Master Optique, Image, Vision, Multimédia

Apprentissage non supervisé

COMPTE RENDU (final)

Réalisé par :

- **Nassim BATTACHE**

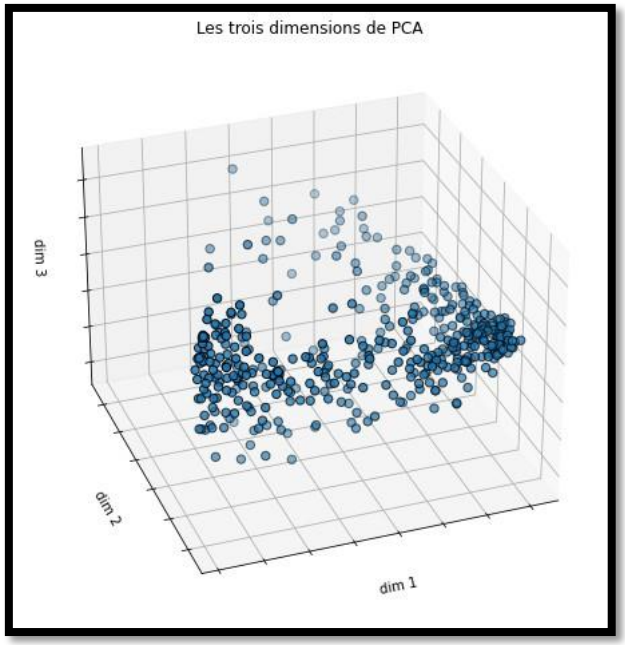
Introduction:

Le but de ce TP est d'exploiter tous les types de clusters vus en cours(KMeans, GMM, CHA, DBSCAN, SpectralClustering) sur deux datasets différentes, afin de comparer ces méthodes et leurs efficacités sur les différents types de données. Pour ce faire, on va étudier chacune de ces méthodes séparément et de Ploter les résultats de chacune d'eux.

Pour bien exploiter les données une normalisation de ces dernières est nécessaire. Les données seront comprises entre 0 et 1.

I. Plot des données de la première dataset « Wholesale customers data» avant application des méthodes de clustering :

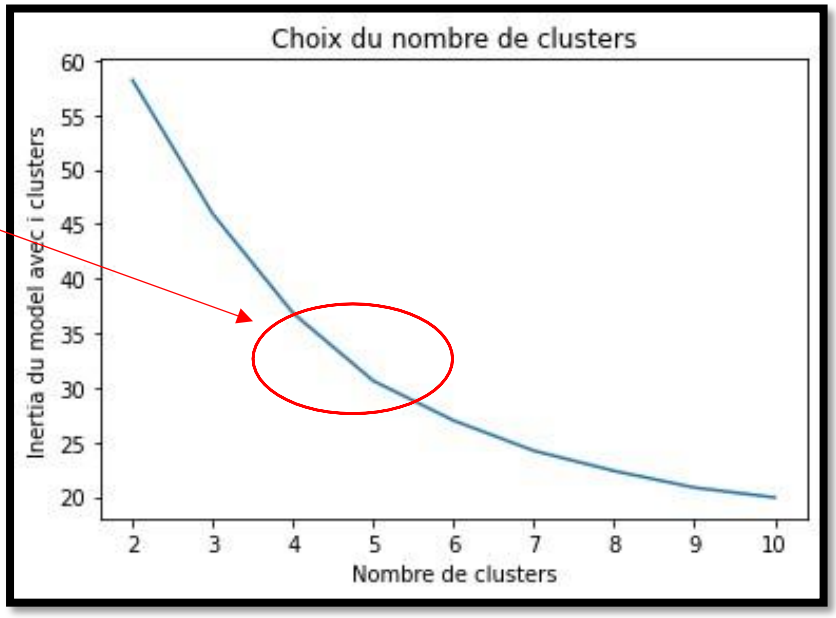
Pour ploter les données on utilisera une APC qui permet de décomposer les données en 3 dimensions.



