

“Segmentez des clients d’un site de e-commerce”

Projet n°5

Léa ZADIKIAN
Parcours Data Scientist
6 Février 2023

Segmentation des clients du marketplace Olist

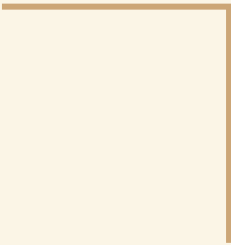
- **Olist** est un **site de e-commerce brésilien** qui propose des solutions de vente sur des marketplaces en ligne.
- Pour **aider ses équipes marketing dans leurs campagnes de communication**, Olist souhaite **mieux connaître ses différents types d'utilisateurs** à partir de leur comportement et de leurs données personnelles.



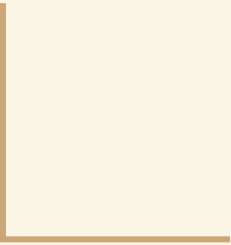
- **Notre mission :**
 1. Réaliser une **segmentation des clients** (regrouper les clients de profils similaires) et en fournir une **description actionnable** pour les équipes marketing.
 2. Recommander la **fréquence à laquelle cette segmentation doit être mise à jour** pour rester pertinente (devis contrat de maintenance).

Segmentez des clients d'un site de e-commerce

- I. Présentation des données, du feature engineering et de l'exploration**
- II. Les différentes approches de modélisation**
- III. Simulation pour le contrat de maintenance**



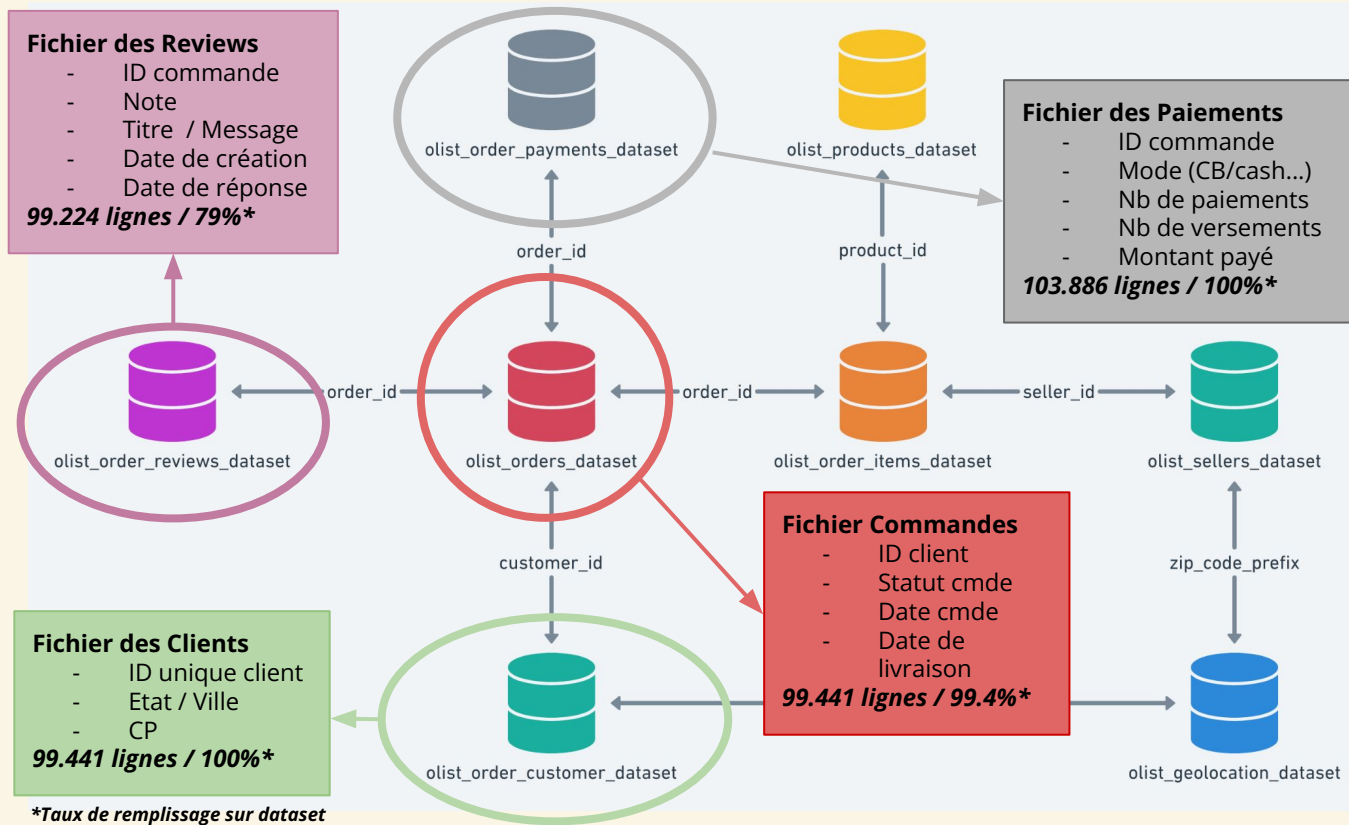
I. Présentation des données, feature engineering et exploration



