+

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE 20 AOUT 1955 SKIKDA
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

Mémoire de fin d'étude en vue de l'obtention du Diplôme de

Master en Informatique

Spécialité : Réseaux et systèmes distribués

THEME

CLUTERING NON SUPERVISE PAR APPROCHE DU CHAMP POTENTIEL.

Présenté par : *M. IGHIL Sami.*

Dirigé par :

M. LAROUM Toufik, enseignant Université 20 août 1955 skikda.

Devant le Jury :

M. BOUCHEHAM Bachir, professeur Université 20 août 1955 skikda.

M. MAZOUZI Smaine, professeur Université 20 août 1955 skikda.

Juillet 2024

Ce thème est dédié

à ma mère et à mon père,

à mes trois enfants Ishak, Eline et Yakoub et mon épouse,

à ma sœur, son marie et ses enfants Racim et Ania,

à mon oncle paternel, sa femme qui est très proche à mon cœur et ses enfants Fatima Zohra, Lina Malek et mon petit frère Iskandar.

Remerciement

Tous mes remerciements à ceux qui m'ont aidé pour arriver à terme de ce thème, et en particulier, le Professeur Smaine Mazouzi par sa personnalité et ces qualités exceptionnelles, reste mes mots ne peuvent pas d'écrire la personne. Le professeur Bachir Boucheham pour l'écoute et la pensée critique et philosophique digne d'un grand professeur, le mercredi pour moi représente une journée ou le savoir est au premier rond. Le docteur Djamel Zeghida pour les discussions et débats académiques qui m'ont poussé a donné de mon mieux. Toufik Laroum un ami qui est très proche, plus qu'un encadreur ou binôme un frère.

Je tiens également à exprimer ma sincère gratitude à tous les enseignants du département informatique, pour l'écoute, la collaboration et la contribution envers notre réussite, est surtout le respect en matière de comportement. Un plaisir de revenir après temps d'année.

Merci.

ملخص

تهتم هذه الاطروحة الي التطرق لأحد أكبر الفروع الهامة في التعلم الالي والتنقيب عن البيانات، وهو التصنيف غير الخاضع للإشراف أو بالتحديد التجميع الذاتي، هذا النوع من التصنيف يهدف الى تجميع واكتشاف البنية الكامنة والأنماط في البيانات بطريقة يستعمل فيها نوع من الذكاء الاصطناعي، بحيث تكون البيانات المتشابهة الى حد كبير في نفس المجموعة، ومن ناحية اخري تكون فيه المجموعات المختلفة متباينة قدر الإمكان. على خلاف التجميع أو التحديد غير الذاتي أين يعرف فيه مسبقا عدد التجمعات، وهنا يكمن مربط الخيل والاشكال في التصنيف الجيد. أبرزها خوارزمية K-Means التي لا يمكننا ان لا نذكرها لشيوعها وشعبيتها.

معظم التقنيات والخوارزميات يتم تصنيف البيانات فيها إلى مجموعات بناءً على التشابه بين عناصرها وحساب المسافات اعتمادا علي بحر من المفاهيم في الرياضيات والهندسة. بدءا بالمسافة الإقليدية التي مفهومها بسيط لكن تطبيقها قوي، وصولا في الأعوام الأخيرة وظهور مقاييس جديدة للمسافات في كل من مجالات الرياضيات وعلوم البيانات، من أبرزها مسافة فاسيرشتاين، والمعروفة أيضًا باسم مسافة النقل الأمثل أومترية كانتوروفيتش روبنشتاين. هذا المقياس يقدم طريقة لقياس المسافة بين التوزيعات الاحتمالية ويستخدم بشكل واسع في التعلم الآلي ومعالجة الصور.

التصنيف غير الخاضع للإشراف لا يعتمد فقط على ما تم سرده سابقا، ففلسفة التجميع اعتمادا على حساب أوجه التشابه والمسافات تطبيقاته تتعلق بطبيعة المعطيات، وهذا ما يشكل ضعف كبير. ما ارضناه من خلال هذه الاطروحة هو تطبيق مفهوم مغاير لحد ما وهو مفهوم حقل الجهد (أو المجال الكامن)، وهو من المفاهيم الأساسية في الفيزياء. هذا مفهوم يساعد على وصف التفاعلات عن بُعد بين الأجسام بدون الحاجة إلى تفصيل القوى المؤثرة مباشرة. مجالات استخدامه متعددة مثل الكهرباء الساكنة والجاذبية، ويوفر طريقة أنيقة ورياضية دقيقة لتحليل مختلف القوى والطاقة في الانظمة.

بعد تطبيقنا لمفهوم حقل الجهد الشيء الملاحظ هو التفاعل، التجمع والتجاذب المثير لمجموعة البيانات المتشابهة لحد ما من حول بعضها، وما هذا العمل والنتائج المتوصل اليها سوي بداية لرؤية مختلفة ونمط مغاير في هذا الميدان الساحر، كمثال اخر ورؤية اخري، لا يمكننا ان نعبر دون ذكر فزياء الكم وتطبيقاتها التي تخرج عن الواقع، والتي سوف تحدث قفزة كبيرة في هذا المجال. في الأخير نرجو ان يكون العمل والنتائج المتحصل عليها لبنة لمشاريع مستقبلية يكون فيها التصنيف غير الخاضع للإشراف من المشاكل التي تم التوصل الى حلها على صعيد واسع.

Résumé

Ce thème s'intéresse à l'un des domaines les plus importants de l'apprentissage automatique et de la fouille de données, à savoir le clustering non supervisé, ce type de clustering vise à découvrir les structures sous-jacentes qui sont dissipées dans les données en utilisant une forme d'intelligence artificielle, de sorte que les données très similaires se retrouvent dans le même groupe, tandis que les groupes qui différent sont aussi distincts que possible. Cela diffère du clustering ou de la classification supervisée où le nombre de groupes est connu à l'avance, ce qui constitue un défi pour un bon clustering. L'un des algorithmes les plus connus dans ce domaine est l'algorithme K-Means, que nous ne pouvons pas omettre en raison de sa popularité et de son utilisation répandue.

La plupart des techniques et algorithmes classent les données en groupes s'appuyant sur la similarité entre leurs éléments et le calcul des distances, en se basant sur un large éventail de concepts en mathématiques et en ingénierie. Cela commence par la distance euclidienne, dont le concept est simple, mais l'application est puissante, et se poursuit avec l'émergence de nouvelles mesures de distances dans les domaines des mathématiques et des sciences des données, telles que la distance de Wasserstein, également connue sous le nom de distance de transport optimal aussi sous le nom de métrique de Kantorovich-Rubinstein. Cette mesure offre une méthode pour mesurer la distance entre des distributions de probabilités, elle est largement utilisée dans l'apprentissage automatique et dans le traitement d'images.

Le clustering non supervisé ne repose pas uniquement sur les concepts mentionnés ci-dessus, la philosophie du clustering basée sur le calcul des similarités et des distances dépend de la nature des données, ce qui constitue une grande faiblesse. Ce que nous cherchons à réaliser à travers ce thème, c'est l'application d'un nouveau concept, à savoir le concept du champ potentiel, qui est l'un des concepts fondamentaux en physique. Ce concept aide à décrire les interactions à distance entre les objets sans avoir besoin de détailler les forces directement influentes. Il est utilisé dans divers domaines comme l'électricité statique et la gravité, il offre une méthode élégante et mathématiquement précise pour analyser les différentes forces et énergies dans les systèmes.

Après avoir appliqué le concept du champ potentiel, nous avons observé l'interaction, le regroupement et l'attraction intrigants des ensembles de données similaires autour les uns des autres. Le travail et les résultats obtenus ne sont que le début d'une vision différente et une approche nouvelle dans ce domaine fascinant. À titre d'exemple, aussi, nous évoquons comme approche captivante et innovante que nous ne pouvons pas passer sans mentionner la physique quantique et ses applications, qui dépassent la réalité et qui provoqueront une avancée majeure dans ce domaine. En fin de compte, nous espérons que ce travail et les résultats obtenus seront une pierre angulaire pour des projets futurs où le clustering non supervisé sera un problème largement résolu.

Abstract

This theme focuses on one of the most important areas of machine learning and data mining, namely unsupervised clustering. This type of clustering aims to discover the underlying structures scattered in the data using a form of artificial intelligence, so that highly similar data ends up in the same group, while different groups are as distinct as possible. This differs from supervised clustering or classification where the number of groups is known in advance, posing a challenge for proper clustering.

One of the most well-known algorithms in this field is the K-Means algorithm, which we cannot omit due to its popularity and widespread use. Most techniques and algorithms classify data into groups based on the similarity between their elements and distance calculations, relying on a wide range of concepts in mathematics and engineering. This starts with the Euclidean distance, whose concept is simple but application powerful, and continues with the emergence of new distance measures in mathematics and data science, such as the Wasserstein distance, also known as the optimal transport distance or the Kantorovich-Rubinstein metric. This measure provides a method for measuring the distance between probability distributions and is widely used in machine learning and image processing.

Unsupervised clustering does not rely solely on the concepts mentioned above, the clustering philosophy based on similarity and distance calculations depends on the nature of the data, which is a major weakness. What we seek to achieve through this theme is the application of a new concept, namely the concept of potential field, which is one of the fundamental concepts in physics. This concept helps to describe remote interactions between objects without the need to detail directly influencing forces. It is used in various fields such as static electricity and gravity; it provides an elegant and mathematically precise method for analyzing different forces and energies in systems.

After applying the concept of the potential field, we observed intriguing interactions, clustering, and attraction among similar data sets. This work and the results obtained represent only the beginning of a different vision and a new approach in this fascinating field. As an example, we cannot overlook the captivating and innovative approach of quantum physics and its applications, which transcend reality and will lead to major advancements in this domain. Ultimately, we hope that this work and the results obtained will serve as a cornerstone for future projects, where unsupervised clustering will become a widely resolved problem.

Table des matières

Table des figures

Liste des tableaux

Introduction générale

Au cours de ces dernières années, il a été constaté une augmentation fulgurante de la capacité à recueillir des données issues d'un ensemble varié de dispositifs, capteurs, appareils et en différents formats. L'apport de données étant tellement fort et important qu'il a dépassé au large les antiques technologiques, et ce, sur plusieurs plans à savoir le traitement, l'analyse, le stockage et surtout leurs compréhensions. Cet essor fulgurant accéléré encore plus par l'utilisation des réseaux sociaux qui vont permettre aux usagers un ajout inconditionnel d'informations (textes, images, vidéos, etc.) sur le web et ceci va encore accentuer sa montée en puissance. Dou la naissance du phénomène nommé le Big Data, en conséquence, nous pouvons dire pour déduction que le phénomène a changé d'une manière radicale, la manière de gérer des données, car il introduit de nouvelles problématiques concernant la volumétrie, la vitesse de transfert et le type de données.

Le développement et la croissance assez rapide des données collectées dans les bases de données avec la nécessité d'une réactivité efficace vont avoir pour conséquence le développement du Knowledge Discovery in Data base en anglais, processus non-trivial d'identification de structures inconnues, valides et potentiellement utiles dans les bases de données. Son but est essentiellement de venir en aide à l'être humain pour extraire des informations utiles (connaissances) à partir de données très volumineuses avec une croissance très rapide. Les étapes de ce processus sont l'acquisition de données multiformes (textes, images, vidéos, etc.), la préparation de données (prétraitement), le Data Mining (la fouille de données), et enfin la validation et mise en forme des connaissances. La Fouille de Données va se situer dans le cadre de l'apprentissage inductif. Cette étape a recours alors à différentes techniques permettant la découverte de connaissances auparavant cachées dans les données et va permettre la possibilité de prendre une décision, pour cela, on distingue deux grandes sous-familles. Les approches supervisées, on cherche à prédire la valeur d'une variable cible continue (régression) ou nominale (classification supervisée), à partir des attributs descriptifs des données. À titre d'exemple dans le domaine des télécommunications, on utilise notamment ce type d'études pour attribuer un score à un client en fonction de ses caractéristiques. L'autre grande famille qui fera l'objet de notre recherche regroupe les approches non-supervisées, parmi lesquelles on peut citer l'estimation de densité, la détection de motifs et la classification. Cette approche sans supervision est la force du

clustering. Contrairement à l'apprentissage supervisé qui nécessite des données étiquetées (comme des courriels déjà classés en spam ou non-spam), le clustering non supervisé part de l'inconnu. C'est un explorateur intrépide qui découvre des segments et des modèles cachés que les méthodes traditionnelles pourraient manquer.

Le champ d'application du clustering est vaste et fertile. En marketing, il permet d'identifier des segments de clientèle distincts, révélant des préférences et des comportements d'achat spécifiques. Dans la lutte contre la fraude, il aide à détecter des anomalies suspectes dans les transactions financières. La recherche médicale l'utilise pour classer des gènes ou des protéines, ouvrant la voie à une meilleure compréhension des maladies.

Pour mener à bien son investigation, le clustering dispose d'un arsenal d'algorithmes sophistiqués, chacun ayant ses spécialités. K-means, simple et rapide, partitionne les données en un nombre prédéfini de clusters. Le clustering hiérarchique, tel un arbre généalogique, dévoile les relations entre les groupes à différents niveaux de détail. Le clustering par mélange de modèles gaussiens, plus complexe, identifie des clusters de formes et de tailles variées, s'adaptant ainsi à la nature hétérogène des données.

Le succès de l'enquête du clustering repose sur une évaluation minutieuse. On analyse la cohérence des clusters, mesurant la distance entre les points de données d'un même groupe. On compare les résultats à des structures connues, s'il en existe, pour valider la pertinence du regroupement. Enfin, la visualisation des données et des clusters permet d'identifier d'éventuels problèmes de répartition et d'ajuster l'analyse.

En conclusion, le clustering joue un rôle clé dans l'exploration et l'analyse des données non étiquetées. Il agit comme un détective perspicace, dévoilant des structures et des relations cachées. En choisissant l'algorithme adapté et en évaluant rigoureusement les résultats, le clustering permet de transformer les données massives en une mine de connaissances exploitables. Grâce à lui, les secrets des données non étiquetées ne sont plus un mystère.

Notre contribution consiste à proposer une approche algorithme non supervisé en appliquant le champ potentiel sur les individus, en suivant ce champ les éléments qui sont

similaires sont attirés les unes aux autres, l'application de la technique reviens à bénéficier des caractéristiques du champ potentiel attractive.

Notre mémoire comprend trois grands chapitres, appui par une introduction et une conclusion générale, les chapitres du thème sont organisés comme suit :

Le chapitre 1 reprend les grandes lignes du clustering, un état de l'art sur le processus et différentes méthodes du clustering, les différentes mesures de similarités et de distances, nous pencherons aussi sur la taille, la dimensionnalité et types de données dans le clustering qui posent des problèmes, à la fin nous citons les différents critères et indices pour évaluer la qualité du clustering.

Chapitre 2 est consacré au clustering suivant l'approche du champ potentiel, un concept essentiel et fondamental en physique et en ingénierie, l'application du champ potentiel permet dans notre cas le regroupement des individus similaires par attraction.

Chapitre 3 est consacré à la conception et à l'implémentation de l'approche du clustering inspirée du champ potentiel. Nous évoquons aussi l'architecture générale de l'algorithme.

Chapitre 4 est consacré à la réalisation et exposition des chiffres, mettant en valeur la qualité de notre contribution, une comparaison entre les résultats de notre approche est d'autres travaux similaires dans le monde du clustering.

Finalement, nous achevons ce thème par une conclusion générale dans laquelle nous résumons nos contributions. Nous traçons de futurs travaux, en vision le clustering non supervisé sur une gamme large de data set avec un paramétrage et une convergence instinctive.

Chapitre 1

Etat de l'art du Clustering un travail d'un détective dans la science des données

1.1 Introduction

D'une façon générale le clustering a pour but de regrouper des individus similaires pour créer des groupes d'individus ayant le même profil en fonction d'attributs, comme l'âge, la catégorie socioprofessionnelle, les habitudes de consommation, etc. Cela en explorant les similarités entre les points de données le clustering pourra les rassembler en groupes distincts, appelés clusters (figure 1.1).

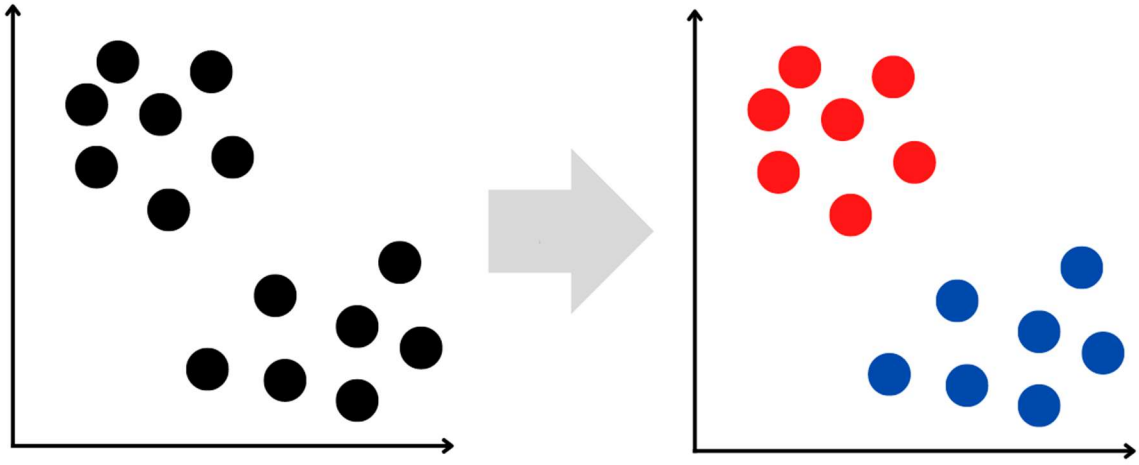

Fig. 1.1 – Le clustering pour regroupement.

Dans le monde de l'analyse de données, deux approches majeures s'affrontent pour découvrir les secrets cachés des informations, le clustering supervisé un champion polyvalent. Il

excelle dans la classification d'images, la reconnaissance faciale, la prédiction de la demande et bien d'autres domaines. L'autre approche intrigante et le clustering non supervisé qui brille dans l'analyse de marché, la détection d'anomalies et la découverte de patterns cachés. Chacun possède ses forces et ses faiblesses, les deux techniques ne sont pas des rivaux, mais plutôt des complémentaires, chacun offre des capacités uniques pour décrypter les données et révéler leurs secrets et le choix du champion dépend de la nature de la recherche et la quête.

1.2 Les étapes du processus de clustering

Le processus de clustering, qu'il soit supervisé ou non supervisé, implique généralement les étapes suivantes [1] [2] [3] :

1.2.1 Préparation des données

Le nettoyage et le prétraitement des données comprend la suppression des valeurs manquantes, la correction des erreurs et la normalisation des données si nécessaire, aussi l'exploration des données pour obtenir une compréhension de base des données en utilisant des techniques de visualisations et de statistiques descriptives.

1.2.2 Choix de l'algorithme de clustering

Sélectionner un algorithme approprié en fonction de la nature des données, des objectifs d'analyses et des ressources disponibles, choisir un algorithme adapté comme K-means, clustering hiérarchique, modèles de mélange gaussiens, etc., revient à définir les paramètres de l'algorithme en fonction des données et des besoins spécifiques, comme le nombre de clusters souhaité pour K-means ou le nombre de composants pour les modèles de mélange gaussiens.

1.2.3 Application de l'algorithme de clustering

Exécuter l'algorithme de clustering choisi sur les données préparées et analyser les clusters générés pour s'assurer qu'ils sont pertinents, significatifs et correspondent aux objectifs de l'analyse.

1.2.4 Interprétation et visualisation des résultats

Représenter si possible les clusters graphiquement pour identifier des patterns, des tendances et des relations spatiales entre les points de données, aussi il est important d'attribuer un sens aux clusters en fonction du contexte du problème et les caractéristiques des données, afin de prévoir la prédiction, la segmentation ou la prise de décision.

1.2.5 Affinement et itération

Evaluer la qualité des clusters générés en utilisant des mesures de validité internes ou externes, cela contribue a ajusté et affiné les paramètres de l'algorithme ou estimé de partir sur un autre algorithme, aussi itérer les étapes de préparation des données jusqu'à obtention des clusters optimaux différent pour des résultats plus satisfaisants (figure 1.2).

Il est important de noter que le processus de clustering est souvent itératif et que les étapes peuvent être répétées plusieurs fois jusqu'à obtenir des résultats satisfaisants. Le choix de l'algorithme, des paramètres et des techniques d'interprétation dépend du contexte spécifique du problème et des objectifs de l'analyse.

