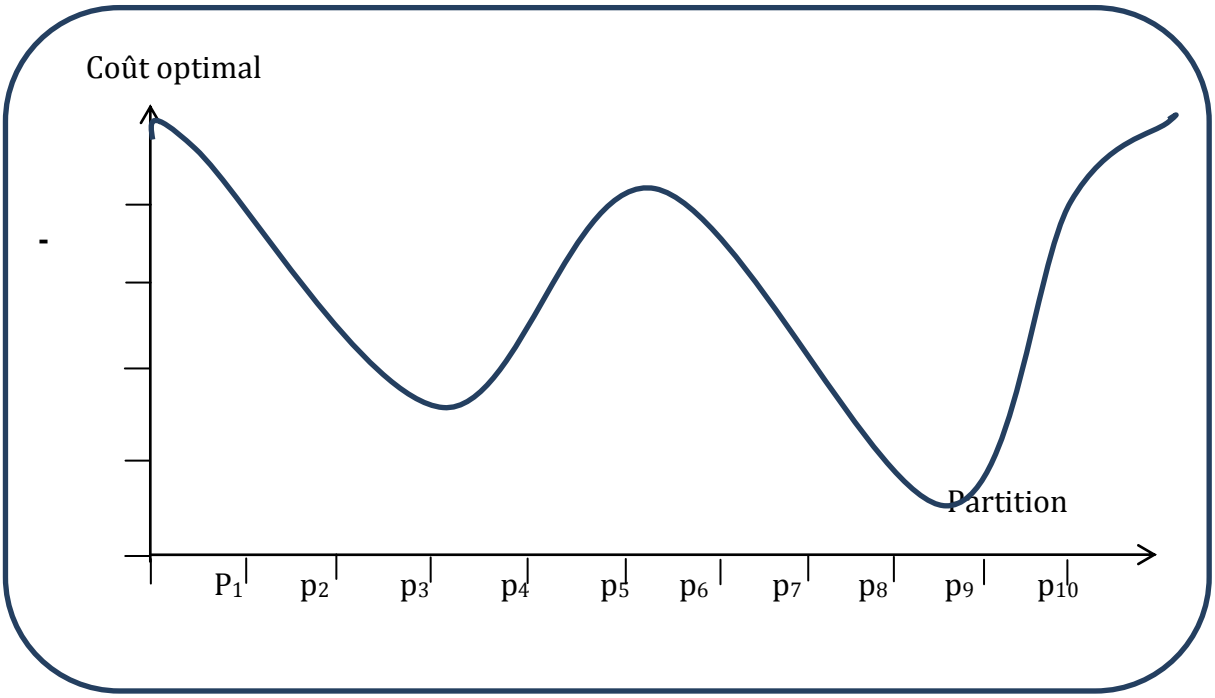



**Figure III.6** : Initialisation des centres vs coût optimal

- Si la partition initiale est $p_1$ l'algorithme k-medoid prend la partition optimale $p_3$ quel que soit le nombre d'itération.

-Nombre d'itération : 20

| | Nombre d'exemples | Inertie | Temps d'exécution | Cout optimal |
|---|---|---|---|---|
| Cluster 1<br>X=56.0<br>Y=350.0 | 5 | 276.0 | 125 ms | 3009.0 |
| Cluster 2<br>X=327.0<br>Y=333.0 | 5 | 219.0 | | |
| Cluster 3<br>X=98.0<br>Y=145.0 | 9 | 683.0 | | |
| Cluster 4<br>X=308.0<br>Y=76.0 | 7 | 414.0 | | |
| Cluster 5<br>X=255.0<br>Y=149.0 | 2 | 54.0 | | |
| Cluster 6<br>X=75.0<br>Y=319.0 | 5 | 209.0 | | |
| Cluster 7<br>X=93.0<br>Y=34.0 | 4 | 283.0 | | |
| Cluster 8<br>X=234.0<br>Y=132.0 | 3 | 252.0 | | |
| Cluster 9<br>X=175.0<br>Y=261.0 | 6 | 423.0 | | |
| Cluster 10<br>X=347.0<br>Y=192.0 | 4 | 196.0 | | |

**Table III.3 :** Nombre de cluster =10

- Par comparaison de la table II.1 avec la table II.3, on remarque que le temps d'exécution a augmenté et le coût optimal a diminué.

## III.6 Conclusion

Nous avons présenté dans ce chapitre la conception et l'implémentation de K-MEDOIDS avec java. Nous notons que K-MEDOIDS est plus robuste en présence de bruit et valeurs aberrantes car un medoid est moins influencé par les points aberrants ou autres valeurs extrêmes.

Selon l'expérimentation il est clair que K-MEDOIDS fonctionnes efficacement pour les petits ensembles de données mais son temps d'exécution sera très prohibitif pour les grandes bases d'exemples.

# Conclusion générale

Dans ce mémoire on a présenté une version de l'algorithme de k-medoid qui permet de regrouper les individus dans un ensemble des clusters homogènes, Il est plus robuste que les autres méthodes en présence de bruit.

Mais, on note aussi que cet algorithme a une complexité de l'ordre de ($O[K*(n-K)^2*i]$) **[20]** . Sachant que K est le nombre maximum de classe, n est le nombre d'instances dans l'ensemble de données et i est le nombre d'itérations .Ce ci devient plus coûteux en cas de K et n assez grand. Par conséquent nous pouvons dire que cet algorithme est intéressant et efficace pour des bases ayant une petite taille.

Comme perspectives, On peut comparer les performances des autres algorithmes avec k-medoid 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 Maximisation) ou FuzzyCMeans, 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.

# Références Bibliographiques

**[1]** Lebart L., Morineau A. & piron M.statistique exploratoire Multidimensionnelle. Dunod, 3ème édition, paris, 2000.

**[2]** 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.

**[3]** 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.

**[4]** Benzécri J.P. L'analyse des données. Dunod, Paris, 197.

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

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

**[7]** 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.

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

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

**[10]** 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

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

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

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

**[14]** 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.

**[15]** 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.

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

**[17]** 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.

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

**[19]** Jean Pierre Nakach.Josiane Confais Approche pragmatique de la classification, livre.

**[20]** Nicolas Creff, Clustering à l'aide d'une représentation supervisée, Rapport de Stage .Promotion 2011 Majeure SCIA, Du 1er février au 31 juillet 2011.

## Liste de figures



## Liste des tableaux