Environnement technique

- Notebook Jupyter 6.4.8
- Python 3.9.12
- Librairies utilisées :
 - Pandas
 - Numpy
 - Missingno
 - Matplotlib, Seaborn, Plotly
 - Scikit-Learn
 - Yellowbrick

Base de données anonymisée Olist



CLIENTS

- 42% des clients proviennent de l'état de Sao Paulo, 13% de l'état de Rio de Janeiro.

COMMANDES :

- 97% des cmde au statut "livré".
- Historique de cmde : oct 2016→sept 2018.

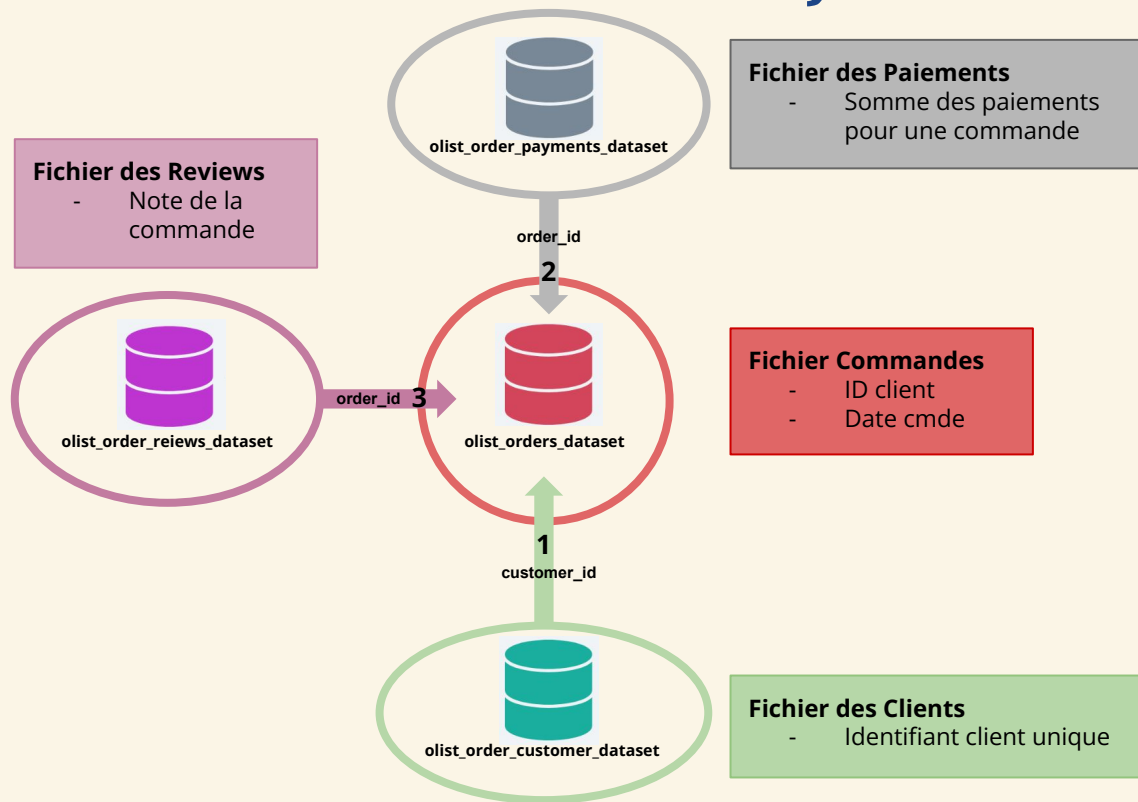
PAIEMENTS

- Près de 75% des paiements sont réalisés en CB.
- Paiement en 1 à 24 versements, 50 % des paiements effectués en 1 fois.
- Si plusieurs modes de paiements cumulés sur une même commande, alors plusieurs lignes de paiements associées à la commande.
- Montant des paiements de 0 à 13.600€, en moyenne 154€.