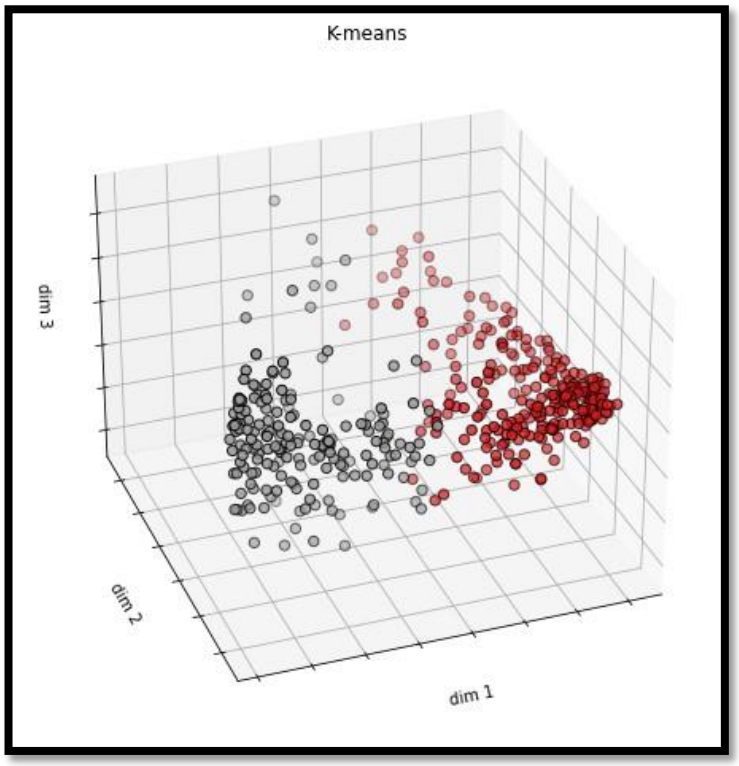
1) K-means :

Cette méthode permet de séparer les données linéairement, le nombre de clusters doit être choisi délicatement. Pour la dataset « Wholesale_customers_data.csv », on remarque qu'on dispose pas de label du coup le choix du nombre de cluster n'est pas connu au préalable, pour cela l'utilisation d'une technique qui permet de prédire ce nombre sera utile. Dans notre cas on trace la courbe des « inertia » en fonction du nombre de clusters, après à partir de la courbe on décide le nombre de clusters adéquat. Pour cet exemple un nb_clusters = 2 et 4 est jugé bon.
La courbe qui l' illustre c'est la suivante :

Le nombre de clusters adéquat



Résultat de plot après utilisation de K-means :

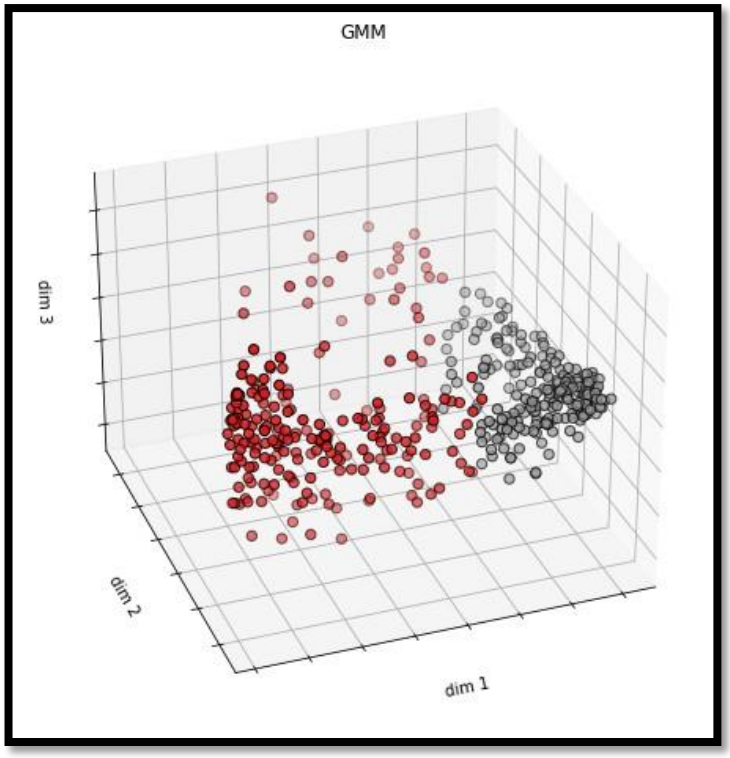


2) Gaussian Mixture Model (GMM):

Il permet d'estimer les distributions paramétriquement la distribution de variables aléatoires en les modélisant comme une somme de plusieurs gaussiennes.

Ces paramètres sont optimisés selon le critère de maximum de vraisemblance afin d'approcher le plus possible la distribution recherchée. Cette optimisation est souvent effectuée en utilisant la procédure itérative appelée Expectation-Maximisation(EM).

