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

Université Mira Abderrahmane de Béjaïa
Faculté des Sciences Exactes
Département de Recherche Opérationnelle

Mémoire de fin de cycle

Pour l'Obtention du Diplôme de Master en Recherche Opérationnelle
Option : Modélisation Mathématiques et Techniques de Décision

L'analyse de clustering dans une dynamique de jeu non coopératif

Présenté Par :
-Mr Abdelkrim REZAG

Soutenu publiquement à l'Université de Béjaïa, le 04/07/2019, devant le jury composé de :

Mr N. KHIMOUM	M.C. Classe B	Président	à l'UAM - Béjaïa.
Mme S. KHERBACHI	M.C. Classe B	Encadreur	à l'UAM - Béjaïa.
Mr S. ZIANI	M.C. Classe B	Examinateur	à l'UAM - Béjaïa.
Mr K. MEZIANI	Doctorant	Examinateur	à l'UAM - Béjaïa.

Année Universitaire 2018 – 2019

Louange A Dieu, le miséricordieux, sans Lui rien de tout cela n'aurait pu être.

Je tiens tout d'abord à remercier **M^me^ S. KHERBACHI** pour l'honneur qu'elle ma fait en acceptant de m'encadrer. Ses conseils précieux ont permis une bonne orientation dans la réalisation de ce modeste travail.

Je teins également à remercier **M^r^ N. KHIMOUM** d'avoir accepté de présider le jurry de ce mémoire.

Je remercie **M^r^ S. ZIANI** et **M^r^ K. MEZIANI** d'avoir accepté de faire partie du jurry et consacré leur temps à la lecture et à la correction de ce mémoire.

Mes remerciements les plus vifs vont tout particulièrement à mes parents.

Ces remerciements ne seraient pas complets sans y avoir associé **M^r^ I. KHALFAOUI** et **M^r^ T. ZIANI**. Je les remercie, pour ses soutiens morals, ses présences et ses écoutes.

Finalement, je réserve une pensée toute particulière et profonde à mes parents et mes soeurs et leur exprime ici toute ma gratitude pour leur soutien sans limite.

De manière, plus particulière, mes pensées les plus intimes vont vers ma mère (la voix de la sagesse et de l'amour) a qui je dois beaucoup, malgré, je sais que je pourrais jamais te rendre ce que tu m'as donnée.

Enfin, merci à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Je dédie ce modeste travail en principe aux personnes les plus proches de ma vie, mes parents qui ont été présents pour moi pendant tout mon cursus et ma vie, mais surtout qui ont toujours su trouver les mots qui m'encourageaient et qui me poussaient à aller de l'avant.

Je dédie aussi ce travail à toute ma famille qui était toujours derrière moi pour me fortifier pendant mes moments difficiles, en particulier :

⋆ *Mes sœurs :* ***ROMAÏSSA, SARAH, NADA.***

⋆ *Mes petits :* ***LIMA, MOHAMED.***

Je veux aussi dédier ce travail à mes chers amis qui ne m'ont jamais oubliés et qui étaient présents en cas de besoin :

Ahlil, Akram, Ali, Amar, Brahim, Midou, Salim.

Tabch, Lol friends, tous mes proches amis.

Djamel, Fateh, Fodil

ABDELKRIM

Table des matières

Table des figures

Liste des Algorithmes

Liste des tableaux

Introduction Générale

L'analyse de cluster ou simplement le clustering est un processus de partitionnement d'un ensemble de données en sous-ensembles appelés clusters. Ces clusters sont caractérisés idéalement par une forte similarité à l'interieur et une forte dissimilarité entre les membres de différents clusters [**Kaufman and Rousseeuw, 1990**].

Le clustering a été largement utilisé dans de nombreux domaines, tels que le marketing, la segmentation d'images et la biologie.

Plusieurs familles de méthodes de clustering ont vu le jour et parmi ces méthodes, les méthodes basées sur k-means. 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.

Cependant, la principale limite de cette méthode est la dépendance des résultats des centroïdes iniales. 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 de ce problème consiste à lancer l'algorithme plusieurs fois avec différentes initialisations et retenir le meilleur clustering. L'usage de cette solution reste limité et que nous pouvons trouver une meilleure partition en une seule exécution.

Notre objectif est de proposer une technique basée sur la classification non supervisée. Pour cela, et afin de trouver un modèle, nous utiliserons la théorie des jeux. Nous cherchons à trouver la meilleure distribution des données dans les clusters.

Notre thème de recherche vise à apporter des éclaircissements théoriques et pratiques sur cette problématique :

Comment déterminer une analyse de clustering dans une dynamique de jeu ?

Pour y répondre, nous avons formulé les questions suivantes :

– Comment détérminer le nombre de clusters ?
– Comment est définie l'utilité d'une donnée ?

Des réponse sont formulées à trouver les hypothèses suivantes :

Hypothèse 1 : L'existence d'une méthode qui calcule le nombre de clusters.

Hypothèse 2 : Est de définir l'utilité d'une donnée selon ses attributs.

Le plan du mémoire s'articule autour de trois chapitres :

- Dans le premier chapitre, nous présenterons les concepts de base du clustering.
- Le deuxième chapitre, nous donnerons un aperçu sur K-means et des notons de base de la théorie des jeux .

- Dans le dernier chapitre, nous présenterons notre approche, ainsi que les expérimentations menées dans ce cadre, pour comparer les résultats obtenu.
- Enfin, ce mémoire se termine par une conclusion générale.

1

Généralités sur le clustering

Contents

Introduction

Le clustering est une tâche dont l'objectif est de trouver des clusters au sein d'un ensemble de données. Ces données sont décrites par des attributs qui définissent leurs propriétés. Les clusters recherchés forment des groupes homogènes de données partageant des caractéristiques communes. Il existe de nombreuses méthodes de clustering permettant de créer ces groupements de manière automatique, chacune utilisant une stratégie et ayant un objectif propre pour les construire.

Pour réaliser cette opération de clustering, nous faisons fréquemment appel à la notion de similarité (ou de distance) entre les données. La notion de similarité (ou de distance) occupe une classe importante en clustering car elle permet d'évaluer à quel point deux données sont similaires ou dissimilaires pour les regrouper ou les séparer.

L'objectif de ce chapitre est d'introduire les concepts de base du clustering qui seront utilisés dans ce mémoire. Nous allons décrire les différents types de clusters (section 2) et de clustering (section 3) et les méthodes de clustering (section 4). Nous analyserons ensuite la mesure de distance et de similarité (section 5) et nous allons terminer, décrire la procédure d'évaluation du clustering.

1.1 Définition et domaine d'application du clustering

La classification est une discipline reliée de près ou de loin a plusieurs domaines. Elle est connue aussi par plusieurs appellations (classification, clustering, segmentation,etc.) selon les données qu'elle traite et les objectifs qu'elle vise.

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

En mathématique, elle est appelée la catégorisation algorithmique de données. Elle consiste à attribuer une classe ou une catégorie à chaque donnée (ou individu) à classer, en se basant sur des données statistiques. Elle fait couramment appel aux méthodes d'apprentissage et est largement utilisée pour définir les formes existances.

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

S'il y d'affecter dans un classement, nous avons un ensemble de données qui sont unis sauf forme de groupes préétablis, le but de l'analyse discriminante est tout simplement de fixer des règles pour déterminer la classe des données. Autrement dit, la classification consiste à rechercher des classes "naturelles" dans le domaine étudié, c'est "Cluster Analysis" en anglais.

Nous pouvons résumer tous les termes déjà évoqués comme suit :

Français	Anglais
Classification	Clustering
Classement	Classification

Table 1.1 – Noms attributés à la classification en Fr/Eng

D'une manière générale, à partir de ces définitions, nous déduisissions que la classification se définit comme une méthode mathématique d'analyse de données, pour faciliter l'étude d'une population d'effectifs importants, généralement des bases d'observations caractérisant un domaine particulier (animaux, plantes, malades, gènes, etc.), où nous les regroupons en plusieurs classes.

Le clustering possède des domaines d'application extrêmement variés, parmi lesquels :

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

L'environnement : Identification des zones terrestres similaires dans une base de données contenant des informations (en termes d'utilisation) de la terre.

Les assurances : Identification de groupes d'assurés distincts associés à un nombre important de déclarations.

La planification des villes : Identification de groupes d'habitations suivant leurs types, valeur, localisation géographique.

La médecine : Localisation de tumeurs dans le corps humain. Par exemple, dans un nuage de points fournis par le scanner du cerveau, les points définissant une tumeur peuvent être définis.

La segmentation d'images : Détection des zones homogènes dans une image.

Web log annalysis : Identification de profils d'utilisateurs à travers leur de clies (Clickstream).

Text mining : Classification des textes selon leur similitude dans des dossiers automatiques.

Dans tous les chapitres, nous utiliserons les notations suivantes :

Soit $O = \{o_1, \cdots, o_n\}$ un ensemble de n données, où chaque donnée o_i est décrite par un ensemble d'attributs d formant ainsi un vecteur à d-dimensions, $O_i = (o_{i1}, \cdots, o_{id}) \in \mathbb{R}^d$. Soit $\mathcal{C} = \{C_1, \cdots, C_K\}$ un résultat de clustering comportant K clusters.

1.2 Différentes approches en clustering

Le résultat d'un algorithme de clustering peut se présenter sous différentes formes selon qu'il soit possible ou non que deux clusters se chevauchent, c'est-à-dire qu'une donnée puisse appartenir ou non à plusieurs clusters en même temps [**Forestier, 2010**].

Clustering dur : Dans un clustering dur, chaque donnée o_i ; $i = \overline{1, n}$ appartient à un seul cluster C_k ; $k = \overline{1, K}$:

$$O = \bigcup_{k=1}^{K} C_k \ ; \ C_k \cap C_l = \varnothing \ ; \ k \neq l \ ; \ \forall k, l \in \{1, \cdots, K\}. \tag{1.1}$$

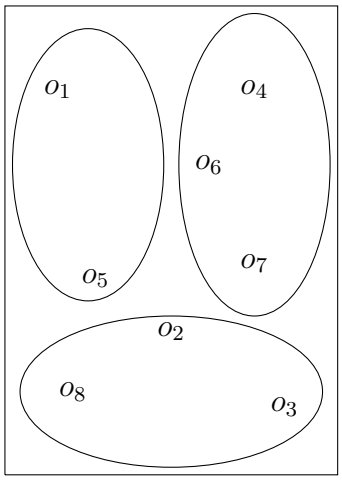

FIGURE 1.1 – Exemple de clustering dur

Clustering dur partiel : Dans un clustering dur partiel, chaque donnée o_i ; $i = \overline{1,n}$ appartient à un seul cluster comme il peut n'y appartenir à aucun :

$$O \leq \bigcup_{k=1}^{K} C_k \ ; \ C_k \cap C_l = \varnothing \ ; \ k \neq l \ ; \ \forall k, l \in \{1, \cdots, K\}. \tag{1.2}$$

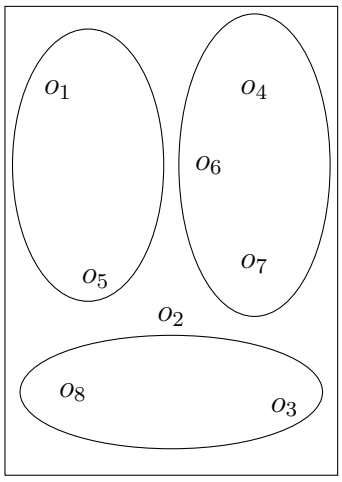

FIGURE 1.2 – Exemple de clustering dur partiel

Clustering doux : Dans un clustering doux, chaque donnée o_i ; $i = \overline{1,n}$ appartient à un ou plusieurs clusters :

$$O = \bigcup_{k=1}^{K} C_k \ ; \ \exists l \in \{1, \cdots, K\} \ ; \ C_k \cap C_l \neq \varnothing \ ; \ k \neq l \ ; \ \forall k \in \{1, \cdots, K\}. \tag{1.3}$$

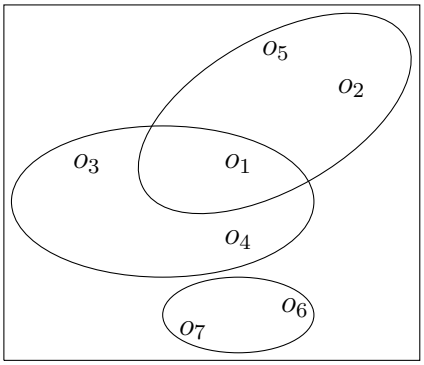

FIGURE 1.3 – Exemple de clustering doux

Clustering doux partiel : Dans un clustering doux partiel, chaque donnée o_i ; $i = \overline{1,n}$ n'appartient à aucun cluster, mais dans certain cas un ou plusieurs clusters peuvent le renfermer :

$$O \leq \bigcup_{k=1}^{K} C_k \ ; \ \exists l \in \{1, \cdots, K\} \ ; \ C_k \cap C_l \neq \varnothing \ ; \ k \neq l \ ; \ \forall k \in \{1, \cdots, K\}. \tag{1.4}$$

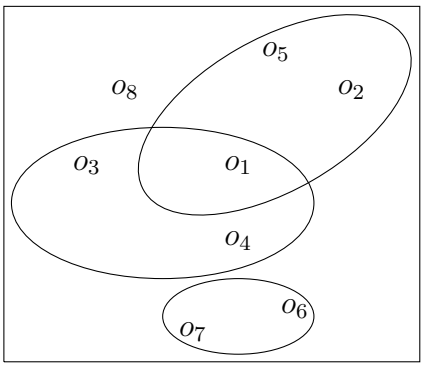

FIGURE 1.4 – Exemple de clustering doux partiel

1.3 Type de clustering

Parmi tous les types de clustering, nous distinguons principalement :

1.3.1 Classification supervisée

Dans laquelle un expert a fourni le modèle exacte des classes à obtenir. Donc, le classifieur est entrainé à l'aide d'un ensemble de données connues à priori, cet entraînement à pour but d'adapter les sorties du classifieur en fonction des entrées qui lui sont soumises. Ainsi, les distances permettent d'évaluer la similarité entre les données. D'ailleurs, les termes similarité et dissimilarité sont équivalents, respectivement, à ressemblance et dissemblance.

1.3.2 Classification non-supervisée

Pour laquelle le classifieur doit se débrouiller seul pour classer les données sans aide extérieure. Cette classification regroupe des éléments ayant les mêmes propriétés statistiques, géométriques, etc. Elles utilisent un critère de regroupement qui peut être basé sur des distances entre les données ou sur des appartenances floues. Dans ce type de classification, le nombre de classes, inconnu à priori, est déduit directement des données.

1.3.3 Classification semi-supervisée

La semi-supervision intervient lorsqu'on dispose à la fois d'un ensemble de données étiquetées et non étiquetées. Généralement, les données non-supervisées sont disponibles en grand nombre car peu coûteuses à produire. Au contraire, les données supervisées qui nécessitent l'expertise humaine sont plus rares mais également plus riches en informations.

1.4 Méthodes de clustering

Il est difficile de fournir une catégorisation précise des méthodes de clustering, car ces catégories peuvent se chevaucher de sorte qu'une méthode peut avoir des caractéristiques de plusieurs catégories. Néanmoins, nous présentons les principales familles de méthodes de clustering [**Kaufman and Rousseeuw, 1990**]. Les méthodes peuvent être classées dans les catégories suivantes :

- Les méthodes de partitionnement.
- Les méthodes hiérarchiques.
- Les méthodes basées sur une dansité.
- Les méthodes basées sur une grille.

1.4.1 Méthodes de partitionnement

Une méthode de partitionnement construit $K \leq n$ partitions de données où chaque partition représente un cluster. Autrement dit, il divise les données en K clusters de sorte que chaque cluster doit contenir au moins une donnée. Les méthodes de partitionnement de base adaptent généralement un clustering dur.