REVIEWS

- 58 % des reviews ont une note de 5/5, 19% des reviews ont une note de 4/5. Près de 15% des reviews avec 1 ou 2.

Base de données anonymisée Olist



JOINTURES

1. Fichiers des clients avec le fichier des Commandes.
2. Ajout du montant payé pour une commande (somme des paiements sur une commande) du fichier des Paiements.
3. Ajout du review score issu du fichier des Reviews.

Nettoyage du jeu de données

Opération de nettoyage	Commentaires
1. Filtrage sur les commandes au statut "Livré"	2.963 commandes supprimées (3%)
2. Suppression des doublons sur les reviews (plusieurs reviews sur même commande)	1.098 reviews supprimées (1%)
3. Suppression des clients sans review score	765 clients supprimés (0,8%)
4. Typage des dates (object → datetime)	/

RÉSULTAT

Fichier de 96.478 commandes sans valeur manquante avec :

- Identifiant client unique
- Montant payé pour la commande
- Date de commande
- Review score de la commande

Feature engineering

1. Segmentation RFM

La **segmentation RFM** est une méthode de **segmentation comportementale** fréquemment utilisée pour analyser la valeur des clients, les regrouper **en segments homogènes** et **définir une stratégie commerciale et marketing adaptée** à chaque segment :

- **Récence** (ou Recency) : nombre de jours depuis le dernier achat.
- **Fréquence** (ou Frequency) : nombre d'achats réalisés par le client sur une période donnée.
- **Montant** (ou Monetary value) : montant dépensé par le client sur la période.

Concrètement :

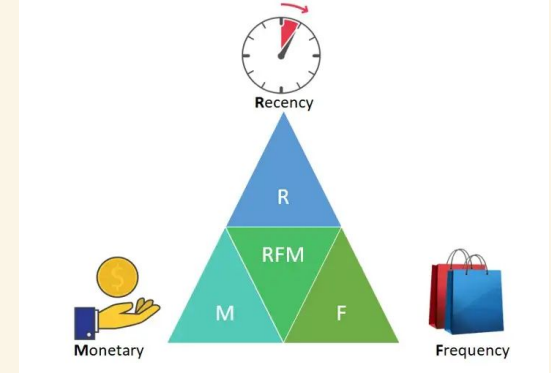
- Regroupement des commandes par client unique (*groupby()* et *fonction d'agrégation pour chaque feature*).
- Prise en compte de toute la période à disposition : septembre 2016 à octobre 2018.
- Pour le calcul de la récence, on se place au lendemain de la date du dernier achat.

⇒ **Fichier de 93.358 clients uniques avec les features RFM.**

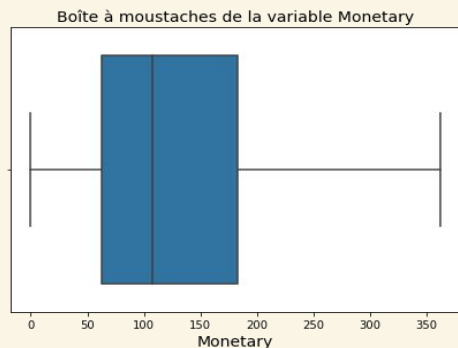
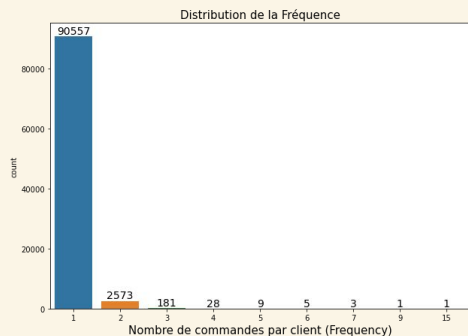
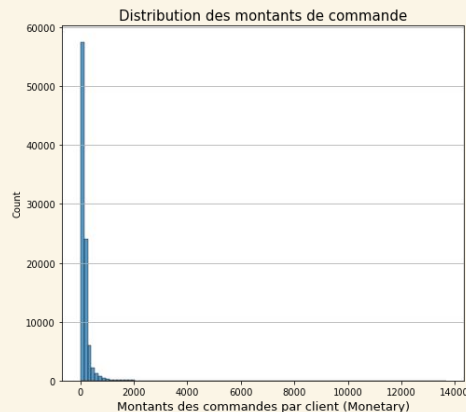
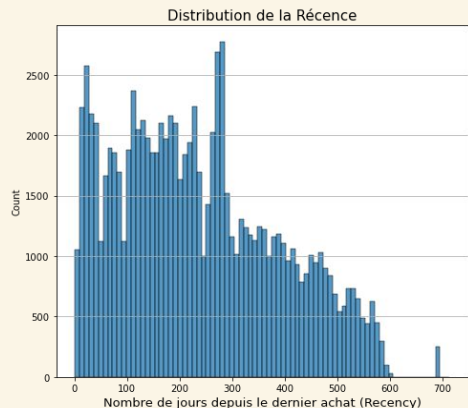
2. Ajout du Review Score

Les review scores sont agrégés en effectuant la moyenne des notes attribuées par un client à ses différentes commandes.