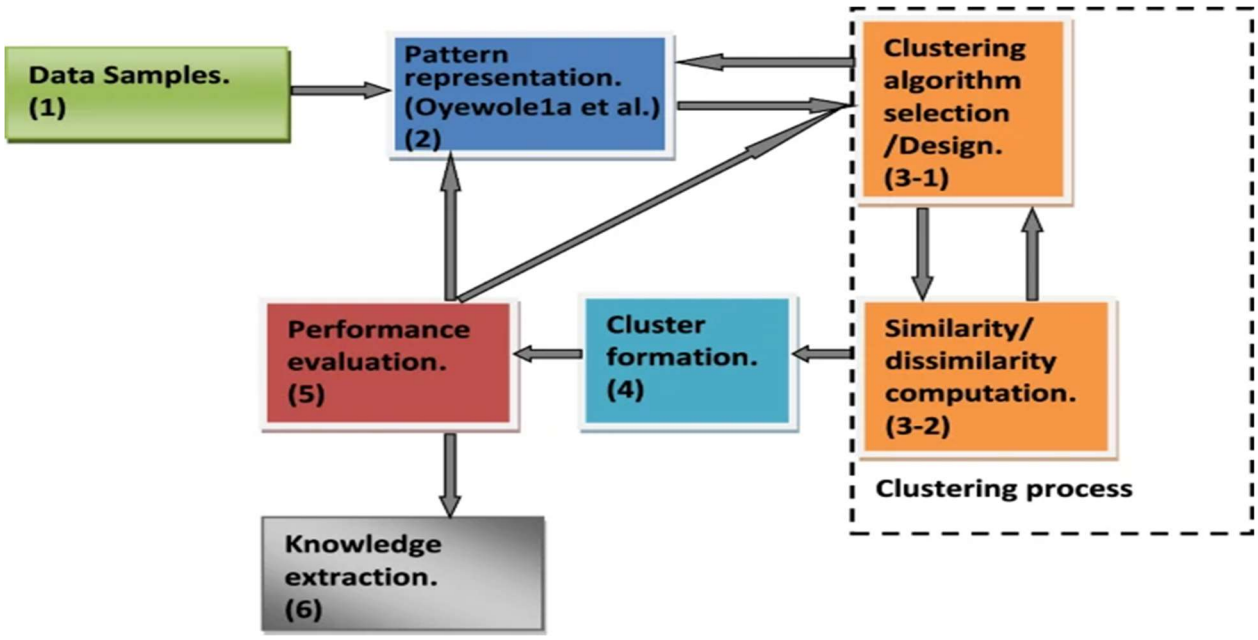

Fig. 1.2 – Les étapes du processus de clustering.

1.3 Distance et mesure de similarité et dissimilarité - Quantifier la ressemblance pour analyser et organiser les données

Dans le vaste univers des données, la capacité à mesurer et à comprendre la similarité est essentielle. La mesure de similarité permet de quantifier la ressemblance ou la dissemblance entre des éléments [3] [4], de ce fait la similarité devra satisfaire les propriétés de non-négativité elle doit être une valeur non négative, symétrique entre deux éléments, transitive (facultatif selon la mesure) et la métrique de distance doit satisfaire l'inégalité triangulaire. Qu'il s'agisse de nombres, de textes, d'images ou d'autres types d'informations. Cette quantification devient la pierre angulaire de nombreuses techniques d'analyse de données et d'apprentissage automatique. La mesure a pour but la **c**omparaison, le regroupement et recherche d'information donc identifier des éléments similaires et les regrouper en fonction de leurs caractéristiques communes (tableau 1.1). Cela joue un rôle crucial dans le clustering où des données non étiquetées sont regroupées en fonction de leur similarité.

Les définitions mathématiques de base de certaines de ces mesures [5], On suppose que l'ensemble de données $\boldsymbol{X}$ se compose de $\boldsymbol{n}$ objets de données ou observations et $\boldsymbol{d}$ fonctionnalités. Notation $\boldsymbol{D}(.,.)$ une fonction de distance entre deux objets dans l'ensemble de données, et $\boldsymbol{S}(.,.)$ une fonction de similarité entre deux objets de l'ensemble de données.

TAB. 1.1 – Des fonctions de similarités notées dans la littérature [5, 7, 14, 15].

Mesure	Pertinence
Minkowski	Pour les attributs numériques. La similarité entre les paires de données correspond à la proximité de la distance entre les paires de données
Distance euclidienne	Le plus couramment utilisé pour les attributs numériques. Un exemple particulier de Minkowski, par exemple algorithme k-means
Mesure du cosinus	Varie davantage avec les transformations linéaires qu'avec les transformations rotationnelles. Plus couramment utilisé pour le regroupement de documents
Mesure de corrélation de Pearson	Convient aux variables numériques et à la différence d'ampleur de deux variables. Utilisé pour analyser les données d'expression génique
Mesure Jaccard	Convient à la recherche d'informations et à la mesure de la similarité des mots. Peut détecter une erreur d'orthographe mais ne peut pas détecter les mots écrasés
Mesure du coefficient de dés	Similaire à la mesure Jaccard pour la recherche d'informations

1.3.1 La distance Minkowski

Nommée d'après le mathématicien allemand Hermann Minkowski [6]. Est une distance dans un espace vectoriel normé, et qui est donnée par la distance mesurée entre deux points dans un espace à N dimensions. Il s'agit essentiellement d'une généralisation de la distance euclidienne et de la distance de Manhattan. Il est largement utilisé dans le domaine de l'apprentissage automatique, notamment dans le concept de recherche de la corrélation ou de la classification optimale des données, la distance est formulée comme suit :

$$\boldsymbol{D\ (X\ ,Y)\ =\left(\sum_{i=1}^{n}|x_i-\ y_i|^p\right)^{1/p}}$$

Où ***P*** est une valeur numérique générique supérieur ou égale à 1.

1.3.2 La distance Euclidienne

Le nom provient du mathématicien grec ancien Euclide, la distance est appelée aussi parfois la distance de Pythagore référence aussi au mathématicien grec *Pythagore*. En mathématiques, la distance euclidienne entre deux points de l'espace euclidien est la longueur du segment de droite qui les sépare [7], elle peut être calculée à partir des coordonnées cartésiennes des points à l'aide du théorème de Pythagore, un théorème qui met en relation les longueurs des côtés dans un triangle rectangle [8], la distance est un cas particulier de la distance de Minkowski ou le ***p*** est égale à 2.

$$\boldsymbol{D\ (X\ ,Y)\ =\left(\sum_{i=1}^{n}|x_i-\ y_i|^2\right)^{1/2}}$$

1.3.3 La distance Manhattan

La géométrie de Manhattan [9], le nom fait référence à l'île de *Manhattan*, ou géométrie des taxis, est une géométrie dans laquelle la distance euclidienne familière est ignorée et la distance entre deux points est définie comme étant la somme des différences absolues de leurs coordonnées cartésiennes respectives [10], la distance est un cas particulier de la distance de Minkowski ou le ***p*** est égale à 1.

$$\boldsymbol{D\ (X\ ,Y)\ =\sum_{i=1}^{n}|x_i-\ y_i|}$$

1.3.4 Distance Canberra

Définie par *Godfrey N. Lance* et *William T. Williams* en 1967. La distance est une mesure robuste, qui mesure la dissimilarité pour comparer des ensembles de données contenant des attributs numériques [11]. Particulièrement, elle est utile pour comparer des données qui peuvent contenir des valeurs nulles ou manquantes. La mesure est souvent utilisée dans le domaine de l'apprentissage automatique, les statistiques, la médecine et d'autres domaines [12]. La distance se calcule de la manière suivante :

$$D(X,Y) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{|x_i| + |y_i|}$$

1.3.5 Distance de Wasserstein (Distance de Kantorovitch)

Le nom de la distance vient du mathématicien russe *Léonid Wasserstein*, mais cette distance avait été défini par *Léonid Kantorovitch* dans son célèbre rapport de 1939.

La distance quantifie la différence entre deux distributions de probabilité en termes de la quantité minimale de travail nécessaire pour transformer l'une en l'autre en déplaçant la masse de probabilité. Elle est interprétée comme la distance géodésique entre les deux distributions dans l'espace des mesures de probabilité [13]. La distance se calcule de la manière suivante :

$$W(\mu, \nu) = \inf_{\gamma \in \Gamma(\mu,\nu)} \int_{X \times X} d(x,y)\, d\gamma(x,y) \quad \text{où :}$$

- μ, ν deux distributions de probabilité.
- $d(x, y)$ est la distance entre les points x et y dans l'espace X.
- $\Gamma(\mu, \nu)$ l'ensemble des mesures de probabilité γ sur $X \times X$.

L'interprétation de l'équation est de quantifie le coût minimal de transport pour transformer la distribution μ en la distribution ν. Le coût de transport pour une unité de masse entre deux points x et y est donné par $d(x, y)^p$.

1.3.6 La similitude de Cosinus

Mesurer la proximité entre objets représentés dans un espace métrique est un problème fondamental, lorsque ces objets sont représentés par des vecteurs dans un espace métrique réel. Un des indices de similarité des plus populaires est la mesure du Cosinus de l'angle formé par deux vecteurs, la mesure de cosinus est formulée comme suit [14] :

$$\boldsymbol{S(x,y) = cos(\theta) = \frac{(x,y)}{\sqrt{||x||^2\,||y||^2}}}$$

Où $\boldsymbol{\theta}$ est la mesure angulaire entre $\boldsymbol{x}$ et $\boldsymbol{y}$, $\boldsymbol{(x, y)}$ est le produit scalaire canonique et $\boldsymbol{||x||}$ est la norme de $\boldsymbol{x}$. Dans cette contribution, l'indice de similarité qui généralise la mesure de Cosinus est introduit.

1.3.7 La mesure de Jaccard

Le nom de la mesure provient du botaniste suisse *Paul Jaccard* [15], est appelée aussi coefficient de communauté. L'indice est une mesure simple de similarité entre deux ensembles d'objets proposée par le botaniste en 1901 [16, 17]. Il consiste à diviser le nombre d'objets en commun par le nombre d'objets distincts dans les deux ensembles, autrement dit le cardinal de l'intersection divisé par le cardinal de l'union, Il permet d'évaluer la similarité entre les ensembles, ou encore la distance est formulée comme suit :

$$\boldsymbol{D(x,y) = \frac{|x \cap y|}{|x \cap y|}}$$

La mesure est interprétée autrement, par la mesure de Jaccard **é**tendue et qui est formulée comme suit :

$$\boldsymbol{D(x,y) = \frac{(x,y)}{||X||^2 + ||Y||^2 - (x,y)}}$$

1.3.8 La mesure de Gower

Gower a proposé un indice de similarité qui porte son nom en 1971 [18]. L'objectif de cet indice consiste à mesurer à quel point deux individus sont semblables. L'indice de Gower varie entre 0 et 1, si l'indice vaut 1 les deux individus sont identiques. À l'opposé, s'il vaut 0 les deux individus considérés n'ont pas de point commun.

L'indice de similarité de Gower entre deux individus se calcule de la manière suivante :

$$S_{ij} = \frac{\sum_{k=0}^{n} w_{ijk}\, s_{ijk}}{\sum_{k=0}^{n} w_{ijk}}$$

Où ***Sijk*** est une composante de similarité pour la $k^{ème}$ variable, et le poids ***Wijk*** vaut 1 si la comparaison est valable pour la $k^{ème}$ variable, ou 0 sinon. Ainsi ***Wijk*** égale à 0 si une ou les deux observations sur la $k^{ème}$ variable sont manquantes.

En conclusion la mesure de similarité est un concept fondamental qui sous-tend de nombreuses techniques d'analyse de données et d'apprentissage automatique. En quantifiant la ressemblance entre des éléments, elle permet d'extraire des informations précieuses à partir de vastes ensembles de données, de résoudre des problèmes complexes d'organisation et de classification, et d'améliorer la performance des algorithmes, tout on respectons le choix de la mesure de similarité appropriée qui dépend de plusieurs facteurs, selon la nature des données (nombres, texte, images, etc.), la tâche à accomplir (clustering, recherche d'information, etc.), et les propriétés souhaitées pour la mesure (symétrie, transitivité, etc.).

1.4 La taille, dimensionnalité et types de données dans le clustering posent des problèmes

L'une des approches pour classer les algorithmes de clustering est le type de données d'entrée. *Liao*. [19] a observé que les données pouvant être saisies dans n'importe quelle tâche de clustering, elles peuvent être classées comme binaires, catégoriques, numériques, par intervalles, ordinales, relationnelles, textuelles, spatiales, temporelles, spatio-temporelles, d'images, multimédia ou des mélanges de ces données. Cette classification peut également être sous-classée. Par exemple, les données brutes numériques destinées au clustering peuvent être statiques, chronologiques ou sous forme de flux de données. Les données statiques ne changent pas avec le temps, tandis que les objets des données de séries chronologiques changent avec le temps. *Aggarwal et coll*. [20] ont décrit le flux de données comme de grands volumes de données arrivant à un taux de croissance illimité. Comme l'ont noté *Mahdi et al*. [21] types de données vastes et complexes à stocker, telles que les données des réseaux sociaux (appelées big data) et les données à haut débit (flux de données) telles que les flux de clics Web. De plus, ils ont souligné que le type de données considéré influence souvent le type de techniques de regroupement sélectionnées.

L'application de certains algorithmes de regroupement directement aux données brutes s'est avérée problématique à mesure que la taille des données augmente (*Gordon*. [22], *Parsons et al*. [23]). Deux raisons ont été avancées pour expliquer ce problème observé. La première raison indiquée était basée sur le type d'algorithme de clustering utilisé. C'est tel que certains algorithmes de clustering prennent pleinement en compte toutes les dimensions des données lors du processus de clustering. En conséquence, ils masquent des groupes potentiels d'objets de données aberrants. La seconde raison est que, à mesure que la dimensionnalité augmente dans les données, la mesure de distance pour calculer la similarité ou la dissemblance entre les objets de données devient moins efficace. L'extraction et la sélection de caractéristiques ont été suggérées comme méthode générique pour résoudre ce problème en réduisant la dimensionnalité des données avant l'application des algorithmes de clustering. Cependant, ils ont noté que cette méthode basée sur les caractéristiques pourrait omettre certains clusters cachés dans des sous-espaces des ensembles de données. Le regroupement de sous-espaces était la méthode suggérée pour surmonter ce problème.

Étant donné que les résultats du clustering sont fortement liés au type et aux caractéristiques des données représentées, leurs performances sont améliorées grâce aux méthodes actuelles d'apprentissage automatique supervisé telles que les réseaux de neurones profonds (DNN). Comme l'ont noté *James et al*. [24] et *Ni et al*. [25], les DNN ont eu de meilleures performances (par exemple dans la modélisation de la parole et du texte, la classification de vidéos et d'images) par rapport aux réseaux de neurones développés précédemment, comme le montrent (*Hastie et al*. [26]) en raison du moins de bricolage de formation requis et de la disponibilité croissante. De grands ensembles de données de formation. DNN pourrait être utilisé pour obtenir une représentation améliorée des fonctionnalités utile pour le clustering avant que le clustering réel ne soit effectué. C'est ce qu'on appelle le clustering profond dans le domaine de l'apprentissage automatique (*Aljalbout et al*. [27]).

1.5 Les différentes méthodes de clustering

Dans la littérature, un panorama des différentes méthodes de clustering rencontrées est invoqué. Du fait qu'il soit difficile de proposer une classification logique de ces méthodes, les différentes études de synthèses proposent chacune leur propre organisation. Ils présentent successivement les méthodes de clustering hiérarchique, par partitionnement et enfin les approches probabilistes [28], tandis que d'autres choisissent de considérer principalement les approches hiérarchiques et de partitionnement, en incluant les méthodes probabilistes dans cette dernière catégorie [29]. Enfin, d'autres classifications distinguent les méthodes de clustering paramétriques et non paramétriques. Ces différentes présentations des méthodes de clustering sont dues, d'une part, au fait que les classes d'algorithmes se recouvrent (certaines méthodes s'appuyant, par exemple, sur des modèles probabilistes proposent un partitionnement.) et d'autre part, diffèrent selon que l'on s'intéresse plutôt aux résultats du clustering (Hiérarchie vs. Partitionnement, Clustering dur vs. Clustering flou, etc.), ou à la méthode utilisée pour parvenir à ce résultat (Utilisation de fonctions probabilistes vs. Utilisation de graphes, etc.).

Pour ce qui suit nous choisissons de présenter d'abord les méthodes hiérarchiques et de partitionnement, puis en nous intéressons aux découpages en grilles, les densités et des descriptions conceptuelles.

1.5.1 Le clustering Hiérarchique

Le principe des algorithmes hiérarchiques est de construire un arbre de clusters ou dendrogramme, comme il est illustré dans la figure 1.3 :

- La racine de l'arbre est formée par le cluster $\boldsymbol{X}$ contenant l'ensemble des objets.
- Chaque nœud de l'arbre constitue un cluster $\boldsymbol{C_i \subset X}$.
- Les feuilles de l'arbre correspondent aux singletons $\{\boldsymbol{x_1}\}$, …, $\{\boldsymbol{x_n}\}$.
- L'union des objets contenus dans les fils d'un nœud donné, correspond aux objets présents dans ce nœud.
- Les "paliers" sont indicés relativement à l'ordre de construction.

Ces hiérarchies sont généralement appréciées puisqu'elles permettent une visualisation de l'organisation des données et du processus de clustering. À partir de ce dendrogramme, il est possible d'obtenir une partition de $\boldsymbol{X}$ en coupant l'arbre à un niveau donné.

On distingue deux approches pour parvenir à un tel arbre hiérarchique, les algorithmes agglomératifs et divisifs. Un regroupement agglomératif construit l'arbre en partant des feuilles (singletons) et procède par fusions successives des plus proches clusters jusqu' à obtenir un unique cluster racine, contenant l'ensemble des objets. Par opposition, les algorithmes divisifs considèrent d'abord la racine contenant tous les objets, puis procèdent par divisions successives de chaque nœud jusqu'à obtenir des singletons. Notons que pour chacune de ces deux méthodes, l'arbre hiérarchique n'est pas nécessairement construit totalement. Le processus peut être stoppé lorsque le nombre de clusters désire est atteint ou lorsqu'un seuil de qualité est dépassé.

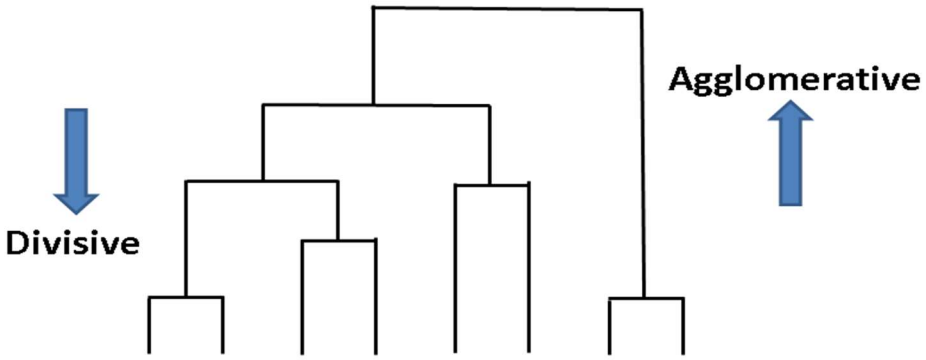

Fig. 1.3 – Dendrogramme pour clustering (Agglomératifs vs. Divisifs).

Nous présentons ci-dessous les deux principaux algorithmes de clustering hiérarchique, l'algorithme **DIANA** pour l'approche divisive représentée dans la figure 1.4, et l'algorithme **SAHN** (**S**equential **A**gglomerative **H**ierarchical and **N**onoverlapping) pour l'approche agglomérative qui est représentée dans la figure 1.5.

Algorithme DIANA (**DI**visive **ANA**lysis)

Entrée : Une matrice de similarité S sur l'ensemble des objets à traiter X

Sortie : Une hiérarchie P

1. Initialisation a 1 cluster (racine), $C = \{\{x_1, \ldots, x_n\}\}$, et $P = C$

2. Sélectionner le cluster $C \in C$ de diamètre maximum

3. Identifier dans C l'objet (ou l'un des objets) x^* ayant la plus faible similarité moyenne avec les autres objets :

$$x* = \ arg\ min\ \frac{1}{|C|-1}\sum_{j\neq i} s(xi, xj)$$

$x*$ Initialise un nouveau cluster C^*,

4. Pour chaque objet $x_i \notin C^*$ calculer :

Si = [moyenne des $S(x_i ; x_j)$, $x_j \in C \setminus C^*$] - [moyenne des $S(x_i ; x_j)$, $x_j \in C^*$]

5. Soit x_k l'objet pour lequel s_k est minimal. Si s_k est négatif alors ajouter x_k à C^*

6. Répéter les étapes 3 et 4 jusqu'à $s_k > 0$

7. Remplacer C par $C \setminus C^*$ et C^* dans C puis ajouter $C \setminus C^*$ et C^* dans P

8. Répéter les étapes 2 à 7 jusqu'à ce que chaque cluster de C soit réduit à un singleton

9. Retourner P, un ensemble de parties non vides sur X, correspondant aux nœuds de la hiérarchie

Fig. 1.4 – Algorithme divisif hiérarchique DIANA.

Algorithme SAHN (**S**equential **A**gglomerative **H**ierarchical and **N**onoverlapping)

Entrée : Une matrice de similarité $\boldsymbol{S}$ sur l'ensemble des objets à traiter $\boldsymbol{X}$

Sortie : Une hiérarchie $\boldsymbol{P}$

1. Initialisation a $\boldsymbol{n}$ singletons, $\boldsymbol{C} = \{\{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}\}$, et $\boldsymbol{P} = \boldsymbol{C}$

2. Identifier dans $\boldsymbol{C}$ les deux objets $\boldsymbol{C_k}$ et $\boldsymbol{C_l}$ les plus proches selon $\boldsymbol{S}$:

   $(\boldsymbol{C_k}, \boldsymbol{C_l}) = \arg\max \{sim\ (\boldsymbol{C_i}, \boldsymbol{C_j})\}$

3. Remplacer $\{\boldsymbol{C_k}\}$ et $\{\boldsymbol{C_l}\}$ par $\{\boldsymbol{C_k} \cup \boldsymbol{C_l}\}$ dans $\boldsymbol{C}$ et ajouter $\{\boldsymbol{C_k} \cup \boldsymbol{C_l}\}$ à $\boldsymbol{P}$

4. Recalculer la matrice $\boldsymbol{S}$ en conséquence

5. Si $\boldsymbol{C} \neq \{\{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}\}$, revenir à l'étape 2

6. Retourner $\boldsymbol{P}$, un ensemble de parties non vides sur $\boldsymbol{X}$, correspondant aux nœuds de la hiérarchie

Fig. 1.5 – Algorithme agglomératif hiérarchique SAHN.