La plupart des méthodes de partitionnement sont basées sur la similarité (ou la distance), les méthodes K-means et Fuzzy C-means en sont les exemlpes connues de cette famille de méthodes. Une méthode de partitionnement crée un partitionnement initial. Il utilise ensuite une téchnique de relocalisation itérative qui tente d'améliorer le partitionnement en déplaçant des données d'un cluster à un autre. Le critère général d'un bon partitionnement est de sorte que les données d'un même cluster soient proches, alors les données de différents clusters soient éloignés. Il existe différents types de critères permettant de juger la qualité des partitions.

1.4.2 Méthodes hiérarchiques

Dans un clustering hiérarchique, un cluster peut être divisé en sous clusters, l'ensemble des clusters étant généralement représenté par un arbre. Une donnée appartient à une et une seule feuille dans la hiérarchie, mais également à son nœud père, et ainsi de suite jusqu'à la racine. Les méthodes hiérarchiques permettent d'obtenir ce type de résultats. Il existe deux types d'approches de clustering hiérarchique : Les approches par agglomération (ascendantes) et les approches par division (descendantes).

1.4.2.1 Approches par agglomération

Cette approche commence par des clusters formés d'une seule donnée puis les fusionne successivement jusqu'à ce que le critère d'arrêt soit atteint [**Day and Edelsbrunner, 1984**]. Nous pourrons citer l'algorithme Chameleon [**George et al., 1999**].

1.4.2.2 Approches par division

Cette approche commence par un cluster formé de toutes les données, qui sera ensuite divisé en petits clusters jusqu'à atteindre une condition d'arrêt donnée par l'utilisateur.

1.4.3 Méthodes basées sur la densité

La plupart des méthodes de partitionnement regroupent des données en fonction de la distance entre eux. De telles méthodes ne peuvent détecter que des clusters de forme sphérique et rencontrent des difficuttés pour découvrir des clusters de forme arbitraire. D'autres méthodes de clustering ont été développées sur la base de la notion de densité. Leur idée générale est de continuer à développer un cluster donné tant que la densité (nombres de données ou de points de donnée) du voisinage dépasse un certain seuil

[**Heller and Ghahramani, 2005**]. Par exemple, pour chaque point de données d'un cluster donné, le voisinage d'un rayon donné doit contenir au moins un nombre minimal de points. Une telle méthode peut être utilisée pour filtrer le bruit ou les valeurs aberrantes et découvrir des clusters de forme arbitraire. Nous pourrons citer l'algorithme DBSCAN [**Ester et al., 1995**].

1.4.4 Méthodes basée sur les grilles

Les méthodes basées sur une grille quantifient l'espace de données en un nombre fini de cellules qui forment une structure de grille [**Wang et al., 1997**]. Toutes les opérations de clustering sont effectuées sur la structure de grille (c'est-à-dire l'espace quantifié). L'avantage principal de cette approche est son temps de traitement rapide, qui est généralement indépendant du nombre de cellules dans chaque dimension de l'espace quantifié.

L'utilisation de grilles constitue souvent une approche efficace pour résoudre de nombreux problème, y compris le clustering. Par conséquent, les méthodes basées sur une grille peuvent être intégrées à d'autres méthodes de clustering telles que les méthodes basées sur la densité et les méthodes hiérarchiques. Nous pourrons citer un algorithme CLIQUE [**Agrawal et al., 1998**].

1.5 Distance et Similarité

Plusieurs méthodes de clustering utilisent les mesures de distance pour statuer sur la similitude ou la dissimilitude de n'imoprte quelle paire de données. La mesure de distance entre deux données o_i et o_j est une fonction d définie par :

$$\begin{aligned} d : O \times O &\rightarrow \mathbb{R}^+ \\ (o_i, o_j) &\rightarrow d(o_i, o_j). \end{aligned} \tag{1.5}$$

Une mesure de distance est appelée mesure métrique de distance si elle satisfait également les propriétés suivantes :

- Positivité :

$$d(o_i, o_j) > 0 \quad ; \quad \forall o_i, o_j \in O.$$

- Inégalité triangulaire :

$$d(o_i, o_k) \leq d(o_i, o_j) + d(o_j, o_k) \quad ; \quad \forall o_i, o_j, o_k \in O.$$

- Séparation :

$$d(o_i, o_j) = 0 \Rightarrow o_i = o_j \quad ; \quad \forall o_i, o_j \in O.$$

- Symétrie :

$$d(o_i, o_j) = d(o_j, o_i) \quad ; \quad \forall o_i, o_j \in O.$$

Dans ce qui suit, nous posserons en revue les principales mesures de distance utilisées en clustering.

1.5.1 Mesures de similarité pour les attributs numériques

1.5.1.1 Distance entre deux données

Mesure de distance	Notation	Formule mathématique
Distance de Minkowski	d_α^{M}	$\sqrt[\alpha]{\sum_{m=1}^{d} \lvert o_{im} - o_{jm}\rvert^\alpha}$
Distance de Manhattan	$d_1^{\mathrm{M}} = d^{\mathrm{Mh}}$	$\sum_{m=1}^{d} \lvert o_{im} - o_{jm}\rvert$
Distance Euclidienne	$d_2^{\mathrm{M}} = d^{\mathrm{E}}$	$\sqrt{\sum_{m=1}^{d} (o_{im} - o_{jm})^2}$
Distance de Tchebyschev	$d_\infty^{\mathrm{M}} = d^{\mathrm{T}}$	$\max_{1 \le m \le d} \{\lvert o_{im} - o_{jm}\rvert\}$

TABLE 1.2 – Quelques mesures de distance

1.5.1.2 Distance entre deux classes

Soit $D(C_k, C_l)$ la distance entre deux clusters C_k et C_l, pour cela il y a certaines fonctions permettant de mesurer cette distance comme :

- **Plus Proche Voisin** :

$$D_{\min}(C_k, C_l) = \min_{\substack{o_i \in C_k \\ o_j \in C_l}} d(o_i, o_j). \tag{1.6}$$

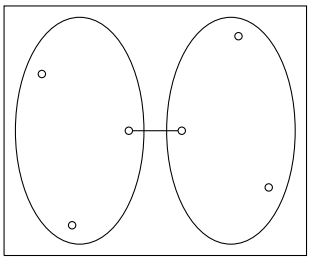

FIGURE 1.5 – Plus Proche Voisin

- **Diamètre Maximum** :

$$D_{\max}(C_k, C_l) = \max_{\substack{o_i \in C_k \\ o_j \in C_l}} d(o_i, o_j). \tag{1.7}$$

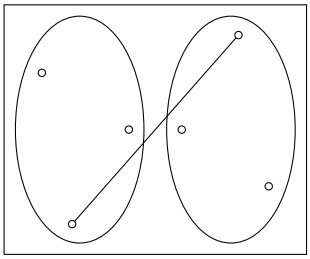

FIGURE 1.6 – Diamètre Maximum

- **Distance Moyenne** :

$$D_{\text{moy}}(C_k, C_l) = \frac{1}{n_k n_l} \sum_{i=1}^{n_k} \sum_{j=1}^{n_l} d(o_i, o_j). \tag{1.8}$$

Où :
– $n_k = |C_k|$;
– $n_l = |C_l|$.

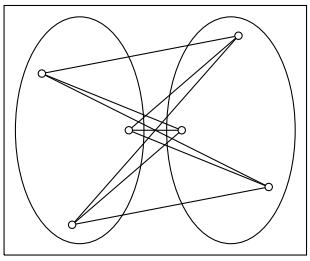

FIGURE 1.7 – Distance Moyenne

- **Distance barycentrique** :

$$D_\mu(C_k, C_l) = d(\mu_k, \mu_l). \tag{1.9}$$

D'où μ_k est le centroïde de cluster k avec :

$$\mu_k = \frac{1}{n_k} \sum_{o_i \in C_k} o_i. \tag{1.10}$$

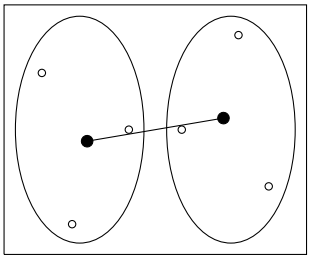

FIGURE 1.8 – Distance barycentrique

1.5.2 Mesure de similarité pour les attributs binaires

Dans le cas des attributs binaires, la similarité entre deux données est calculée en terme de nombre de fréquences. En effet la similarité se calcule à partir des informations du tableau de contingence. Ce tableau permet de compter le nombre de concordances et le nombre de discordances entre les attributs de données. L'interprétation de ces nombres est la suivante :

- a : le nombre d'attribut 1 que possèdent o_i et o_j ;
- b : le nombre d'attribut 1 que possèdent o_i et pas o_j ;
- c : le nombre d'attribut 1 que possèdent o_j et pas o_i ;
- a : le nombre d'attribut 1 que o_i et ni o_j ne possèdent.

		o_j	
		1	0
o_i	1	a	b
	0	c	d

TABLE 1.3 – Tableau de contingence

Le tableau 1.4 présente les mesures de similarité les plus usuelles de la littérature spécifique aux attributs binaires. Chacune de ces mesures de similarité possède ses propres propriétés qui influencent les résultats de clustering. Les indices de similarité présentés dans le tableau 1.4 sont proposés en combinant de diverses manières les quatres nombres du tableau 1.3.

Mesures de similarité	Notation	Définition
[Jaccard, 1908]	S_{J}	$\frac{a}{a+b+c}$
[Kulczynski, 1927]	S_{K}	$\frac{1}{2}(\frac{a}{a+b}+\frac{a}{a+c})$
[Dice, 1945]	S_{D}	$\frac{2a}{2a+b+c}$
[Sørensen, 1948]	S_{Sor}	$\frac{4a}{4a+b+c}$
[Ochiai, 1957]	$S_{\cos}$	$\frac{a}{\sqrt{a+b}\sqrt{a+c}}$
[Anderberg, 1973]	S_{A}	$\frac{8a}{8a+b+c}$
[Sneath et al., 1973]	S_{SS}	$\frac{a}{a+2b+2c}$

TABLE 1.4 – Quelques mesures de similarité

Remarque :
Nous pouvons utiliser la mesure de Jaccard dans les attributs numériques, sous la formule :

$$S_{\mathrm{J}}(o_i, o_j) = \frac{o_i^T.o_j}{\|o_i\|^2 + \|o_j\|^2 - o_i^T.o_j}. \tag{1.11}$$

1.5.3 Mesures de similarité pour les données nominales

Pour les bases de données catégorielles, il n'y a pas une mesure inhérente de distance, telle que la distance Euclidienne, qui peut être directement appliquée pour calculer la dissimilarité entre deux données catégoriques. Ceci est parce qu'il n'y a pas d'ordre naturel parmi les valeurs catégorielles. Par conséquent, il est difficile de mesurer la dissimilarité entre deux données catégoriques. La distance est largement utilisée pour calculer la distance entre les données catégoriques nominales de Hamming [**Ng et al., 2007**]. La distance de Hamming entre deux données o_i et o_j est donnée par le nombre de caractéristiques de o_i qui diffèrent de celles de o_j comme suit :

$$d^{\mathrm{H}}(o_i, o_j) = \sum_{m=1}^{d} \gamma(o_{im}, o_{jm}); \tag{1.12}$$

où

$$\gamma(o_{im}, o_{jm}) = \begin{cases} 0 & \text{si} \quad o_{im} = o_{jm}; \\ 1 & \text{sinon} \quad o_{im} \neq o_{jm}. \end{cases} \tag{1.13}$$

1.5.4 Mesures de similarité pour les attributs ordinaux

Lorsque les attributs sont ordinaux, alors une transformation de ces attributs est nécessaire [**Elghazel, 2007**]. Les attributs o_{im} ; $m = \overline{1,d}$ sont remplacées par leurs rangs $r_{im} = 1, 2, \cdots, p_m$, où p_m est le nombre de valeurs distinctes du $m^{\text{ème}}$ attribut. Par la suite, ces valeurs sont transformées en utilisant la formule suivante qui fournit une variation des valeurs des attributs r_{im} entre 0 et 1 :

$$z_{im} = \frac{r_{im} - 1}{p_m - 1}. \tag{1.14}$$

La distance entre deux données peut être calculée par la suite en utilisant une des formules développées pour les attributs numériques.

1.5.5 Mesures de similarité pour les attributs mixtes

Dans le cas où les attributs sont mixtes (des attributs numériques et des attributs catégoriques) [**Huang, 1998**] et [**Lebbah et al., 2005**] ont montré que les formalismes habituels des problèmes de clustering restent valables sous condition d'utilisation d'une distance adaptée. Une méthode classique consiste à combiner la distance Euclidienne à celle de Hamming :

$$d(o_i, o_j) = d^{\mathrm{E}}\big((o_{i1}, \cdots, o_{id_r}), (o_{j1}, \cdots, o_{jd_r})\big) + \alpha d^{\mathrm{H}}\big((o_{i1}, \cdots, o_{id_c}), (o_{j1}, \cdots, o_{jd_c})\big). \tag{1.15}$$

où d_r est le nombre des attributs numériques et d_c est le nombre des attributs catégoriques. Par conséquent, le premier terme est la distance Euclidienne sur les attributs numériques et le second terme est la distance de Hamming sur les attributs catégoriques. Cependant dans cette combinaison, il est important de prendre en considération l'influence de la partie numérique des données par rapport à la partie catégorique et inversement. Ainsi, l'utilisation d'un paramètre de pondération α pour éviter de favoriser un type d'attribut à l'autre est nécessaire.

1.6 Évaluation du clustering

En général, pour un cluster il faut évaluer la faisabilité de l'analyse de clustering sur un ensemble de données. Les tâches principales de l'évaluation de clustering sont les suivantes [**Jain et al., 1988**] :

1.6.1 Évaluer la tendance au clustering

L'évaluation de la tendance de clustering détermine un ensemble de données donné à une structure non aléatoire, ce qui peut conduire à des clusters significatifs. Considérons un ensemble de données qui ne possèdent aucune structure non aléatoire, tel qu'un ensemble

de points uniformément répartis dans un espace de données. Même si un algorithme de clustering peut renvoyer des clusters pour les données, ces clusters sont aléatoires et n'ont pas de sens.

1.6.2 Nombre de clustering

Quelques algorithmes exigent le nombre de clusters dans un ensemble de données en tant que paramètre. De plus, le nombre de clusters peut être considéré comme une statistique sommaire intéressante et importante d'un ensemble de données. Par conséquent, il est souhaitable d'estimer ce nombre avant même qu'un algorithme de clustering ne soit utilisé pour obtenir des clusters détaillées.

Déterminer le nombre de clusters est loin d'être facile, souvent parce que le bon nombre est ambigu. Déterminer le nombre exact de clusters à utiliser dépend souvent de la forme et l'échelle de la distribution dans l'ensemble de données, ainsi que de la résolution du clustering requise par l'utilisateur. Il existe de nombreuses façons d'estimer le nombre de clusters.

Des méthodes plus avancées peuvent déterminer le nombre de clusters à l'aide de critères d'information ou d'approches théoriques de l'information. Nous pouvons citer :

- Le critère de longueur minimale de message (MML) ;
- Le modèle de mélange gaussien (GMM) ;
- Le critère de longueur minimale de description (MDL) ;
- Le critère d'information Bayes (BIC) ;
- Le critère d'information Akiake (AIC) ;
- Gap statistics (GS).