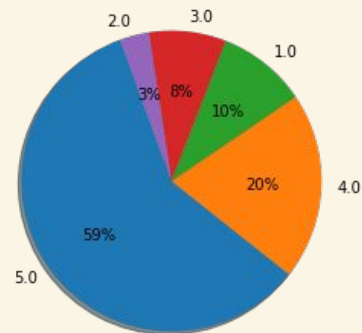
⇒ **Fichier de 92.593 clients uniques avec les features RFM + le Review Score**



Exploration des données



Distribution de la moyenne des notes (1 à 5) attribuée par un client à ses commandes

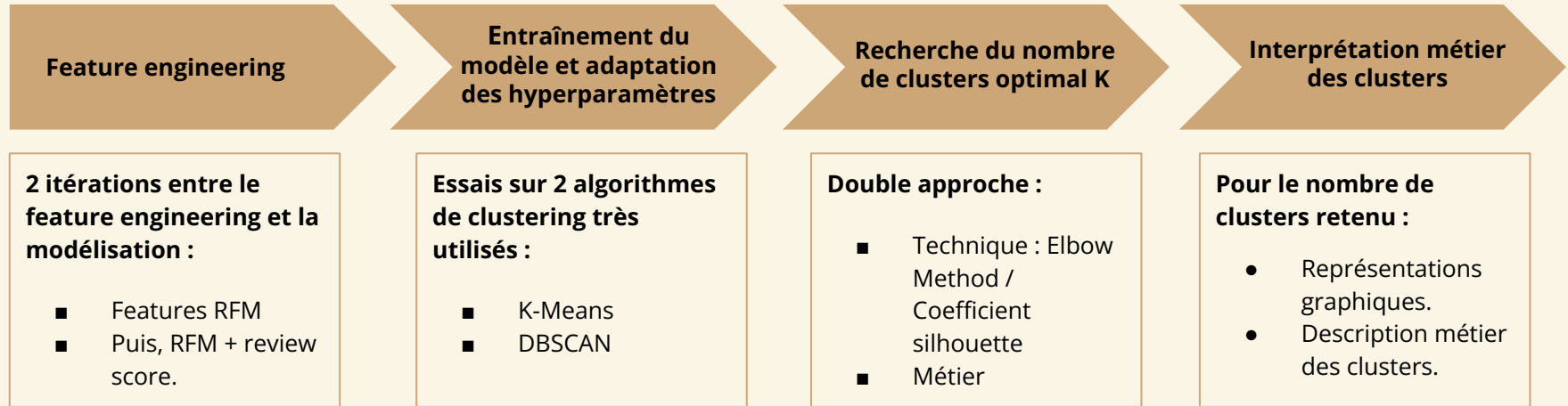


COMMENTAIRES

- Historique des commandes : octobre 2016 à septembre 2018, récence de 1 à 700 jours.
- Environ 97 % des clients n'ont passé qu'une seule commande, 3 % entre 2 et 15 commandes.
- Le montant dépensé par les clients varie de 0 à 13.600€, en moyenne 165€.
- Les clients satisfaits (note de 5/5 et 4/5) représentent près de 80%. Les clients non satisfaits (1/5 et 2/5) représentent 13%.

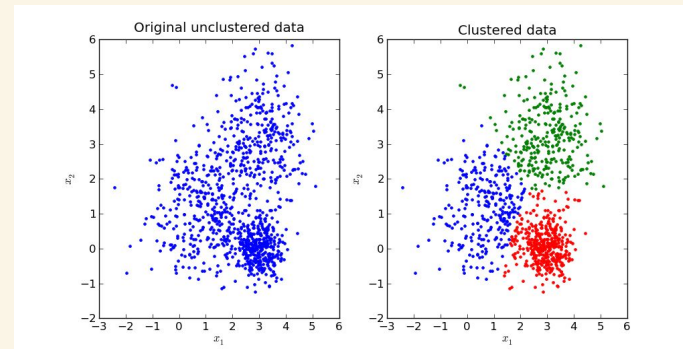
II. Les différentes approches de modélisation