1.5.2 Le clustering par Partitionnement

Contrairement aux approches hiérarchiques précédentes, les algorithmes de partitionnement proposent, en sortie, une partition de l'espace des objets plutôt qu'une structure organisationnelle du type dendrogramme. Le principe est alors de comparer plusieurs schémas de clustering (plusieurs partitionnements) afin de retenir le schéma qui optimise un critère de qualité. En pratique il est impossible de générer tous les schémas de clustering pour des raisons évidentes de complexité. On cherche alors un bon schéma correspondant à un optimum. Cet optimum est obtenu de façon itérative, en améliorant un schéma initial choisi plus ou moins aléatoirement, par réallocation des objets autour de centres mobiles. Sans aucun doute la méthode de partitionnement la plus connue et la plus utilisée dans divers domaines d'application et qui ne cesse d'évoluer d'une partition stricte vers une pseudo-partition, puis vers une partition floue est l'algorithme des k-

moyennes (k-Means) [30]. Ce succès est dû au fait que cet algorithme présente un rapport coût/efficacité avantageux, comme il est illustré dans la figure 1.6.

Algorithme k-moyennes (K-Means)

Entrées : k le nombre de clusters désiré, d'une mesure de dissimilarité sur l'ensemble des objets à traiter X

Sortie : Une partition $C = \{C_1, \ldots, C_k\}$

Etape 0 : 1. Initialisation par tirage aléatoire dans X, de k centres $x^*_{1.0}, \ldots, x^*_{k.0}$

2. Constitution d'une partition initiale $C_0 = \{C_1, \ldots, C_k\}$ par allocation

De chaque objet $x_i \in X$ au centre le plus proche :

$$C_l = \{x_i \in X \setminus d(x_i, \ldots, x^*_{l.0}) = \min_{h=1,\ldots,k} d(x_i, \ldots, x^*_{h.0})\}$$

3. Calcul des centroïdes des k classes obtenues $x^*_{1.1}, \ldots, x^*_{k.1}$

Etape t : 4. Constitution d'une nouvelle partition $Ct = \{C_1, \ldots, C_k\}$ par allocation de

Chaque objet $x_i \in X$ au centre le plus proche :

$$C_l = \{x_i \in X \setminus d(x_i, \ldots, x^*_{l.t}) = \min_{h=1,\ldots,k} d(x_i, \ldots, x^*_{h.t})\}$$

5. Calcul des centroïdes des k classes obtenues $x^*_{1.t+1}, \ldots, x^*_{k.t+1}$

6. Répéter les étapes 4 et 5 tant que des changements s'opèrent d'un schéma

Ct a un schéma $Ct+1$ ou jusqu'à un nombre t d'itérations

7. Retourner la partition finale C_{finale}

Fig. 1.6 – Algorithme de partitionnement k-moyennes (K-Means).

1.5.3 Le clustering par Grilles

Le clustering par grilles procède par découpage de l'espace de représentation des données en un ensemble de cellules (ex. hyper-cubes, hyper-rectangles). De ce fait, ces méthodes visent principalement le traitement de données spatiales. Le résultat d'une telle méthode est une partition des données via une partition de l'espace de représentation des données. Les clusters formes correspondent à un ensemble de cellules denses et connectées.

La principale difficulté de ces méthodes concerne la recherche d'une taille appropriée pour les cellules construites (problème de granularité). De trop petites cellules conduiraient à un sur-partitionnement, a l'inverse de trop grandes cellules entraineraient un sous-partitionnement. Nous citons, STING, WaveCluster et GIZMO, trois algorithmes de clustering utilisant un découpage en grilles, ces trois méthodes résument les principales approches existantes dans ce domaine. STING une approche qui consiste à construire une hiérarchie de grilles, en revanche WaveCluster est une approche de détection de limites entre zones à forte et à faible densités, et GIZMO la fusion des cellules de densités comparables et connectées. Pour ce qui suit nous nous inclinons sur l'algorithme WaveCluster (Wavelet-Based clustering) [31] qui utilise les ondelettes une méthode de transformation pour le traitement de signaux sur les données synthétisées par un découpage de l'espace des attributs en grille. L'algorithme (figure 1.7) procède en premier lieu à un découpage de l'espace des attributs (espace d-dimensionnel) tel que chaque dimension i est divisée en m_i intervalles. La grille construite possède alors $\prod_i m_i$ cellules, chaque objet étant affecté à une cellule.

La répartition des objets dans les cellules compose un signal d-dimensionnels, dont les parties à haute fréquence correspondent aux frontières des clusters tandis que les parties à basse fréquence permettent de distinguer les régions de l'espace où les objets sont concentrés.

L'utilisation de la méthode des ondelettes permet d'obtenir la décomposition appropriée du signal et de détecter ainsi la position des clusters. L'analyse du signal transformé passe par la recherche de cellules connexes (clusters) dans les différentes bandes de fréquence. Les objets sont finalement étiquetés en fonction des clusters obtenus par cette technique. L'algorithme WaveCluster se limite à des applications où les données multidimensionnelles sont décrites par

des attributs numériques. La méthode est de complexité linéaire sur le nombre d'objets dans le cas bidimensionnel mais cette complexité augmente de façon exponentielle avec la dimension de l'espace des attributs.

Algorithme WaveCluster (Wavelet-Based Clustering)

Entrée : Un ensemble ***X*** de données multidimensionnelles

Sortie : Une partition de **X**

1. Découper l'espace des attributs et affecter chaque objet a une cellule

2. Appliquer la transformation par ondelettes sur le signal multidimensionnel induit par les densités des cellules

3. Rechercher les composants connectés (clusters) dans les différentes bandes de fréquence

4. Affecter les objets aux clusters extraits

Fig. 1.7 – Algorithme WaveCluster.

1.5.4 Le clustering par Densités

Les algorithmes présentés précédemment utilisent, pour la plupart, la notion de densité d'une cellule qui est définie relativement au nombre d'objets contenus dans cette cellule et qui font références à un seuil fixé par l'utilisateur. Les algorithmes de clustering par densités se basent sur une notion similaire, complétée par d'autres concepts fondamentaux tels que le voisinage d'un objet, l'objet noyau, l'accessibilité ou la connexion entre objets. Ou le voisinage d'un objet est précisé comme suit, soit un objet $x_i \in X$, *le voisinage* $N_\epsilon(xi)$ de x_i (de rayon ϵ) est défini par l'ensemble des points de X, distants d'au plus ϵ de x_i, est qui interprété par $N_\epsilon(xi) = \{xi \in X \mid d(x_i, ..., x_j) \leq \epsilon\}$, Aussi le noyau est précisé comme suit, soit un objet $x_i \in X$, ϵ et M deux paramètres fixés, x_i est un objet *noyau* dans X si et seulement si le voisinage de x_i contient au moins M objets

ce qui est interprété par *noyau*(x_i) $\Leftrightarrow$ $|N_\epsilon(x_i)| \geq M$. La construction d'un tel cluster revient à rechercher d'abord un objet noyau puis à agglomérer, autour de ce noyau, tous les objets ***d***-accessibles par ce noyau. L'algorithme DBSCAN (**D**ensity-**B**ased **S**patial **C**lustering of **A**pplications with **N**oise) [32] résume assez bien cette stratégie de recherche (figure 1.8).

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

Entrées : Un ensemble X de n objets, ϵ et M deux paramètres fixés

Sortie : Une partition $C = \{C_1, \ldots, C_k\}$ de X en k d-clusters.

1. Initialisation $id = 1$ et $C_{id} = \Phi$

2. Pour i allant de **1** à n :

   3. Si x_i n'est pas un noyau ou si $x_i \in \bigcup_{j=1..id} Cj$, alors retourner à l'étape 2

   4. construire-cluster (x_i, X, C_{id}, ϵ, M)

   5. $id = id + 1$ et $C_{id} = \Phi$

6. Retourner l'ensemble des d-clusters : C_1, ..., C_{id-1}

Fig. 1.8 – Algorithme DBSCAN.

1.5.5 Le clustering Conceptuel

Le clustering conceptuel a été introduit au début des années 1980 par [33]. Cette approche du clustering est alors présentée comme une manière de découvrir des schémas (clusters) compréhensibles à partir des données. Plutôt que de définir une mesure de similarité puis d'organiser les objets de façon à minimiser les inerties intra-clusters et maximiser l'inertie inter-clusters, *Michalski* propose de générer une structure (hiérarchique) de concepts (figure 1.9). Dans ce type de structure, chaque concept se définit à la fois en extension (ensemble des objets placés

dans ce concept) et en intension (règles descriptives sur les objets). La définition en intension permet une caractérisation des clusters (ou concepts) et fournit alors à l'utilisateur une explication compréhensible de ces concepts. Plusieurs systèmes de clustering conceptuel ont été proposés, afin de traiter des données plus ou moins complexes et de construire différents types de structures conceptuelles. Nous citons quelques algorithmes tels que CLUSTER/2 et CLUSTER/S, et l'algorithme incrémental COBWEB [34] qui a été adapté dans CLASSIT pour traiter des données décrites par des attributs numériques. Contrairement à ces dernières méthodes, qui génèrent chacune un arbre hiérarchique, d'autres approches consistent à organiser les concepts dans des graphes conceptuels, s'apparentant davantage à des treillis de Galois.

Pour ce qui suit nous choisissons de présenter, l'algorithme COBWEB (figure 1.10), et ces sous fonctions (figure 1.11), afin d'illustrer le fonctionnement des méthodes de clustering conceptuel. Cet algorithme est très utilisé et se présente souvent comme une référence dans ce domaine. COBWEB est un système incrémental de clustering conceptuel hiérarchique. Le processus de formation de concepts a pour objectif de construire, à partir d'un ensemble de données et de leur description, une hiérarchie de concepts telles que nous la présentons en algorithme.

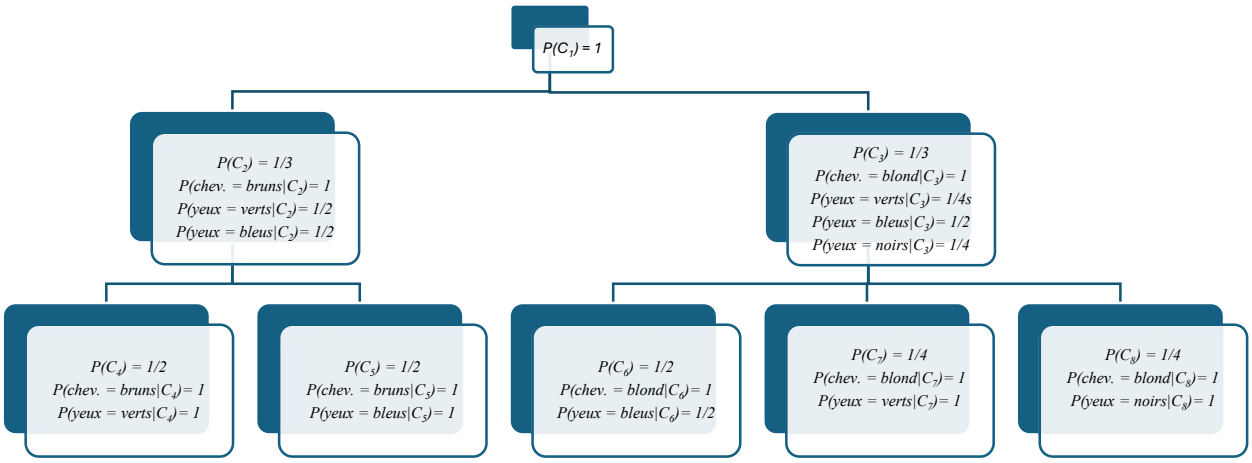

Fig. 1.9 – Une structure hiérarchique de concepts.

COBWEB (Algorithme incrémental de classification conceptuelle hiérarchique)

Entrées : Le nœud courant ***N*** de la hiérarchie de concepts et x_i une nouvelle instance à incorporer

Sortie : Une hiérarchie de concepts classifiant l'instance x_i.

Procedure ***COBWEB*** **(*N, xi*)**

Si ***N*** est un nœud terminal (feuille) alors :

Créer-nouveaux-nœuds-terminaux (N, x_i)

Incorporer (N, xi)

Sinon ***Incorporer (N, x_i)***

Pour chaque nœud fils ***C*** de ***N*** :

Calculer le score d'intégration de x_i dans ***C***

Soit ***P*** le nœud avec le plus haut score (S_I)

Soit ***R*** le nœud avec le deuxième plus haut score

Soit ***SC*** le score correspondant à la création d'un nouveau nœud ***Q*** = {***xi***}

Soit ***SF*** le score correspondant à la fusion des nœuds ***P*** et ***R***

Soit ***SD*** le score correspondant à la division du nœud **P** (remplacé par ses fils)

Si S_I est le meilleur score alors : ***COBWEB (P, xi)***

Sinon si ***SC*** est le meilleur score alors : initialiser ***Q*** relativement a ***xi***

Sinon si ***SF*** est le meilleur score alors : O = ***Fusionner (P, R, N)*** et ***COBWEB (O, x_i)***

Sinon si ***SD*** est le meilleur score alors : ***Fiviser (P, N)*** et ***COBWEB (N, xi)***

Fig. 1.10 – Algorithme COBWEB.

Fonction Créer-nouveaux-nœuds-terminaux (N, xi)

Créer un nouveau nœud fils ***M*** au nœud ***N***

Initialiser les probabilités de ***M*** à partir des probabilités de ***N***

Créer un nouveau fils ***O*** au nœud ***N***

Initialiser les probabilités de ***O*** à relativement à ***xi***

Fonction Incorporer (N, xi)

Mettre à jour les probabilités de ***N***

Pour chaque attribut ***A*** de x_i :

Pour chaque valeur ***V*** de ***A*** :

Mettre à jour la probabilité ***p(V\|N)***

Fonction Fusionner (P, R, N)

Créer un nouveau nœud fils ***O*** au nœud ***N***

Mettre à jour les probabilités de ***O*** par moyenne des probabilités de ***P*** et ***R***

Supprimer les nœuds fils ***P*** et ***R*** de ***N***

Ajouter ***P*** et ***R*** comme nœud fils du nœud ***O***

Retourner ***O***

Fonction Diviser (P, N)

Supprimer le nœud fils ***P*** de ***N***

Ajouter chaque nœud fils de ***P*** comme nouveau nœud fils de ***N***

Fig. 1.11 – Fonctions *Créer-nouveaux-nœuds-terminaux, Incorporer, Fusionner et Diviser*.

Il est impérativement de cité et d'invoquons quelques autres approches de clustering, autres que nous avons cité jusqu'ici, des approches incontournables permettant de constituer des classes d'objets, à partir de leur description ou seulement d'une matrice de (dis)similarité, dans un contexte non-supervisé. Ces algorithmes hiérarchiques ou de partitionnement, constituent une synthèse assez large de ce domaine qui est la reconnaissance de formes. Cependant, cet état de l'art ne saurait être complet sans mentionner certaines techniques ou formalismes provenant d'autres domaines de l'apprentissage ou plus généralement de l'informatique, tels que les réseaux de neurones artificiels ou la théorie des graphes.

En premier les réseaux de neurones artificiels, inspirés des réseaux de neurones biologiques [35], sont célèbres pour leur utilisation en classification (supervisée). Cependant, les caractéristiques de ces réseaux ont également été adaptées dans un cadre non-supervisé [36], notamment l'aspect parallélisation du processus, permettant ainsi le traitement de grandes bases de données. L'algorithme le plus connus est la méthode SOM, ou cartes auto-organisatrices de Kohonen (Self-Organizing Map) [37]. Il s'agit d'un réseau monocouche, où chaque objet d'entré (neurone d'entrée) est associé à un neurone de sortie par projection,

La deuxième approche, le clustering basé sur les graphes est apparu dans les années 80. Cette approche est fondée sur le formalisme théorique des graphes, constitués de sommets et d'arêtes éventuellement évaluées sur lequel une mesure de distance d est définie. Il existe différentes stratégies pour générer ce type de graphes, appelés graphes de proximités. L'une des techniques est le graphe du plus proche voisin NNG(X) (***N**earest **N**eighbor **G**raph*) [38], qui est défini par l'ensemble X de sommets et l'ensemble V d'arêtes tel que chaque objet $x_i \in X$ est relié à son plus proche voisin dans X relativement à d, L'approche fixe un nombre de clusters (k) et choisit un critère (ex : nombre d'arêtes entre les clusters). L'algorithme de clustering recherche alors une partition du graphe en k ensembles équilibrés (homogènes en taille) de sommets et tels que le nombre d'arêtes inter-clusters soit minimal. La complexité d'une telle méthode augmente de façon exponentielle avec le paramètre k, c'est pourquoi la plupart des algorithmes proposent une approche récursive de partitionnement en 2 telle que $k = 2^p$ [39].

1.6 Évaluer la qualité du clustering

Évaluer la qualité d'un clustering est crucial pour garantir que les groupes formés sont pertinents et significatifs. Deux cheminements sont alors envisagés pour évaluer ou comparer entre les schémas de clustering. Deux grandes catégories d'indices ont été mises en évidence dans la littérature. Ce sont les indices internes et externes. Certains auteurs ont indiqué une répartition de ces indices de validation en trois catégories, cette problématique est plutôt bien synthétisée dans les récents travaux de *M. Halkidi.* [40, 41], mais comme *Xu et Wunsch*. [42] et *Sekula et al*. [43] ont indiqué que ceux-ci pouvaient encore être intégrés dans l'esprit des indices internes et externes. Selon *Baidari et Patil*. [44], les indices internes mesurent la compacité des clusters en appliquant des techniques de mesure de similarité, la séparabilité des clusters et l'homogénéité intra-cluster, ou une combinaison de ces deux. Des critères externes sont appliqués pour faire correspondre la structure du cluster à une classification prédéfinie des instances afin de valider les résultats du clustering. Ils ont cependant noté l'utilisation courante de la validité interne avec les algorithmes de clustering.

1.6.1 Mesures de validité externe

L'évaluation externe s'agit de confronter un schéma avec une classification prédéfinie. L'évaluation porte donc sur l'adéquation entre le schéma obtenu et une connaissance externe sur les données (schéma attendu).

1.6.2 Mesures de validité interne

L'évaluation interne n'utilise pas de connaissances externes mais uniquement les données d'entrées (matrice de (dis)similarité, descriptions des données etc.) comme référence. Ainsi, par exemple, parmi plusieurs schémas, le meilleur sera celui qui conserve un maximum d'information relativement à l'information contenue dans la matrice de (dis)similarité.

1.6.3 Critères externes

Les mesures externes pour comparer l'adéquation entre un schéma de clustering $\boldsymbol{C}$ et une classification préétablie $\boldsymbol{P}$. les critères externes jouent un rôle crucial dans l'évaluation et la

validation des modèles de clustering, en offrant une perspective objective basée sur des données étiquetées. Leur utilisation aide à améliorer la fiabilité et l'efficacité des algorithmes de clustering. Notons :

- ***a*** Le nombre de paires (x_i, x_j) telles que x_i et x_j se retrouvent dans une même classe dans ***C*** et dans ***P*** (liaison correcte).

- ***b*** Le nombre de paires (x_i, x_j) telles que x_i et x_j se retrouvent dans une même classe dans ***C*** mais non dans ***P*** (liaison incorrecte).

- ***c*** Le nombre de paires (x_i, x_j) telles que x_i et x_j se retrouvent dans une même classe dans ***P*** mais non dans ***C*** (non-liaison incorrecte).

- ***d*** Le nombre de paires (x_i, x_j) telles que x_i et x_j ne se retrouvent pas dans une même classe ni dans ***C*** ni dans ***P*** (non-liaison correcte).

- ***N*** Le nombre d'objets à traiter.

- ***K*** le nombre de cluster.

- ***m*** le nombre de classe.

- C_i l'ensemble des points dans le cluster i.

- L_j l'ensemble des points dans le cluster j.

- $|C_i \cap L_j|$ le nombre de points du cluster i qui appartiennent à la classe j.

- N_T le nombre total de cas.

Pour ce qui suit, nous allons présenter en détail des indices et mesures d'évaluation qui sont des outils essentiels pour évaluer la qualité et la performance des modèles de clustering.

1.6.3 a. Indice de Rand

L'indice est interprété par la formule : $\boldsymbol{R\ (C,P) = \frac{a+d}{a+b+c+d}}$

1.6.3 b. Indice de Jaccard

L'indice est interprété par la formule : $\boldsymbol{J\ (C,P) = \frac{a}{a+b+C}}$

1.6.3 c. Indice de Fowlkes et Mallows

L'indice est interprété par la formule : $\boldsymbol{FM\ (C,P) = \sqrt{\frac{a}{a+b} + \frac{a}{a+C}}}$

La principale différence entre les indices de Rand et de Jaccard est la prise en compte, ou non, des non-liaisons correctes. Une dernière mesure d'évaluation concerne la pureté des clusters obtenus dans ***C*** relativement à ***P***. Cette mesure n'est, bien sûr, pas indépendante des indices déjà mentionnés précédemment. La pureté d'un ensemble de clusters peut s'évaluer par la mesure d'entropie telle que nous la définissons.

1.6.3 d. Mesure d'Entropie

$$\boldsymbol{E\ (C,P) = \sum_{i=1}^{kc} \frac{|C_i|.\varphi(C_i)}{N}} \quad \text{Avec} \quad \boldsymbol{\varphi(C_i) = \sum_{j=1}^{kp} P_j.\, log\ (P_j)}$$

Dans cette dernière définition, $\boldsymbol{k_C}$ et $\boldsymbol{k_P}$ désignent le nombre de clusters respectivement dans ***C*** et dans ***P***, $\boldsymbol{P_j}$ correspond à la proportion (dans ***Ci*** de ***C***) d'objets appartenant à la classe ***j*** de ***P***. Le schéma d'entropie minimale, correspond au schéma de pureté maximum relativement à la classification attendue.

