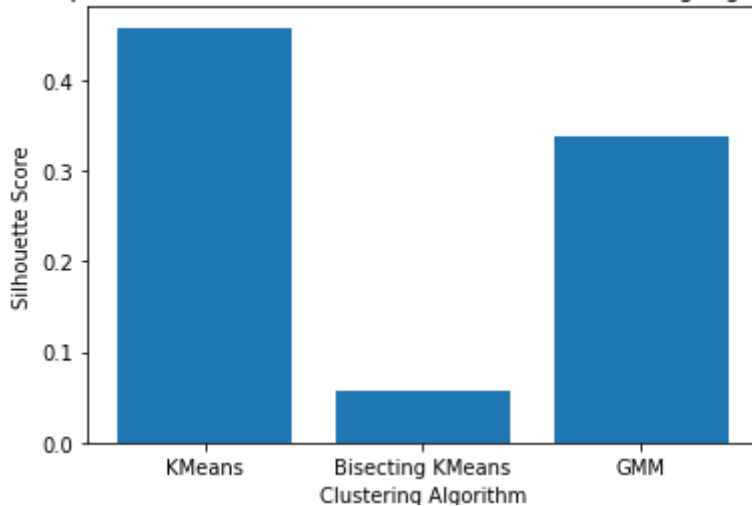


Rapport : M2-BigData-MLIB-UnsupervisedLearning

Première approche :

- Réglage du modèle : Utilisation de la validation croisée avec KMeans, BisectingKMeans, et Gaussian Mixture Models (GMM). Hyperparamètres ajustés incluent le nombre de clusters, le nombre maximal d'itérations, et la graine pour la reproductibilité.
- Interprétation des résultats : Les scores de silhouette indiquent une bonne séparation des clusters. KMeans obtient le meilleur score suivi de GMM, et BisectingKMeans a le score le plus bas.

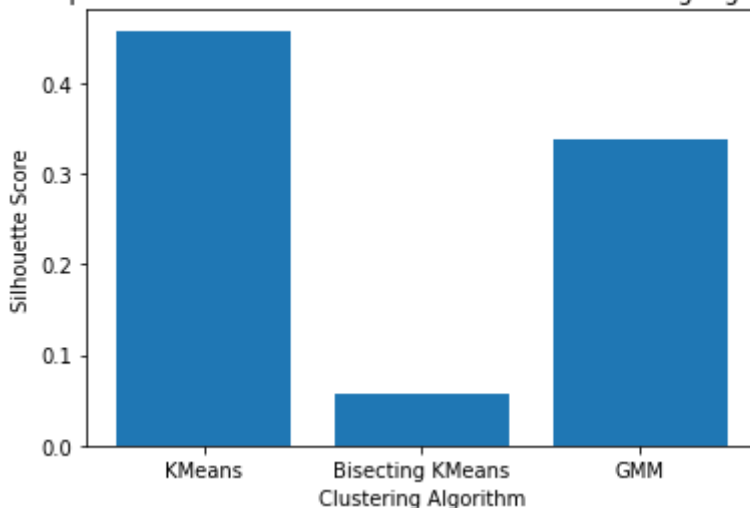
Comparison of Silhouette Scores for Different Clustering Algorithms



Seconde approche :