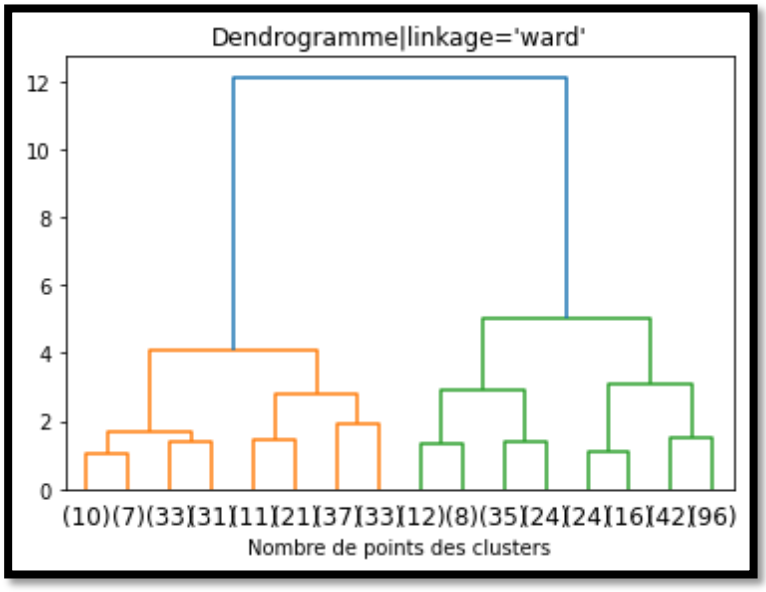
Résultat de plot après utilisation de GMM:



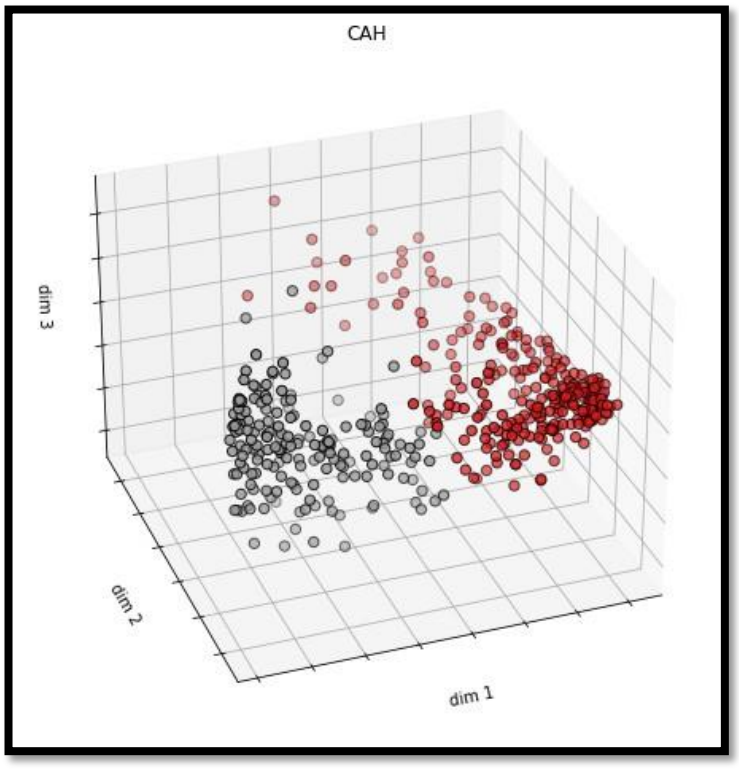
3) Classification Ascendante Hiérarchique (CAH) :

Le but de cette méthode est de rassembler les données selon des clusters en partant d'un groupe de clusters qui est égal aux nombres de données en entrée, en sortie il retourne le nombre de clusters après découpage suivants quelques critères de découpage afin d'obtenir le bon nombre de clusters.