1.6.3 e. La Pureté

La pureté (ou purity en anglais) définie la qualité d'un cluster en termes de proportion de points qui appartiennent à la même classe. Un cluster pur est un cluster qui contient des points d'une seule classe, tandis qu'un cluster impur contient des points de plusieurs classes. La pureté est

généralement définie comme la proportion du point de données le plus fréquent dans le cluster. Formellement, la pureté est définie comme suit :

$$\boldsymbol{Purity = \frac{1}{N}\sum_{i=1}^{k} \max_{j} |C_i \cap L_j|}$$

1.6.3 f. L'Homogénéité

L'homogénéité (ou homogeneity en anglais). Une mesure utilisée pour déterminer à quel point les clusters contiennent des points de données appartenant à une seule classe. Contrairement à la pureté, qui évalue directement la proportion de points de la classe dominante dans chaque cluster, l'homogénéité prend en compte l'entropie et la distribution des classes au sein des clusters. Formellement, l'homogénéité est définie comme suit :

$$\boldsymbol{Homogénéité = 1 - \frac{H(C)}{H(L)}}$$ Où :

- Entropie totale des clusters : $\boldsymbol{H(C) = \sum_i \frac{|C_i|}{N} H(C_i)}$
- Entropie des classes : $\boldsymbol{H(L) = -\sum_j \frac{|L_j|}{N} log\left(\frac{|L_j|}{N}\right)}$
- Entropie de chaque cluster : $\boldsymbol{H(C_i) = -\sum_j \frac{|C_i \cap L_j|}{|C_i|} log\left(\frac{|C_i \cap L_j|}{|C_i|}\right)}$

1.6.3 g. La Complétude

La complétude (ou completness en anglais). Une mesure qui évalue dans quelle mesure tous les points d'une même classe sont regroupés dans un même cluster. Autrement dit, elle vérifie si les membres de chaque classe sont bien regroupés ensemble. Formellement, la complétude est définie comme suit :

$$\boldsymbol{Completude = 1 - \frac{H(L)}{H(C)}}$$

1.6.3 h. La V-Mesure

La V-Mesure (ou V-Measure en anglais). Une mesure qui est définie comme la moyenne harmonique pondérée de l'homogénéité (***H***) et de la complétude (***C***), connu sous le nom de score NMI (Normalized Mutual Information). Formellement, la V-Mesure est définie comme suit :

$$\boldsymbol{V - Mesure = 2 \times \frac{Homogénéité \times Completude}{Homogénéité + Completude}}$$

1.6.3 i. La Précision

La précision (ou precision en anglais). La mesure évalue la proportion de paires de points qui sont dans le même cluster et appartiennent à la même classe par rapport à toutes les paires de points qui sont dans le même cluster. Formellement, elle peut être définie comme suit :

$$\boldsymbol{Précision = \frac{\sum_{i=1}^{k} \max_{j} |C_i \cap L_j|}{N}}$$

1.6.3 j. Le Rappel

Le rappel (ou recall en anglais). Une mesure de la qualité du clustering qui évalue la proportion de paires de points d'une même classe qui sont correctement regroupées dans le même cluster. En d'autres termes, il mesure la capacité du clustering à regrouper tous les points d'une même classe dans le même cluster. Formellement, elle peut être définie comme suit :

$$\boldsymbol{Rappel = \frac{\sum_{i=1}^{k} \max_{j} |C_i \cap L_j|}{\sum_{j=1}^{m} |L_j|}}$$ Où :

$\boldsymbol{\sum_{j=1}^{m} |L_j|}$: est le nombre total de points dans toutes les classes.

1.6.3 k. La G-Mesure

La G-Mesure (ou G-Measure en anglais). Une mesure de performance qui combine la précision et le rappel en prenant leur moyenne géométrique. Cette mesure est utilisée pour évaluer la qualité du clustering en mettant en balance la capacité du modèle à regrouper correctement les points (rappel) et à éviter les regroupements incorrects (précision). Formellement, elle peut être définie comme suit :

$$\boldsymbol{G - Mesure} = \sqrt{\boldsymbol{Précision} \times \boldsymbol{Rappel}}$$

1.6.3 l. La F-Mesure

La F-mesure (ou F-Measure en anglais). Une mesure qui est connue également sous le nom de F1-score, qui mesure la précision globale d'un modèle qui combine la précision et le rappel en une seule métrique. Elle est particulièrement utile dans les cas où il est important d'équilibrer la précision et le rappel, car elle prend en compte à la fois les faux positifs et les faux négatifs. Formellement, elle peut être définie comme suit :

$$\boldsymbol{F - Mesure} = \frac{\boldsymbol{2 \times Précision \times Rappel}}{\boldsymbol{Précision + Rappel}}$$

Le plus souvent, il n'existe pas de classification connue à l'avance et pouvant servir de référence. Quand bien même une telle connaissance existe, cette dernière peut être fondée sur des informations qui ne se retrouvent ni dans la description proposée des données ni, indirectement, dans la matrice de (dis)similarité calculée à partir de ces descriptions. Il est donc nécessaire de pouvoir évaluer un schéma de façon autonome, c'est-à-dire en utilisant uniquement les informations dont l'algorithme dispose. Ce sont les critères d'évaluation interne ou relative qui sont alors utilisés.

1.6.4 Critères internes

Comparativement aux deux autres approches, il existe peu de mesures d'évaluation interne. Dans le cas d'un partitionnement, c'est l'indice $\boldsymbol{\Gamma}$ (interne) qui est généralement utilisé. En revanche, lorsqu'il s'agit de l'évaluation d'une méthode de clustering hiérarchique, on utilise de

coefficient de corrélation Co-phénétique (*CPCC*). Pour cela on construit la matrice Co-phénétique ($\boldsymbol{P_C}$) de la hiérarchie produite par l'algorithme de clustering, que l'on compare à la matrice de proximité ($\boldsymbol{P}$). Pour $\boldsymbol{M} = \frac{\boldsymbol{n(n-1)}}{\boldsymbol{2}}$, $\boldsymbol{\mu_p}$ et $\boldsymbol{\mu_c}$ les moyennes des matrices respectives $\boldsymbol{P}$ et $\boldsymbol{P_C}$.

1.6.4 a. Coefficient de corrélation Co-phénétique (*CPCC*)

Le coefficient *CPCC* est défini comme suit :

$$\boldsymbol{CPCC(PC, P)} = \frac{\frac{\boldsymbol{1}}{\boldsymbol{M}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n-1}} \sum_{\boldsymbol{j=i+1}}^{\boldsymbol{n}} \boldsymbol{P(i,j).PC(i,j)} - \mu\boldsymbol{p}.\mu\boldsymbol{c}}{\sqrt{[\frac{\boldsymbol{1}}{\boldsymbol{M}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n-1}} \sum_{\boldsymbol{j=i+1}}^{\boldsymbol{n}} \boldsymbol{P(i,j)^2} - \mu^2\boldsymbol{p}] + [\frac{\boldsymbol{1}}{\boldsymbol{M}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n-1}} \sum_{\boldsymbol{j=i+1}}^{\boldsymbol{n}} \boldsymbol{PC(i,j)^2} - \mu^2\boldsymbol{c}]\ ]}}$$

Cet indice prend ses valeurs entre -1 et 1, une valeur proche de 0 est significative d'une forte similarité entre les deux matrices et donc d'un bon schéma de clustering. La matrice Co-phénétique ($\boldsymbol{P_C}$), une matrice qui est définie par les valeurs $\boldsymbol{P_C}$ ($\boldsymbol{x_i, x_j}$) (ultra métriques), où chaque valeur correspond au premier niveau (similarité) de la hiérarchie, pour lequel les deux objets se retrouvent dans un même cluster.

Pour ce qui suit, il est a noté pour deux indices Calinski- Harabasz et Davies-Bouldin, $\boldsymbol{n}$ objets de données dans l'ensemble de données $\boldsymbol{X}$, avec $\boldsymbol{K}$ partitions indexées à partir de $\boldsymbol{i}$=1 jusqu'à $\boldsymbol{K}$. Notons :

- $\boldsymbol{n_i}$ est le nombre d'objets de données affectés au cluster $\boldsymbol{C_i}$.
- $\boldsymbol{m_i}$ est le centroïde lié au cluster $\boldsymbol{C_i}$.
- $\boldsymbol{m}$ est un vecteur centroïde total (moyen) de l'ensemble de données.
- $\boldsymbol{e_i}$, $\boldsymbol{e_j}$ sont l'erreur moyenne pour le cluster $\boldsymbol{C_i}$ et $\boldsymbol{C_j}$ respectivement.
- $\boldsymbol{D\ (C_i,\ C_j)}$ une fonction de distance entre les clusters $\boldsymbol{C_i}$ et $\boldsymbol{C_j}$ dans l'ensemble de données.

1.6.4 b. Indice de Calinski et Harabasz

L'indice est formulé comme suit :

$$\boldsymbol{CH\ (K)} = \frac{\boldsymbol{Tr(SB)}}{\boldsymbol{K-1}} \Big/ \frac{\boldsymbol{Tr(Sw)}}{\boldsymbol{n-K}} \quad \text{Avec}$$

$$\boldsymbol{Tr(SB)} = \sum_{i=1}^{k} \boldsymbol{n_i}||\boldsymbol{m_i} - \boldsymbol{m}||^2 \quad \text{et} \quad \boldsymbol{Tr(Sw)} = \sum_{i=1}^{k} \sum_{i=1}^{n_i} ||\boldsymbol{x_j} - \boldsymbol{m_i}||^2$$

1.6.4 c. Indice Davies-Bouldin

L'indice est formulé comme suit :

$$\boldsymbol{DB\ (K)} = \frac{1}{K}\sum_{i=1}^{k} \boldsymbol{R_i} \text{ Avec } \boldsymbol{R_i} = \max_{j,\ j \neq i} \left(\frac{e_i + e_j}{||m_i - m_j||^2}\right)$$

1.6.4 d. Indice Dunn

L'indice est formulé comme suit :

$$\boldsymbol{DI\ (K)} = \min_{i\ =\ 1..k} \left(\min_{j=1..k,\ j \neq i} \left(\frac{\mathrm{D}\ (C_i,\ C_j)}{\max\limits_{l=1..k} \delta\ (C_l)} \right) \right) \quad \text{Avec}$$

$$\mathrm{D}\ (C_i, C_j) = \min_{x \in C_i,\ y \in C_j} \boldsymbol{D(x,\ y)} \qquad et \quad \delta(C_l) = \max_{x,\ y \in C_l} \boldsymbol{D(x,\ y)}$$

1.7 Clustering basé sur l'apprentissage profond

Le clustering basé sur l'apprentissage profond est un domaine de recherche en pleine évolution avec un large éventail d'applications potentielles. Au fur et à mesure que les méthodes de clustering profond continuent de se développer et de s'améliorer, elles devraient jouer un rôle encore plus important dans l'analyse et la compréhension des données complexes.

Plusieurs approches de clustering basées sur l'apprentissage profond ont été proposées dans la littérature. Ces approches peuvent être classées en deux grandes familles :

a. Les méthodes pipeline pour l'apprentissage de la représentation à l'aide d'architectures DNN (Deep Neural Network), comme exemple, les perceptrons multicouches, les réseaux de neurones convolutifs (Convolutional Neural Network), ces méthodes pipeline sont largement discutées dans la littérature.

b. Les méthodes à modèle unique pour clustering de bout en bout.

1.7.1 Principe de fonctionnement du clustering basé sur l'apprentissage profond

Le développement d'un modèle d'apprentissage profond repose sur quatre étapes principales :

a. La première étape consiste à sélectionner et à préparer un ensemble de données d'entraînement**.** Ces données seront utilisées pour nourrir le modèle d'apprentissage profond pour apprendre à résoudre le problème pour lequel il est conçu. Les données doivent être soigneusement préparées, organisées et nettoyées. Dans le cas contraire, l'entraînement du modèle risque d'être biaisé.

b. La deuxième étape consiste à sélectionner un algorithme à exécuter sur l'ensemble de données d'entraînement. Le type d'algorithme à utiliser dépend du type et du volume de données d'entraînement et du type de problème à résoudre.

c. La troisième étape est l'entraînement de l'algorithme. Il s'agit d'un processus itératif. Des variables sont exécutées à travers l'algorithme, et les résultats sont comparés avec ceux qu'il aurait dû produire. Ces résultats peuvent ensuite être ajustés pour accroître la précision. On exécute ensuite de nouveau les variables jusqu'à ce que l'algorithme produise le résultat correct.

d. La quatrième et dernière étape est l'utilisation et l'amélioration du modèle. On utilise le modèle sur de nouvelles données, dont la provenance dépend du problème à résoudre. A titre d'exemple un modèle conçu pour détecter les spams sera utilisé sur des courriels.

1.8 Conclusion

Dans ce chapitre, nous avons, dans un premier temps, abordé le développement et la croissance assez rapide des données collectées dans les bases de données, est la naissance du Knowledge Discovery in Data base, un concept qui a mené à la naissance aux différentes approches de clustering. Nous avons aussi passé en revue les principales étapes du processus de clustering, un processus qui consiste à regrouper un ensemble d'objets de manière que les objets d'un même groupe soient plus similaires entre eux que ceux appartenant à d'autres groupes.

Dans un second temps, nous avons présenté la problématique générale du clustering en définissant les cinq étapes majeures de ce processus. Pour chacune de ces étapes, nous avons focalisé notre attention sur les distances et mesures de similarité, les différentes méthodes de clustering, et l'évaluation de la qualité du clustering à savoir interne ou externe pour une excellente évaluation et validation des schémas de clustering obtenus.

Pour le prochain chapitre, nous allons, introduire un nouveau concept pour le clustering non supervisé, un concept qui est fondamental en physique, mais qui est innovant, radical au clustering, nous parlons bien du champ potentiel qui permet de décrire le potentiel d'énergie d'un objet dans un champ, pour notre cas d'étude, l'ensemble des individus constituant les Data_Set.

Chapitre 2

Clustering par approche du champs potentiel

2.1 Introduction

La notion de champ a été introduite par les physiciens pour tenter d'expliquer comment deux objets peuvent interagir à distance, sans que rien ne les relie. À la fois, la loi de la gravitation universelle de Newton. [45] et la loi de Coulomb en électrostatique [46], impliquent une telle interaction à distance. Il n'y a pas de fil qui relie la terre au soleil, mais celui-ci exerce son attraction à distance. De même, deux charges électriques s'attirent ou se repoussent dans le vide sans que rien ne les relie, sans aucun support matériel. Pour tenter d'expliquer cela, *Michael Faraday.* [47] a introduit la notion de champ électrique. Si une charge q_A a un effet à distance sur une charge q_B qui se trouve éloignée, c'est parce que la charge q_A met tout l'espace qui l'entoure dans un état particulier, cette charge q_A par sa présence, produit en tout point de l'espace qui l'entoure, un champ électrique et c'est l'interaction de ce champ électrique avec la charge q_B qui produit la force que cette dernière ressent (figure 2.1). Cette notion de champ s'est révélée très utile et très pratique. Elle a pu être utilisée pour décrire d'autres forces fondamentales que la force électrique et elle permet de décrire les phénomènes de manière élégante.

2.2 La loi de Coulomb

Dans les années 1780 [48], le physicien français *Charles-Augustin de Coulomb* découvre expérimentalement l'expression décrivant le module de la force électrique que s'exercent deux charges électriques immobiles disposées sur des sphères. De nos jours, nous savons que la loi de Coulomb s'applique à toutes les particules pouvant être considérées comme étant ponctuelles. *Coulomb* réalise que le module de la force électrique est impliqué et dépend des paramètres suivants :

a. $F_e \propto q_A q_B$: La force électrique est proportionnelle au produit des deux charges q_A et q_B en attraction ou en répulsion.

b. $F_e \propto 1/r^2$: La force électrique est inversement proportionnelle au carré de la distance entre les deux charges.

c. $F_e \propto k$: La force électrique est proportionnelle à une constante afin d'évaluer la force électrique en newton.

2.2.1 La loi de Coulomb sous forme scalaire

La formule scalaire de la loi exprime la grandeur de la force électrostatique F_e entre deux charges ponctuelles q_A et q_B, situées à une distance r l'une de l'autre.

$$F_e = k\frac{|q_A q_B|}{r^2} \quad \text{Où :}$$

- F_e : Force électrique en newton (N).
- q_A : Charge #1 qui applique la force électrique sur la charge #2 en coulomb (C).
- q_B : Charge #2 qui applique la force électrique sur la charge #1 en coulomb (C).
- r : Distance entre les deux charges ponctuelles en mètre (m).
- k : Constante de la loi de Coulomb, $k = 9.00\times10^9\ N.m^2/C^2$.

2.2.2 La loi de Coulomb sous forme vectorielle

La formule vectorielle de la loi exprime non seulement la grandeur de la force électrostatique F_e, mais également sa direction. La force électrique nécessite un vecteur unitaire $\hat{r}$ désignant l'orientation radiale de la force électrique. Dans cette définition, il faut préciser quelle charge Q applique la force et quelle charge q subit la force.

$$F_e = k\frac{q\,Q}{r^2}\hat{r} \quad \text{Où :}$$

- Q : Charge qui applique la force électrique en coulomb en coulomb (C).
- q : Charge qui subit la force électrique en coulomb (C).

- $\hat{r}$: Vecteur unitaire orientation de Q source à q cible.

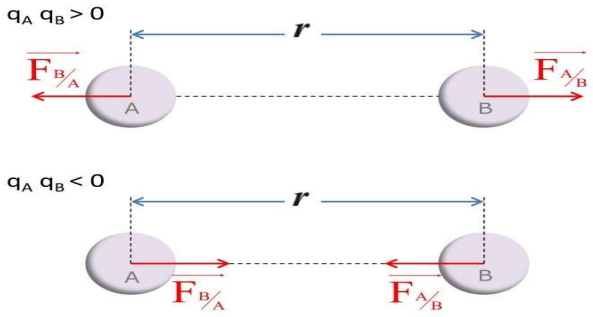

Fig. 2.1 – Force d'attraction et de répulsion.

2.3 Topographie du champ électrique.

La présence de charges sources dans une région de l'espace modifie les propriétés électriques de celle-ci en créant, en chaque point, un champ électrique [49]. On introduit alors le concept de lignes de champ. Le tracé de ces lignes donne une représentation spatiale du champ. Les figures 2.2 et 2.3 représentent les lignes de champ dues à une seule charge source Q. Si celle-ci est positive (+) le champ est dirigé de la charge vers l'extérieur (figure 2.2). Si la charge est négative (-), le champ est dirigé de l'extérieur vers la charge (figure 2.3). Chaque charge source crée des lignes de champ telles qu'elles sont représentées.

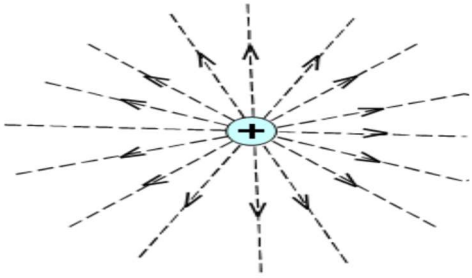

Fig. 2.2 – Les lignes de champ (Charge positive).

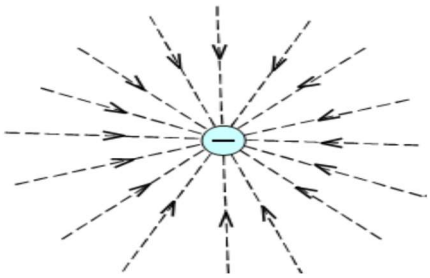

Fig. 2.3 – Les lignes de champ (Charge négative).

La mise en présence de deux charges, d'égale ou distincte valeur, entraîne une déformation des lignes de champ et on obtient une nouvelle topographie (figures 2.4). En chaque point, la ligne de champ est tangente au champ résultant.

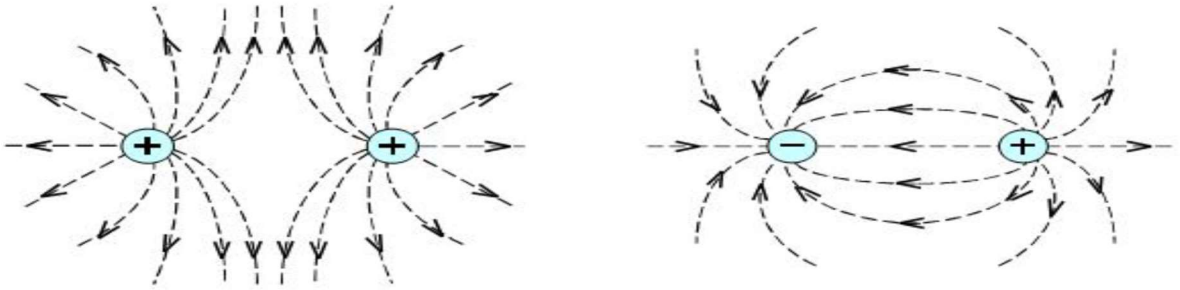

Fig. 2.4 – Les lignes de champ (Deux charges d'égale ou distincte valeur).

2.4 Champ et potentiel créés par des charges électriques

Précédent (figure 2.1), nous avons utilisé le concept de force d'interaction. Dans le cas de deux charges électriques par exemple, chacune des charges exerce sur l'autre une force dont l'expression mathématique est donnée par la loi de Coulomb. En vertu du principe de l'action et de la réaction de Newton, la seconde charge exerce sur la première une force égale et opposée. Ainsi les deux charges jouent le même rôle. Avec le concept de champ, le problème est posé d'une façon différente. Une charge électrique $\boldsymbol{Q}$, appelée '*charge source'*, crée dans l'espace environnant, appelé '*champ*', un '*état*' qui est mis en évidence par son action sur toute autre charge $\boldsymbol{q}$ placée en un point $\boldsymbol{M}$ de cet espace [50].