1.6.3 Mesurer la qualité de clustering

- **L'indice de Dunn (Du)**
  L'indice de Dunn [**Dunn, 1974**] permet d'identifier les clusters compacts et bien séparés. L'objectif principal de l'indice de Dunn est d'augmenter en maximum les distances inter-clusters (séparation) et en revanche de diminuer au maximum les distances intra-clusters (augmenter la compacité), il est définit par :

$$\mathrm{Du}(\mathcal{C}) = \min_{1\leq k\leq K}\Big\{\min_{\substack{1\leq l\leq K\\ k\neq l}}\Big\{\frac{D(C_k,C_l)}{\max\limits_{1\leq h\leq K} D(C_h,C_h)}\Big\}\Big\}. \tag{1.16}$$

  Où $D(C_h, C_h)$ mesure la distance à l'interieure du cluster

$$D(C_h,C_h) = \max_{o_i,o_j\in C_h} d(o_i,o_j). \tag{1.17}$$

  Une faible valeur indique une forte compacité et une forte réparation des clusters.

- **L'indice de Davies et Bouldin (DB)**
  L'objectif de cet indice [**Davies and Bouldin, 1979**] est de minimiser la similarité moyenne entre chaque cluster et le cluster qui lui est le plus similaire, cet indice est défini par :

$$\mathrm{DB}(\mathcal{C}) = \frac{1}{K}\sum_{k=1}^{K} \max_{\substack{1\leq l\leq K \\ k\neq l}} \Big\{\frac{\lambda(C_k)+\lambda(C_l)}{d(\mu_k,\mu_l)}\Big\}. \tag{1.18}$$

  Où $\lambda(C_k)$ est la distance moyenne entre chacune donnée de C_k et son centroïde μ_k

$$\lambda(C_k) = \frac{1}{n_k}\sum_{o_i\in C_k} d(o_i,\mu_k). \tag{1.19}$$

  Une valeur optimale de K est celle qui minimise DB
  Plus les clusters sont compacts et $\lambda(C_k)$ est petite. Plus les clusters sont séparés et $d(\mu_k,\mu_l)$ est élevée. Une valeur faible de l'indice de DB indique un clustering de bonne qualité.
- **Le coefiicient silhouette (CS)**
  Le coefficient silhouette [**Rousseeuw, 1987**] permet d'évaluer la compacité des clusters que la séparabilité de ceux-ci. Il peut être calculé pour chaque donnée, pour chaque cluster et pour le clustering entier. Pour une donnée o_i, il est défini comme :

$$\mathrm{CS}(o_i) = \frac{b_{o_i}-a_{o_i}}{\max\{a_{o_i},b_{o_i}\}}. \tag{1.20}$$

  avec a_{o_i} la distance moyenne entre la donnée o_i et toutes les autres données appartenant au même cluster que o_i.

$$a_{o_i} = \frac{1}{n_k-1}\sum_{\substack{o_j\in C_k \\ o_j\neq o_i}} d(o_i,o_j) \text{ avec } o_i\in C_k \quad ; \quad k\in\{1,\cdots,K\}. \tag{1.21}$$

  b_{o_i} la distance moyenne minimale entre la donnée o_i et toutes les autres données n'appartenant pas à ce même cluster.

$$b_{o_i} = \min_{\substack{1\leq l\leq K \\ l\neq k}} \{\frac{1}{n_l}\sum_{o_j\in C_l} d(o_i,o_j)\}. \tag{1.22}$$

  Le coefficient $\mathrm{CS}(o_i)$ varie entre -1 et 1. Une valeur positive ($a_{o_i} < b_{o_j}$) signifie que les données appartenant au même cluster que o_i sont plus proches de o_i que des données d'autres clusters.
  Pour un cluster, le coefficient silhouette est la moyenne des coefficients des données appartenant à ce cluster :

$$\mathrm{CS}(C_k) = \frac{1}{n_k}\sum_{o_i\in C_k} \mathrm{CS}(o_i). \tag{1.23}$$

Enfin, pour un clustering, le coefficient silhouette est égal à la moyenne des coefficients de ses clusters :

$$\mathrm{CS}(\mathcal{C}) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{CS}(C_k). \tag{1.24}$$

Le coefficient pour le clustering varie également entre -1 et 1, une valeur positive indiquant que les clusters sont très compacts et bien séparés. Il est à noter que le calcul de cet indice est relativement coûteux en temps car de nombreux calculs de distance sont nécessaires à son évaluation.

- **L'indice de Wemmert et Gançarski (WG)**
  L'indice de Wemmert et Gançarski [**Wemmert et al., 2000**] évalue la séparabilité et la compacité des clusters. Pour un cluster, il est définit par :

$$\mathrm{WG}(C_k) = \begin{cases} 0 & \text{si} \quad \dfrac{1}{n_k}\displaystyle\sum_{o_i\in C_k}\frac{d(o_i,\mu_k)}{d(o_i,\mu_l)} > 1; \\ 1 - \dfrac{1}{n_k}\displaystyle\sum_{o_i\in C_k}\frac{d(o_i,\mu_k)}{d(o_i,\mu_l)} & \text{sinon} \quad . \end{cases} \tag{1.25}$$

  Où $l = \arg\min_{j\neq k}\big(d(o_i,\mu_j)\big)$.
  L'indice de Wemmert et Gancarski prend ses valeurs entre 0 et 1, 1 indiquant une très bonne compacité et séparabilité des clusters.
  Pour un clustering, il est défini par :

$$\mathrm{WG}(\mathcal{C}) = \frac{1}{n}\sum_{k=1}^{K} n_k \mathrm{WG}(C_k). \tag{1.26}$$

- **La somme des erreurs au carré (SSE)**
  La somme des erreurs au carré est l'une des façons les plus simples d'évaluer la qualité d'un résultat. Elle est définie comme [**Desgraupes, 2013**] :

$$\mathrm{SSE}(\mathcal{C}) = \sum_{k=1}^{K}\sum_{o_i\in C_k} d^2(o_i,\mu_k). \tag{1.27}$$

1.7 Problèmes et limites du clustering

Malgré l'existence d'un grand nombre de méthodes de clustering ainsi que leur utilisation avec succès dans de nombreux domaines, le clustering pose encore de nombreux problèmes. Ces problèmes sont liés d'une part au manque de précision dans la définition de ce qui est réellement un cluster mais également dans la difficulté de définir une mesure de similarité entre les données ou encore dans la définition d'une fonction objective pour un problème donné. [**Jain et al., 1988**] ont listé un ensemble de questions qui sont

nécessaire de se poser lorsqu'on entreprend d'effectuer une tâche de clustering. Cette liste de questions met en premier la multiplicité et surtout la nature différente des paramètres à prendre en compte dans ce type d'approche :

- Qu'est-ce qu'un cluster ?
- Quels attributs doivent être utilisés ?
- Les données doivent-elles être normalisées ?
- Les données contiennent-elles des données atypiques ?
- Comment est définie la similarité entre deux données ?
- Combien de clusters sont présents ?
- Quelle méthode de clustering doit-on utiliser ?
- Est-ce que les données contiennent des cluster ?
- Est-ce que la partition découverte est valide ?

Nous avons étudié dans les sections suivantes certains problèmes et limites récurrents en clustering qui sont soulevés par ces question.

Conclusion

Le clustering est une tâche dont l'objectif est de trouver des groupes au sein d'ensemble de données. Dans ce chapitre, nous avons étudié les grands concepts du clustering, les principales méthodes existantes. Nous avons également présenté les nombreux problèmes et limites en clustering. Le chapitre suivant sera consacré à la présentation de quelques concepts fondamentaux en théorie des jeux et K-means qui seront mis en oeuvre ultérieurement dans la résolution du problème de clustering.

2

K-means et concepts de la théorie des jeux

Contents

Introduction

L'objectif de ce chapitre est d'introduire les concepts et les notions nécessaires à la compréhension du chapitre 3. Nous commencerons par mentionner et détailler l'algorithme de K-means ainsi que quelques méthodes basées sur K-means. Par la suite, nous rappellerons quelques concepts de la théorie des jeux et nous présenterons quelques travails proposés dans le clustering et la théorie des jeux.

2.1 K-means

2.1.1 Objective de K-means

K-means est un algorithme de clustering partitionnement qui se base sur une mesure de similarité et qui tente de trouver K clusters ne se chevauchant pas. Quoiqu'il ne fonctionne pas très bien pour les attributs catégoriels, il a un bon sens numérique.

K-means est généralement exprimé par une fonction objective qui dépend des proximités des points d'objets par rapport aux centroïdes du cluster. La fonction objective de K-means est formulée comme la somme des erreurs au carré, comme suit [**Wu, 2012**] :

$$E(\mathcal{C}) = \sum_{k=1}^{K} \sum_{o_i \in C_k} d^2(o_i, \mu_k). \tag{2.1}$$

Cette fonction est utilisée pour évaluer la qualité du partitionnement, de sorte que les objets d'un cluster soient semblables les uns aux autres, mais différents des objets des autres clusters.

2.1.2 Algorithme de K-means

L'algorithme de clustering K-means [**MacQueen et al., 1967**] est le suivant :

Étape initiale : Nous sélectionnons K centroïdes aléatoires $\mu_k^{(0)}$, $k = \overline{1, K}$

Étape d'affectation : Pour chaque point d'objets, nous calculons les distances entre un objet et les K centroïdes ($d(o_i, \mu_k)$, $k = \overline{1, K}$) et regroupons cet objet avec un centroïde qui réalise la distance minimale, comme suit :

$$o_i \in C_l \text{ tel que } l = \arg \min_{k=\overline{1,K}} d(o_i, \mu_k). \tag{2.2}$$

Étape de recalage des centroïdes : Nous recalculons les nouveaux centroïdes.

Cluster stables : Nous répétons la deuxième et la troisième étape jusqu'à ce qu'aucun objet ne change de cluster :

$$\mu_k^{(t)} = \mu_k^{(t+1)} \ ; \ \forall k = \overline{1, K} \ ; \ t \geq 0; \tag{2.3}$$

où t est le nombre d'itération.

Algorithme 1 Algorithme de K-means

Données : O, K, $\mu^{(0)}$
Sorties : C
Début
  $t = 1$;
  Tant que $\mu^{(t-1)} \neq \mu^{(t)}$ **faire**
    Pour chaque $o_i \in O$ **faire**
      Pour chaque $k = 1$ jusqu'à K **faire**
        Calculer $d(o_i, \mu_k)$;
      Fin pour ;
      Regrouper l'objet o_i du centroïde le plus proche ;
    Fin pour ;
    Pour chaque $k = 1$ jusqu'à K **faire**
      Calculer $\mu_k^{(t)}$;
    Fin pour ;
    $t = t + 1$;
  Fin tant que ;
Fin.

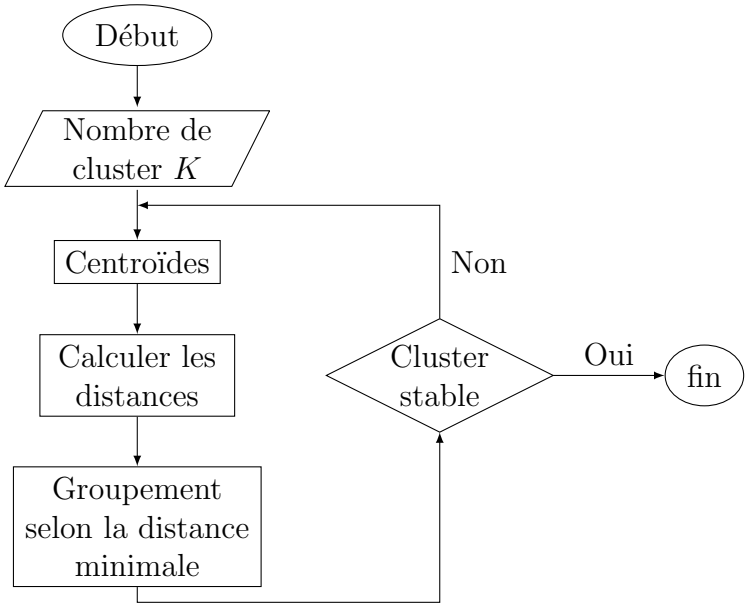

FIGURE 2.1 – Algorigramme de K-means

2.1.3 Avantages et inconvénients

2.1.3.1 Avantages de K-means

1. L'algorithme de K-means est très populaire du fait qu'il soit très facile à comprendre et à mettre en oeuvre [**D'Hondt Frédéric, 2004**] ;
2. K-means est rapide et efficace en termes de coût de calcul. En effet sa complexité est $O(K \times n \times d)$;
3. K-means convient à un grand nombre d'ensembles de données et est calculé beaucoup plus rapidement. Il peut également produire des clusters plus élevés ;
4. Donne les meilleurs résultats lorsque les ensembles d'objets sont distincts ou bien séparés les uns des autres ;
5. L'algorithme utilisé permet de partitionner les gros de datasets. Son efficacité est fonction de la forme des clusters. Les K-Means fonctionnent bien dans les clusters hyper-sphériques ;
6. La segmentation en K-Means est linéaire en nombre d'objets de données, ce qui augmente le temps d'exécution. Il ne faut pas plus de temps pour classer des caractéristiques similaires dans des données.

2.1.3.2 Inconvénients de K-means

1. Le nombre de cluster doit être fixé au départ [**D'Hondt Frédéric, 2004**] ;
2. S'il y a deux clusters qui se chevauchent fortement, alors l'algorithme ne pourra pas résoudre le problème ;
3. K-means ne permet pas de développer un ensemble optimal de clusters et nous devons choisir les clusters avant pour des résultats effectifs ;
4. L'algorithme K-means ne fonctionne pas bien pour les objets catégorielles ;
5. Le choix aléatoire de centroïdes ne peut nous conduire toujours à un résultat fructueux ;
6. L'algorithme échoue pour les objets non linéaires.

2.1.4 Quelques méthodes basées sur K-means

2.1.4.1 Méthode des K-médoïdes

Dans la méthode des K-médoïdes, un cluster est représenté par un de ses objets. Cet objet représentatif est appelé médoïde, et est le plus placé au centre en terme de distance [**Kaufman and Rousseeuw, 1990**]. Les méthodes K-médoïdes présentent l'avantage d'être applicables à tout type de données puisqu'il n'est pas nécessaire de définir la moyenne des objets. ils sont dans l'ensemble plus robustes aux points aberrants que les méthodes des K-means, d'autant qu'elles recourent aux médoïdes plutôt qu'aux moyennes (centroïdes) pour évaluer la distance aux centres. Cependant, la complexité temporelle est

$O(K \times (n-K)^2)$. Cette dégradation de la complexité est relative à la phase de repérage d'un médoïde approprié qui doit tester tous les objets, qui ne sont pas des médoïdes, ce qui est coûteux en temps de calcul. Nous citons l'algorithme PAM.

2.1.4.2 Méthode de fuzzy C-means

Fuzzy C-means (FCM) est une méthode de clustering détermine par [**Ball and Hall, 1967**]. Dans FCM, il est possible qu'un objet appartienne 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 flou 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.

2.2 Théorie des jeux

La théorie des jeux est un outil mathématique pour analyser et prévoir comment les humains se comportent dans des situations stratégiques [**Osborne and Rubinstein, 1994**], appelées jeux. Un jeu peut se voir comme une modélisation d'une situation d'interactions décisionnelles.

Les travaux de recherches développés en théorie des jeux peuvent en premier lieu être classés entre jeux coopératifs et jeux non-coopératifs, chaque branche ayant ses propres applications et concepts de solutions.

2.2.1 Définitions élémentaires

Jeu :

Un jeu est une représentation formelle d'une situation dans laquelle un certain nombre de joueurs sont conduits à faire des choix parmi un certain nombre de stratégies, et où l'utilité de chaque joueur, ne dépend pas que de ses choix, mais également des choix effectués par les autres.

Les jeux sont principalement caractérisés par les éléments suivants :

Joueur :

Un joueur est défini comme étant l'unité fondamentale de décision pouvant être une personne, un groupe de personne, une société,.... qui agit dans le but de maximiser son utilité tout en étant rationnel.