Démarche de modélisation



Présentation de l'algorithme K-Means

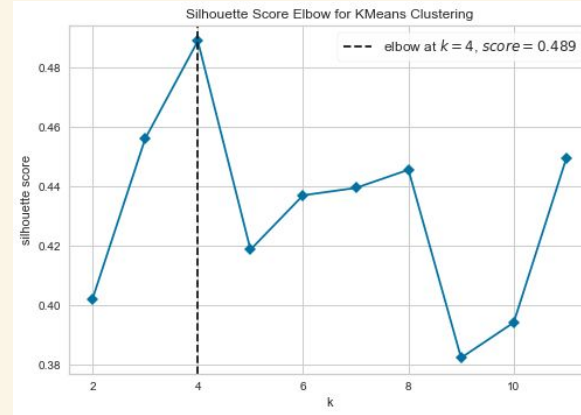
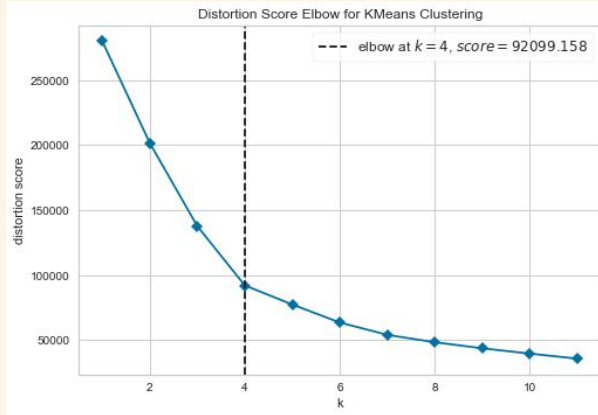
- Le K-Means est un algorithme de ML non supervisé (de clustering) couramment utilisé dans de nombreux domaines.
- Le K-Means cherche à séparer les données en k groupes, en **minimisant l'inertie** (*somme des carrés des distances intra-cluster*).
- **Initialisation des centroïdes avec *k-mean++*** (distants les uns des autres), plutôt qu'une initialisation complètement aléatoire.
- **Avantages :**
 - Assez simple à comprendre et à implémenter.
 - Converge toujours et rapidement.
 - Adapté aux formes convexes de clusters.
 - Adapté aux dataset de grandes tailles.
- **Inconvénients :**
 - **Nécessite de spécifier le nombre de clusters k .**



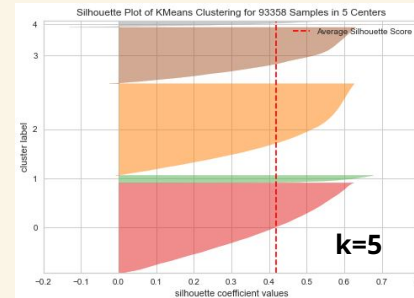
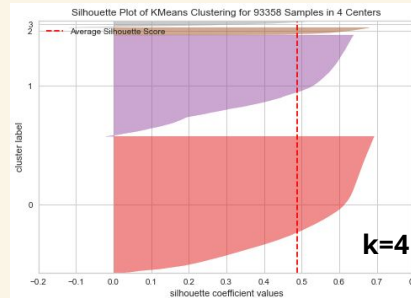
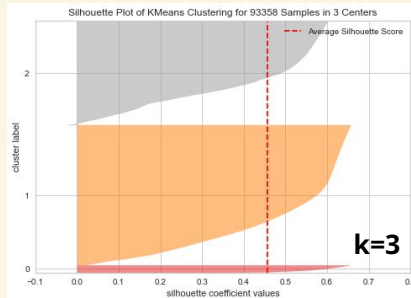
Recherche du nombre optimal de clusters K

➔ Approche double : technique (Elbow Method / coefficient de silhouette) et métier.