2.4.1 Charge ponctuelle unique

a. Soit une charge ponctuelle $\boldsymbol{Q}$ immobile, dans son voisinage, toute charge $\boldsymbol{q}$ subit une force électrique $\boldsymbol{\vec{F}}$.

$$\vec{F} = K\frac{Qq}{r^2}\vec{u}$$ (Loi de Coulomb) Où :

- $\vec{u}$ le vecteur unitaire pointant de Q vers q, K est la *c*onstante de la loi de Coulomb.
- r la distance entre les deux charges.

Les charges peuvent être positives ou négatives. Deux charges positives (ou négatives) se repoussent (figure 2.1), deux masses s'attirent, ce qui explique la différence de signe par rapport à la loi de Newton.

b. Soit un champ électrique $\vec{E}$, une région de l'espace où une charge électrique subit une force. Il est défini par la force électrique $\vec{F}$ agissant sur une charge q placée en un point donné.

$$\vec{E} = K\frac{Q}{r^2}\vec{u} \quad \text{Soit aussi} \quad \vec{E} = \vec{F}/q$$

c. Soit le potentiel électrique $\vec{V}$, qui est défini comme le travail nécessaire pour déplacer une charge q d'un point de référence à un point donné dans le champ sans accélération.

$$\vec{V} = K\frac{Q}{r} \quad \text{Soit aussi} \quad \vec{E} = -\,grad(\vec{V})$$

2.4.2 Deux charges ponctuelles (Principe de superposition)

Considérons le cas de deux charges ponctuelles fixes q_1 et q_2 agissant sur une troisième charge q.

a. Lorsque plusieurs forces agissent sur une charge q, la force résultante est la somme vectorielle de toutes les forces individuelles.

$$\vec{F} = \vec{F}_1 + \vec{F}_2 = K\frac{q_1 q}{r_1^2}\vec{u_1} + K\frac{q_2 q}{r_2^2}\vec{u_2}$$

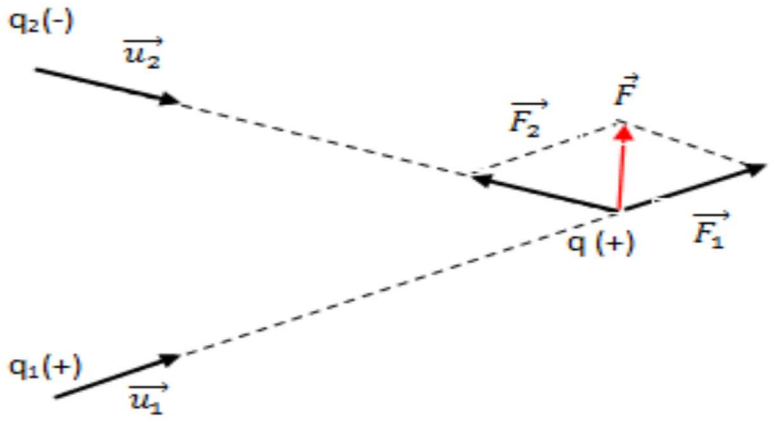

Fig. 2.5 – La superposition des forces.

b. Soit deux charges électriques q_1 et q_2, le champ électrique ressenti à ce point est la somme vectorielle des champs électriques créés par chacune de ces charges.

$$\vec{E} = \vec{E}_1 + \vec{E}_2 = K\frac{q_1}{r_1}\vec{u_1} + K\frac{q_2}{r_2}\vec{u_2}$$

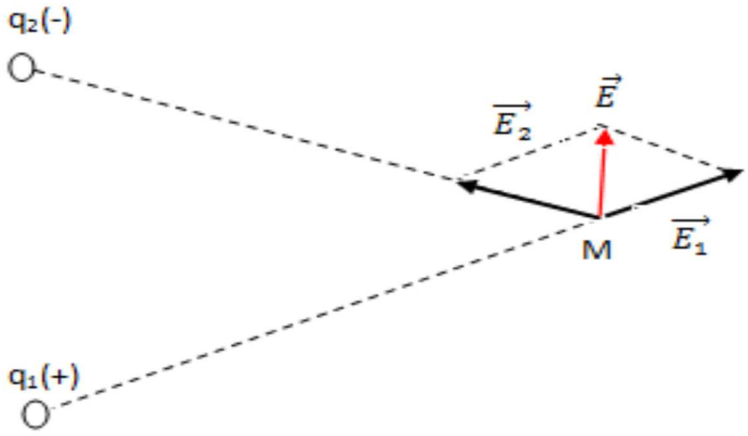

Fig. 2.6 – La superposition des champs.

c. Soit deux charges électriques q_1 et q_2, le potentiel ressenti à ce point est la somme algébrique des potentiels créés par chacune de ces charges.

$$\vec{V} = V_1 + V_2 = K\frac{q_1}{r_1} + K\frac{q_2}{r_2}$$

2.5 Clustering et champ potentiel

La relation entre le clustering et le concept du champ potentiel réside dans une idée qui est simple mais puissante, d'attirer des points de données similaires entre eux. En clustering, l'objectif est de regrouper les points de données qui se ressemblent plus qu'ils ne ressemblent aux points d'autres groupes. Cela peut être visualisé comme un champ potentiel, où des points de données similaires agissent comme des attracteurs, attirant d'autres points similaires proches vers eux. Le principe est que chaque point donné génère un champ potentiel qui influence les autres points proches. Les points de données similaires s'attireront les uns les autres, formant des clusters naturels.

Plus clairement, pour chaque élément qui est représenté par un point, une coordonnée cartésienne ou plus complexe un individu avec un vecteur d'attributs, l'application et l'attribution du champ potentiel revient a calculé l'influence de la force générée par l'ensemble des voisins. L'application du champ change et modifie les valeurs de départ jusqu'à obtention d'une superposition entre les individus similaires.

2.6 Conclusion

Dans ce chapitre, nous avons, évoqués les concepts de la force électrique $\vec{F}$, qui est générée par une ou plusieurs charges électriques, le champ électrique $\vec{E}$, un champ qui décrit l'influence de la force d'une ou de plusieurs charges sur son environnement. Les deux principes qui sont fondamentaux pour la physique nous conduisent à l'étude du champ potentiel, qui est défini

comme un travail qu'il faut effectuer pour déplacer une charge ponctuelle d'une référence donnée à un point donné pour le cas du potentiel électrique, ou une masse pour le cas du potentiel gravitationnel.

Pour ce qui suit, le phénomène du champ potentiel nous fascine pour l'étudier de prêt et l'appliquer sur l'ensemble des données constituant les bases de données, pour notre cas d'expérimentations les fleurs, spécifiquement iris une fleur qui est caractérisée par la longueur et largeur de ces pales et pétales.

Pour le prochain chapitre, nous allons aborder la conception et l'implémentation d'une idée abstraite a une solution concrète. Nous exposons le diagramme de classe, cas d'utilisation et diagramme d'activité.

À la fin, nous formulons l'implémentation de l'algorithme, une approche différente pour résoudre la question du clustering non supervisé.

Chapitre 3

Conception et implémentation

3.1 Introduction

Le fait qu'aucun algorithme de clustering ne puisse résoudre tous les problèmes de clustering a conduit au développement de plusieurs algorithmes de clustering avec des applications diverses.

Dans ce chapitre, nous essayons d'introduire et de présenter un algorithme de clustering non supervisé, ou nous utilisons comme technique de classification un phénomène qui est très connu en physique, le champ potentiel, l'application de ce phénomène nous mène à bénéficier de ces concepts et caractéristiques pour regrouper des données non étiquetées afin d'extraire des informations significatives. Pour ce qui suit aussi, nous invoquons les différentes classes est attributs constituant l'architecture de l'algorithme. Le langage UML s'impose ici, un langage de modélisation unifié, qui a été pensé pour être un langage de modélisation visuelle commun. Il est destiné à l'architecture, la conception et la mise en œuvre des systèmes. Nous utilisons comme diagramme décrivons le système les trois diagrammes, diagramme de classes, cas d'utilisation et diagramme d'activité, ensuite, nous présentons l'algorithme en détail, ces différents parties et l'utilisation du dite concept, champs potentiel, sur l'ensembles des données d'entrées.

3.2 Conception

3.2.1 Digramme de classes

Il permet de modéliser les classes, leurs attributs, leurs opérations et les relations entre elles. Les diagrammes de classe sont utilisés pour modéliser la structure d'un système, visualiser les

relations et les interactions entre les différentes classes, documenter les attributs et opérations des classes, et planifier et conceptualiser la conception orientée objet avant le codage.

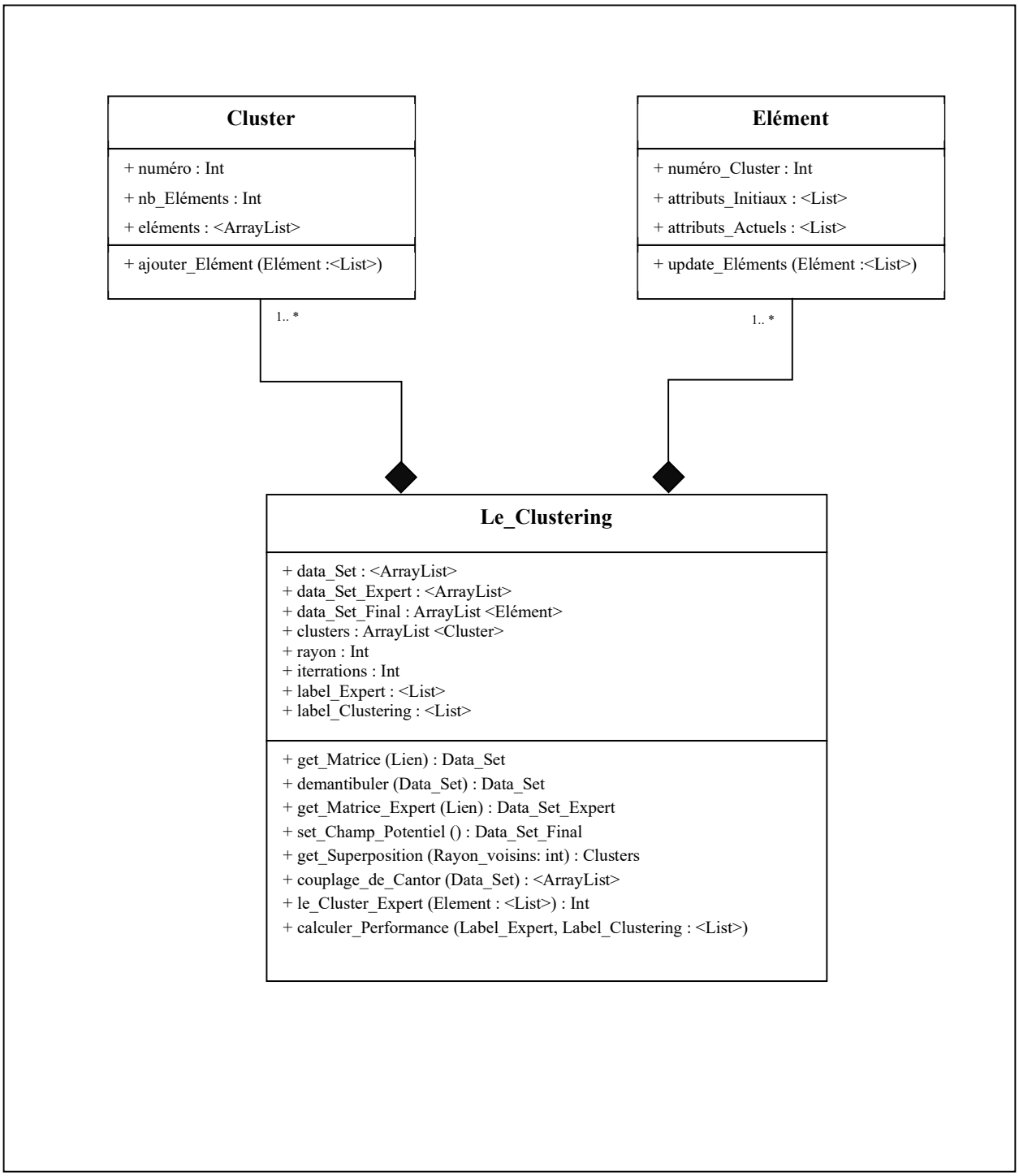

Fig. 3.1 – Diagramme de Classes.

3.2.2 Digramme de cas d'utilisation

Un diagramme de cas d'utilisation est outil pour modéliser les interactions entre un système logiciel et ses utilisateurs externes. Il permet de décrire les fonctionnalités du système du point de vue de l'utilisateur et d'identifier les acteurs qui interagissent avec le système.

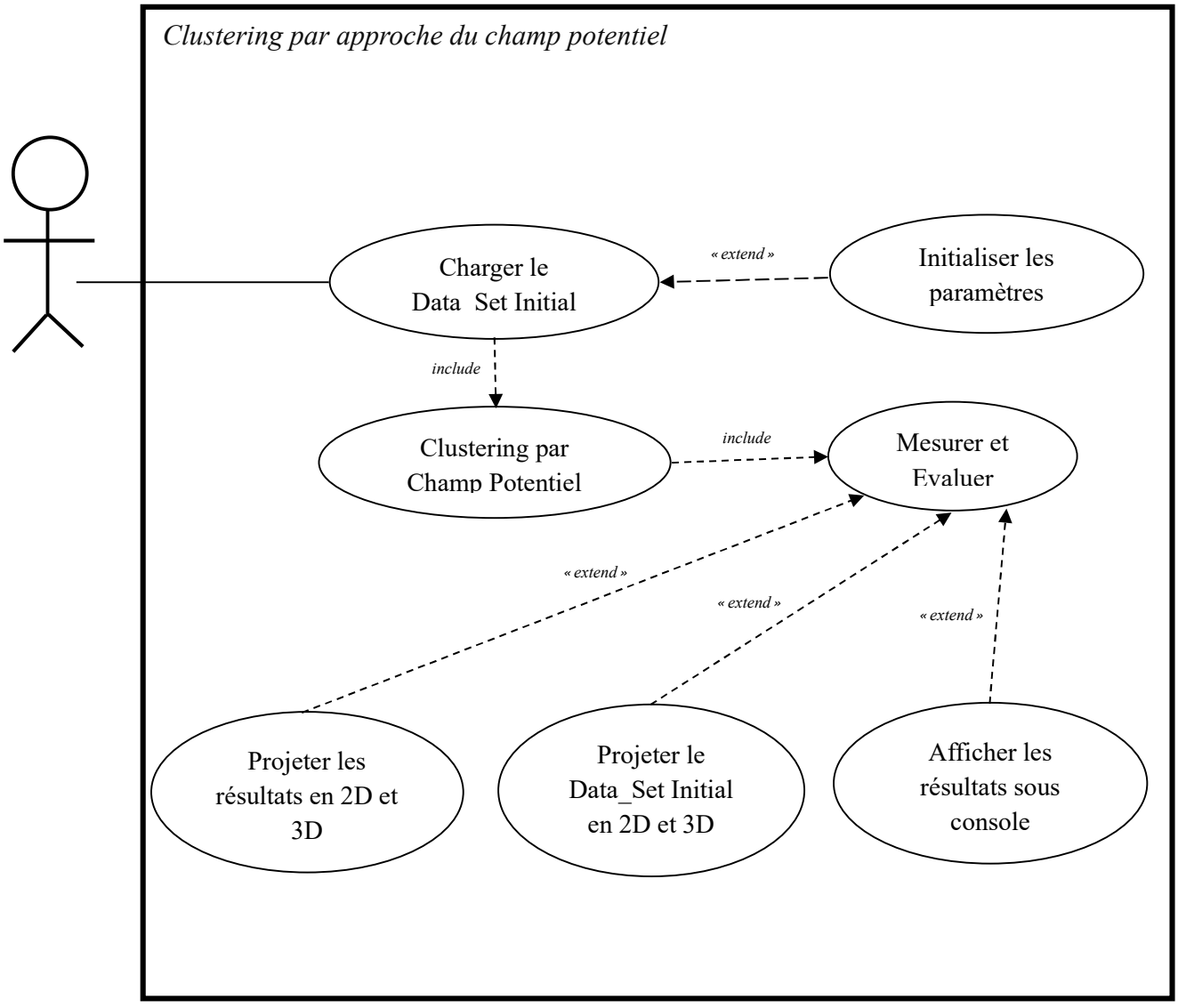

Fig. 3.2 – Digramme cas d'utilisation.

3.2.3 Diagramme d'activité

Un diagramme d'activité est un outil qui modélise le comportement dynamique du système, il peut représenter des actions, des décisions, des boucles, des synchronisations. Il permet de vérifier la complétude d'un processus.

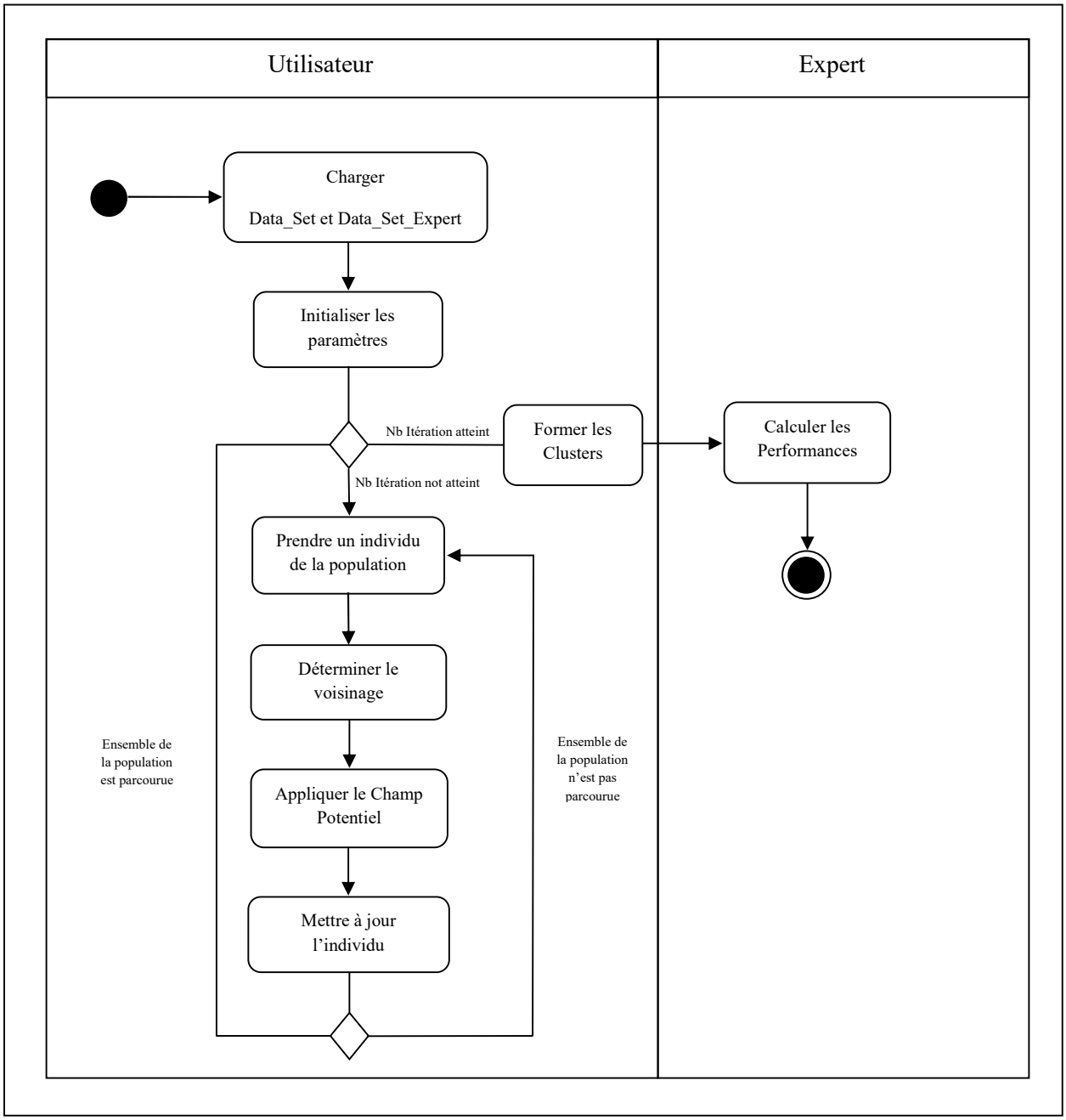

Fig. 3.3 – Digramme d'activité.

3.3 Implémentation

3.3.1 Environnement et développement

a. Environnement matériel : nous utilisons une machine DELL LATITUDE 7430 avec12th Gen Intel(R) Core (TM) i7-1265U 1.80 GHz, et une RAM 32,0 Go (31,4 Go utilisable).

b. Environnement logiciel : nous exploitons la machine sous un système d'exploitation Windows 11 Professionnel 64 bits, version 23H2 installé le 05/06/2023.

c. Environnement de développement : l'utilisation du langage Python sous l'interpréteur 3.12.2 64-bit est imminente, un langage de programmation généraliste et multiparadigme, le choix été pour ça puissance de calcul et ces librairies.

d. Environnement de conception graphique : nous utilisons Qt Designer, un outil essentiel pour les développeurs utilisant le framework Qt, un outil WYSIWYG (What You See Is What You Get), facilitant la création rapide et efficace d'interfaces utilisateur interactives et visuellement attrayantes. Grâce à son interface intuitive et ses nombreuses fonctionnalités, il permet de concevoir des applications multiplateformes de haute qualité sans avoir à écrire de code d'interface utilisateur manuel.

