Rapport Machine Learning
Mathias Predon / Pierre Bonnecaze

Introduction :

Le but de ce TP est d'effectuer une analyse expérimentale sur plusieurs méthodes de clustering pour l'apprentissage non supervisé. Ce dernier appartient à la grande famille du machine learning (sous catégorie de l'intelligence artificielle) qui regroupe par exemple l'apprentissage supervisé, l'apprentissage semi supervisé mais également le deep learning. L'apprentissage non supervisé a pour but à partir de données non labellisées de regrouper par similarité ces données en cluster. La cible est également inconnue en apprentissage non supervisé. Ainsi au cours de ce rapport nous analyserons par l'intermédiaire de plusieurs types de Data sets les méthodes de clustering k-means et agglomératif. Il s'agira d'analyser les points forts et points faibles pour les deux méthodes étudiées. Dans une deuxième partie nous effectuerons une analyse comparative des deux méthodes mais sur des nouvelles données fournies (qualité des solutions obtenues, performance des méthodes,..).
Vous trouverez le code python associé à ce TP sur le lien suivant : https://github.com/mathprd/Clustering.git
Les fichiers des codes pythons sont Data_reader.py et analyse_dataset_rapport.py.

I. Clustering k-means

k-means est un algorithme itératif. L'idée centrale de l'algorithme K-means est d'itérer entre l'affectation des points aux clusters et la mise à jour des centres jusqu'à ce que le regroupement converge vers une solution stable. L'algorithme part de centre de clusters aléatoires et calcule la distance d'un point avec chaque centre à chaque itération afin d'attribuer le point à un cluster.

Afin d'analyser la méthode de clustering k-means nous allons appliquer itérativement la méthode pour déterminer le bon nombre de clusters à l'aide de métriques d'évaluation. Ces dernières correspondent au coefficient de silhouette / indice de Davies-Bouldin / indice de Calinski-Harabasz. Nous allons ainsi pour trois datasets chercher le meilleur nombre de clusters k trouvés suivant le calcul de ces métriques. A noter qu'il est possible qu'un meilleur k pour une métrique peut être différent du meilleur k pour une autre. En ce qui concerne ces métriques il est également important de savoir que plus le coefficient de silhouette et indice de Calinski-Harabasz sont élevés plus la solution trouvée est optimale. A l'inverse plus l'indice de Davies-Bouldin est faible plus la solution trouvée est optimale. L'ensemble des résultats obtenus par notre code optimisé se trouve dans les tableaux ci-dessous :

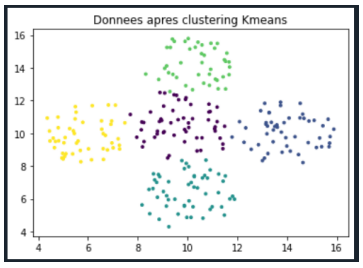
Nom du fichier	Coefficient de silhouette			
	Score	itération	Temps d’exécution (ms)	Nombre de Clusters
Spherical_5_2.arff	0.5123446895187646	7	4.32	5
xclara.arff	0.6945587736089913	2	4.84	3
rings.arff	0.4388335296073186	10	5.72	5

Nom du fichier	indice de Davies-Bouldin			
	Score	itération	Temps d’exécution (ms)	Nombre de Clusters
Spherical_5_2.arff	0.6674633006423455	6	9.71	5
xclara.arff	0.42056158508478453	3	6.26	3
rings.arff	0.7828926773157083	12	7.04	5

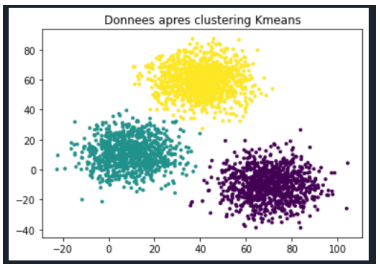
Nom du fichier	indice de Calinski-Harabasz			
	Score	itération	Temps d’exécution (ms)	Nombre de Clusters
Spherical_5_2.arff	389.178126129042	6	3.89	5
xclara.arff	10826.600579461161	3	5.41	3
rings.arff	655.6856311034065	16	27.18	5