1^{ère} itération :
features RFM

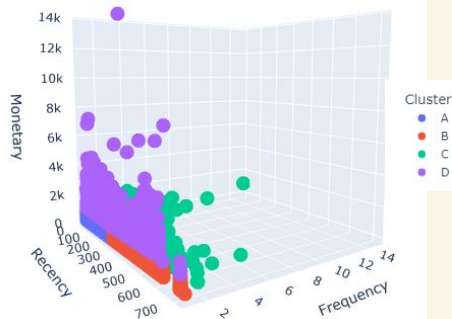


⇒ Nombre de clusters
retenu : $K = 4$



Interprétation métier des clusters (K-Means, k=4)

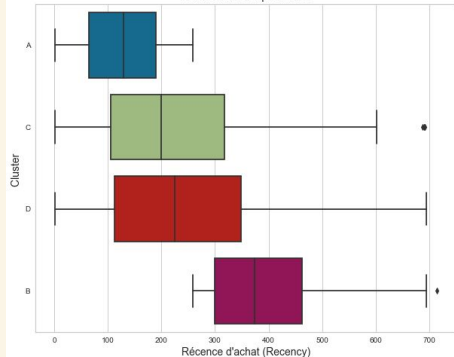
Segmentation RFM avec K-Means des clients Olist



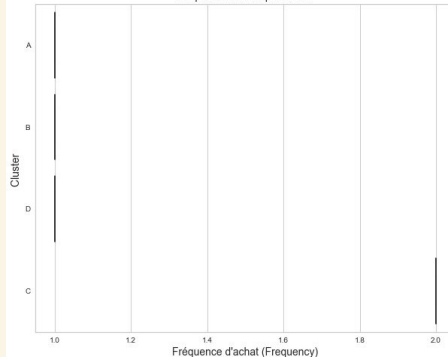
1^{ère} itération : features RFM

Cluster	Nb de clients	Interprétation métier
A	50.644	Les clients qui ont acheté récemment.
B	37.526	Les clients qui n'ont pas acheté depuis longtemps.
C	2.772	Les clients qui ont fait plus d'un achat.
D	2.416	Les clients qui dépensent le plus.

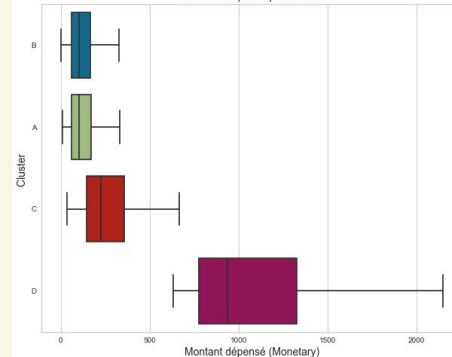
Récence d'achat par cluster



Fréquence d'achat par cluster

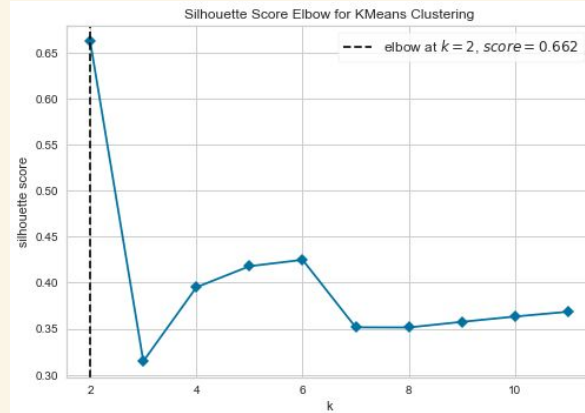
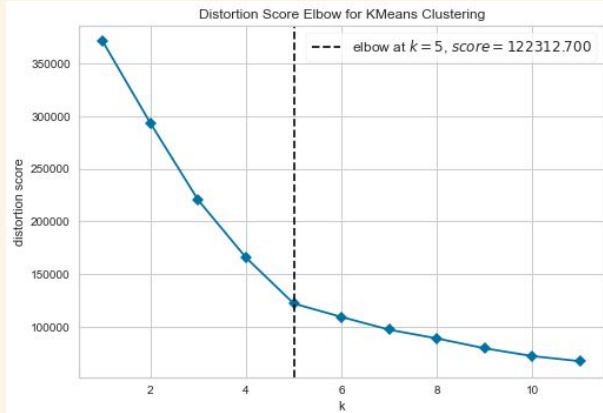


Montant dépensé par cluster

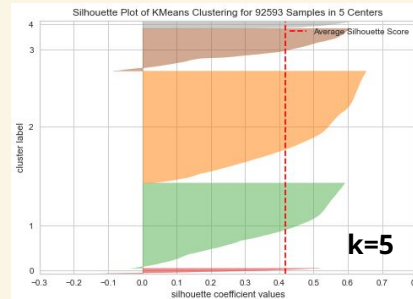
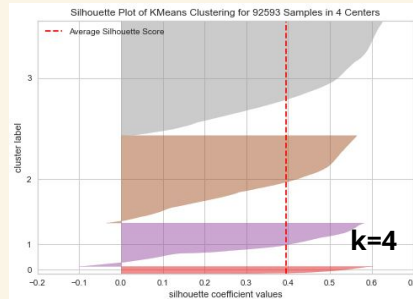
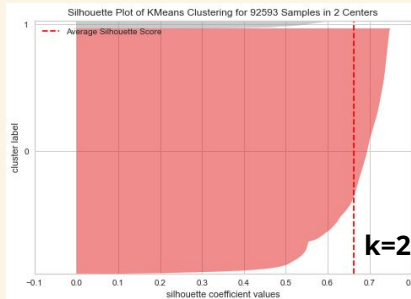