3.3.2 Présentation de l'algorithme

Pour ce qui suit nous allons exprimer l'implémentation des différentes étapes du processus, en premier lieu nous chargeons le Data_Set sous forme de tableau bidimensionnel numpy. L'initialisation des paramètres qui s'impose en second lieu, un rayon pour déterminer si un élément est assez proche ou non, aussi la valeur de déplacement et une distance minimale pour déterminer la superposition des individus.

L'algorithme exécute une boucle d'itération. Ce nombre reflète la convergence entre éléments, pour chaque itération l'algorithme calcul le cumul des forces qui est généré par l'ensemble des voisins, à la fin de chaque itération le principe du champ est appliqué sur l'individu ensuite ces coordonnées respectives seront mis à jour, cette mise à jour permette de déplacer les éléments entre eux jusqu'à convergence sur un point de superposition.

La sortie de la boucle d'itération fournit un résultat sous forme de points, nous précisons que le point représente un ensemble d'éléments superposé. Comme exemple, le Data_Set iris des fleurs de cent cinquante éléments, l'algorithme superpose les dits éléments de sept à neuf ensemble de points, ces points seront assez proches pour former des clusters naturels, un simple regroupement permet de créer les clusters effectifs.

Algorithme Clustering par Approche du Champ Potentiel

```
Input: Number of Iterations, Neighborhood Radius.
Output: The_Clusters.
Import Data_Set and Data_Set_Expert.
Initialization of parameters.
Repeat
    For each Element_I in Data_Set do
        For each Element_J in Data_Set do
            if Element_J is in neighborhood radius of Element_I then
                Calculer le cumul des forces générés par les voisins.
        End for.
        Calculer le champ potentiel appliqué par le voisinage.
        Mettre à jour les attributs de l'Element_I.
        Mettre à jour le Data_Set.
    End for.
until number of iterations is reached.
Regrouper_Les_Voisins.
Return: Data Set Clustered.
```

Fig. 3.4 – Algorithme du clustering par approche du champ potentiel.

3.3.3 Exemple illustratif

Nous allons illustrer un cas concret pour clarifier le concept, l'idée principale pour l'analyse des données en apposant un algorithme du clustering par application du champ potentiel. Il s'agit d'une instance spécifique qui aide à rendre la notion abstraite du champ potentiel plus compréhensible.

Nous saisissons un data set suffisamment simple, disperser et explicatif, les coordonnées respectives sont Data_Set = [[-2, 2], [-1, 2], [-0.5, 1.5], [0, 0.5], [1, 0], [1, 1]].

La figure 3.5 représente une projection en 3D du Data_Set, avec une fixation de l'axe Z à zéro.

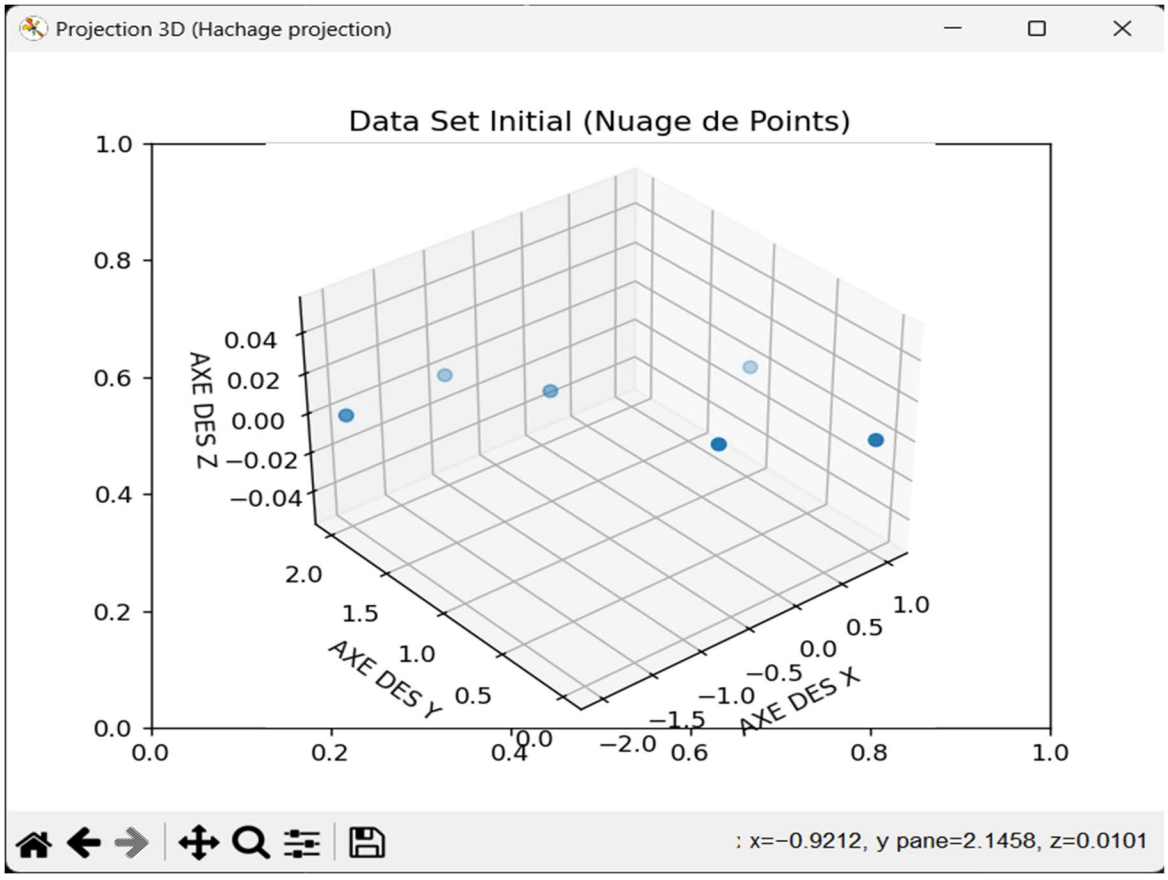

Fig. 3.5 – Data_Set test.

L'initialisation des paramètres, revient à définir un rayon pour le calcul de l'ensemble des voisins, une valeur minimale pour décider si un élément est si proche, le pad de déplacement et le nombre d'itération fixé à deux cents.

La figure 3.6 illustre le phénomène d'attraction entre éléments, après 30 itérations les éléments les plus proches et les plus similaires commencent a voyagé entre eux pour se rapprocher.

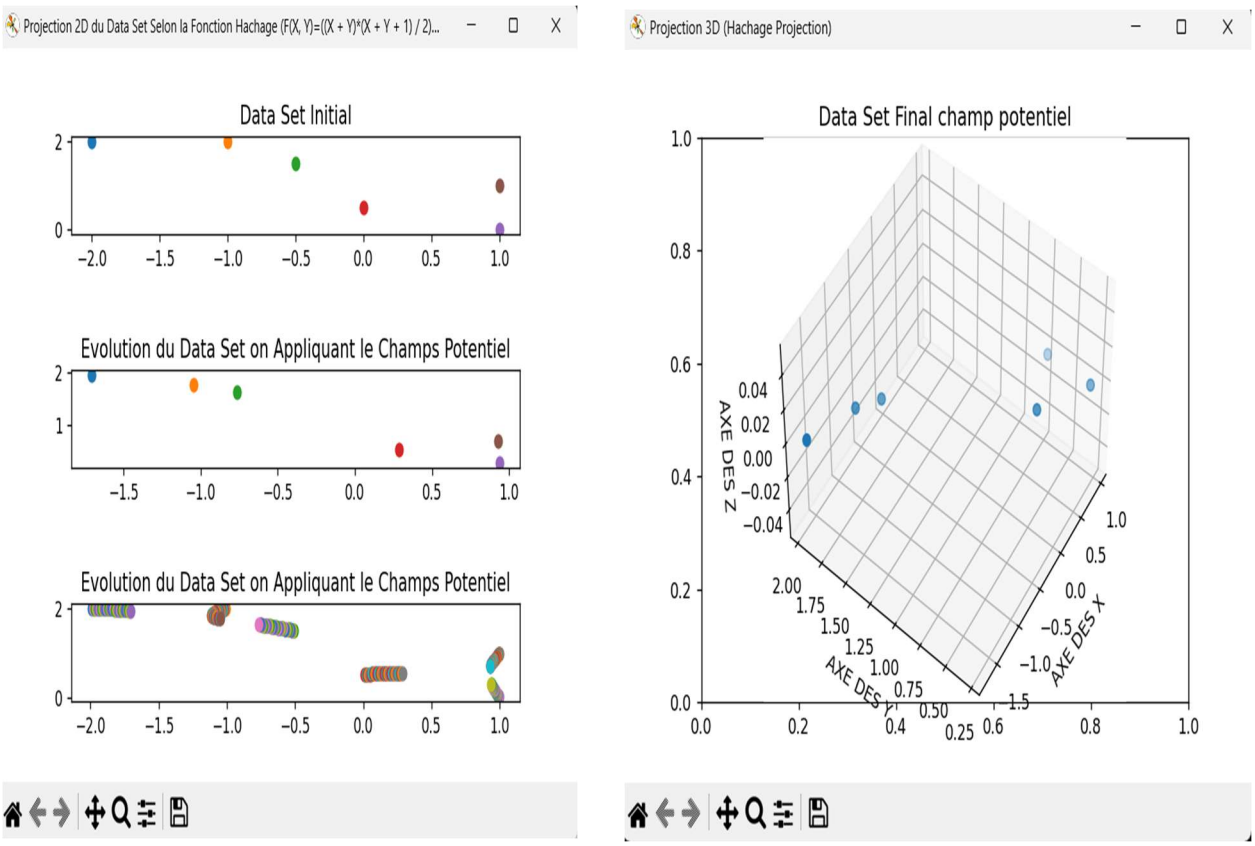

Fig. 3.6 – Evolution du Data_Set test (Après 30 itérations).

La figure 3.7 illustre aussi les attractions entre éléments, après 50 itérations les éléments commencent à se regrouper en clusters cohérents. Le nombre d'éléments passent de six à quatre, les plus similaires commencent à se superposer pour former un seul point.

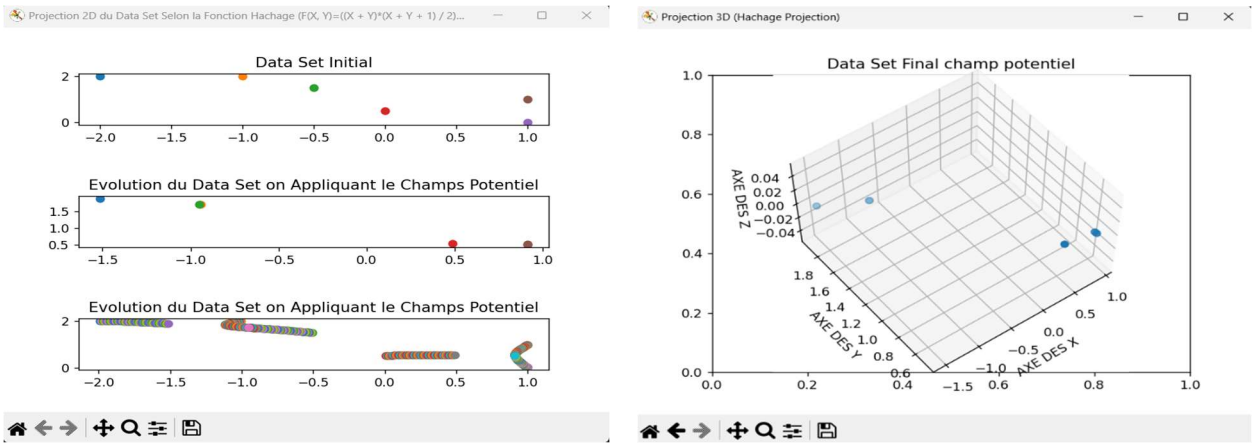

Fig. 3.7 – Evolution du Data_Set test (Après 50 itérations).

La figure 3.8 illustre la convergence du Data Set après 80 itérations, le système passe de six à deux individus superposés. La superposition et la transformation du Data_Set sont visibles. Les deux clusters sont représentés par deux nouvelles coordonnées respectivement [1.22, 1.80] et [0.70, 0.52].

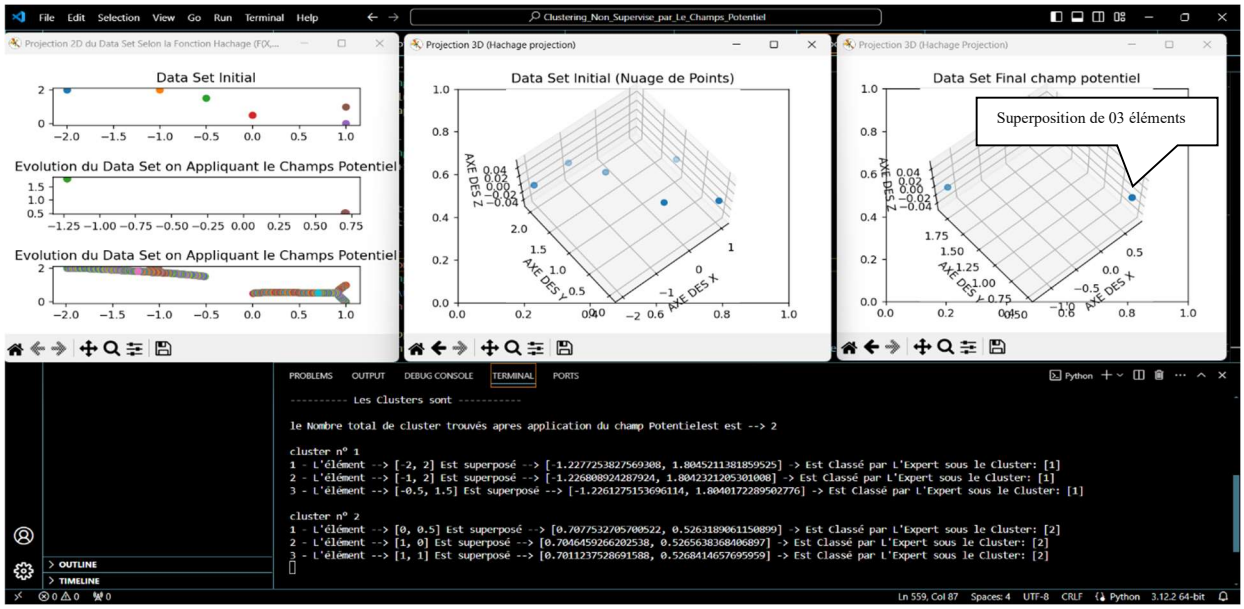

Fig. 3.8 – Superposition du Data_Set test (Après 80 itérations).

3.4 Application de l'algorithme sur la fleur Iris

Iris est l'un des ensembles de données les plus connus et les plus utilisés en statistique et en science des données, le Data Set est constitué de cent cinquante fleurs avec quatre attributs. Les fleurs sont classées par l'expert en trois classes, *Setosa*, *Versicolor* et *Virginica*.

La figure 3.9, nous montre la répartition des éléments constituant le Data Set complet en trois 3D. Une meilleure projection exige la transformation des quatre attributs constituant la population des fleurs en deux attributs tout en gardant l'emprunte initial des éléments. La transformation suit une technique communément appelée la fonction de couplage de Cantor, une formule utilisée pour mapper une paire d'entiers non négatifs ($\boldsymbol{X}$, $\boldsymbol{Y}$) à un seul entier non négatif de manière unique. Formellement :

$$\boldsymbol{f(X,Y) = \frac{(X+Y)*(X+Y+1)}{2} + Y}$$

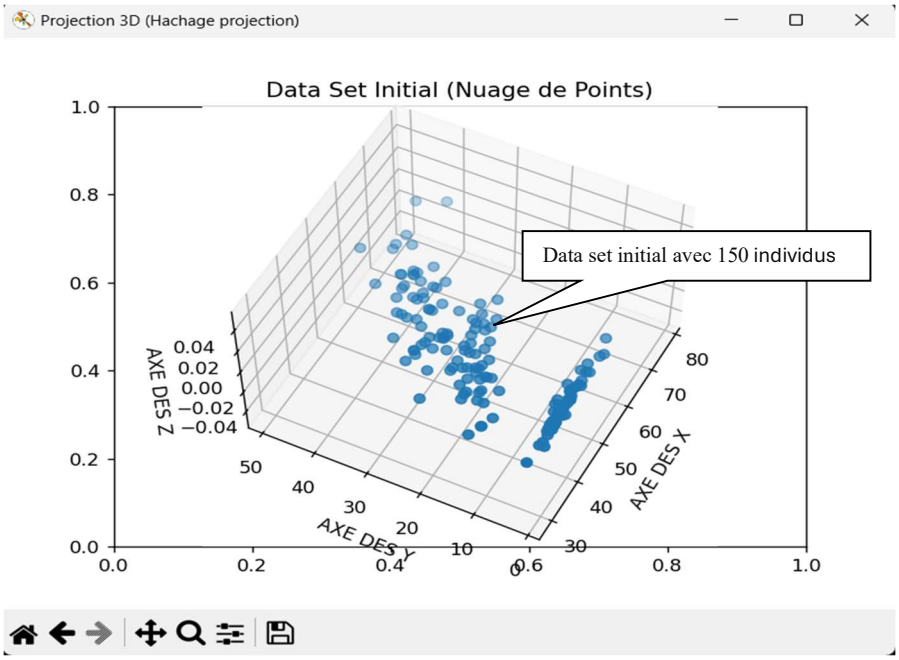

Fig. 3.9 – Projection du Data_Set (Fleur Iris).

Après deux cent cinquante itérations, la convergence du system nous révèle les résultats exposés dans la figure 3.10, la projection suit aussi la formule de couplage de cantor, la formation des trois clusters est claire, les éléments finissent à se regrouper entre eux pour former des clusters naturels.

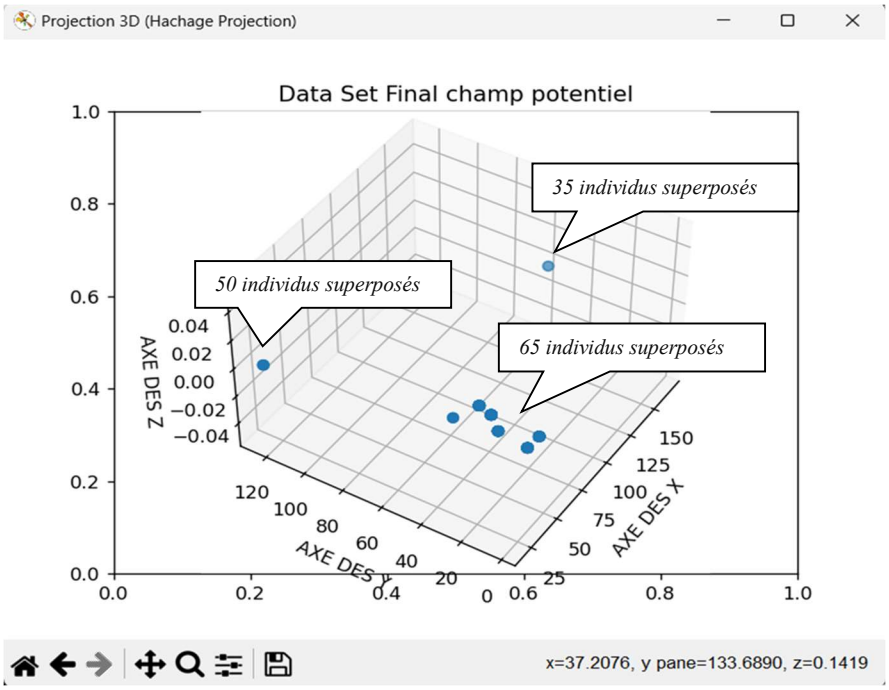

Fig. 3.10 – Application du champ potentiel et superposition du Data_Set (Fleur Iris).

3.5 Conclusion

Le long de ce chapitre, nous avons présenté une conception générale du système avec les différents diagrammes de classe, de séquence et cas d'utilisation. L'architecture est simple, basic ou les interactions entre le système et l'utilisateur externe n'est pas considérable, de même, la structure interne du système, les relations et les interactions entre les différentes classes est néo-moins simple. Pour les futurs travaux, nous envisageons une architecture en parallèle avec en

concurrence l'exécution de différents processus pour un résultat plus optimal et une convergence rapide.

Le point clé de ce chapitre, est la présentation de l'algorithme décrivant l'ensemble des étapes, et le développement de l'approche pour un clustering non supervisé.

À la fin du chapitre, nous exposant des scénarios réels de l'algorithme sur un ensemble de données qui sont simples et explicatives, ensuite, vient une application puissante sur un Data_Set qui est réputé dans la communauté est qui iris. Nous arguments les résultats obtenus par l'appui des projections dans différents plans et perspectifs en 2D et 3D, le but est l'obtention d'une meilleure visualisation. Ces techniques sont essentielles pour transformer l'ensemble de données brutes en informations compréhensibles et exploitables. En choisissant la bonne technique de visualisation, nous pouvons faciliter l'interprétation des données et améliorer la prise de décision.

Chapitre 4

Evaluation et discussion

4.1 Introduction

Dans ce chapitre, nous allons aborder un point qui est le fruit de toute étude, l'évaluation et la discussion, qui sont des éléments essentiels d'un article scientifique. L'évaluation permet de présenter les données recueillies lors de l'expérimentation et de les analyser en profondeur, couplée à une présentation des données qui se fait d'une manière claire et consistante, en utilisant différents outils afin de faciliter la compréhension. En second lieu, la discussion et l'interprétation des résultats obtenus en les reliant aux hypothèses de recherche et aux objectifs de l'étude. À la fin la validation des résultats en comparant les résultats obtenus aux travaux antérieurs en tenant compte des limites de l'expérimentation.