Le dendrogramme :



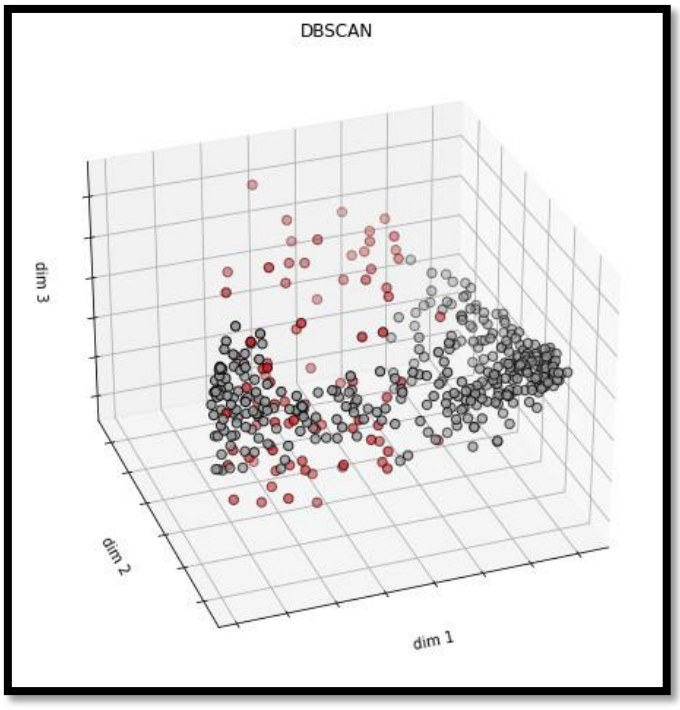
Résultat de plot après utilisation de CAH:



4) **DBSCAN :**

Il est basé sur la densité des clusters pour effectuer le regroupement des données .
Son principe est de trouver dans un rayon « r » un nombre minimum de points qui vont être assignés à un cluster.

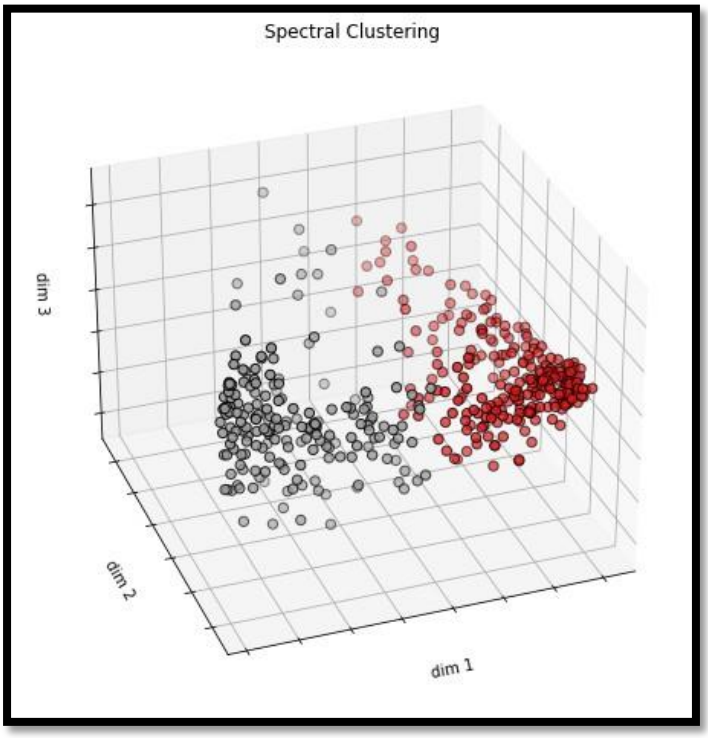
Résultat de plot après utilisation de DBSCAN:



5) **Spectral Clustering :**

Son but est de partitionner les data en K groupes en minimisant un critère de coupe.

Résultat de plot après utilisation de Spectral Clustering:



Afin d'obtenir le nombre adéquat de clusters, on sait bien que le choix de ce dernier lors de l'application d'un modèle de clustering ne se fait pas aléatoirement d'où l'importance d'utiliser le **silhouette score** qui permet d'estimer la qualité d'un cluster de données, un coefficient de silhouette est la distance moyenne entre un point avec les autres points intra classe et la distance moyenne de ce point avec les points qui appartiennent aux autres clusters.