Sur les figures suivantes sont représentés les trois Data sets utilisés précédemment avec en couleur les différents clusters trouvés par la méthode de clustering k-means

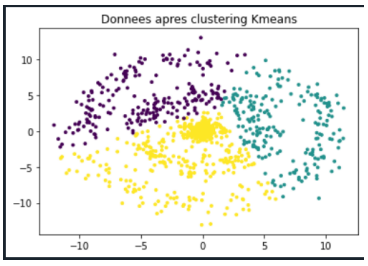
Spherical_5_2.arff :



xclara.arff :



rings.arff :



En analysant cette méthode de clustering on remarque que k-means est un algorithme simple à mettre en œuvre et efficace. L'algorithme est rapide, c'est la raison pour laquelle il est grandement utilisé. De plus, c'est un algorithme qui marche particulièrement lorsque les clusters ont une forme convexe ou globulaire comme c'est le cas pour le Data set Spherical_5_2. Il marche également très bien lorsque les points sont très bien répartis.

En revanche, malgré ses points positifs, la méthode de clustering de k-means possède aussi ses inconvénients. En effet, c'est un algorithme qui est forcément biaisé par le choix des centres initiaux. En effet, plusieurs initialisations peuvent mener à des résultats différents. C'est donc un algorithme non déterministe.

De plus, comme dit précédemment, k-means assume que les clusters ont une forme sphérique et donc il fonctionne beaucoup moins bien lorsque les clusters ont une forme plus complexe.

II. Clustering agglomératif

L'algorithme de clustering agglomératif est une méthode utilisée en apprentissage non-supervisée. Il commence par traiter chaque point de données comme un cluster individuel, puis fusionne progressivement les clusters les plus proches les uns des autres jusqu'à ce qu'un seul cluster contenant toutes les données soit formé.

Afin d'analyser la méthode de clustering agglomératif nous allons tout d'abord considérer plusieurs méthodes de combinaison des clusters à savoir : single / ward / average / complete uniquement pour la distance euclidienne.

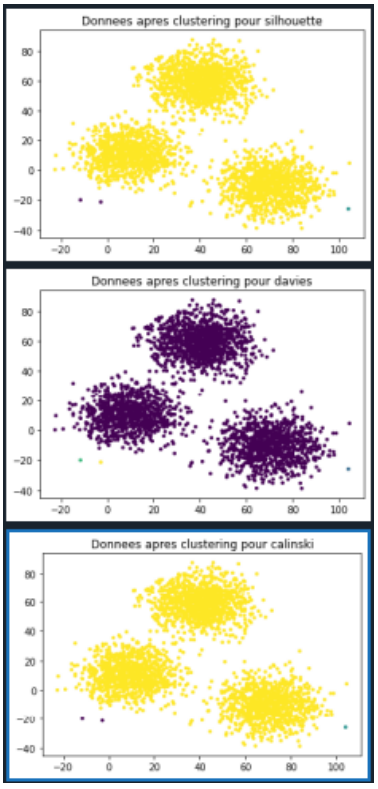
- Single Fusionne les deux clusters les plus proches en termes de distance minimale entre leurs points les plus proches.

- Ward : Minimise la variance totale au moment de fusionner les clusters, favorisant la formation de clusters compacts.

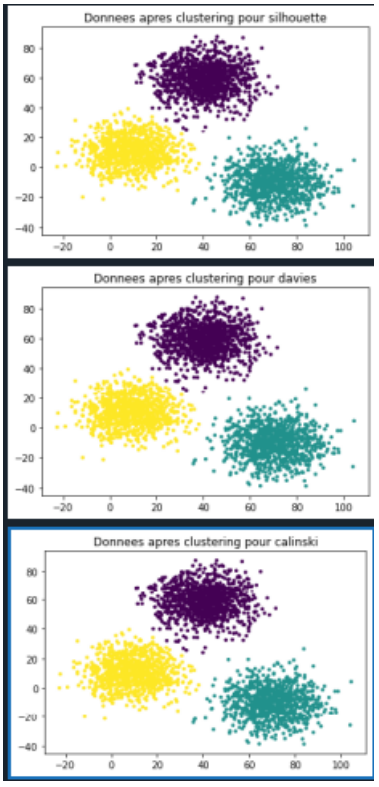
- Complete : Fusionne les deux clusters les plus proches en termes de distance maximale entre leurs points les plus éloignés.
- Average : Fusionne les clusters en utilisant la moyenne des distances entre tous les points des clusters, favorisant des regroupements équilibrés.