4.2 Benchmarks

Qu'est-ce qu'un benchmark ?

Le benchmark ou benchmarking est une technique marketing développée dans les années 80, qui signifie repère en anglais. Le benchmark désigne tous les moyens permettant d'évaluer les performances par comparaison avec les concurrents. Pour notre cas d'étude, nous se référençons aux deux benchmarks Iris pour les fleurs et Seeds pour les grains de blé, les Data_Set sont référencés au UC Irvine Machine Learning Repository [51].

4.2.1 La fleur Iris

Les iris sont des plantes vivaces à bulbes ou à rhizome de la famille des iridacées, l'espèce comprends, pas moins de deux cents qualités. Pour notre étude (tableau 4.1), le Data Set récupéré est de cent cinquante fleurs, regroupées en trois classes cinquante chacune, *Setosa*, *Versicolor* et

la dernière *Virginica* (figure 4.1), avec quatre attributs pour chacune, les attributs ne sont que la longueur et largeur des sépales et pétales.

TAB. 4.1 – Benchmark Iris.

Benchmark	Nombre d'instances	Nombre d'attributs	Nombre de classes
Iris	150	4	3

Iris Setosa **Iris-versicolor** **Iris-virginica**

Fig. 4.1 – La fleur Iris.

4.2.2 Grains de blé

Le blé est un terme générique qui désigne plusieurs céréales appartenant au genre triticum. Ce sont des plantes annuelles de la famille des graminées. Pour notre étude (tableau 4.2), le Data_Set récupéré est de deux cent dix graines, regroupées en trois classes soixante-dix chacune, *Kama*, *Rosa* et la dernière *Canadian* (figure 4.2), avec sept attributs pour chacune, les attributs ne sont que la surface, le périmètre, la compacité $\boldsymbol{C = \frac{4\pi A}{P^2}}$, la longueur et largeur du grain, le coefficient d'asymétrie, et la longueur de la rainure du noyau.

TAB. 4.2 – Benchmark Seeds.

Benchmarks	Nombre d'instances	Nombre d'attributs	Nombre de classes
Seeds	210	7	3

Canadian **Kama** **Rosa**

Fig. 4.2 – Grains de blé.

4.3 Algorithmes et paramétrages utilisés pour l'analyse

Les performances de l'approche par champ potentiel sont comparées aux algorithmes les plus récents bien connus rapportés dans la littérature [52]. Nous allons procéder à une comparaison étalée sur deux étapes.

a- Pour ce qui suit la première étape, nous allons exposer un comparatif entre l'algorithme du clustering par champ potentiel avec les quatre approches évolutives, Deferential Evolution (DE) [53], Particle Swarm Optimization (PSO) [54], Genetic Algorithm (GA) [55], Multi-verse Optimizer (MVO) [56]. L'évaluation comporte l'utilisation de quatre mesures populaires implémentés dans le module scikit-learn de python :

- L'homogénéité est définie sous la fonction metrics.homogeneity_score().
- La complétude est définie sous la fonction metrics.completness_score().
- La V-Mesure est définie sous la fonction metrics.v_measure_score().
- La pureté : a été implémentée.

Les paramètres des techniques utilisées sont répertoriés dans tableau 4.3. Le nombre maximum d'itérations pour les algorithmes d'optimisations est fixé à 200, la taille de la population est réglée sur 50 et les résultats sont collectés sur 10 exécutions indépendantes [57].

TAB. 4.3 – Paramètres des algorithmes suivant la littérature (1).

Algorithme	Paramètres	Valeurs
PSO	*F*acteur d'inertie C_1 C_2	0.1 2 2
DE	*C C*roisement *F F*acteur d'échelle	0.5 [0.2 - 0.8]
GA	*C*roisement *M*utation	0.8 0.02
MVO	WEP_{max} WEP_{min}	1 0.2

b- Pour la deuxième étape, les performances de l'approche par champ potentiel sont comparées aux algorithmes les plus récents bien connus rapportés dans la littérature [58], notamment, K-Means (KM) [59], Genetic Algorithm Clustering (GAC) clustering qui est basé sur le génétique[d], Harmony Search Clustering (HSC) clustering qui est basé sur la recherche d'harmonie [60], Multimodal Harmony Search Clustering (MHSC) des algorithmes qui sont modifiés sur la base des algorithmes d'harmonies, Particle Swarm Optimization Clustering (PSOC) clustering qui est modifié sur la base des algorithmes d'optimisation par essaim de particules [61], Flower Pollination Algorithm Clustering (FPAC) clustering basé sur un algorithme de pollinisation des fleurs [62], et le dernier, Bat Algorithm Clustering (BATC) clustering basé sur un algorithme de chauve-souris [63]. L'évaluation comporte l'utilisation de trois mesures populaires implémentés dans le module scikit-learn de python :

- La précision est définie sous la fonction metrics.precision_score().
- Le rappel est défini sous la fonction metrics.recall_score().
- La G_Mesure est déduite, la racine carrée de la précision * le rappel

Les paramètres des techniques utilisées sont répertoriés dans tableau 4.4. La taille de la population et le nombre maximum d'itérations pour tous les algorithmes sont définis sur trente et cinq cents. Les réglages des paramètres de tous les algorithmes mentionnés ci-dessus sont les suivants.

TAB. 4.4 – Paramètres des algorithmes suivant la littérature (2).

Algorithme	Paramètres	Valeurs
GAC	*P*robabilité de *C*roisement	0.8
	*P*robabilité de *M*utation	0.01
HSC	*T*aux de *C*onsidération de la *M*émoire *H*armonique (HMCR)	0.9
	Taux d'*A*justement de *P*as (PAR)	0.5
	*L*argeur de la *B*ande (BW)	0.01
MHSC	HMCR (*H*armony *M*emory *C*onsideration *R*ate)	[0.5 - 0.95]
	PAR (Pitch Adjustment Rate)	[0.01 - 0.99]
	BandWidth (BW)	[0.001 - 0.1]
PSOC	*P*oids d'inertie (w)	0.72
	*C*onstantes d'accélération	1.49
	*V*itesse maximale (V_{max})	255
FPAC	*P*robabilité de *C*hangement	0.8
BATC	*V*olume Sonore	[1-2]
	Taux d'*E*mission des impulsions	[0-1]
	*C*oefficients constants α et γ	0.9
GWAC	Centre des K clusters	K aléatoire

4.4 Résultats et analyses.

4.4.1 Champ potentiel vs MVO

Comme déjà mentionné dans le point 4.3.a, ce qui suit est la première comparaison qui s'effectue sur les deux benchmarks iris de 150 instances, un rayon égal à 1 et 180 est le nombre

d'itérations. Pour le benchmark seeds, 210 instances, un rayon égal à 10 et 180 est le nombre itérations. Le score élevé sera indiqué en ***italique gras***.

TAB. 4.5 – Algorithmes dans la littérature vs Champ potentiel (Benchmark Iris) (1).

Jeu de données Iris	Homogénéité	Complétude	V-Mesure	La pureté
DE	0.72778	0.75507	0.74096	0.86733
PSO	0.65750	***0.82877***	0.72629	0.77133
GA	0.60002	0.69056	0.64046	0.75333
MVO	0.73642	0.74749	0.74191	0.88667
Champ Potentiel	***0.78692 (fréquent)*** ***0.88458 (Meilleur)***	0.80937 ***(fréquent)*** ***0.88555 (Meilleur)***	***0.79799 (fréquent)*** ***0.88506 (Meilleur)***	***0.90000 (fréquent)*** ***0.96667 (Meilleur)***

Nous exposons dans le tableau 4.5, les quatre paramètres chiffrés homogénéité, complétude, V-Mesure et la pureté pour l'analyse et comparaison du Data_Set iris. On peut voir l'application du champ potentiel réalise un score meilleur par apport aux rivaux, pour qu'il soit de l'homogénéité, la V-Mesure et la pureté (figure 4.3), à l'encontre d'une, qui est suffisamment proche à la complétude de l'algorithme PSO. Nous prenons comme référence la comparaison avec l'algorithme Multi-verse Optimizer (MVO), un algorithme moderne d'intelligence inspiré par la théorie du multi-univers en astrophysique, le concept clé de l'algorithme MVO repose sur l'interaction de différents univers (les trous blancs, les trous noirs et les trous de ver) [57], chacun représente une solution potentielle au problème d'optimisation.

1- Pour l'homogénéité, l'approche du champ potentiel réalise un score fréquent de ***0.78692*** contre 0.73642 pour l'algorithme MVO. Lorsque on compare les deux scores, on peut conclure que l'algorithme du champ potentiel a une meilleure capacité à créer des clusters homogènes que celui de l'algorithme MVO. En pratique, une homogénéité plus élevée est généralement souhaitable, car elle reflète une meilleure séparation des classes dans les clusters produits, ce qui est souvent l'objectif dans les tâches de clustering supervisé.

2- Pour la complétude l'algorithme PSO réalise un score de ***0.82877*** contre 0.80937 pour l'approche du champ potentiel. Lorsque l'on compare les deux scores, on peut conclure que l'algorithme PSO est légèrement meilleur pour regrouper les éléments de la même classe ensemble dans un seul cluster par rapport à l'approche du champ potentiel avec un score fréquent de 0.80937. Bien que les deux scores soient élevés, le deuxième score montre aussi une meilleure performance en termes de complétude. Un score de complétude élevé est important, car il garantit que les clusters formés par l'algorithme contiennent le maximum possible d'éléments d'une même classe, réduisant ainsi la dispersion des éléments similaires à travers différents clusters. Cela est particulièrement crucial dans les applications où une forte séparation des classes est nécessaire.

3- Pour la V-Mesure l'approche du champ potentiel réalise un score fréquent de ***0.79799*** contre 0.74191 pour l'algorithme MVO. Lorsque on compare les deux scores, on peut conclure que l'approche du champ potentiel produit des clusters de haute qualité. Un score de ***0.79799*** est assez élevé, ce qui signifie que les clusters formés sont à la fois relativement homogènes et complets. Autrement dit, chaque cluster contient principalement des éléments d'une seule classe (homogénéité), et les éléments d'une même classe sont principalement regroupés dans un seul cluster (complétude). En résumé, le résultat obtenu indique que l'approche du champ potentiel est plus efficace pour créer des clusters de haute qualité, équilibrant mieux l'homogénéité et la complétude. Ce score reflète aussi une meilleure capacité à former des clusters utiles et significatifs pour l'analyse des données.

4- Pour la pureté l'approche du champ potentiel réalise un score fréquent de ***0.90000*** contre 0.88667 pour l'algorithme MVO. Lorsque on compare les deux scores, on peut conclure que 90% des éléments des clusters correspondent à la classe majoritaire au sein de chaque cluster. Cette performance indique que les clusters produits par l'algorithme sont très purs. Un tel score reflète une grande efficacité pour former des clusters qui reflètent bien les classes réelles des données.

Le meilleur score réalisé avec une homogénéité égal à ***0.88458***, complétude égale à ***0.88555***, V-Mesure égale à ***0.88506*** et une pureté égale à ***0.96667***. Seulement, un mal

positionnement de trois individus, un résultat qui reflète la puissance et la robustesse de l'approche (figure 4.4).

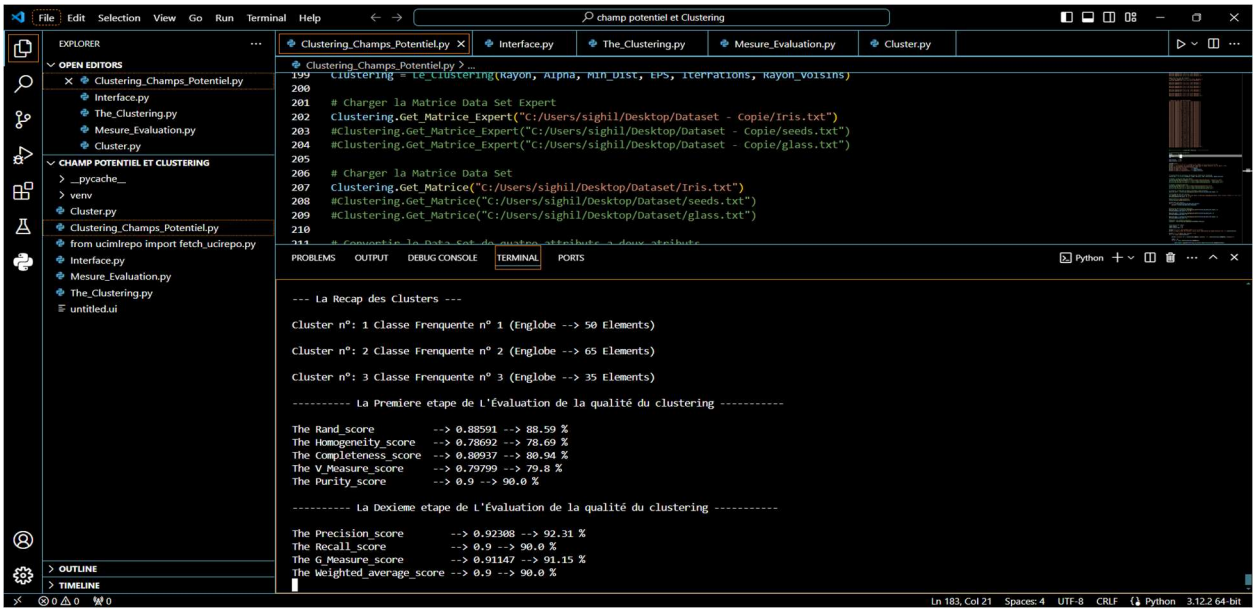

Fig. 4.3 – Résultat pour le Benchmark Iris (fréquent résultat).

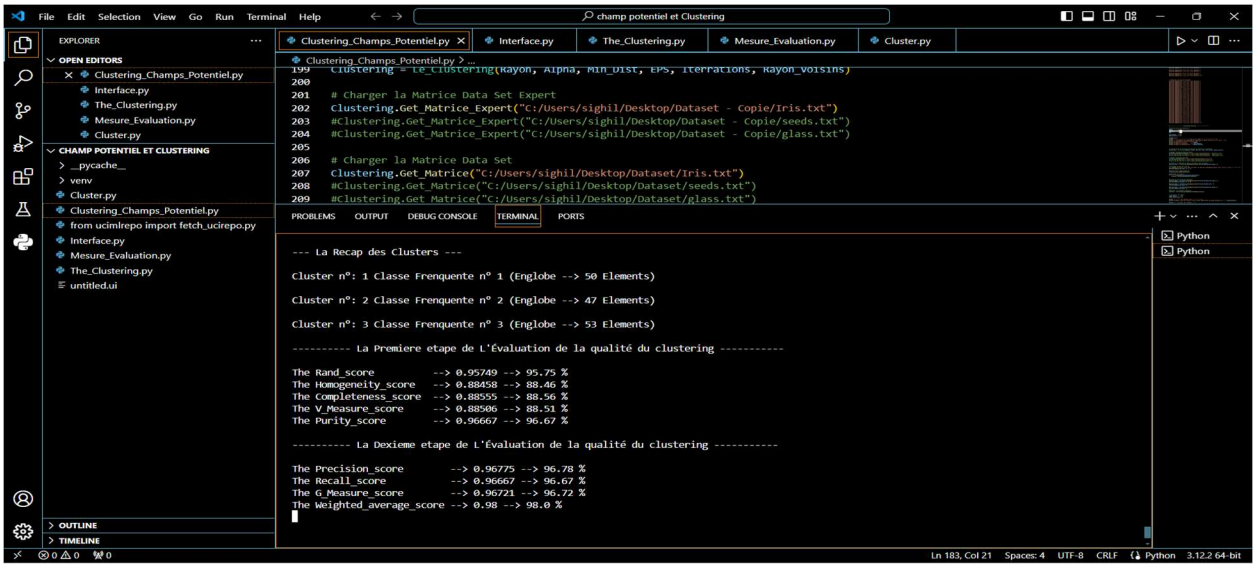

Fig. 4.4 – Résultat pour le Benchmark Iris (meilleur résultat).

TAB. 4.6 – Algorithmes dans la littérature vs Champ potentiel (Benchmark Seeds) (1).

Jeu de données Seeds	**Homogénéité**	**Complétude**	**V-Mesure**	**Pureté**
DE	0.55015	0.64305	0.58691	0.77048
PSO	0.54263	0.68222	0.59593	0.76095
GA	0.54015	0.61663	0.57184	0.76762
MVO	0.61098	0.67855	0.63709	0.82810
Champ Potentiel	***0.69643 (fréquent)*** ***0.71015 (Meilleur)***	***0.70947 (fréquent)*** ***0.71107 (Meilleur)***	***0.70289 (fréquent)*** ***0.71061 (Meilleur)***	***0.89048 (fréquent)*** ***0.90952 (Meilleur)***

Nous exposons dans le tableau 4.6, les quatre paramètres chiffrés homogénéité, complétude, V-mesure et la pureté pour l'analyse et comparaison du Data_Set seeds, on peut voir l'application du champ potentiel réalise un meilleur score pour les quatre mesures (figure 4.5). Nous prenons comme référence une autre fois la comparaison avec l'algorithme Multi-verse Optimizer (MVO), vu qu'il réalise le score meilleur par rapport aux autres algorithmes.

1- Pour l'homogénéité l'approche du champ potentiel réalise un score fréquent de ***0.69643*** contre 0.61098 pour l'algorithme MVO. On peut conclure que l'algorithme du champ potentiel a une meilleure capacité à créer des clusters homogènes que celui de l'algorithme MVO. Ce qui s'explique comme une meilleure séparation des classes dans les clusters produits.

2- Pour complétude l'approche du champ potentiel réalise un score fréquent de ***0.70947*** contre 0.67855 pour l'algorithme MVO. On peut conclure que l'approche du champ potentiel est légèrement meilleur pour regrouper les éléments de la même classe ensemble dans un seul cluster. Une garantie que les clusters formés contiennent le maximum possible d'éléments d'une même classe.

3- Pour la V-Mesure l'approche du champ potentiel réalise un score fréquent de ***0.70289*** contre 0.63709 pour l'algorithme MVO. On peut conclure que l'approche du champ

potentiel produit des clusters de haute qualité, ce qui signifie que les clusters formés sont à la fois relativement homogènes et complets. En résumé, le résultat obtenu indique que l'approche est plus efficace pour créer des clusters de haute qualité.

4- Pour la pureté l'approche du champ potentiel réalise un score de ***0.89048*** contre 0.82810 pour l'algorithme MVO. On peut conclure que 86% des éléments des clusters correspondent à la classe majoritaire au sein de chaque cluster. Un tel score reflète une grande efficacité pour former des clusters qui reflètent bien les classes réelles des données.

Le meilleur score réalisé avec une homogénéité égal à ***0.71015***, complétude égale à ***0.71107***, V-Mesure égale à ***0.71061*** et une pureté égale à ***0.90952***. Seulement, un mal positionnement de cinq individus (figure 4.6).

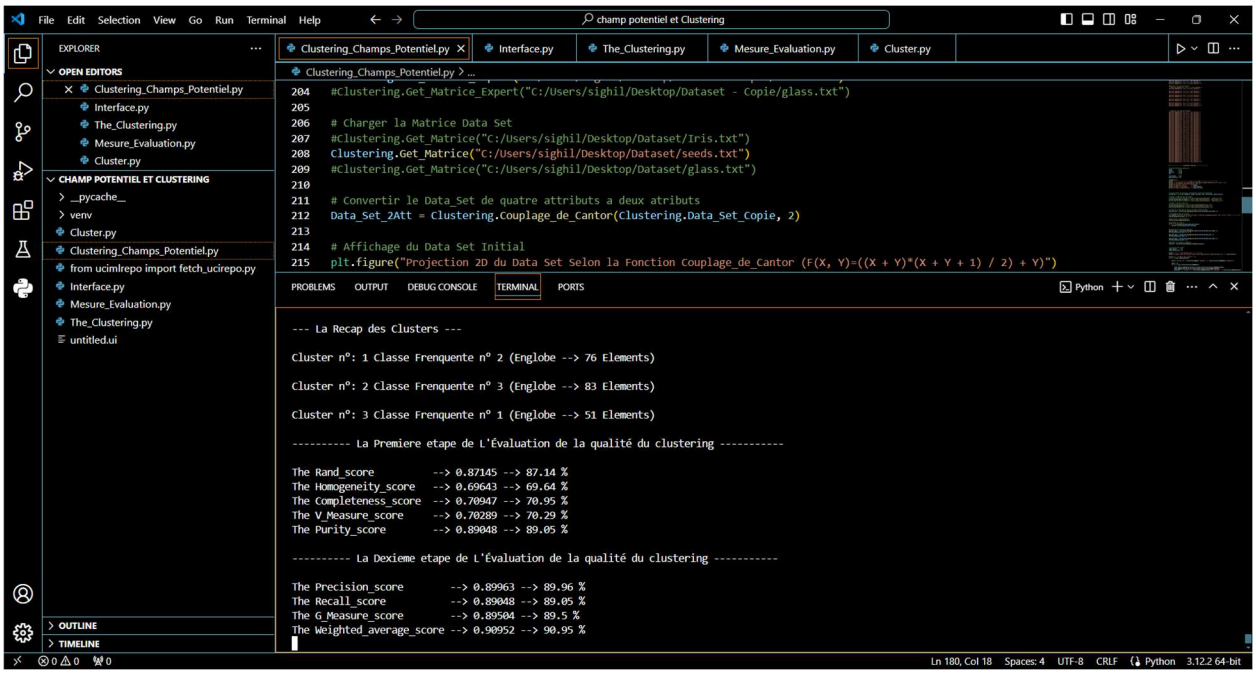

Fig. 4.5 – Résultat pour le Benchmark Seeds (fréquent résultat).