Nous noterons l'ensemble des joueurs par : $I = \{1, 2, \cdots, n\}$ où n est le nombre de joueurs participant au jeu.

Stratégie :

une stratégie est un plan d'actions complet pour chaque joueur, spéciffant ce que fera ce dernier, à chaque étape du jeu et face à chaque situation pouvant survenir au cours du jeu. Une stratégie décrit donc le comportement d'un joueur.

Nous distinguons deux types de stratégies :

- Une stratégie pure du joueur i est l'action qu'il choisit à chaque fois qu'il est susceptible de joueur, c'est-à-dire, toutes les options possibles qu'a le joueur. Nous noterons par X_i, l'ensemble de toutes les stratégies pures du joueur i avec $i \in I$, et x_i un élément de X_i tel que : $|X_i| = m_i$. On pose $X = \prod_{i=1}^{n} X_i$ l'ensemble de toutes les issues en stratégies pures du jeu.
- Une stratégie mixte du joueur i est une distribution de probabilités $\alpha = (\alpha_1, \cdots, \alpha_{m_i})$ définie sur l'ensemble des stratégies pures du joueur i.
  Si l'ensemble des stratégies X_i est fini, alors on définit l'ensemble des stratégies mixtes comme suit :

$$\Delta_{m_i} = \{\alpha = (\alpha_1, \cdots, \alpha_{m_i}) \in \mathbb{R}^{m_i} \ ; \ \sum_{j=1}^{m_i} \alpha_j = 1 \ ; \ \alpha_j \geq 0 \ ; \ \forall j = \overline{1, m_i}\}. \tag{2.4}$$

  Où α_j est le probabilité que le joueur i joue sa stratégie pure $x_j \in X_i$.

Fonction d'utilité : Une fonction d'utilité est une fonction associée à chaque joueur i reflétant ses préférences c'est-à-dire, la satisfaction qu'il éprouve en utilisant une stratégie donnée. Cette satisfaction ne dépend pas seulement de son choix, mais également du choix des autres joueurs.

Nous noterons par u_i la fonction d'utilité du joueur i définie comme suit :

$$u_i : X = \prod_{i=1}^{n} X_i \to \mathbb{R}. \tag{2.5}$$

Chaque joueur souhaite maximiser sa fonction d'utilité.

2.2.2 Classification des jeux

2.2.2.1 Jeux coopératifs / jeux non-coopératifs

La théorie des jeux coopératifs se focalise sur une valeur, c'est-à-dire la valeur qu'un ensemble de joueurs peut obtenir en coopérant, sans préciser les actions spécifiques que les joueur doivent entreprendre afin de créer cette valeur. Les jeux coopratifs modélisent les situations où les joueurs peuvent se grouper en coalitions. Les actions des joueurs seront alors menées conjointement de façon à atteindre un objectif commun.

Les jeux non-coopératifs modélisent les interactions où les joueurs sont libres de choisir leurs actions et où un joueur rationnel cherche à maximiser son propre bien-être.

2.2.2.2 Jeux simultanés / jeux séquentiels

Dans un jeu, si les joueurs décident de leurs actions simultanément, alors nous parlons dans ce cas de jeu simultané. à l'inverse, si les joueurs décident de leur actions l'une après l'autre alors, nous somme dans le cas d'un jeu séquentiel.

2.2.2.3 Jeux à information complète / incomplète

Un jeu est dit à information complète si chacun des joueurs connaît la structure du jeu, c'est-à-dire : l'ensemble des joueurs, les préférences des joueurs, les règles du jeu et le type d'information qu'a chaque moment du je chaque joueur possède sur les actions entreprises par les autres joueurs au cours des phases précédentes. Donc, chaque joueur peut se mettre à la place de tous les autres joueurs et du modélisateur. Si au moins un des joueurs ne connaît pas entrièrement la structure du jeu, le jeu est dit à information incomplète.

2.2.2.4 Jeux à information parfaite / imparfaite

Un jeu est dit à information parfaite si chacun des joueurs, au moment de choisir son action, a une connaissance parfaite de l'ensemble des décisions prises antérieurement par les autres joueurs. Un jeu est à information imparfaite si un des joueurs ne connaît pas, à un moment du déroulement du jeu, ce qu'a joué un autre joueur. Ceci peut arriver dans le cas où nous cachons l'information aux joueurs ou parce que les joueurs jouent simultanément.

2.2.2.5 Jeux répétés

Les jeux répétés sont des jeux qui sont joués plus d'une fois. Les joueurs peuvent choisir des actions différentes en considérant l'histoire du jeu étant donné que l'expérience que les joueurs ont acquise à travers la répétition est cruciale pour définir leurs actions futures.

2.2.3 Forme extensive et forme stratégique d'un jeu

2.2.3.1 Jeu sous forme extensive

Lorsque le jeu est à information complète, chaque joueur connaît toutes les données du problème, pour lui et pour les autres. Toutefois, pour qu'un jeu soit totalement défini, il faut que ses règles précisent l'ordre des coups. Trois types de situations peuvent alors être envisagées :

- Soit les joueurs font leurs choix de façon séquentielle, dans un ordre précis fixé à l'avance.
- Soit ils prennent leur décision simultanément.
- Soit ils font face à des situations mixtes, avec des coups successifs et des coups simultanés.

Lorsque les règles du jeu stipulent que les joueurs interviennent les uns après les autres, dans un ordre précis et que le nombre d'actions parmi lesquelles leur choix s'exerce est fini,

la représentation qui semble la plus appropriée consiste à tracer un arbre (appelé arbre de Kuhn). Une telle représentation est dite sous forme extensive du jeu.

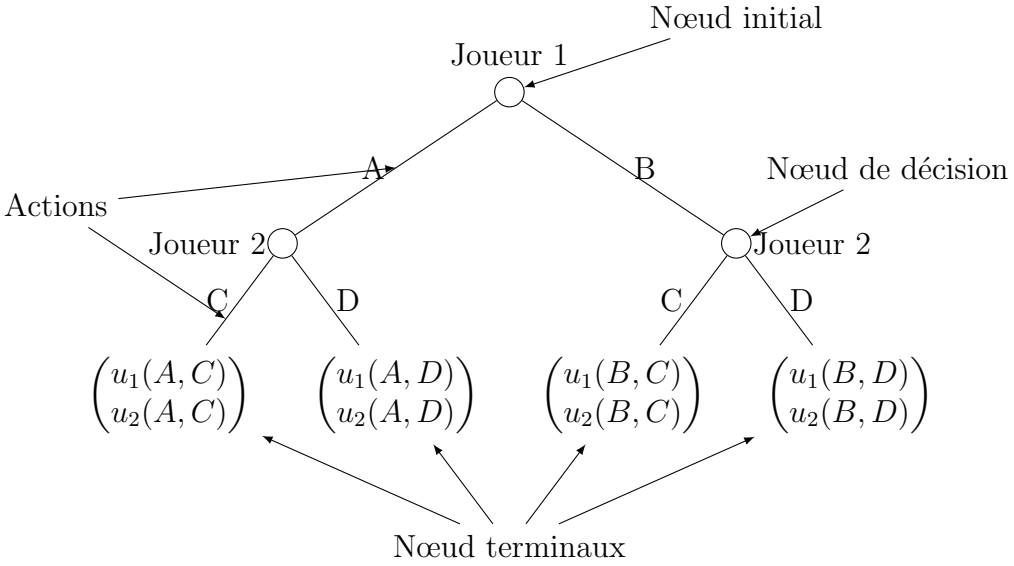

FIGURE 2.2 – Jeu sous forme extensive à deux joueurs

2.2.3.2 Jeu sous forme stratégique

Lorsque le jeu est à coups simultanés, la représentation par la forme extensive devient particulièrement lourde et compliquée. Pour cela, la modélisation qui apparaît comme la plus appropriée est la forme stratégique, ou normale, qui appelle à un (ou des) tableau(x) de nombres donnant les gains des joueurs pour chacune des issues possibles, les lignes et les colonnes correspondent aux diverses stratégies. Dans ce contexte, nous supposons que la satisfaction d'un joueur peut être représentée par des nombres réels. Plus le nombre est évelé, plus la satisfaction est importante. Ces préférences sont définies par une fonction d'utilité ou de satisfaction des résultats.

2.2.4 Définitions

Nous présenterons dans cette section les concepts de solutions les plus utilisés dans la branche non-coopérative de la théorie des jeux.

1. **Équilibre de Nash en stratégies pures**
   Une situation $x^* = (x_1^*, \cdots, x_n^*) \in X$ est un équilibre de Nash du jeu non coopératif, si pour chaque joueur $i \in I$ on a :

$$\forall i \in I \ ; \ \forall x_i \in X_i \ ; \ u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*). \tag{2.6}$$

équilibre de Nash est une situation dans laquelle aucun joueur n'a intérêt unilatéralement de s'écarter de la situation d'équilibre.

2. **Équilibre de Nash en stratégie mixtes**
   L'issue en stratégies mixtes $\alpha^* = (\alpha_1^*, \cdots, \alpha_n^*) \in \prod_{i=1}^{n} \Delta(X_i)$ est un équilibre de Nash si pour tout joueur $i \in I$ et pour toute stratégie mixte $\alpha \in \Delta(X_i)$, on a :

$$u_i(\alpha_i^*, \alpha_{-i}^*) \geq u_i(\alpha_i, \alpha_{-i}^*) \tag{2.7}$$

   où u_i est l'espérance de gains du joueur i quand il joue sa stratégie mixte α_i et les autres joueurs jouent leurs stratégies mixtes α_{-i}^*.
   Un équilibre en stratégies mixtes est donc une situation dans laquelle tous les joueurs choisissent leurs stratégies mixtes de façon à rendre leurs adversaires indifférents entre les gains espérés de chacune de leurs stratégies pures.

3. **Équilibre de Pareto-Nash**
   Un équilibre de Nash pur x^* est un équilibre de Pareto-Nash du jeu non-coopératif, si pour chaque joueur i on a :

$$\forall i \in I \ ; \ \forall x_i \in X_i \ ; \ u_i(x_i^*, x_{-i}) > u_i(x_i, x_{-i}). \tag{2.8}$$

   Un équilibre de Nash pur est Pareto optimal si le jeu n'admet aucun autre équilibre de Nash pur pour lequel chaque joueur a une utilité strictement plus élevée.

2.3 Clustering et la théorie des jeux

Dans cette section, nous allons présenter quelques travails proposés au niveau de l'université de Béjaïa :

1. **Application de la théorie des jeux pour la définition, développement et implémentation d'un algorithme de clustering** : [**Sabri, 2011**] qui est utilisée les jeux coopératifs sous forme stratégique pour représenter le problème de clustering de données numériques.
2. **Clustering : Approche par la théorie des jeux** : [**Hamidouche and Idjeraoui, 2013**] qui sont utilisés les jeux non coopératifs pour représenter le problème de clustering de données numériques.
3. **Application de la théorie des jeux dans les problèmes de clustering multi-critères** : [**Heloulou, 2017**] qui est proposée trois approche, la première basée sur la théorie des jeux séquentiels et les données numériques, la deuxième basée sur la théorie des jeux séquentiels et les données catégorielles et la dernière basée sur la théorie des jeux coopératifs.

Conclusion

Dans ce chapitre, nous avons présenté l'un des méthodes de partitionnement, nous avons commencé par l'objectif de K-means. Par la suite, nous avons présenté l'algorithme de K-means, les avantages et inconvénients et quelques méthodes basées sur K-means. Nous avons également présenté les notions de base de la théorie des jeux, qui est comme un outil pour résoudre les problème de clustering. Nous avons terminé par présentation de quelques travails proposés dans ce domaine.

Dans le chapitre suivant, nous allons présenter notre approche qui est de modéliser un problème du clustering sous forme d'un jeu.

3

Clustering et jeu non coopératif

Contents

Introduction

Dans cette section, nous présenterons une approche qui modélise le problème de clustering dans un contexte de jeu non coopératif.

Notre approche consiste à trouver une partition de l'espace de départ telles que les données appartenant à un même groupe soient plus similaires entre elles qu'avec les données issues d'un autre groupe. Elle nous permet de construire K clusters initiaux de données similaires et les améliorer afin d'obtenir des clusters correspondant à un équilibre de jeu associé.

3.1 Idée globale de notre proposition

Le modèle que nous mettons au point dans la partie d'un jeu non coopératif s'applique sur des données qui sont décrites par un vecteur d'attributs numériques à d dimensions, soit : $o_i = (o_{i1}, \cdots, o_{id}) \in \mathbb{R}^d$, et en notant l'ensemble des données par $O = \{o_i \mid i = \overline{1,n}\}$, et nous partitionnons l'ensemble O en K clusters de telle sorte qu'un clustering dur partiel, K étant connu à priori, soit un paramètre d'entrée de l'algorithme.

Le jeu que nous proposerons est un jeu non coopératif entre les clusters, où chacun d'entre eux est considéré comme un joueur. Le but de chaque cluster est d'améliorer sa fonction d'homogénéité. Les données qui forment le cluster ont des utilités non nulles qui affectent la fonction d'homogénéité.

Chaque cluster est représenté par une donnée unique qui a une meilleure utilité par rapport aux autres données du cluster même, appelé une donnée idéale. Nous notons l'ensemble des données idéales par $OI = \{oi_k \mid k = \overline{1,K}\}$.

Nous testerons notre approche non coopérative sur deux différentes mesures : la distance Euclidienne et la similarité de Jaccard afin de comparer les résultats.

3.2 Modélisation du problème

Un jeu non coopératif sous forme stratégique :

$$< \mathcal{C}, \{X_{C_k}\}_{C_k \in \mathcal{C} \text{ et } k=\overline{1,K}}, \{E(C_k)\}_{C_k \in \mathcal{C} \text{ et } k=\overline{1,K}} > . \tag{3.1}$$

Avec les éléments suivants :

1. $\mathcal{C}$: l'ensemble des K clusters.
2. X_{C_k} : l'ensemble des stratégies d'un cluster k avec $k = \overline{1,K}$. Nous posons une collection de stratégies décrivant les actions de chaque cluster. Les stratégies sont les suivantes :
   - La stratégie "Libérer une donnée" notée x_1. Cela signifie qu'un cluster a décidé de libérer cette donnée qui n'améliore pas son homogénéité ;
   - La stratégie "Récruter un donnée" notée x_2. Cela signifie qu'un cluster a décidé de récruter l'une des données libres, car il améliore son homogénéité ;
   - La stratégie "Ne rien faire" notée x_3. Le cluster est satisfait des données dont il dispose.
3. $E(C_k)$: une fonction d'homogénéité pour chaque cluster. Elle associe un vecteur dénoté $\sigma(C_k)$. Ce vecteur interprète comment les données du C_k sont regroupées et interagissent ensemble.

   La distance Euclidienne :

   La valeur $\sigma_j(C_k)$ d'un attribut est calculé comme suit :

$$\sigma_j(C_k) = \frac{1}{n_k} \sqrt{\sum_{o_i \in C_k} (o_{ij} - \mu_{kj})^2}; \tag{3.2}$$

alors, le vecteur σ d'un cluster :

$$\sigma(C_k) = \big(\sigma_1(C_k), \cdots, \sigma_d(C_k)\big);$$

donc, la fonction d'homogénéité d'un cluster :

$$E(C_k) = \max \sigma(C_k) - \min \sigma(C_k). \tag{3.3}$$

Notre objectif dans ce cas est de minimiser la fonction d'homogénéité ($E(C_k) \to \min$).