En appliquant les quatre méthodes on remarque que le choix de la méthode de linkage est très important. Prenons quelques exemples :

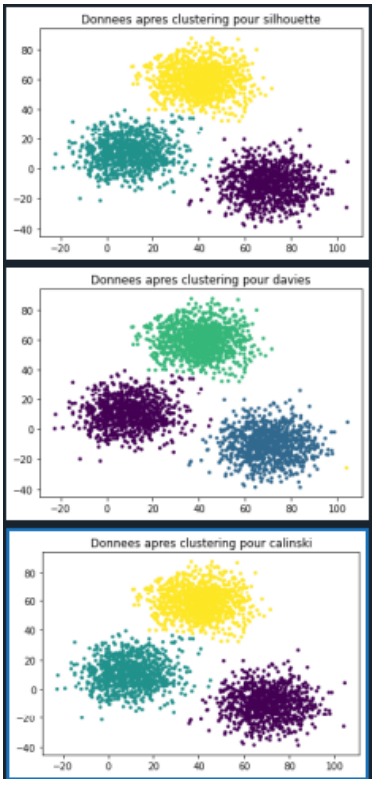
- xclara : Pour ce fichier, voici les solutions de clustering trouvées par les quatre différentes méthodes de combinaison :



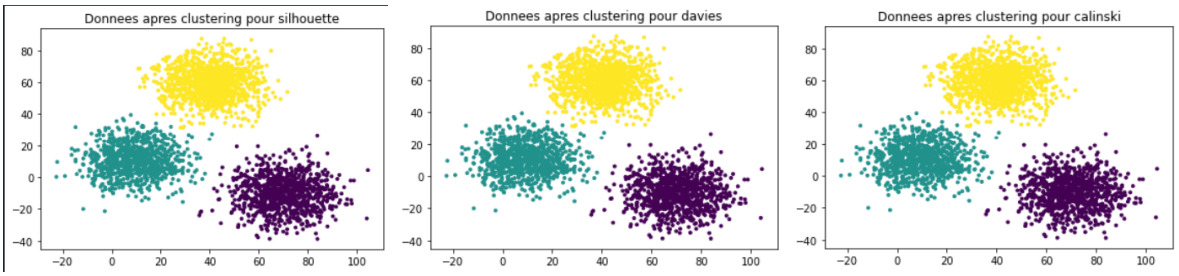
‘Single’



‘Ward’