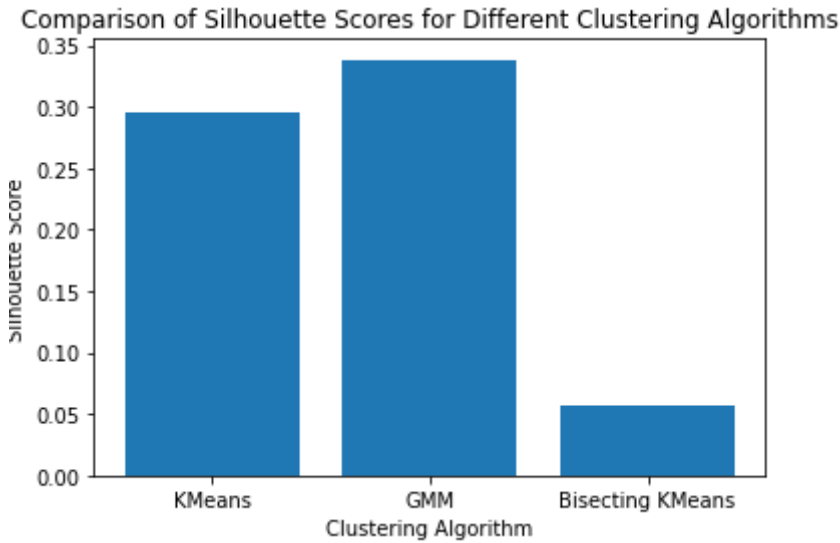
- Réglage du modèle : Similarité à l'approche 1, mais avec l'adoption d'un pipeline et des évaluateurs intégrés. ParamGridBuilder est utilisé pour définir une grille d'hyperparamètres.
- Interprétation des résultats : Les résultats sont identiques à l'approche 1, ce qui confirme la cohérence des résultats malgré la simplification du processus.

Comparison of Silhouette Scores for Different Clustering Algorithms



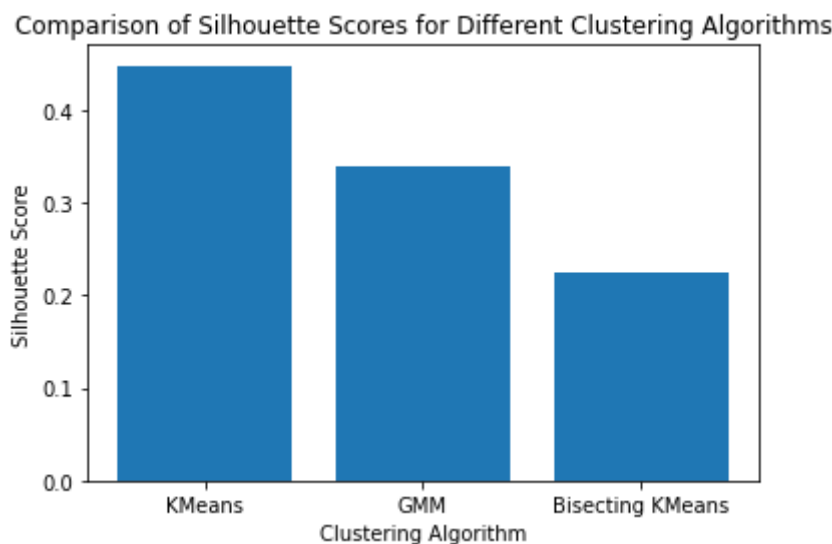
Troisième approche :

- Réglage du modèle : Structurée avec des fonctions définies pour optimiser les paramètres de KMeans et de BisectingKMeans. Évaluation modulaire.
- Interprétation des résultats : GMM obtient le meilleur score de silhouette, suivi de KMeans, et BisectingKMeans a le score le plus bas.



Quatrième approche :

- Réglage du modèle : Optimisation des hyperparamètres spécifiques à chaque algorithme avec ParamGridBuilder.
- Interprétation des résultats : KMeans obtient le meilleur score, suivi de GMM, et BisectingKMeans a le score le plus bas.



Comparaison Générale :

- Les approches 1 et 2 donnent des résultats similaires, ce qui montre que l'utilisation d'un pipeline n'a pas d'impact significatif sur la performance.
- L'approche 3, bien qu'elle ait des scores de silhouette plus bas pour KMeans et BisectingKMeans, obtient le meilleur score global avec GMM.

- L'approche 4, en optimisant directement les hyperparamètres spécifiques, semble fournir des résultats compétitifs avec KMeans en tête.

En général, l'approche 3, en adoptant une approche structurée et modulaire, semble offrir un bon équilibre entre performance et clarté du code.