La similarité de Jaccard :

La valeur $\sigma_j(C_k)$ d'un attribut est calculé comme suit :

$$\sigma_j(C_k) = \frac{1}{n_k} \sum_{o_i \in C_k} (o_{ij}); \tag{3.4}$$

alors, le vecteur σ d'un cluster :

$$\sigma(C_k) = \big(\sigma_1(C_k), \cdots, \sigma_d(C_k)\big);$$

donc, la fonction d'homogénéité d'un cluster :

$$E(C_k) = \max \sigma(C_k) - \min \sigma(C_k). \tag{3.5}$$

Notre objectif dans ce cas est de maximiser la fonction homogénéité ($E(C_k) \to \max$).

3.2.1 Utilités des données

- **Les données dans clusters**
  Pour chaque donnée dans un cluster, nous définissons une fonction de gain qui représente l'utilité de cette donnée.

  La distance Euclidienne : L'utilité de donnée o_i par rapport à un cluster C_k est donnée par :

$$u(o_i, C_k) = \frac{1}{n_k - 1} \sum_{\substack{o_j \in C_k \\ o_i \neq o_j}} d^{\mathrm{E}}(o_i, o_j). \tag{3.6}$$

  La donnée idéale pour un cluster C_k est d'avoir l'utilité la plus petite :

$$oi_k = \arg\min_{o_i \in C_k} u(o_i, C_k) \ ; \ \forall k = \overline{1, K}. \tag{3.7}$$

La similarité de Jaccard : L'utilité de donnée o_i par rapport à un cluster C_k est donnée par :

$$u(o_i, C_k) = \frac{1}{n_k - 1} \sum_{\substack{o_j \in C_k \\ o_i \neq o_j}} S_{\text{J}}(o_i, o_j). \tag{3.8}$$

La donnée idéale de cluster C_k est celui ayant l'utilité la plus grande :

$$oi_k = \arg\max_{o_i \in C_k} u(o_i, C_k) \ ; \ \forall k = \overline{1, K}. \tag{3.9}$$

- **Les données libres**
  L'utilité des données libres (les données qui n'appartiennent à aucun cluster) est égale à zéro :

$$u(o_i, OL) = 0 \ ; \ \forall o_i \in OL; \tag{3.10}$$

d'où OL l'ensemble des données libres.

3.2.2 Fonction globale de clustering

La fonction globale de clustering est calculée de la manière suivante :

$$EG = \sum_{j=1}^{d} \big(\max_{1 \leq k \leq K} \sigma_j(C_k) - \min_{1 \leq k \leq K} \sigma_j(C_k) \big). \tag{3.11}$$

Nous pouvons ajouter au critère d'arrêt de la fonction globale de clustering la contrainte suivante : si les clusters s'améliorent alors la fonction globale change sinon elle reste inchangée.

3.2.3 Choix du nombre de clusters

Pour obtenir de bons résultats, il faut choisir un vrai nombre K. Pour cela nous appliquerons notre algorithme sur différents nombres K ($2 \leq K \leq [\frac{n}{3}]$), et nous choisirons l'un de ces nombres dont la fonction globale du clustering est meilleure. Nous pouvons ajouter une autre critère secondaire (dans le cas où nous obtiendrions plusieurs résultats similaires) qui est de minimiser le nombre de données libres.

Nous prenons la taille des clusters TC dans l'itération initiale égale ou presque égale. De plus, nous ne pouvons pas obtenir un cluster vide, pour cela nous ajoutons un critère pour éviter ce cas. Notons, le nombre minimum de données dans un cluster.

3.3 Ensemble de données

Notre ensemble de données est présentée comme une matrice de dimension $(n \times d)$, où :
- n est le nombre de données. Ces données peuvent être des personnes, des pays, des entréprises, des projets, etc.
- d est le nombre d'attributs d'une donnée. Ces attributs peuvent être des activités, des caractéristiques, etc.

3.4 Algorithme proposé

L'algorithme de notre approche pour un jeu non coopératif est le suivant :

Étape initiale : Nous choisissons une donnée qui n'appartient à aucun cluster et la regroupons avec les données les plus proches (ou les plus similaires) de taille TC. Nous répétons cette opération K fois pour former des clusters initiaux. Nous pouvons maintenant calculer les centroïdes et regrouper les données les plus proches de chaque centroïde (ou choisir les données les plus similaires dans chaque cluster).

Nous calculons les utilités des données, sélectionnons les données idéales et calculons les fonctions d'homogénéité.

Étape d'amélioration : Chaque cluster choisit séquentiellement sa stratégie selon les utilités de leurs données et sa fonction d'homogénéité. Après tous les changements opérés dans le cluster, nous recalculons les utilités et la fonction d'homogénéité.

Critère d'arrêt : En général, nous fixerons un nombre maximum d'itérations, mais nous pouvons obtenir un autre critère d'arrêt qui est le changement de la fonction globale de clustering. En d'autres termes, si la fonction n'est pas changée, il y a lieu d'arrêter.

Algorithme 2 Notre approche

Données : O, K, MI, $MinC$, TC
Sortées : $\mathcal{C}$, OL, EG
Début
  Pour chaque $o_i \in O$ **faire**
    Calculer la distance entre o_i et tous les autres données ;
  Fin pour ;
  Pour chaque $C_k \in \mathcal{C}$ **faire**
    Regrouper chaque donnée o_i à son cluster le plus proche C_k ;
    Pour chaque $o_i \in C_k$ **faire**
      Calculer $u(o_i, C_k)$;
    Fin pour ;
    Calculer oi_k ;
    Calculer $E(C_k)$;
  Fin pour ;
  Calculer $EG(0)$;
  $t = 1$;
  Tant que $t \leq MI$ et $EG(t-1) \neq EG(t)$ **faire**
    Pour chaque $C_k \in \mathcal{C}$ **faire**
      Choisir l'une des stratégies x_1, x_2 et x_3 ;
      Pour chaque $o_i \in C_k$ **faire**
        Calculer $u(o_i, C_k)$;
      Fin pour ;
      Calculer oi_k ;
      Calculer $E(C_k)$;
    Fin pour ;
    Calculer $EG(t)$;
    $t = t + 1$;
  Fin tant que ;
Fin.

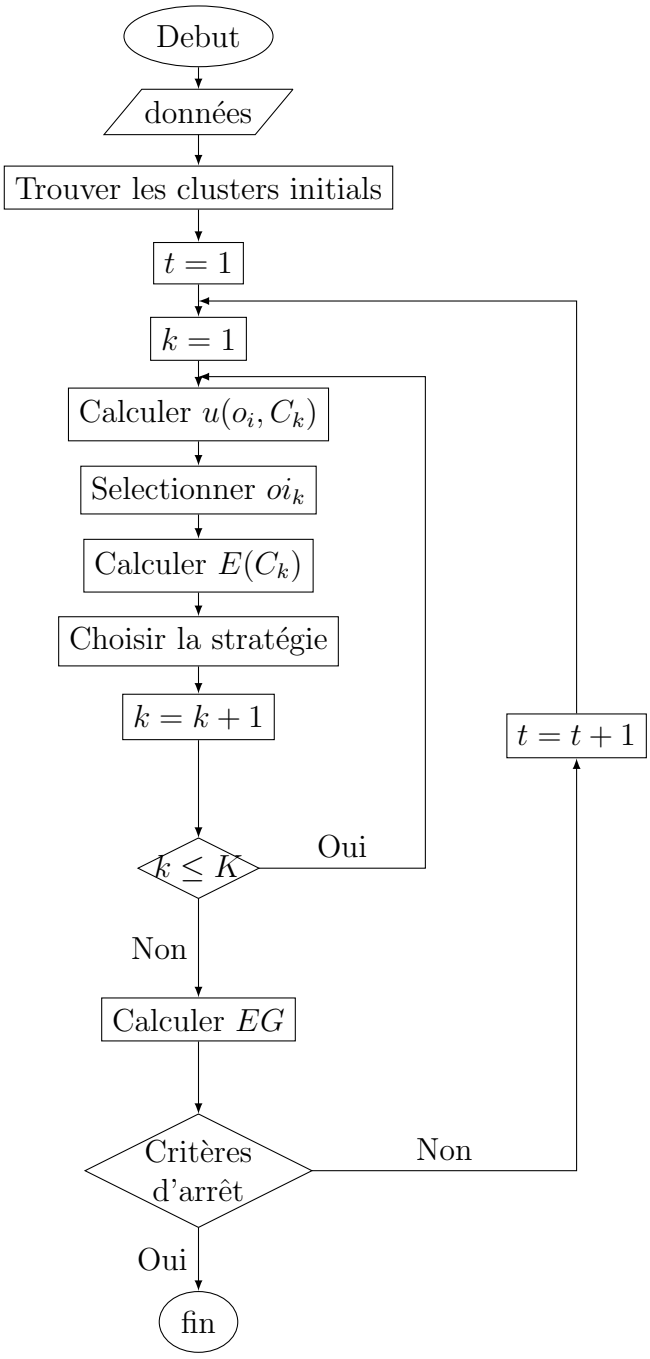

FIGURE 3.1 – Algorigramme proposé

3.5 Exemples d'applications sur plusieurs bases de données

3.5.1 Présentation de MATLAB

Actuellement, il existe en pratique des outils permettant de résoudre de tels problèmes. Mais évidement, la théorie montre combien il est difficile d'obtenir une solution optimale lorsque le modèle est limité en nombre de contraintes. L'outil informatique utilisé dans la recherche de solution du problème étudié est le MATLAB.

MATLAB est un langage de programmation de quatrième génération émulé par un environnement de développement du même nom, il est utilisé par la société 'The MathWorks', MATLAB permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en oeuvre des algorithmes, de créer des interfaces utilisateurs.

Les utilisateurs de MATLAB sont de milieux très différents comme l'ingénierie, les sciences et l'économie dans un contexte aussi bien industriel que pour la recherche.

3.5.2 Ensemble d'Iris

L'ensemble de données Iris connu aussi sous le nom de Iris de Fisher est un ensemble de données multivariées présenté par [**Fisher,1936**]. Il contient 150 données et chacune donnée est de 4 dimensions (largeur des sépales, longueur des sépales, largeur des pétales et longueur des pétales). Nous appliquerons notre approche proposé sur cet ensemble de données dans les cas Euclidien et Jaccard. (Voir l'annexe A).

Distance Euclidienne :

Après avoir exécuté notre algorithme plusieurs fois, plusieurs valeurs pour K, nous avons obtenu la figure suivante :

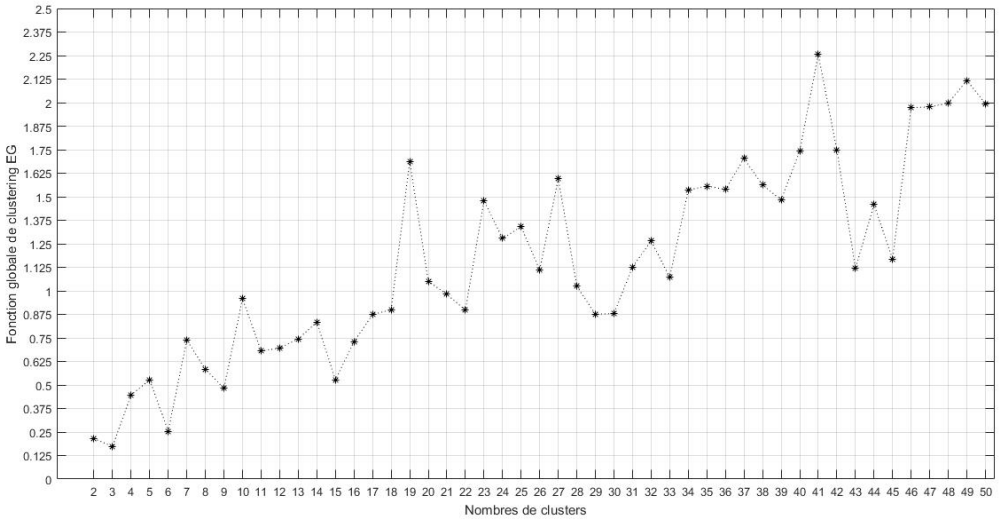

FIGURE 3.2 – Évolution de EG en fonction du nombre de clusters - Euclidien - Iris

Nous remarquons que le nombre de clusters qui minimise la fonction globale est égale à $K = 3$ avec $EG = 0.1732$.

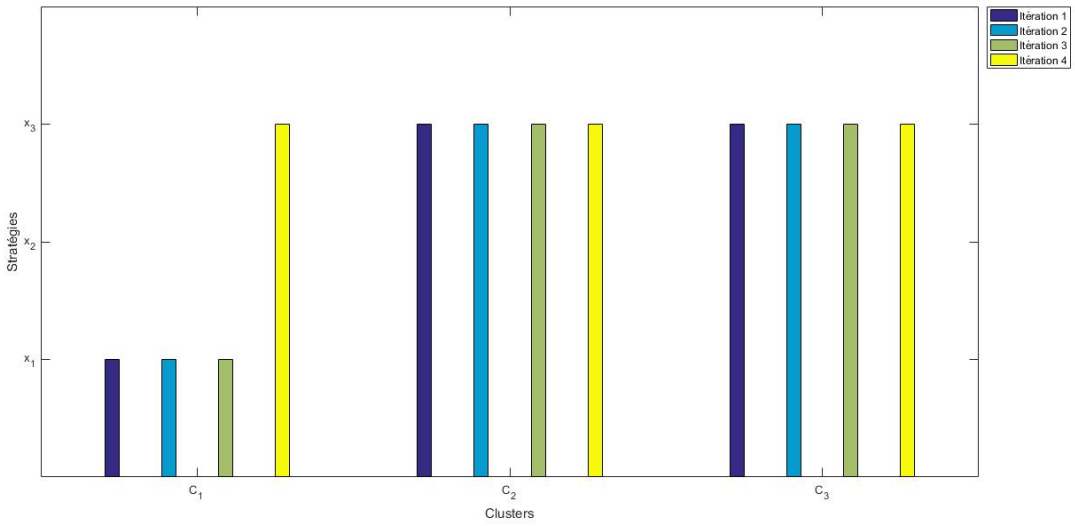

FIGURE 3.3 – Stratégies de chaque cluster - Euclidien - Iris

La figure 3.3 montre les stratégies jouées par chaque cluster dans les 4 itérations.