Recherche du nombre de clusters optimal K

→ 2^{ème} itération : Features RFM + Review score

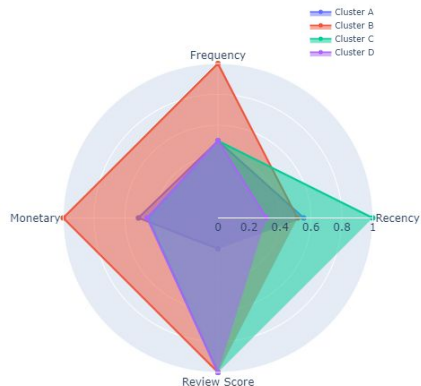


⇒ Nombre de clusters retenu : $K = 4$



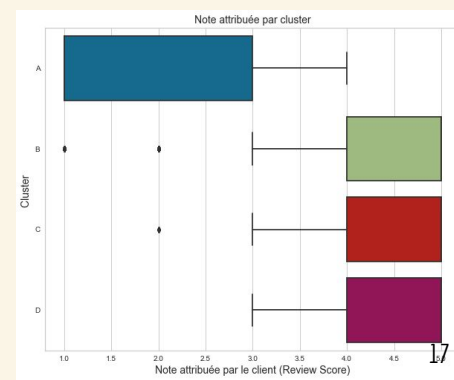
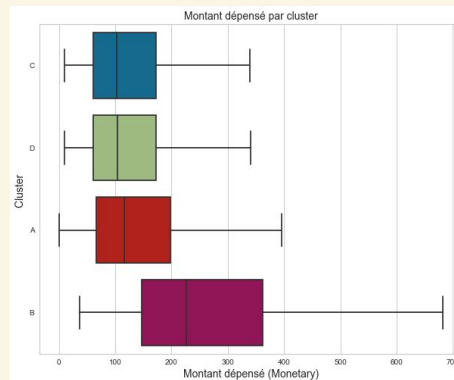
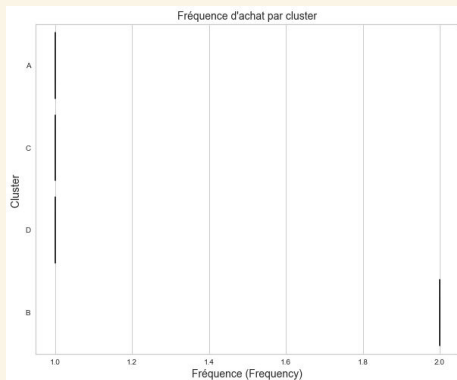
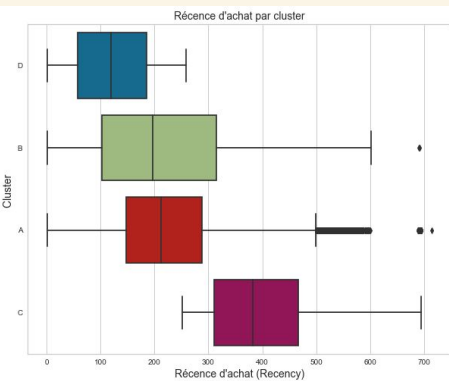
Interprétation métier des clusters (K-Mean, k= 4)

Segmentation par K-Means des clients Olist



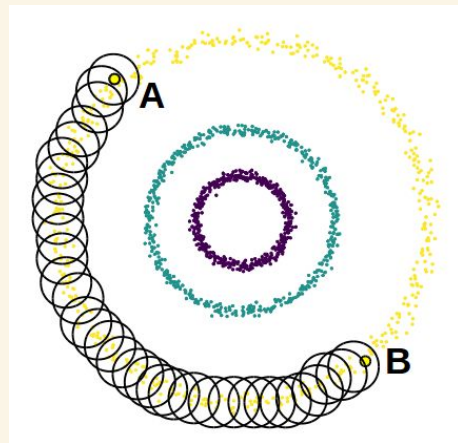
2^{ème} itération : features RFM + Review score

Cluster	Nb clients	Interprétation métier
A	41.960	Les clients moins satisfaits qui n'ont pas acheté depuis longtemps.
B	32.007	Les clients qui achètent le plus fréquemment et dépensent le plus.
C	15.902	Les clients satisfaits qui n'ont pas acheté depuis longtemps.
D	2.724	Les clients qui ont acheté le plus récemment.