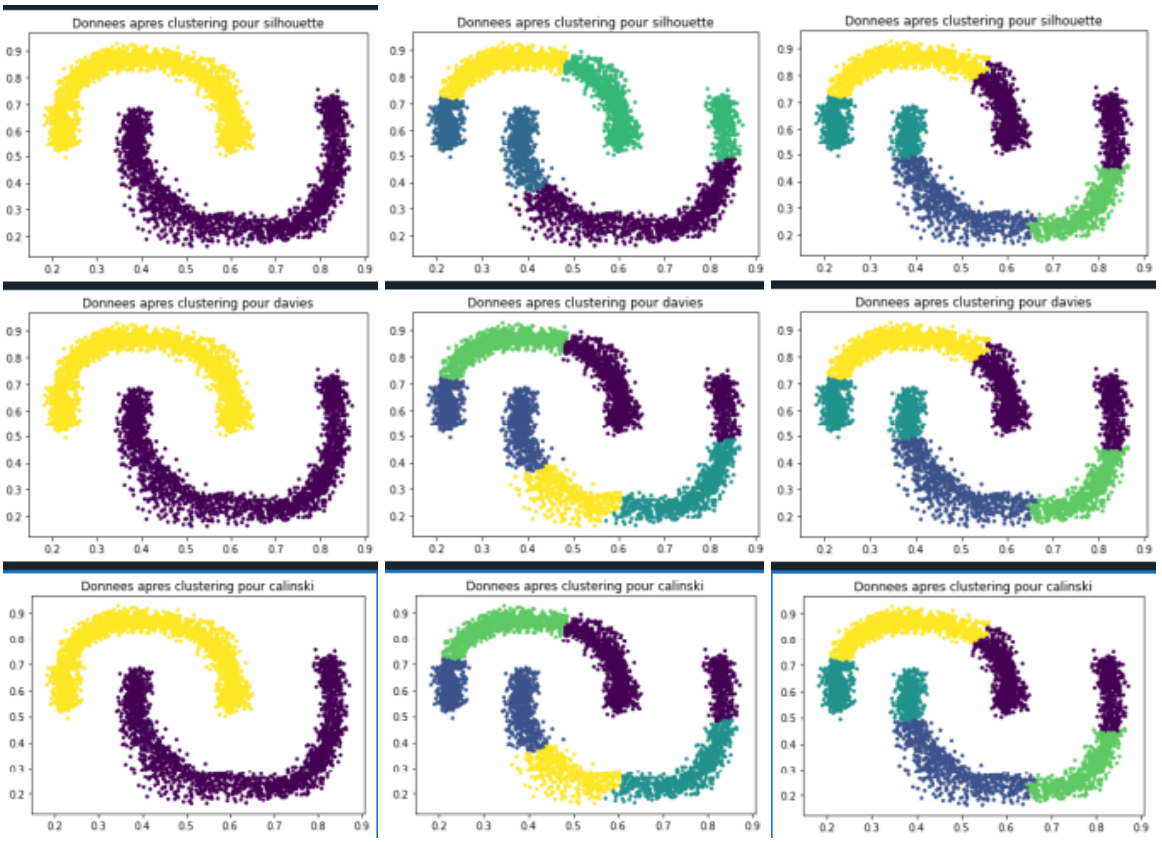
‘Average’



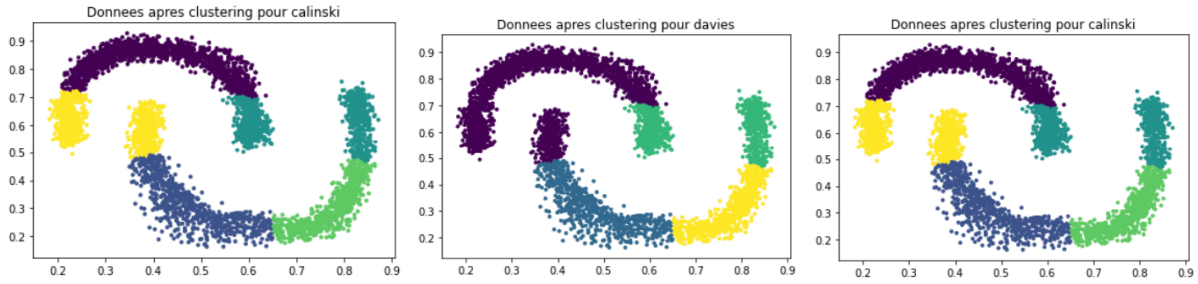
‘Complete’

On remarque qu’avec la méthode single, le programme n’est pas capable d’identifier les 3 clusters présents. Avec les 3 autres méthodes, le programme est capable de trouver 3 clusters, même s’ils ne sont pas exactement les mêmes selon les méthodes. On peut en déduire que lorsque les clusters ont bien une forme sphérique, les méthodes ward, average et complete sont efficaces. Cependant, à cause des données trop excentrées des clusters, single n’est pas efficace.

- banana : Pour ce fichier, voici les solutions de clustering trouvées par les quatre différentes méthodes de combinaison



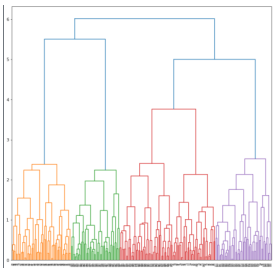
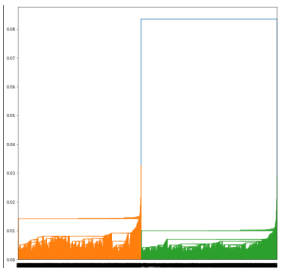
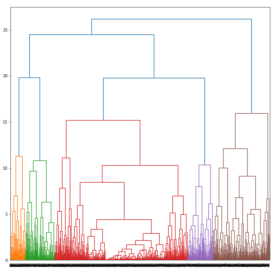
‘Single’ ‘Ward’ ‘Average’