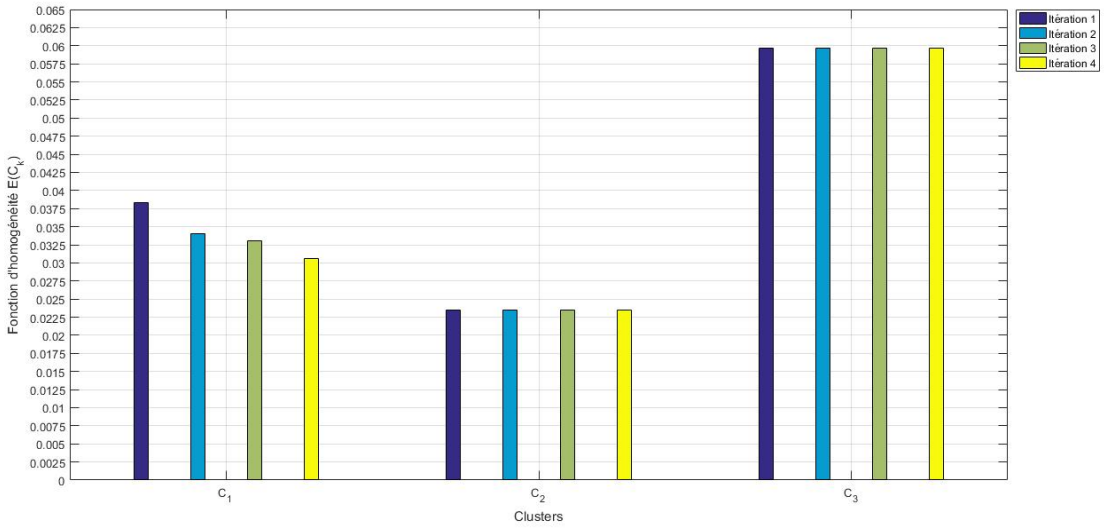

FIGURE 3.4 – Fonction d'homogénéité de chaque cluster - Euclidien - Iris

D'après la figure 3.4, nous remarquons que le cluster C_1 améliore sa fonction d'homogénéité. Par contre, les clusters C_2 et C_3 ne l'améliorent pas.
Nous pouvons résumer notre résultat dans le tableau suivant :

C_1	$\{o_1, o_2, o_3, o_4, o_5, o_6, o_7, o_8, o_9, o_{10}, o_{11}, o_{12}, o_{13}, o_{14}, o_{17}, o_{18}, o_{19}, o_{20}, o_{21}, o_{22}, o_{23}, o_{24}, o_{25}, o_{26}, o_{27}, o_{28}, o_{29}, o_{30}, o_{31}, o_{32}, o_{33}, o_{34}, o_{35}, o_{36}, o_{37}, o_{38}, o_{39}, o_{40}, o_{41}, o_{43}, o_{44}, o_{45}, o_{46}, o_{47}, o_{48}, o_{49}, o_{50}\}$
C_2	$\{o_{52}, o_{54}, o_{55}, o_{56}, o_{57}, o_{59}, o_{60}, o_{62}, o_{63}, o_{64}, o_{65}, o_{66}, o_{67}, o_{68}, o_{69}, o_{70}, o_{71}, o_{72}, o_{73}, o_{74}, o_{75}, o_{76}, o_{79}, o_{80}, o_{81}, o_{82}, o_{83}, o_{84}, o_{85}, o_{86}, o_{87}, o_{88}, o_{89}, o_{90}, o_{91}, o_{92}, o_{93}, o_{94}, o_{95}, o_{96}, o_{97}, o_{98}, o_{100}, o_{107}, o_{114}, o_{120}, o_{122}, o_{127}, o_{128}, o_{139}\}$
C_3	$\{o_{51}, o_{53}, o_{58}, o_{61}, o_{77}, o_{78}, o_{99}, o_{101}, o_{102}, o_{103}, o_{104}, o_{105}, o_{106}, o_{108}, o_{109}, o_{110}, o_{111}, o_{112}, o_{113}, o_{115}, o_{116}, o_{117}, o_{118}, o_{119}, o_{121}, o_{123}, o_{124}, o_{125}, o_{126}, o_{129}, o_{130}, o_{131}, o_{132}, o_{133}, o_{134}, o_{135}, o_{136}, o_{137}, o_{138}, o_{140}, o_{141}, o_{142}, o_{143}, o_{144}, o_{145}, o_{146}, o_{147}, o_{148}, o_{149}, o_{150}\}$
JL	$\{o_{15}, o_{16}, o_{42}\}$

TABLE 3.1 – Clusters stables - Euclidien - Iris

D'après les figures 3.3 et 3.4 et le tableau 3.1, Nous remarquons que les clusters sont bien homogènes avec le nombre des données libres est 3. Ce résultat est accéptable

Similarité de Jaccard :

Après avoir exécuté notre algorithme plusieurs fois, plusieurs valeurs pour K, nous avons obtenu la figure suivante :

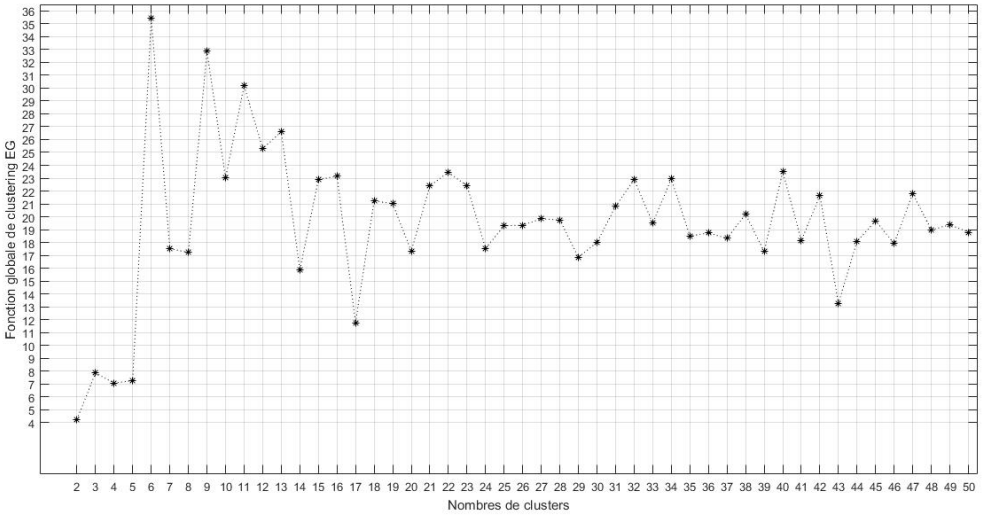

FIGURE 3.5 – Évolution de EG en fonction du nombre de clusters - Jaccard - Iris

Nous remarquons que le nombre de clusters qui maximise la fonction globale est égale à $K = 6$ avec $EG = 35.436$.

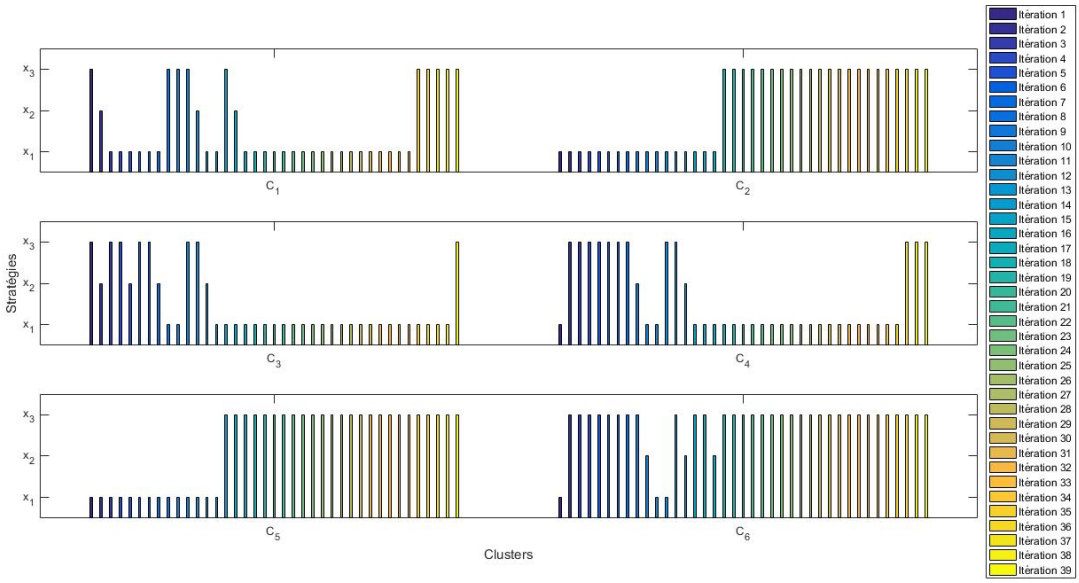

FIGURE 3.6 – Stratégies de chaque cluster - Jaccard - Iris

La figure 3.6 montre les stratégies jouées par chaque cluster dans les 39 itérations.

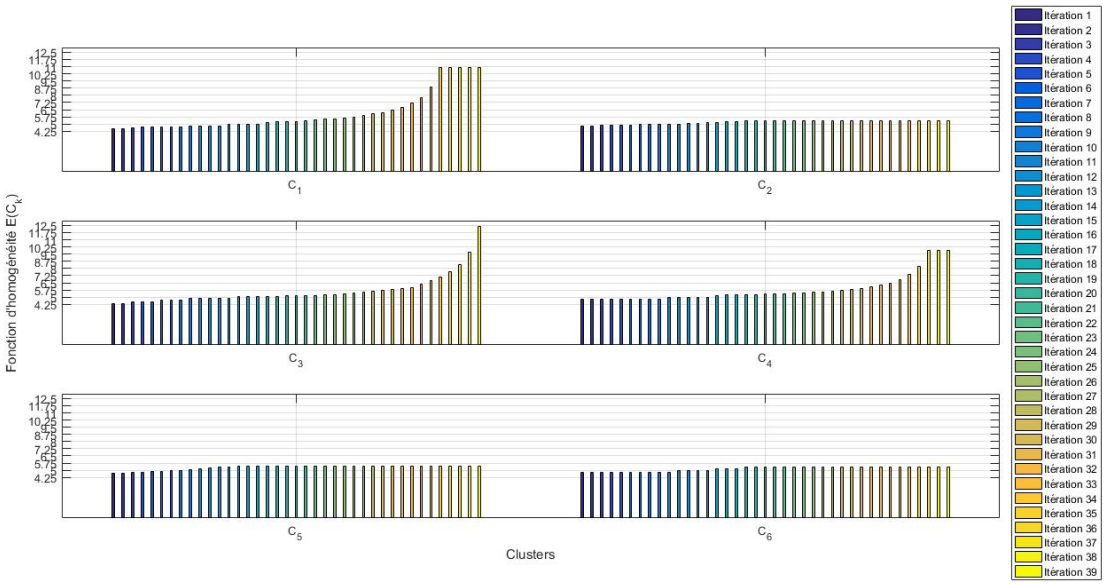

FIGURE 3.7 – Fonction d'homogénéité de chaque cluster - Jaccard - Iris

D'après la figure 3.7, nous remarquons que tous les clusters améliorent leurs fonctions d'homogénéité.

Nous pouvons résumer notre résultat dans le tableau suivant :

C_1	$\{o_{89}, o_{95}, o_{96}, o_{97}, o_{100}\}$
C_2	$\{o_{51}, o_{53}, o_{55}, o_{59}, o_{66}, o_{76}, o_{77}, o_{87}\}$
C_3	$\{o_{84}, o_{102}, o_{128}, o_{139}, o_{143}, o_{150}\}$
C_4	$\{o_1, o_{28}, o_{29}, o_{40}\}$
C_5	$\{o_{103}, o_{106}, o_{108}, o_{110}, o_{118}, o_{123}, o_{126}, o_{130}, o_{131}, o_{132}, o_{136}\}$
C_6	$\{o_2, o_3, o_4, o_6, o_7, o_9, o_{11}, o_{13}, o_{14}, o_{15}, o_{16}, o_{17}, o_{19}, o_{21}, o_{23}, o_{25}, o_{26}, o_{32}, o_{33}, o_{34}, o_{37}, o_{39}, o_{43}, o_{46}, o_{48}, o_{119}, o_{146}, o_{149}\}$
JL	$\{o_5, o_8, o_{10}, o_{12}, o_{18}, o_{20}, o_{22}, o_{24}, o_{27}, o_{30}, o_{31}, o_{35}, o_{36}, o_{38}, o_{41}, o_{42}, o_{44}, o_{45}, o_{47}, o_{49}, o_{50}, o_{52}, o_{54}, o_{56}, o_{57}, o_{58}, o_{60}, o_{61}, o_{62}, o_{63}, o_{64}, o_{65}, o_{67}, o_{68}, o_{69}, o_{70}, o_{71}, o_{72}, o_{73}, o_{74}, o_{75}, o_{78}, o_{79}, o_{80}, o_{81}, o_{82}, o_{83}, o_{85}, o_{86}, o_{88}, o_{90}, o_{91}, o_{92}, o_{93}, o_{94}, o_{98}, o_{99}, o_{101}, o_{104}, o_{105}, o_{107}, o_{109}, o_{111}, o_{112}, o_{113}, o_{114}, o_{115}, o_{116}, o_{117}, o_{120}, o_{121}, o_{122}, o_{124}, o_{125}, o_{127}, o_{129}, o_{133}, o_{134}, o_{135}, o_{137}, o_{138}, o_{140}, o_{141}, o_{142}, o_{144}, o_{145}, o_{147}, o_{148}\}$

TABLE 3.2 – Clusters stables - Jaccard - Iris

D'après les figures 3.6 et 3.7 et le tableau 3.2, nous remarquons que les clusters sont homogènes avec le nombre des données libres est 88. Ce résultat n'est pas accéptable.

D'après les résultats de notre approche dans les cas Euclidien et Jaccard, nous concluons que les ensembles avec un grand nombre de données et un petit nombre d'attributs donnent des bons résultats par rapport la distance euclidienne

3.5.3 Exemple industriel : Danone Algérie

Un exemple industriel, le projet Wrapper Revamping Machine (WRM) du Groupe DANONE, est utilisé pour vérifier les concepts et modèles proposés. La gestion de la communication technique entre les équipes du projet WRM est compliquée en raison du nombre de composants, des équipes, de leurs dépendances et de leurs emplacements. Les équipes techniques sont réparties sur quatre unitété de production et stockage. Le projet WRM comprend 46 activités dirigées par un ingénieur en chef adjoint au siège. Nous avons interrogé 25 personnes, dont le chef de projet, l'équipe de concepteurs et d'autres membres essentiels de l'équipe de projet. Nous avons posé les questions suivantes lors des entretiens :

1. Comment optimiser l'architecture organisationnelle pour coordonner le processus de conception dans les projets de développement distribués ?
2. Comment évaluer la similarité entre les équipes pour répondre aux exigences des sites de délocalisation ?

Les équipes en total forment un nombre de 23. Pour notre algorithme, les équipes de projet représentent les données et leurs activités les attributs. à condition que les attributs ne soient pas nuls sous la forme. (Voir l'annexe B).

Distance Euclidienne :

Après avoir exécuté notre algorithme plusieurs fois, plusieurs valeurs pour K, nous avons obtenu la figure suivante :

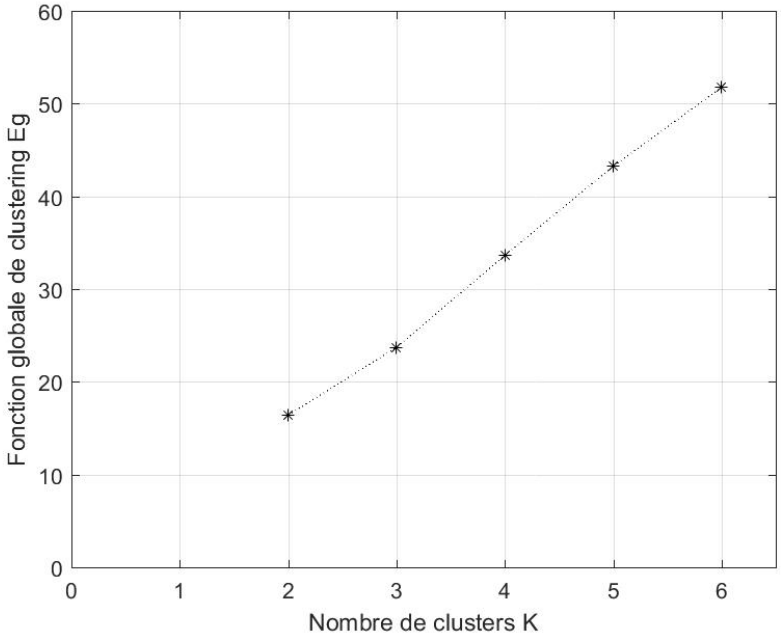

FIGURE 3.8 – Évolution de EG en fonction du nombre de clusters - Euclidien - Exemple industriel

Nous remarquons que le nombre de clusters qui minimise la fonction globale est égale à $K = 2$ avec $EG = 16.3953$.

