# Rapport TP Machine learning

• Plan : – Introduction (0,5 page). Fournir le lien vers le code en fin de l’introduction. – Partie 1 : Points forts et points faibles identifiés pour les différentes méthodes de clustering étudiées(4) – Partie 2 : Etude et Analyse comparative de méthodes de clustering sur de nouvelles données fournies (6) – Conclusion (0,5 page)

## Introduction

Dans cette série de TP nous allons mettre en pratique les connaissances acquises en apprentissage non-supervisé. L’objectif est de comparer différentes méthodes de clustering (k-means, clustering agglomératif et DB Scan Clustering). Ces méthodes seront appliquées sur plusieurs jeux de données en 2 dimensions. Dans une première partie, nous analyserons chaque méthode et dégagerons des points forts et faibles pour chacun d’entre elles. Enfin, dans une seconde partie nous fournirons des nouvelles données à chaque méthode et en tirerons une analyse comparative.

## Particularités des différentes méthodes de clustering

### Jeux de données

L’objectif de cette partie est d’analyser les algorithmes de clustering disponibles. Pour cela on a choisis différents jeux de données pour extraire des informations utiles sur les résultats. Les jeux de données choisies sont donc:

**x-clara**, un jeu de donnée qui montre trois groupes de données qui correspond à une vision “classique” d’un cluster: groupe de points dense et distincts;

**cassini**, là encore, un jeu de donnée qui montre aussi beaucoup de séparation entre les différents clusters. La particularité de celui ci c’est que les groupes ne sont pas très denses et pas forcément circulaires.

**3-spiral**, un jeu de donnée pour lequel la notion de cluster est plus complexe à identifier. Un oeuil humain voudrait grouper chaque branche de la spirale mais elles ne correspond plus à un groupe dense de point.

**birch-rg1**, un jeu de donnée qui ne représente aucun cluster, par contre il comporte un grand nombre de point, ce qui permettra de comparer les vitesse d’exécution des trois algorithmes.

## Méthodes d’évaluation

Pour évaluer la qualité de nos clusterings avec la méthode k-means, nous disposons de 3 métriques d’évaluation :

### Critère de Calinski-Harabasz

Ce critère est le ratio de la somme des dispersions entre les clusters et à l’intérieur des clusters et de la dispersion pour tous les clusters. On définit ici la dispersion comme la somme des distances au carré.

Plus ce critère est élevé, plus les clusters évalués sont bien définis.

### Critère de Davies-Bouldin

Ce critère représente la similarité moyenne entre clusters, où la similarité est une mesure qui compare la distance entre clusters avec la taille des clusters eux-mêmes.

Plus le score est bas, mieux les clusters sont séparés, le score minimum étant zéro.

### Silhouette Coefficient

Ce coefficient est compris entre +1 et -1 et se situe près de zéro s’il y a de “l’overlap” de clusters en assumant une définition “classique” du cluster (groupe de points). Plus ce coefficient est élevé, mieux les clusters du modèle sont bien définis.

Désavantages : Le coefficient de la silhouette is généralement plus élevé pour les clusters convexes que pour les autres concepts de clusters tels que les clusters basés sur la densité comme ceux obtenus via DBSCAN.

## K-Means

L’application de l’algorithme K-Means permet de regrouper les points ensemble par proximité directe. On peut constater que pour le dataset xclara avec 3 cluster, le résultat est assez bluffant, les trois groupes sont bien identifiés !

bon\_exemple\_k-means

Malheureusement cette méthode ne fonctionne pas correctement pour tous les datasets, par exemple avec le dataset cassini, qui comporte 3 groupes mal identifiés :

mauvais\_exemple

Autre exemple de clusters mal identifiés pour 3-spiral :

3-spiral-k=3.png

Ci-dessous, voici une application de la méthode du “coude” pour les 3 jeux de données que nous avons décidé d’utiliser.

3-spiral-score.png cassini-score.png xclara-score.png

Nous avons également mesuré le temps de calcul pour chaque clustering :

3-spiral-compute-time.png cassini-compute-time.png xclara-compute-time.png

### Tableau récapitulatif

| Jeu de données | Nombre de clusters idéal | Temps calcul associé (s) |
| --- | --- | --- |
| 3-spiral | 3 | 0.062 |
| cassini | 5 | 0.058 |
| xclara | 3 | 0.052 |

### Limites de la méthode k-means

La méthode K-means se base sur la distance moyenne des point. Il s’avère être efficace pour les clusters “cirulaires” tels que xclara. On observe une limite pour

## Clustering agglomératif

### Intérêts de la méthode

Le clustering agglomératif consiste à regrouper les données en clusters en utilisant une stratégie hiérarchique. Les avantages de cette méthode sont sa capacité à générer des clusters de tailles et de formes variables. Cependant, il présente également des inconvénients tels que la sensibilité aux choix initiaux et la complexité de l’algorithme. De plus, la vitesse de calcul peut être un problème pour certains jeux de données volumineux.

Pour étudier cette méthode, nous utilisons la méthode AgglomerativeClustering de la bibiothèque scikit-learn. Nous faisons varier le paramètre *distance\_threshold* qui représente la distance à partir de laquelle différents clusters ne seront plus fusionnés

Nous utilisons également le paramètre *linkage* qui va définir la méthode utiliser afin de calculer la distance entre les points. Nous pouvons utiliser les paramètres suivant :

* **ward** : minimise la variance des clusters fusionnés
* **average** : utilise la moyenne des distances de chaque observation de deux sets.
* **complete** : utilise la distance maximum entre toutes les observations de deux sets.
* **single** : utilise le minimum des distances entre toutes les observations de deux sets.

### Limites de la méthode

# Partie 2 - Nouvelles données

Pour visualiser comment seraient classées de nouvelles données, nous avons labelisé les points de l’espace pour les visualiser. Les résultats peuvent être trouvés sur les fichiers {méthode}/{modèle}-extra.png.

## K-Means

Pour K-Means on peut constater sur les différents modèles que la méthode groupe les nouveau points en fonction de la proximité avec les clusters existants. On comprend donc bien que la clusterisation de la spirale n’est pas possible avec cette méthode.

Pour les jeux cassini et xclara le regroupement se fait assez bien. On constate magré tout que la séparation sur xclara n’est pas très claire au niveau du cluster en abs à droite. Cela peut être du à la densité du cluster qui est plus éloigné.

Pour cassini on retrouve exactement le découpage du plan et on peut constater que le groupe de points central n’est pas bien identifié.

Pour 3-spiral on constate là aussi que le découpage proposé n’est pas du tout en adéquation avec les points. Cette méthode ne peut donc pas s’appliquer