Les résultats des silhouettes pour les cinq méthodes (k-means, GMM, CHA, DBSCAN, SpectralClustering) sont les suivants :

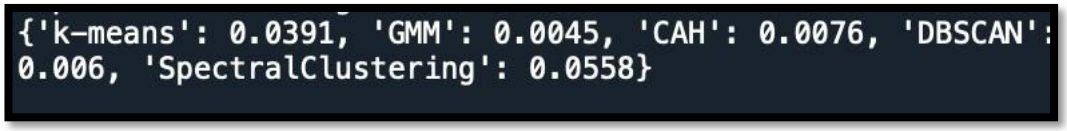
```
{'k-means': 0.5381807664792913, 'GMM': 0.49327783115581836,
'CAH': 0.5282211043672429, 'DBSCAN': 0.18014795840388056,
'SpectralClustering': 0.535468937705174}
```

Remarque : On peut constater que le résultat de score silhouette de 'K-means' est le meilleur parmi tous les autres scores, ce qui veut dire que le 'K-means' est le plus efficace avec 53.8% de précision en terme de prédiction du nombre de clusters par rapport aux autres, mais les autres scores ne sont aussi assez bons, mais dans cette dataset le 'k-means' est bien adéquat pour classifier les données.

Analyse des résultats :

D'après l'étude qu'on a fait sur les différentes méthodes de clustering on constate que chacune d'eux à ces propres avantages et inconvénients, commençant par le k-means qui est très bon en ce qui concerne les données linéairement séparables, comme dans la dataset vue ci-dessus on remarque bien qu'il était le plus accurate en terme du résultat de score silhouette score_silhouette = 53.8%, ensuite, le Spectral Clustering a pu prédire le nombre de clusters avec une précision de 53.5% , d'ailleurs ce modèle est très efficace généralement. Le CAH en partant d'un nombre de clusters très grand il a pu faire sortir le nombre final de clusters = 2 avec une exactitude = 52.8%, et juste après ça vient le GMM avec un score_silhouette = 49.3%. En dernier le DBSCAN était très mouvais ce notre dataset avec une certaine configuration, son score_silhouette = 18%.

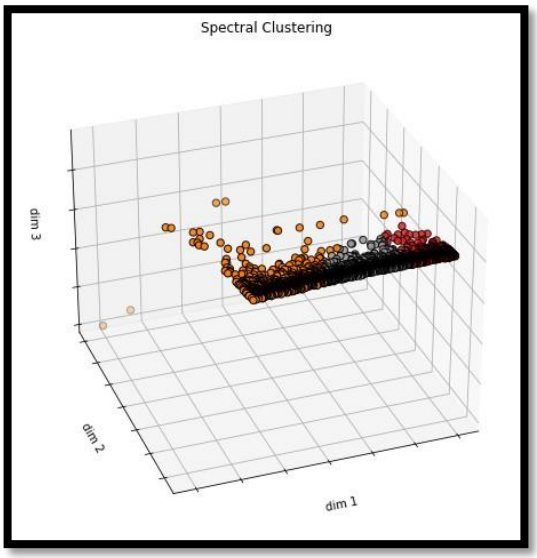
Maintenant en ce qui concerne le temps d'exécution, on constate bien que le GMM est le plus efficace si on prend on considération le score_silhouette car le temps d'exécution ne sert à rien si l'accuracy est mauvaise, après ça vient le CAH, ect… les résultats en terme du temps d'exécution sont montrés ci-dessous :

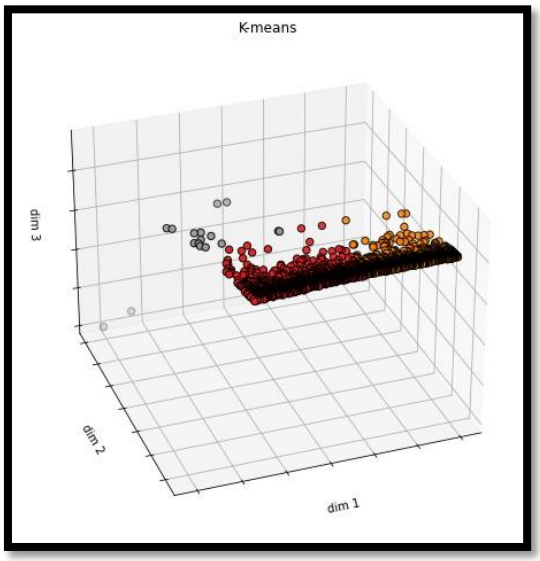


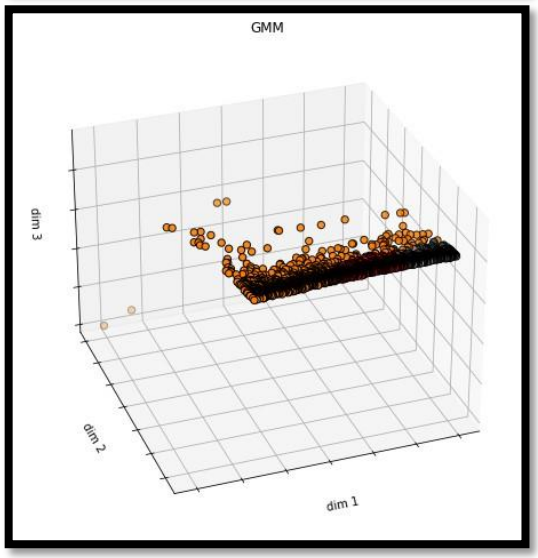
II. Plot des données de la deuxième dataset avant application des méthodes de clustering :