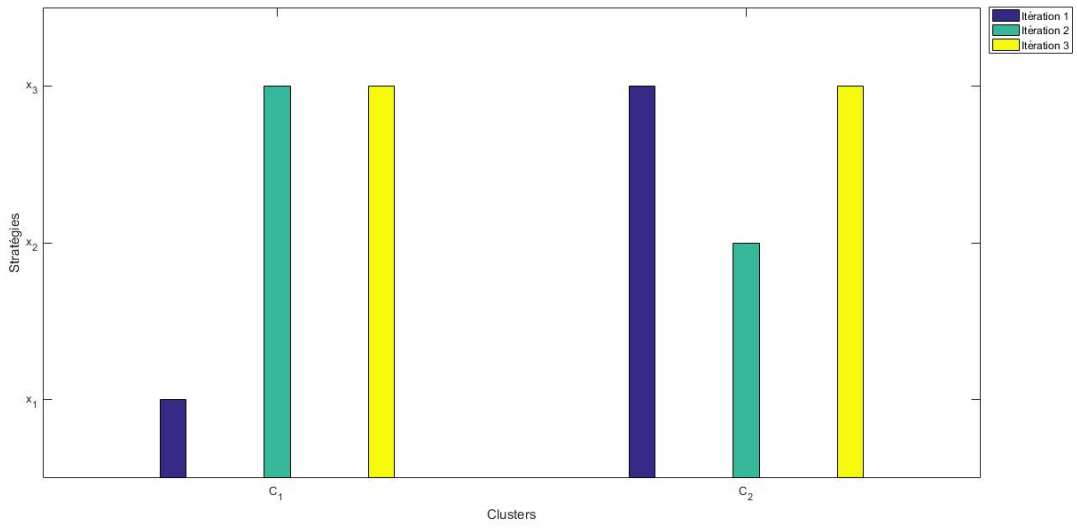

FIGURE 3.9 – Stratégies de chaque cluster - Euclidien - Exemple industriel

La figure 3.9 montre les stratégies jouées par chaque cluster dans les 3 itérations.

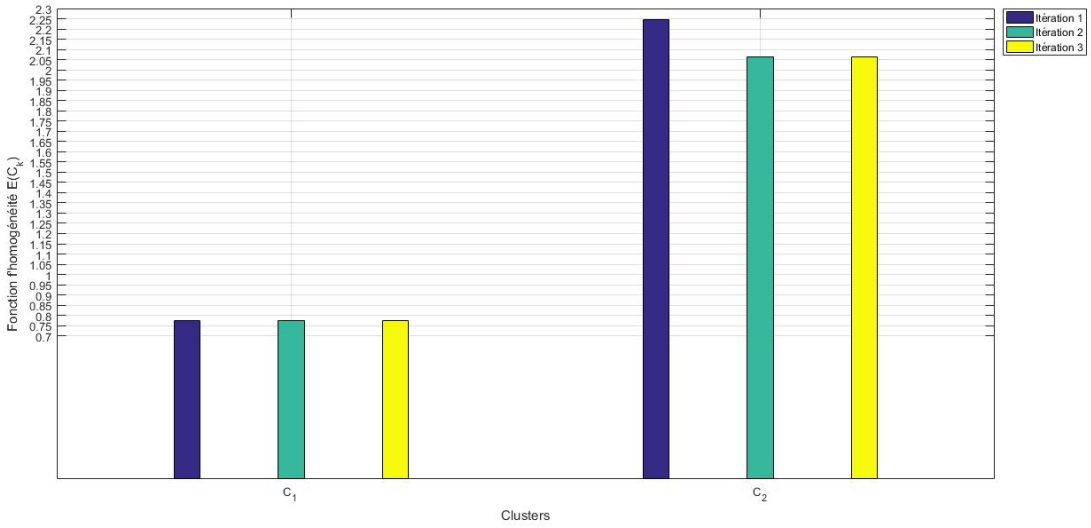

FIGURE 3.10 – Fonction d'homogénéité de chaque cluster - Euclidien - Exemple industriel

D'après la figure 3.10, nous remarquons que les deux clusters C_1 et C_2 améliorent ses fonctions d'homogénéité.
Nous pouvons résumer notre résultat dans le tableau suivant :

C_1	$\{o_1, o_2, o_3, o_4, o_5, o_6, o_7, o_8, o_9\}$
C_2	$\{o_{10}, o_{11}, o_{12}, o_{13}, o_{14}, o_{15}, o_{16}, o_{17}, o_{18}, o_{19}, o_{20}\}$
JL	$\varnothing$

TABLE 3.3 – Clusters stables - Euclidien - Exemple industriel

D'après les figures 3.9 et 3.10 et le tableau 3.3, nous remarquons que les clusters ne sont pas homogènes avec le nombre des données libres égales à 0. Ce résultat n'est pas accéptable.

Similarité de Jaccard :

Après avoir exécuté notre algorithme plusieurs fois (plusieurs valeurs pour K) nous avons obtenu la figure suivante :

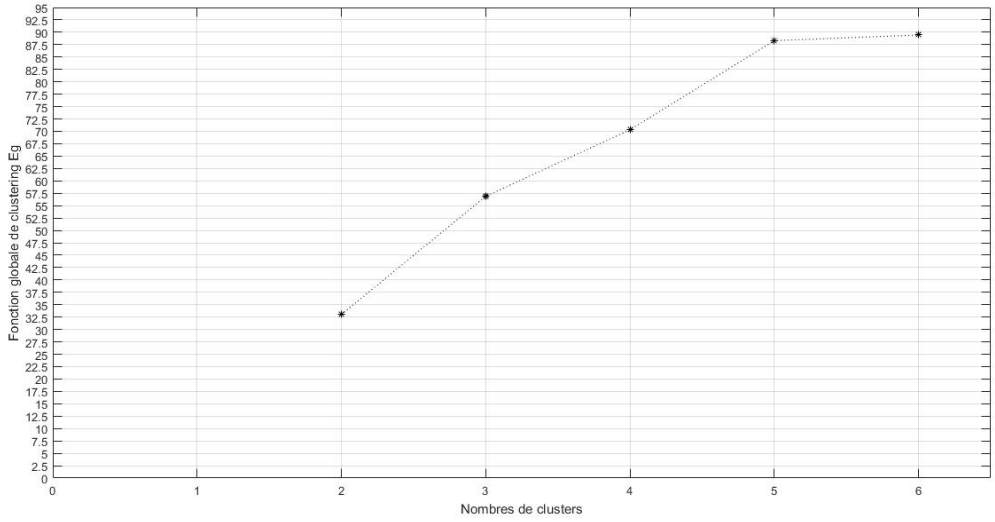

FIGURE 3.11 – Évolution de EG en fonction du nombre de clusters - Jaccard - Exemple industriel

Nous remarquons que le nombre de clusters qui maximise la fonction globale est égale à $K = 6$ avec $EG = 89.4467$.

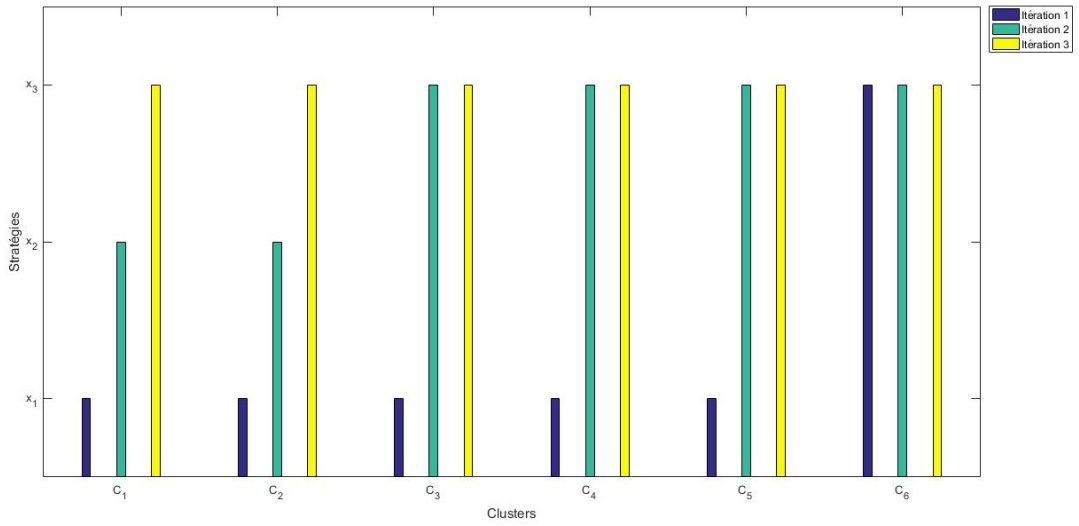

FIGURE 3.12 – Stratégies de chaque cluster - Jaccard - Exemple industriel

La figure 3.12 montre les stratégies jouées par chaque cluster dans les 3 itérations.

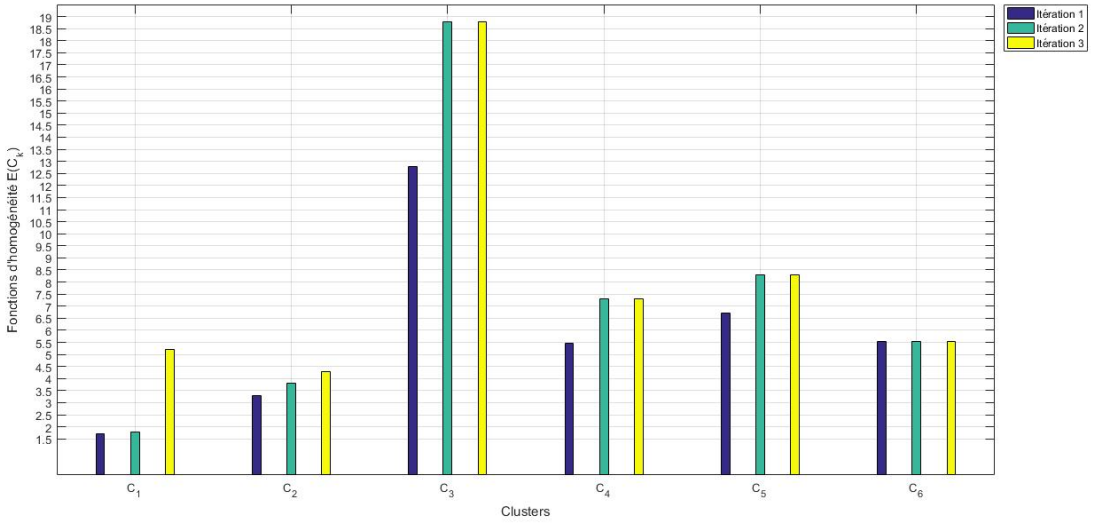

FIGURE 3.13 – Fonction d'homogénéité de chaque cluster - Jaccard - Exemple industeriel

D'après le figure 3.13, nous remarquons que tous les clusters améliorent leurs fonctions d'homogénéité sauf le cluster C_6 qui ne l'améliore pas.

Nous pouvons résumer notre résultat dans le tableau suivant :

C_1	$\{o_1, o_2, o_4, o_{19}\}$
C_2	$\{o_7, o_8, o_{10}, o_{15}\}$
C_3	$\{o_{18}, o_{20}\}$
C_4	$\{o_{13}, o_{14}\}$
C_5	$\{o_{12}, o_{16}\}$
C_6	$\{o_3, o_9, o_{17}\}$
JL	$\{o_5, o_6, o_{11}\}$

TABLE 3.4 – Clusters stables - Jaccard - Exemple industriel

D'après les figures 3.12 et 3.13 et le tableau 3.4, nous remarquons que les clusters sont homogènes avec le nombre des données libres est 3. Ce résultat est accéptable.

D'après les résultats de notre approche dans les cas Euclidien et Jaccard, nous concluons que les ensembles avec un petit nombre de données et un grand nombre d'attributs donnent des bons résultats par rapport la distance euclidienne

conclusion

Dans ce chapitre nous avons proposé une nouvelle approche du problème de clustering. En effet, nous avons modélisé le problème de clustering sous forme d'un jeu non coopératif. Ensuite, nous l'avons résolu en utilisant deux mesures différentes sur deux ensembles de données différents. Nous avons appliqué notre algorithme sous MATLAB. Après l'obtention les résultats, nous avons comparé les résultats.

Conclusion Générale

Dans le cadre de ce mémoire, nous avons essayé d'apporter des solutions aux problèmes de clusterisation des données. Pour y parvenir nous nous sommes basé sur la théorie des jeux comme outil de modélisation et d'aide à la décision.

L'évaluation de la qualité d'un algorithme de clustering reste un problème ouvert. Il n'existe aucune approche reconnue comme étant universellement fiable, et aucune méthode ne peut être qualifiée de meilleure par rapport à une autre dans tout les contextes et les résultats obtenus sont à relativiser.

les résultats de l'expérimentation de notre algorithme sur une base de données réelle, dont les résultats sont connus à priori, ont permis de constater une bonne capacité prédictive de notre approche. Dans le cas Euclidien, il donne des résultats assez satisfaisant dans la base d'IRIS comparativement à l'autre base de données. Et contrairement dans le cas jaccard. Nous avons proposé aussi une technique pour déterminer le nombre de clusters.

En guise de perspectives, nous envisageons d'axer les travaux futurs sur l'amélioration de notre algorithme :

- En incluant le cas de données manquantes qui se présente souvent dans la réalité ;
- En étendant notre approche aux autres types d'attributs autres que numériques ;
- En étudiant la possibiilté d'adapter notre algorithme aux grands nombres de données à haute dimension.

A

Ensembles d'IRIS

	a_1	a_2	a_3	a_4
o_1	5.1	3.5	1.4	0.2
o_2	4.9	3	1.4	0.2
o_3	4.7	3.2	1.3	0.2
o_4	4.6	3.1	1.5	0.2
o_5	5	3.6	1.5	0.2
o_6	5.4	3.9	1.7	0.4
o_7	4.6	3.4	1.4	0.3
o_8	5	3.4	1.5	0.2
o_9	4.4	2.9	1.4	0.2
o_{10}	4.9	3.1	1.5	0.1
o_{11}	5.4	3.7	1.5	0.2
o_{12}	4.8	3.4	1.6	0.2
o_{13}	4.8	3	1.4	0.1
o_{14}	4.3	3	1.1	0.1
o_{15}	5.8	4	1.2	0.2
o_{16}	5.7	4.4	1.5	0.4
o_{17}	5.4	3.9	1.3	0.4
o_{18}	5.1	3.5	1.4	0.3
o_{19}	5.7	3.8	1.7	0.3
o_{20}	5.1	3.8	1.5	0.3
o_{21}	5.4	3.4	1.7	0.2
o_{22}	5.1	3.7	1.5	0.4
o_{23}	4.6	3.6	1	0.2
o_{24}	5.1	3.3	1.7	0.5
o_{25}	4.8	3.4	1.9	0.2
o_{26}	5	3	1.6	0.2
o_{27}	5	3.4	1.6	0.4
o_{28}	5.2	3.5	1.5	0.2
o_{29}	5.2	3.4	1.4	0.2
o_{30}	4.7	3.2	1.6	0.2
o_{31}	4.8	3.1	1.6	0.2
o_{32}	5.4	3.4	1.5	0.4
o_{33}	5.2	4.1	1.5	0.1
o_{34}	5.5	4.2	1.4	0.2
o_{35}	4.9	3.1	1.5	0.1
o_{36}	5	3.2	1.2	0.2
o_{37}	5.5	3.5	1.3	0.2
o_{38}	4.9	3.1	1.5	0.1
o_{39}	4.4	3	1.3	0.2
o_{40}	5.1	3.4	1.5	0.2
o_{41}	5	3.5	1.3	0.3
o_{42}	4.5	2.3	1.3	0.3
o_{43}	4.4	3.2	1.3	0.2
o_{44}	5	3.5	1.6	0.6
o_{45}	5.1	3.8	1.9	0.4
o_{46}	4.8	3	1.4	0.3
o_{47}	5.1	3.8	1.6	0.2
o_{48}	4.6	3.2	1.4	0.2
o_{49}	5.3	3.7	1.5	0.2
o_{50}	5	3.3	1.4	0.2
o_{51}	7	3.2	4.7	1.4
o_{52}	6.4	3.2	4.5	1.5
o_{53}	6.9	3.1	4.9	1.5
o_{54}	5.5	2.3	4	1.3
o_{55}	6.5	2.8	4.6	1.5
o_{56}	5.7	2.8	4.5	1.3
o_{57}	6.3	3.3	4.7	1.6
o_{58}	4.9	2.4	3.3	1
o_{59}	6.6	2.9	4.6	1.3
o_{60}	5.2	2.7	3.9	1.4
o_{61}	5	2	3.5	1
o_{62}	5.9	3	4.2	1.5
o_{63}	6	2.2	4	1
o_{64}	6.1	2.9	4.7	1.4
o_{65}	5.6	2.9	3.6	1.3
o_{66}	6.7	3.1	4.4	1.4
o_{67}	5.6	3	4.5	1.5