‘Complete’

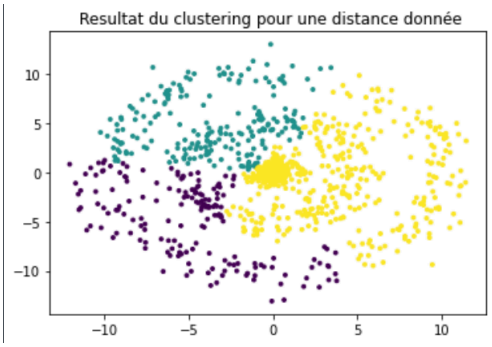
On remarque que cette fois-ci, seule la méthode single arrive à trouver les 2 clusters. On peut en déduire que lorsque que les clusters sont bien distants les uns des autres, alors la méthode single est à privilégier. En effet, comme les clusters ne sont pas en forme sphérique pleine, les autres méthodes ne sont pas efficaces.

Maintenant que nous avons analysé les méthodes de combinaison, nous allons appliquer itérativement la méthode de clustering agglomératif en faisant varier le seuil de distance afin de déterminer une bonne solution de clustering à l'aide des métriques d'évaluation. Nous allons ainsi pour trois datasets chercher le meilleur nombre de clusters k trouvés suivant le calcul de ces métriques. On choisira la meilleure méthode de linkage dans chaque cas. On affiche les dendrogrammes de chaque dataset afin de savoir si le critère distance est pertinent pour différencier les clusters et donc de choisir la méthode de linkage. Les dendrogrammes peuvent nous permettre également d'avoir un bon ordre d'idée de la meilleure distance à trouver.

Nom Fichier	Spherical_5_2.arff	Banana.arff	rings.arff
Méthode de linkage	Average	Single	Complete
Dendrogramme			

Au vu des dendrogrammes, nous pouvons constater que la meilleure distance pour Spherical_5_2.arff est comprise entre 2 et 4 alors que pour le deuxième fichier la meilleure distance est comprise aux environs de 0.05.

Dans le cas de rings.arff, le dendrogramme avec la méthode de linkage complete semble pertinent. Seulement, en affichant la solution de clustering trouvée, elle l'est beaucoup moins comme nous pouvons le constater sur la figure suivante :



La solution de clustering trouvée n'est pas la bonne car avec la méthode de linkage complete, il est nécessaire que les distances maximales des clusters soient assez proches. Ce qui n'est pas le cas pour ce Dataset. De même aucune des trois autres méthodes n'est pertinente pour ce Dataset.

Vous trouverez dans les tableaux suivants l'ensemble de nos résultats.

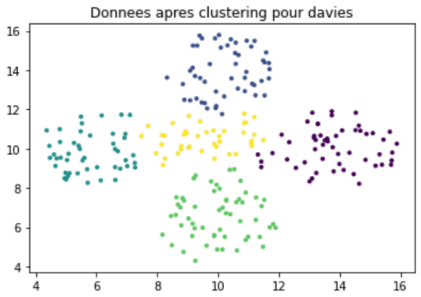
Nom du fichier	Coefficient de silhouette			
	Méthode de linkage	Score	Temps d'exécution (ms)	Meilleure distance
Spherical_5_2.arff	Average	Le score maximal ne renvoie pas le bon nombre de clusters		
banana.arff	Single	0.36733109590793084	77.02	0.04

Nom du fichier	indice de Davies-Bouldin			
	Méthode de linkage	Score	Temps d'exécution (ms)	Meilleure distance
Spherical_5_2.arff	Average	0.6735921348504817	1.66	2.6
banana.arff	Single	1.0988998676613282	75.52	0.04