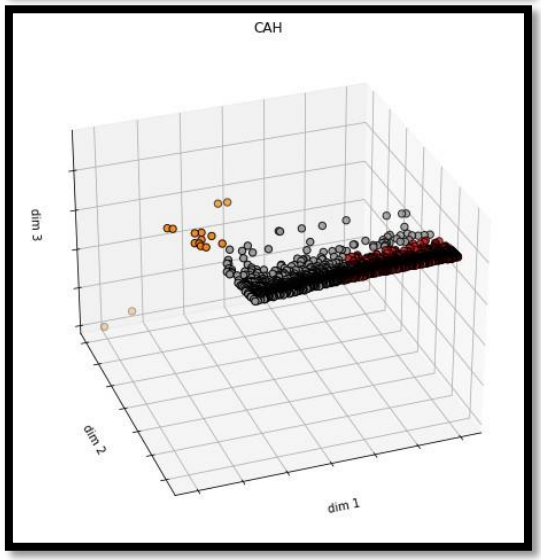
Pour la deuxième dataset, le travail est le même que celui effectué pour la première dataset.

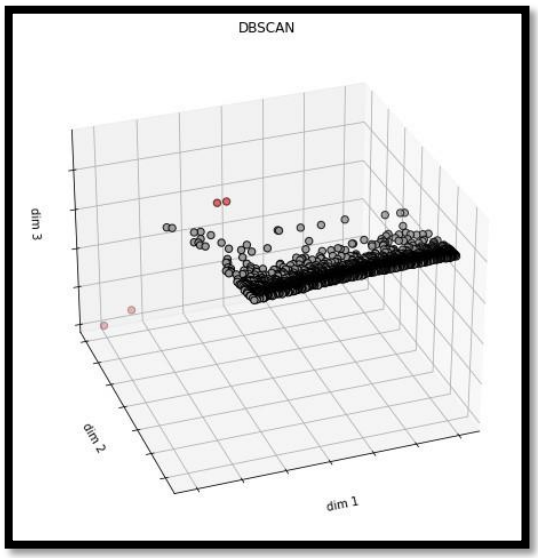
Les plots de tous les modèles sont les suivants :



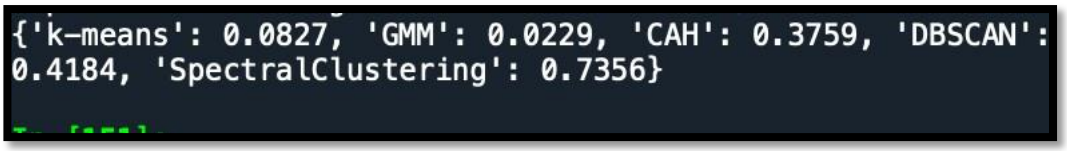








Les résultats de performances :



Analyse des résultats :

Contrairement à la première dataset « Wholesale customers data» qui est moins faible en termes de densité des données, la deuxième dataset « rfm_data» est beaucoup plus dense, laraison pour laquelle DBSCAN est plus efficace parmi tous les autres modèles avec un score_silhouette = 91%, qui est parfait, même le temps d'exécution n'est pas si mauvais avec un temps d'exécution = 0.41 secondes
Le k-means est toujours bon avec une précision de 57.4% et un temps d'exécution = 0.08 secondes, en d'autre part le reste des modèles ont presque les mêmes performances.

Conclusion :
Tout au long les TPs vus on a pu exploiter les modèles de clustering les plus utilisés. On peut conclure que chacun de ces modèles a ses propres avantages et inconvénients, et l'efficacité d'une méthode diffère d'un problème à un autre, et elle s'évalue selon plusieurs critères, parmi ces critères, le score_silhouette, le temps d'exécution, ect…
En fin, l'implémentation de ces modèles sous python nous a permet de bien comprendre ce qui est fait en cours.