o_{68}	5.8	2.7	4.1	1
o_{69}	6.2	2.2	4.5	1.5
o_{70}	5.6	2.5	3.9	1.1
o_{71}	5.9	3.2	4.8	1.8
o_{72}	6.1	2.8	4	1.3
o_{73}	6.3	2.5	4.9	1.5
o_{74}	6.1	2.8	4.7	1.2
o_{75}	6.4	2.9	4.3	1.3
o_{76}	6.6	3	4.4	1.4
o_{77}	6.8	2.8	4.8	1.4
o_{78}	6.7	3	5	1.7
o_{79}	6	2.9	4.5	1.5
o_{80}	5.7	2.6	3.5	1
o_{81}	5.5	2.4	3.8	1.1
o_{82}	5.5	2.4	3.7	1
o_{83}	5.8	2.7	3.9	1.2
o_{84}	6	2.7	5.1	1.6
o_{85}	5.4	3	4.5	1.5
o_{86}	6	3.4	4.5	1.6
o_{87}	6.7	3.1	4.7	1.5
o_{88}	6.3	2.3	4.4	1.3
o_{89}	5.6	3	4.1	1.3
o_{90}	5.5	2.5	4	1.3
o_{91}	5.5	2.6	4.4	1.2
o_{92}	6.1	3	4.6	1.4
o_{93}	5.8	2.6	4	1.2
o_{94}	5	2.3	3.3	1
o_{95}	5.6	2.7	4.2	1.3
o_{96}	5.7	3	4.2	1.2
o_{97}	5.7	2.9	4.2	1.3
o_{98}	6.2	2.9	4.3	1.3
o_{99}	5.1	2.5	3	1.1
o_{100}	5.7	2.8	4.1	1.3
o_{101}	6.3	3.3	6	2.5
o_{102}	5.8	2.7	5.1	1.9
o_{103}	7.1	3	5.9	2.1
o_{104}	6.3	2.9	5.6	1.8
o_{105}	6.5	3	5.8	2.2
o_{106}	7.6	3	6.6	2.1
o_{107}	4.9	2.5	4.5	1.7
o_{108}	7.3	2.9	6.3	1.8
o_{109}	6.7	2.5	5.8	1.8

o_{110}	7.2	3.6	6.1	2.5
o_{111}	6.5	3.2	5.1	2
o_{112}	6.4	2.7	5.3	1.9
o_{113}	6.8	3	5.5	2.1
o_{114}	5.7	2.5	5	2
o_{115}	5.8	2.8	5.1	2.4
o_{116}	6.4	3.2	5.3	2.3
o_{117}	6.5	3	5.5	1.8
o_{118}	7.7	3.8	6.7	2.2
o_{119}	7.7	2.6	6.9	2.3
o_{120}	6	2.2	5	1.5
o_{121}	6.9	3.2	5.7	2.3
o_{122}	5.6	2.8	4.9	2
o_{123}	7.7	2.8	6.7	2
o_{124}	6.3	2.7	4.9	1.8
o_{125}	6.7	3.3	5.7	2.1
o_{126}	7.2	3.2	6	1.8
o_{127}	6.2	2.8	4.8	1.8
o_{128}	6.1	3	4.9	1.8
o_{129}	6.4	2.8	5.6	2.1
o_{130}	7.2	3	5.8	1.6
o_{131}	7.4	2.8	6.1	1.9
o_{132}	7.9	3.8	6.4	2
o_{133}	6.4	2.8	5.6	2.2
o_{134}	6.3	2.8	5.1	1.5
o_{135}	6.1	2.6	5.6	1.4
o_{136}	7.7	3	6.1	2.3
o_{137}	6.3	3.4	5.6	2.4
o_{138}	6.4	3.1	5.5	1.8
o_{139}	6	3	4.8	1.8
o_{140}	6.9	3.1	5.4	2.1
o_{141}	6.7	3.1	5.6	2.4
o_{142}	6.9	3.1	5.1	2.3
o_{143}	5.8	2.7	5.1	1.9
o_{144}	6.8	3.2	5.9	2.3
o_{145}	6.7	3.3	5.7	2.5
o_{146}	6.7	3	5.2	2.3
o_{147}	6.3	2.5	5	1.9
o_{148}	6.5	3	5.2	2
o_{149}	6.2	3.4	5.4	2.3
o_{150}	5.9	3	5.1	1.8

B
Ensemble WRM

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
o_1	1	1	0,2	1	0,89	1	0,2	0,2	0,89	0,2
o_2	1	2	0,2	1	0,89	1	0,2	0,2	0,89	0,2
o_3	0,2	0,2	3	0,2	0,26	0,2	0	0	0,26	0
o_4	1	1	0,2	4	0,89	1	0,2	0,2	0,89	0,2
o_5	0,89	0,89	0,26	0,89	5	0,89	0,26	0,26	1	0,26
o_6	1	1	0,2	1	0,89	6	0,2	0,2	0,89	0,2
o_7	0,2	0,2	1	0,2	0,26	0,2	7	1	0,26	1
o_8	0,2	0,2	1	0,2	0,26	0,2	1	8	0,26	1
o_9	0,26	0,26	0,26	0,26	1	0,26	0,26	0,26	9	0,26
o_{10}	0,2	0,2	1	0,2	0,26	0,2	1	1	0,26	10
o_{11}	0,2	0,2	1	0,2	0,26	0,2	1	1	0,26	1
o_{12}	1	1	0,2	1	0,89	1	0,2	0,2	0,89	0,2
o_{13}	0,2	0,2	1	0,2	0,26	0,2	1	1	0,26	1
o_{14}	0,2	0,2	1	0,2	0,26	0,2	1	1	0,26	1
o_{15}	0,2	0,2	1	0,2	0,26	0,2	1	1	0,26	1
o_{16}	1	1	0,2	1	0,89	1	0,2	0,2	0,89	0,2
o_{16}	1	1	0,2	1	0,89	1	0,2	0,2	0,89	0,2
o_{17}	0,2	0,2	1	0,2	0,26	0,2	1	1	0,26	1
o_{18}	1	1	0,2	1	0,89	1	0,2	0,2	0,89	0.2
o_{19}	0,89	0,89	0,26	0,89	1	0,89	0,26	0,26	1	0,26
o_{20}	1	1	0,2	1	0,89	1	0,2	0,2	0,89	0,2

	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}	a_{19}	a_{20}
o_1	0,2	1	0,2	0,2	0,2	1	0,2	1	0,89	1
o_2	0,2	1	0,2	0,2	0,2	1	0,2	1	0,89	1
o_3	0	0,2	0	0	0	0,2	0	0,2	0,26	0,2
o_4	0,2	1	0,2	0,2	0,2	1	0,2	1	0,89	1
o_5	0,26	0,89	0,26	0,26	0,26	0,89	0,26	0,89	1	0,89
o_6	0,2	1	0,2	0,2	0,2	1	0,2	1	0,89	1
o_7	1	0,2	1	1	1	0,2	1	0,2	0,26	0,2
o_8	1	0,2	1	1	1	0,2	1	0,2	0,26	0,2
o_9	0,26	0,26	0,26	0,26	0,26	0,26	0,26	0,26	1	0,26
o_{10}	1	0,2	1	1	1	0,2	1	0,2	0,26	0,2
o_{11}	11	0,2	1	1	1	0,2	1	0,2	0,26	0,2
o_{12}	0,2	12	0,2	0,2	0,2	1	0,2	1	0,89	1
o_{13}	1	0,2	13	1	1	0,2	1	0,2	0,26	0,2
o_{14}	1	0,2	1	14	1	0,2	1	0,2	0,26	0,2
o_{15}	1	0,2	1	1	15	0,2	1	0,2	0,26	0,2
o_{16}	0,2	1	0,2	0,2	0,2	16	0,2	1	0,89	1
o_{17}	1	0,2	1	1	1	0,2	17	0,2	0,26	0,2
o_{18}	0,2	1	0,2	0,2	0,2	1	0,2	18	0,89	18
o_{19}	0,26	0,89	0,26	0,26	0,26	0,89	0,26	0,89	19	0,89
o_{20}	0,2	1	0,2	0,2	0,2	1	0,2	1	0,89	20

Bibliographie

[Agrawal et al., 1998] Agrawal, R., Gehrke, J., Gunopulos, D., and Raghavan, P. (1998). *Automatic subspace clustering of high dimensional data for data mining applications*, volume 27. ACM.

[Anderberg, 1973] Anderberg, M. (1973). Cluster analysis for researchers. *New York*.

[Ball and Hall, 1967] Ball, G. H. and Hall, D. J. (1967). A clustering technique for summarizing multivariate data. *Behavioral science*, 12(2) :153–155.

[Davies and Bouldin, 1979] Davies, D. L. and Bouldin, D. W. (1979). A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2) :224–227.

[Day and Edelsbrunner, 1984] Day, W. H. and Edelsbrunner, H. (1984). Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of classification*, 1(1) :7–24.

[Desgraupes, 2013] Desgraupes, B. (2013). Clustering indices. *University of Paris Ouest-Lab Modal'X*, 1 :34.

[D'Hondt Frédéric, 2004] D'Hondt Frédéric, E. K. B. (2004). Etude de méthodes de clustering pour la segmentation d'images en couleurs. *Faculté Polytechnique de Mons, 5ème Electricité, Certificat Applicatifs Multimédia*.

[Dice, 1945] Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3) :297–302.

[Dunn, 1974] Dunn, J. C. (1974). Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics*, 4(1) :95–104.

[Elghazel, 2007] Elghazel, H. (2007). *Classification et Prévision des données hétérogenes : Application aux trajectoires et séjours hospitaliers*. PhD thesis, Lyon 1.

[Ester et al., 1995] Ester, M., Kriegel, H.-P., and Xu, X. (1995). Knowledge discovery in large spatial databases : Focusing techniques for efficient class identification. In *International Symposium on Spatial Databases*, pages 67–82. Springer.

[Forestier, 2010] Forestier, G. (2010). *Connaissances et clustering collaboratif d'objets complexes multisources*. PhD thesis, Strasbourg.

[George et al., 1999] George, K., Han, E.-H., and Kumar, V. (1999). Chameleon : a hierarchical clustering algorithm using dynamic modeling. *IEEE computer*, 27(3) :329–341.

[Hamidouche and Idjeraoui, 2013] Hamidouche, S. and Idjeraoui, T. (2013). Clustering : Approche par la théorie des jeux. *Mémoire de Master, Université Abderhmane Mira de Béjaia*.

[Heller and Ghahramani, 2005] Heller, K. A. and Ghahramani, Z. (2005). Bayesian hierarchical clustering. In *Proceedings of the 22nd international conference on Machine learning*, pages 297–304. ACM.

[Heloulou, 2017] Heloulou, I. (2017). *Application de la théorie des jeux dans les problème de clustering multicritères*. PhD thesis.

[Huang, 1998] Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, 2(3) :283–304.

[Jaccard, 1908] Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. *Bull. Soc. Vaud. Sci. Nat.*, 44 :223–270.

[Jain et al., 1988] Jain, A. K., Dubes, R. C., et al. (1988). *Algorithms for clustering data*, volume 6. Prentice hall Englewood Cliffs.

[Kaufman and Rousseeuw, 1990] Kaufman, L. and Rousseeuw, P. J. (1990). *Finding groups in data : an introduction to cluster analysis*, volume 344. John Wiley & Sons.

[Kulczynski, 1927] Kulczynski, S. (1927). Classe des sciences mathématiques et naturelles. *Bulletin International de l'Academie Polonaise des Sciences et des Lettres*, pages 57–203.

[Lebbah et al., 2005] Lebbah, M., Chazottes, A., Badran, F., and Thiria, S. (2005). Mixed topological map. In *ESANN*, volume 17, page 47.

[MacQueen et al., 1967] MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

[Nash, 1950] Nash, J. F. (1950). Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1) :48–49.

[Ng et al., 2007] Ng, M. K., Li, M. J., Huang, J. Z., and He, Z. (2007). On the impact of dissimilarity measure in k-modes clustering algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3) :503–507.

[Ochiai, 1957] Ochiai, A. (1957). Zoogeographic studies on the soleoid fishes found in japan and its neighbouring regions. *Bulletin of Japanese Society of Scientific Fisheries*, 22 :526–530.

[Osborne and Rubinstein, 1994] Osborne, M. J. and Rubinstein, A. (1994). *A course in game theory*. MIT press.

[Rai and Singh, 2010] Rai, P. and Singh, S. (2010). A survey of clustering techniques. *International Journal of Computer Applications*, 7(12) :1–5.

[Rousseeuw, 1987] Rousseeuw, P. J. (1987). Silhouettes : a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20 :53–65.

[Sabri, 2011] Sabri, S. (2011). Application de la théorie des jeux pour la définition, développement et implémentation d'un algorithme de clustering. *Mémoire de Magistère, Université Abderhmane Mira de Béjaia*.

[Sneath et al., 1973] Sneath, P. H., Sokal, R. R., et al. (1973). *Numerical taxonomy. The principles and practice of numerical classification.*

[Sørensen, 1948] Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Biol. Skr.*, 5 :1–34.

[Wang et al., 1997] Wang, W., Yang, J., Muntz, R., et al. (1997). Sting : A statistical information grid approach to spatial data mining. In *VLDB*, volume 97, pages 186–195.

[Wemmert et al., 2000] Wemmert, C., Gançarski, P., and Korczak, J. J. (2000). A collaborative approach to combine multiple learning methods. *International Journal on Artificial Intelligence Tools*, 9(01) :59–78.

[Wu, 2012] Wu, J. (2012). *Advances in K-means clustering : a data mining thinking*. Springer Science & Business Media.

Résumé *: L'analyse de clustering dans une dynamique de jeu non coopératif.*

Dans le cadre de ce mémoire, nous nous sommes intéressés à la résolution des problèmes de clustering de données par la théorie des jeux. Nous avons donc proposé une nouvelle approche. Le problème de clustering est modélisé comme un jeu séquentiel non coopératif. L'algorithme proposé est ensuite implémenté, testé sur deux différentes bases de données réelles et comparé les résultats. La qualité des résultats obtenus montrent la pertinence et l'éfficacité de l'approche choisie.
Mots clés : *Clustering, théorie des jeux, jeux non coopératifs.*

Abstract *: Clustering analysis in a game dynamic non-cooperative.*

As part of this memoir, we are interested in solving data clustering problems through game theory. So we proposed a new approach. The clustering problem is modeled as a non-cooperative sequential game. The proposed algorithm is then implemented, tested on two different real databases and compared with each other. The quality of the results obtained shows the relevance and effectiveness of the chosen approach.

Keywords : *Clustering, game theory, noncooperative game.*