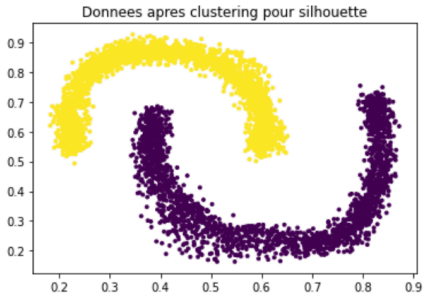
Nom du fichier	indice de Calinski-Harabasz			
	Méthode de linkage	Score	Temps d'exécution (ms)	Meilleure distance
Spherical_5_2.arff	Average	367.9722793637948	1.57	2.6
banana.arff	Single	3751.7304672665023	74.07	0.04

Sur les figures suivantes sont représentés les deux datasets utilisés précédemment avec en couleur les différents clusters trouvés par la méthode de clustering agglomératif

Spherical_5_2 :



banana.arff :



Nous allons maintenant analyser la méthode de clustering agglomératif en faisant varier le nombre de clusters pour trouver une bonne solution de clustering. Tout comme précédemment la méthode de linkage utilisée sera la meilleure. Nous ne présenterons pas les résultats sous la même forme que précédemment pour la simple et bonne raison que les scores sont exactement les mêmes. Les solutions de clustering trouvées sont bien similaires à précédemment.

Avec cette analyse complète de l'algorithme de clustering agglomératif nous sommes maintenant capables d'en trouver les points forts et les points faibles.
Cet algorithme, contrairement au k-means, ne fait pas d'hypothèses quant à la forme des clusters en choisissant le linkage 'single'. Il peut donc s'adapter à différentes formes de clusters selon les méthodes de combinaison. De plus, en utilisant cet algorithme l'utilisateur n'est pas obligé de spécifier le nombre de clusters. On retiendra tout de même son principal avantage de flexibilité de part ses 4 méthodes de combinaison.

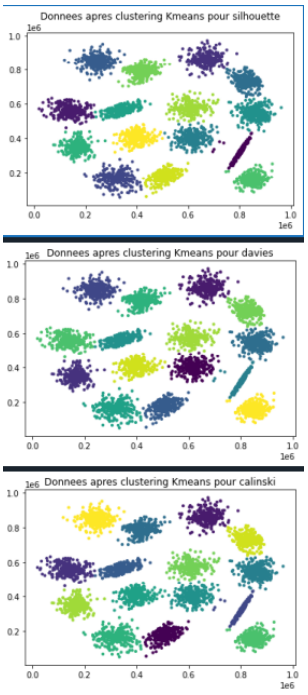
Comme nous avons pu le remarquer, l'algorithme de clustering agglomératif est dépendant des distances. Il est donc sensible aux valeurs un peu éloignées et aberrantes. Ce qui peut influencer grandement la qualité du regroupement. Pour pouvoir utiliser le clustering agglomératif, il faut respecter l'un de ces critères :

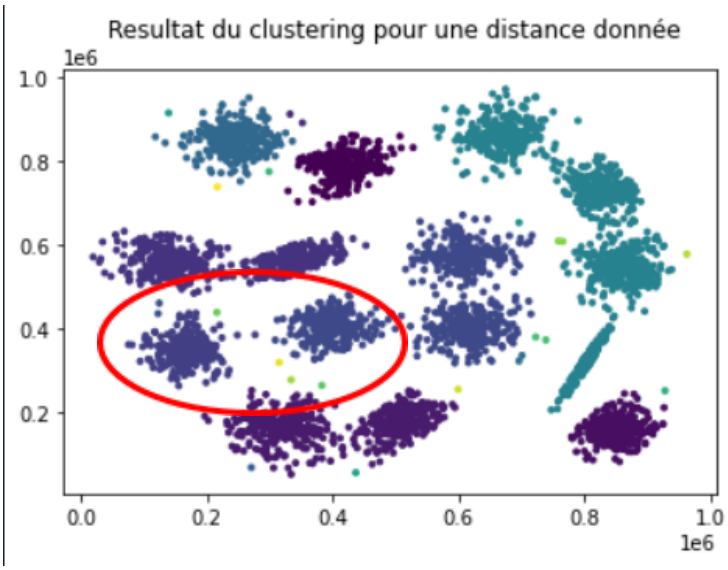
- Les clusters sont biens distants les uns des autres (single)
- Les clusters on des centres (moyennes) bien éloignés (average)
- Les clusters sont compactes et globuleux (ward)
- La distance maximale entre 2 points d'un même cluster est sensiblement la même pour chaque cluster (complete)

III. Evaluation

Sur la base de nos différentes analyses effectuées précédemment sur les deux méthodes de clustering nous allons réaliser une analyse expérimentale comparative des deux méthodes(qualité des solutions obtenues, performances des méthodes).