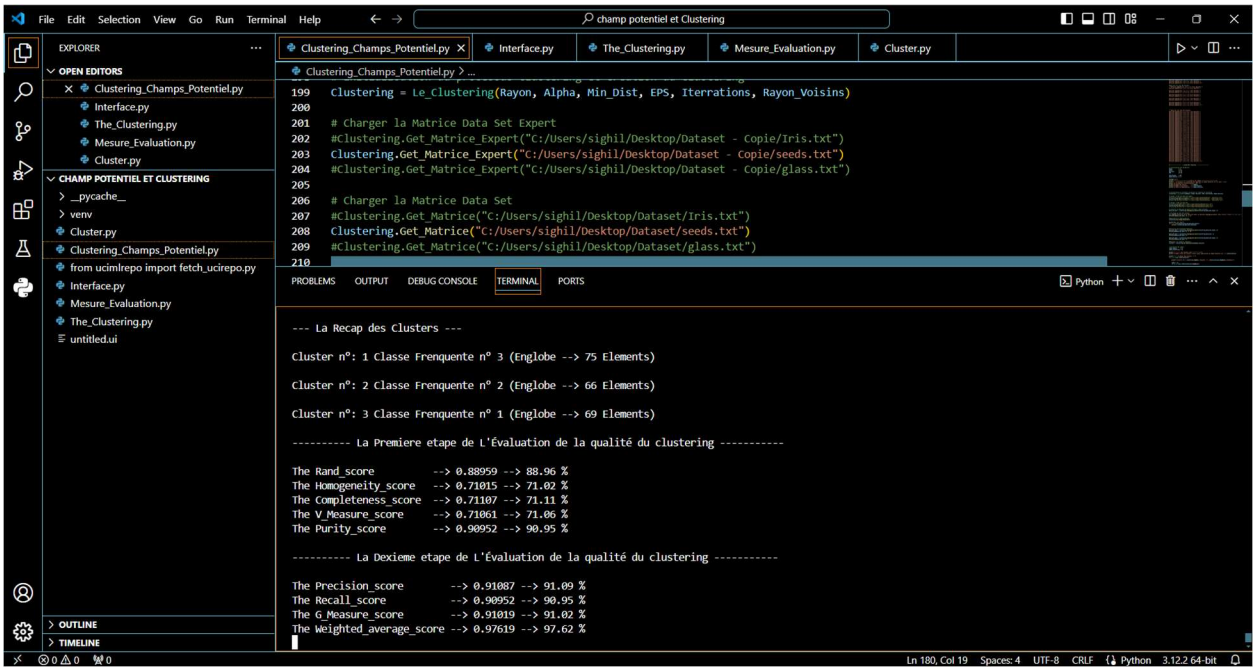

Fig. 4.6 – Résultat pour le Benchmark Seeds (meilleur résultat).

4.4.2 Champ potentiel vs GWAC

Comme déjà mentionné dans le point 4.3.b, ce qui suit est la deuxième comparaison qui s'effectue sur le benchmark iris, les paramètres resterons les mêmes comme nommé dans le point 4.4.1. Le score élevé sera indiqué en ***italique gras***.

TAB. 4.7 – Algorithmes dans la littérature vs Champ potentiel (Benchmark Iris) (2).

Jeu de données Iris	**Précision**	**Rappel**	**G_Mesure**
KM	0.3018	0.3020	0.1144
GAC	0.3534	0.3733	0.2046
HSC	0.2428	0.2393	0.1102
MHSC	0.4586	0.4427	0.4082
PSOC	0.4299	0.4460	0.4364
FPAC	0.4266	0.4385	0.4319
BATC	0.4396	0.4360	0.4359
GWAC	0.5688	0.5813	0.5682
Champ Potentiel	***0.92308 (fréquent)*** ***0.96775 (Meilleur)***	***0.90000 (fréquent)*** ***0.96667 (Meilleur)***	***0.91147 (fréquent)*** ***0.96721 (Meilleur)***

Nous exposons dans le tableau 4.7, les quatre paramètres chiffrés précision, rappel, moyenne pondérée et la G_Mesure pour l'analyse et comparaison du Data_Set iris (figure 4.3), on peut voir l'application du champ potentiel réalise un large score, pour qu'il soit de précision, rappel, moyenne pondérée et la G_Mesure (figure 4.3). Nous prenons comme référence la comparaison avec l'algorithmes de clustering Grey Wolf Algorithm-Based Clustering (GWAC) une technique de clustering basée sur l'algorithme du loup gris (GWA), une métaheuristique inspirée du comportement des loups gris en nature. L'algorithme du loup gris simule la hiérarchie sociale des loups gris, composée de trois types d'individus : les alphas, les bêtas et les deltas [58].

1- Pour précision l'approche du champ potentiel réalise un score fréquent de ***0.92308*** contre 0.5688 pour l'algorithme GWAC. On peut conclure que l'approche garantie une haute précision qui signifie peu de faux positifs par rapport aux vrais positifs. En d'autres termes, un score de précision élevé indique que la majorité des éléments classés comme positifs sont effectivement positifs.

2- Pour le rappel l'approche du champ potentiel réalise un score fréquent de ***0.90000*** contre 0.5813 pour l'algorithme GWAC. On peut conclure que l'approche du champ potentiel est nettement meilleur pour d'identifier une grande proportion des éléments pertinents. En d'autres termes, le système capture efficacement la majorité des éléments recherchés.

3- Pour la G_Mesure l'approche du champ potentiel réalise un score fréquent de ***0.91147*** contre 0.5682 pour l'algorithme GWAC. On peut conclure que l'approche est globalement performante et équilibré dans son traitement des différentes classes présentes dans les données.

Le meilleur score réalisé avec une précision égal à ***0.96775***, rappel égale à ***0.96667*** et une G_Mesure égale à ***0.96721*** (figure 4.4).

4.5 Conclusion

Tout le long du chapitre, l'évaluation et la discussion nous ont révélé que l'approche du champ potentiel mise en œuvre au bon gré du clustering porte ses fruits de manière remarquable et retentissante, dépassant les attentes initiales et générant des résultats concluants sur tous les fronts, il suffit de comparer les chiffres sur les mesures récoltés pour comprendre la puissance et la qualité des résultats. Le succès de l'approche retentissant s'explique par l'adoption du phénomène qui est fondamental en physique, qui permet d'interagir à distance entre individus constituant le Data_Set sans que rien ne les relie. Le phénomène nous a permis de modéliser et d'appliquer efficacement l'approche dans le domaine du clustering, une branche de l'apprentissage automatique et la science des données. Formellement, la qualité des résultats repose sur la précision du modèle mathématique et l'adéquation des méthodes de résolution.

Conclusion générale

Dans ce thème, nous avons en premier lieu présenté la notion de distance et de similarité en détail. Un vaste paradis de notion, ou toujours la distance euclidienne reste la plus ancienne et la plus fascinante malgré sa simplicité, arrivant à l'une des distances récentes, la distance Wasserstein également connue sous le nom de la distance du transport optimal qui provient de la théorie du transport optimal. La notion de distance a endoctriné amplement sur le développement de la recherche sur d'autres axes et perspectives. Dans le monde du data mining, si en dit clustering systématiquement en fait référence à une mesure de distance ou de similarité.

Nous avons fouillé dans l'ensemble des phénomènes observables dans l'univers physique, comme la gravitation, la photosynthèse, la formation des nuages, et d'autre, la chose qui nous a intriqué est la notion d'aimant. Un aimant est un objet qui produit un champ magnétique, qui exerce une force attractive ou répulsive sur d'autres matériaux magnétiques.

Ici vient les travaux réalisés par le professeur *Mazouzi Smaine* en 2008, une thèse intitulée la reconnaissance de formes par les systèmes auto organisée, l'application été sur les images de profondeur [64]. Une grande inspiration par a port au champ potentiel, ces principes et ces caractéristiques. L'idée était de calculer l'influence de la force générée par les individus voisins dans un périmètre qui est fixé à l'avance, ensuite une mise à jour sur l'individu en lui mémé s'impose, pour chaque individu un voyage commence pour des directions imposées par le calcul du champ, chaque individu est attiré à un groupe de voisins, cette influence parmi de les rapprocher les unes aux autres formant à la fin après un nombre déterminé d'itération presque un même individu, ils seront superposés. À la fin, une formation de clusters naturels est ressentie et visualisée.

Dans le cadre de ce travail, plusieurs concepts sont introduits, l'initialisation des paramètres, le calcul des voisins, la superposition des individus. Nous espérons au futur de pousser la recherche pour pouvoir réaliser de meilleurs résultats. L'initialisation des paramètres nécessite une étude sur les éléments d'entrées et le degré de rapprochement entre individus. D'une autre part, le calcul du cumul des forces dépend de la détermination du nombre de voisins, une relation de liaison, l'influence et l'impact est immense sur les résultats.

Le futur est prometteur, nous espérons avoir formulé une résolution pour le problème du clustering en faisons recourir à une approche fondamentale dans la physique classique et qui est le champ potentiel, la manière et le concept sont novatrices qui sursois l'habituel. Il est important d'invoqué que l'introduction d'une hybridation inter concept, fortement, améliore le résultat désiré, surtout le point de la décision des voisins qui inévitable.

Le meilleur est pour la fin, la physique quantique se développe rapidement, une discipline qui révolutionne notre compréhension de la matière, de l'énergie et de l'univers. L'informatique quantique [65] ou pseudo-quantique inspirée de la physique est vue comme un processus d'optimisation. Ces principes fondamentaux aideront sûrement à résoudre les lacunes rencontrées durant le développement d'innombrables algorithmes de clustering. Les concepts comme la représentation quantique un point fort pour représenter l'ensemble des Data_Set, la superposition, un système quantique peut exister dans plusieurs états en même temps jusqu'à ce qu'il soit mesuré, l'intrication, des particules peuvent être intriquées, ce qui signifie que l'état de l'une affecte instantanément l'état de l'autre, quelle que soit la distance qui les sépare, ces principes, vont téléporter le clustering a un autre univers.

Bibliographie

1- M. M. a. P. F. A.K. JAIN, Data Clustering: A Review, The Ohio State University, 1999.

2- Cynthia Pitou. Extraction d'informations textuelles au sein de documents numérisés : cas des factures. Traitement du texte et du document. Université de la Réunion, 2017. Français.

3- Hubert, «in and Max Hierarchical Clustering Using Asymmetric Similarity Measures, » *Psychometrika,* p. 63–72, 1973.

4- Saxena A, Prasad M, Gupta A, Bharill N, Patel OP, Tiwari A, Er MJ, Ding W, Lin CT (2017) Un examen des techniques et des développements de clustering. Neuroinformatique 267 : 664-681.

5- Xu R, Wunsch DC (2010) Algorithmes de clustering dans la recherche biomédicale : une revue. IEEE Rev Biomed Eng 3 : 120–154.

6- Groenen, P. J., & Jajuga, K. (2001). Fuzzy clustering with squared Minkowski distances. Fuzzy Sets and Systems, 120(2), 227-237.

7- Duby, C., & Robin, S. (2006). Analyse en composantes principales. Institut National Agronomique, Paris-Grignon, 80, 53.

8- Winter, M. (1923). Le théorème de pythagore. Revue de Métaphysique et de Morale, 30(1), 23-28.

9- On the Surprising Behavior of Distance Metrics in High Dimensional Space Charu C. Aggarwal 1, Alexander Hinneburg 2, and Daniel A. Keim2 1 IBM T. J. Watson Research Center Yorktown Heights, NY 10598, USA. Page 427.

10- Faisal, M., & Zamzami, E. M. (2020, June). Comparative analysis of inter-centroid K-Means performance using euclidean distance, canberra distance and manhattan distance. In Journal of Physics: Conference Series (Vol. 1566, No. 1, p. 012112). IOP Publishing.

11- Faisal, M., & Zamzami, E. M. (2020, June). Comparative analysis of inter-centroid K-Means performance using euclidean distance, canberra distance and manhattan distance. In Journal of Physics: Conference Series (Vol. 1566, No. 1, p. 012112). IOP Publishing.

12- Yang Y. M., Jia R., Xun H. et al., Determining the number of instars in simulium quinquestriatum (diptera: simuliidae) using k-means clustering via the canberra distance, Journal of Medical Entomology. (2018) 55, no. 4, 808–816, https://doi.org/10.1093/jme/tjy024, 2-s2.0-85061426742.

Bibliographie

13- Panaretos, V. M., & Zemel, Y. (2019). Statistical aspects of Wasserstein distances. Annual review of statistics and its application, 6, 405-431

14- Julien Ah-Pine. Une famille d'indices de similarité généralisant la mesure de cosinus. 17èmes Rencontres de la Société Francophone de Classification (SFC 2010), Jun 2010, Saint-Denis de la Réunion, France. hal-01504528.

15- P. Jaccard, Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines. *Bulletin de la Société Vaudoise des Sciences Naturelles*, Vol. 37, pp. 241–272 (1901).

16- R. Real, and J.M. Vargas, The Probabilistic Basis of Jaccard's Index of Similarity. Systeme Biology, Vol. 45 (3), pp. 385–390 (1996).

17- Using of Jaccard Coefficient for Keywords Similarity Suphakit Niwattanakul*, Jatsada Singthongchai, Ekkachai Naenudorn and Supachanun Wanapu.

18- A GENERAL COEFFICIENT OF SIMILARITY AND SOME OF ITS PROPERTIES J. C. GOWER Rothamsted Experimental Station, Ha~penden, Herts., U.R.

19- Liao TW (2005) Regroupement de données de séries chronologiques : une enquête. Reconnaissance de modèle 38 : 1857-1874.

20- Aggarwal CC, Philip SY, Han J, Wang J (2003) Un cadre pour regrouper des flux de données évolutifs. Dans : Actes de la conférence VLDB 2003, Elsevier, pp 81-92.

21- Mahdi MA, Hosny KM, Elhenawy I (2021) Algorithmes de clustering évolutifs pour le big data : une revue. Accès IEEE. https://doi.org/10.1109/ACCESS.2021.3084057.

22- Gordon AD (1999) Classification. Presse CRC, Boca Raton.

23- Parsons L, Haque E, Liu H (2004) Regroupement de sous-espaces pour les données de grande dimension : une revue. Cote 1(1):5.

24- James G, Witten D, Hastie T, Tibshirani R (2015) Une introduction à l'apprentissage statistique avec des applications chez R. Springer, New York.

25- Ni J, Young T, Pandelea V, Xue F, Cambria E (2022) Progrès récents dans les systèmes de dialogue basés sur l'apprentissage profond : une enquête systématique. Dans : Revue de l'intelligence artificielle, pp 1–101.

26- Hastie T, Tibshirani R, Friedman JH, Friedman JH (2009) Les éléments de l'apprentissage statistique : exploration de données, inférence et prédiction. Springer, New York.

Bibliographie

27- Aljalbout E, Golkov V, Siddiqui Y, Strobel M, Cremers D (2018) Clustering avec apprentissage profond : taxonomie et nouvelles méthodes. Préimpression arXiv arXiv:1801.07648.

28- Jain (A. K.), Murty (M. N.) et Flynn (P. J.). — Data clustering : a review. ACM Computing Surveys, vol.31, No 3, 1999, pp. 264-323.

29- Berkhin (P.). — Survey Of Clustering Data Mining Techniques. — Rapport technique, San Jose, CA, Accrue Software, 2002.

30- MacQueen (J.). — Some methods for classification and analysis of multivariate observations. In : Proceedings of the Fifth Berkeley Symposium on Mathematical statistics and probability. pp. 281-297. - Berkeley, 1967.

31- Sheikholeslami (G.), Chatterjee (S.) et Zhang (A.). — WaveCluster : A Multi-Resolution Clustering Approach for Very Large Spatial Databases. In : Proc. 24th Int. Conf.

32- Ester (M.), Kriegel (H. P.), Sander (J.) et Xu (X.). — A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In : Second International Conference on Knowledge Discovery and Data Mining, ed. par Simoudis (E.), Han (J.) et Fayyad (U.). pp. 226-231. — Portland, Oregon, 1996.

33- Michalski (R. S.). — Knowledge acquisition through conceptual clustering : A theoretical framework and an algorithm for partitioning data into conjunctive concepts. — Technical Report nN° 1026, Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois, 1980.

34- Fisher (D.). — Knowledge acquisition via incremental conceptual clustering. Machine Learning, vol.2, 1987, pp. 139-172.

35- Hertz (J.), Krogh (A.) et Palmer (R. G.). — An Introduction to the Theory of Neural Computation. — Santa Fe Institute Studies in the Sciences of Complexity lecture notes. Addison-Wesley Longman Publ. Co., Inc., Reading, MA., 1991.

36- Bishop (C.M.). — Neural Networks for Pattern Recognition. — Oxford University Press, 1995.

37- Kohonen (T.). — Self-Organization and Associative Memory. Springer, 1984.

38- Paterson (M.S.) et Yao (F.F.). — On Nearest Neighbor Graphs. Automata, Languages and Programming, vol.623, 1992, pp. 416-426.

Bibliographie

39- Pellegrini (F.). — Static mapping by dual recursive bipartitioning of process and architecture graphs. IEEE, 1994, pp. 486{493.

40- Halkidi (M.), Batistakis (Y.) et Vazirgiannis (M.). — Clustering Validity Checking Methods : Part II. ACM SIGMOD, vol.31, No 3, 2002, pp. 19-27.

41- Halkidi (M.) et Vazirgiannis (M.). — Clustering Validity Assessment : Finding the Optimal Partitioning of a Data Set. In : proceedings of IEEE International Conference of Data Mining (ICDM 2001), pp. 187{194. — San Jose, California, USA, 2001.

42- Xu R, Wunsch D (2005) Enquête sur les algorithmes de clustering. Réseau neuronal trans IEEE 16 : 645–678.

43- Sekula M, Datta S, Datta S (2017) optCluster : un package R pour déterminer l'algorithme de clustering optimal. Bioinformation 13:101.

44- Baidari I, Patil C (2020) Un critère pour décider du nombre de clusters dans un ensemble de données en fonction de la profondeur des données. Vietnam J Comput Sci 7 : 417-431.

45- Hermann, C. (2008). La traduction et les commentaires des Principia de Newton par Émilie du Châtelet. Bibnum. Textes fondateurs de la science.

46- Pellat, H. (1896). Électrostatique non fondée sur les lois de coulomb. Forces agissant sur les diélectriques non électrisés. J. Phys. Theor. Appl., 5(1), 244-256.

47- Faraday, M. (1867). *Notice sur Michel Faraday, sa vie et ses travaux: Tiré des Archives des sciences de la Bibliothèque universelle Oct. 1867*. Ramboz.

48- Coulomb, C. A. (1884). *Collection de mémoires relatifs à la physique: mémoires de Coulomb* (Vol. 1). Gauthier-Villars.

49- Unités, S. I., and S. I. Base. "Champ électrique."

50- Domaine, M. I. Electrostatique-électrocinétique.

51- https://archive.ics.uci.edu/ml/index.php [En ligne].

52- Yang, X. S. (2011). Metaheuristic optimization. *Scholarpedia*, *6*(8), 11472.

53- Storn R, Price K (1997) Evolution différentielle : une heuristique simple et efficace.

54- Kennedy J (1997) L'essaim de particules : adaptation sociale des connaissances. Dans : Conférence internationale IEEE sur le calcul évolutif, pp 303–308.

Bibliographie

55- Goldberg David E (1989) Algorithmes génétiques dans la recherche, l'optimisation et l'apprentissage automatique. Addison-Wesley, en train d pour l'optimisation globale sur des espaces continus. J Optimum global 11(4):341–

56- Mirjalili S, Mirjalili SM, Lewis A (2014) Optimiseur de loup gris. Logiciel avancé en angla.

57- I. A. M. A. H. F. Mirjalili, «Multi-verse Optimizer: Theory, Literature Review, and Application in Data Clustering,» 2020.

58- V. Kumar, J. K. Chhabra et D. Kumar, «Grey Wolf Algorithm-based Clustering Thechnique,» 2017.

59- AK Jain, Clustering de données : 50 ans au-delà des K-means, *Pattern Recogn. Lett.* **31** (2010), 651-666.10.1007/978-3-540-87479-9_3

60- Alia, O., & Mandava, R. (2011). The variants of the harmony search algorithm: an overview. *Artificial Intelligence Review*, 36(1), 49-68.

61- Pei, Z., Hua, X., & Han, J. (2008, October). The clustering algorithm based on particle swarm optimization algorithm. In 2008 International conference on intelligent computation technology and automation (ICICTA) (Vol. 1, pp. 148-151). IEEE.

62- Agarwal, P. et Mehta, S. (2016). Algorithme amélioré de pollinisation des fleurs sur le regroupement de données. Journal international des ordinateurs et applications, 38 (23), 144-155. https://doi.org/10.1080/1206212X.2016.1224401.

63- X.-S. Yang, Un nouvel algorithme métaheuristique inspiré des chauves-souris, dans : *Proceedings of Nature Inspired Cooperative Strategies for Optimization, Studies in Computational Intelligence* , JR Gonzalez et al., eds., vol. 284, pp. 65-74, Springer, Berlin, 2010.10.1007/978-3-642-12538-6_6.

64- Kholladi, M. K., & Mazouzi, S. (2008). Reconnaissance de formes par les systèmes auto-organisés.

65- RAMDANE, Chafika, MESHOUL, Souham, BATOUCHE, Mohamed, et al. A quantum evolutionary algorithm for data clustering. International Journal of Data Mining, Modelling and Management, 2010, vol. 2, no 4, p. 369-387.