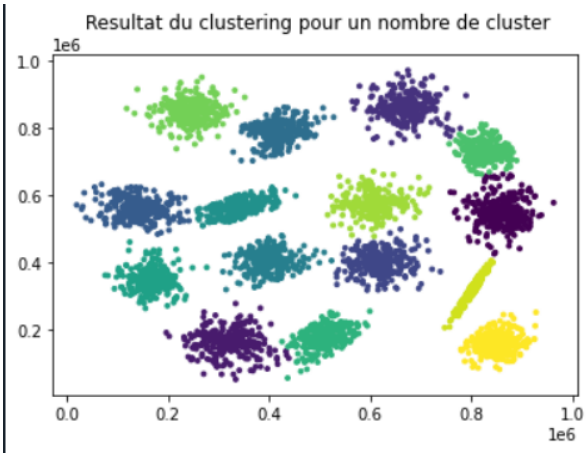
x1.txt :





On remarque que la méthode K Means fonctionne très bien car les clusters sont bien en forme globulaire (figure de gauche).

On remarque que la méthode single ne permet pas d'identifier les différents clusters (figure de droite). Cela est dû au fait que les clusters ne sont pas séparés suffisamment les uns des autres. Par exemple, dans le rond rouge, seul quelques données font le lien entre les 2 clusters, alors que se sont des clusters différents.



Avec un linkage ward pour le clustering agglomératif, on trouve les 15 clusters car chaque cluster est assez compact.

x2.txt

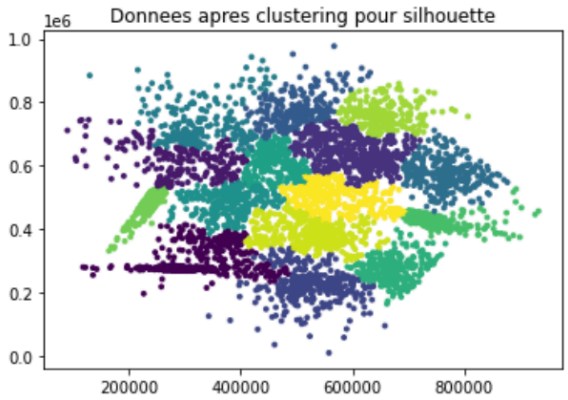
x2 ressemble fortement au premier Dataset, il n'est donc pas pertinent d'en parler puisque nous en ferons la même analyse.

x3.txt

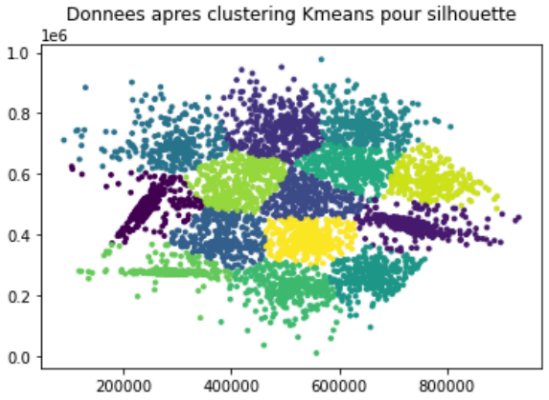
De la même façon x3 et x4 se ressemblent très fortement. Nous choisirons de présenter uniquement x4.

x4.txt

Agglomératif (ward)



k-means

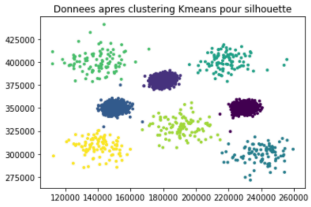


Pour x4, la méthode ward est toujours la plus pertinente. Elle réussit à retrouver les 15 clusters. Quant à la méthode k-Means, elle se débrouille également plutôt bien mais ne trouve pas les 15 clusters (14). De plus, la séparation des clusters trouvés n'est pas cohérente. Concernant les autres méthodes de linkage de la méthode agglomérative, celles-ci ne sont pas pertinentes dans leur solution trouvée. Single : les clusters sont trop proches les uns des autres. Average : les centres des clusters ne sont pas assez éloignés et complete : les distances maximales de chaque cluster ne sont pas vraiment similaires.

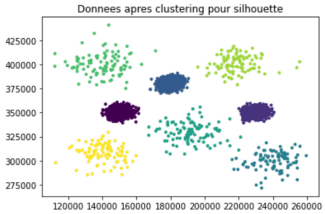
Le fichier y1.txt est trop lourd (environ 100 000 points contre 5000 pour les autres Datasets). Nous ne le traiterons pas dans ce rapport.

zz1.txt:

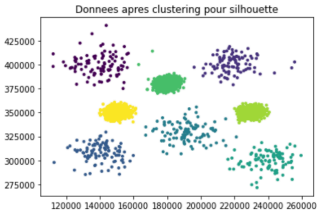
K-Means



agglomératif (ward)

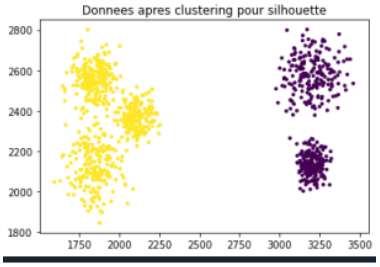


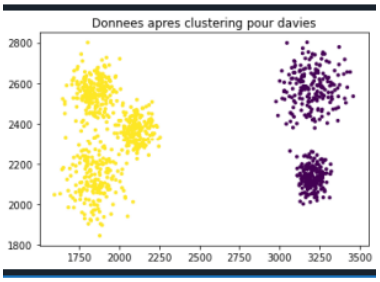
agglomératif (Average)

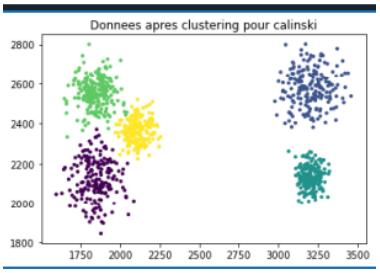


Comme nous pouvons le constater sur les figures ci-dessus les solutions de clustering trouvées sont carrément satisfaisantes pour ces méthodes. Cela se doit à la forme globulaire et compacte des clusters. De plus, les centres des clusters sont éloignés. C'est la raison pour laquelle l'algorithme de clustering agglomératif avec la méthode de linkage average fonctionne correctement. Single (14 clusters trouvés) et complete donnent des solutions de clustering incorrectes pour les mêmes raisons que citées précédemment.

zz2.txt:







Pour les méthodes K Means, ward, complete et average, on obtient sensiblement le même résultat présenté à gauche. Pour les métriques d'évaluation silhouette et davies, le meilleur score ne correspond pas au meilleur nombre de clusters. Cependant pour le score de calinski, le bon nombre de clusters est identifié et les clusters sont bien délimités. En revanche la méthode single ne présente pas de bonne solution de clustering pour aucun des cas.

IV. Conclusion

Pour conclure, au cours de ce rapport nous avons pu effectuer une analyse comparative de deux différentes méthodes de clustering pour l'apprentissage non-supervisée à savoir k-Means et méthode de clustering agglomératif. Nous avons pu à travers une étude expérimentale analyser les points forts et limites de chaque méthode. Concernant la méthode de clustering agglomératif nous nous sommes rendu compte que le choix de la méthode de linkage était très importante vis à vis de la solution de clustering trouvée. Ce choix repose notamment sur l'analyse des dendrogrammes. L'étude par l'intermédiaire de métrique d'évaluation nous a permis de constater qu'il ne faut pas toujours se fier aux scores. En effet un meilleur score ne correspond pas toujours à la meilleure solution de clustering. Enfin dans la dernière partie nous avons pu mettre en œuvre et vérifier nos observations en appliquant les différentes méthodes sur de nouvelles données.