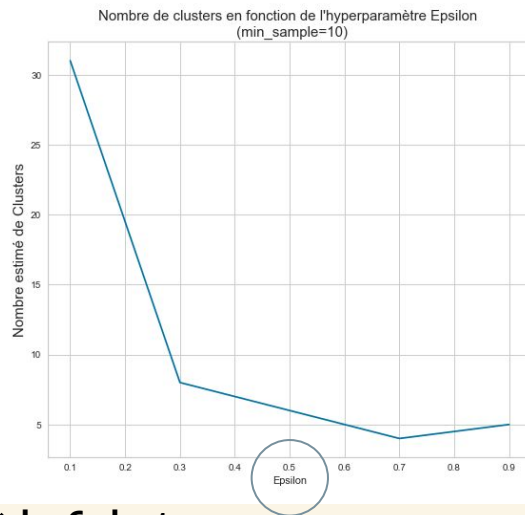
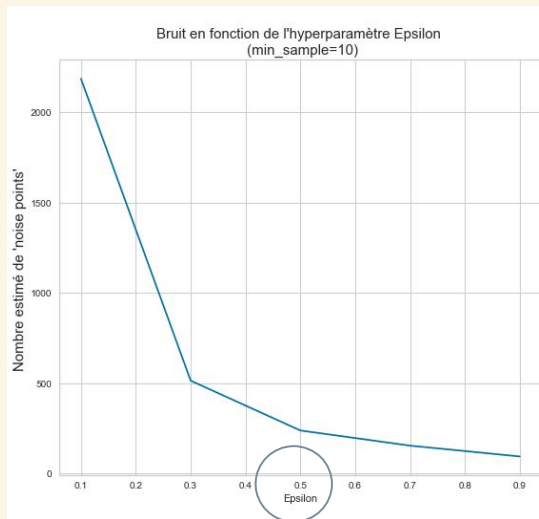


Présentation de l'algorithme DBSCAN

- DBSCAN - Density Based Spatial Clustering of Applications with Noise : **algorithme de clustering par densité.**
- **2 hyperparamètres :**
 - **Min_samples** : le nombre minimal de points de voisinage qu'un point donné a besoin pour être considéré comme dans une zone "dense".
 - **Epsilon** : distance maximale entre 2 points pour être considérés comme appartenant au même cluster.
- **Avantages :**
 - Algorithme déterministe.
 - Contrairement au K-Means, les clusters peuvent être de n'importe quelle forme.

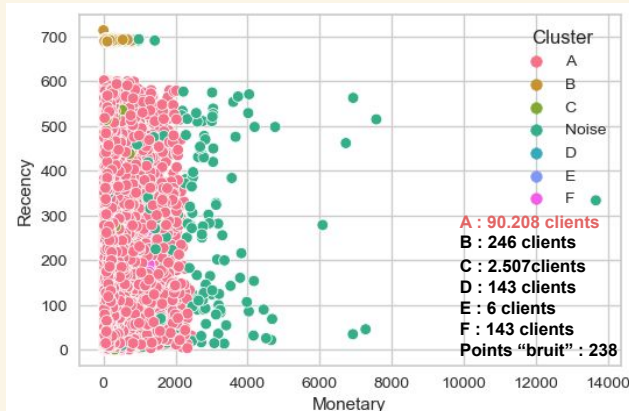


Résultat du DBSCAN (min_sample=10)



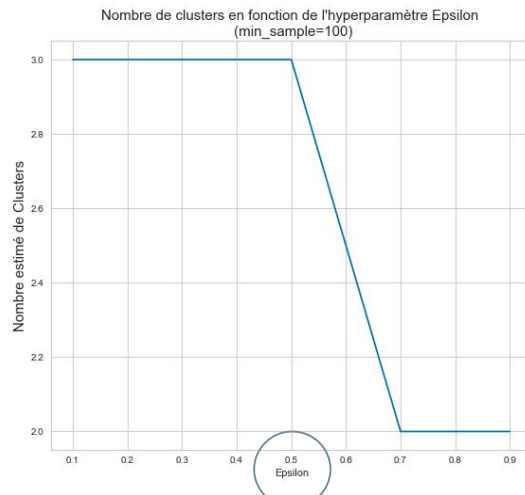
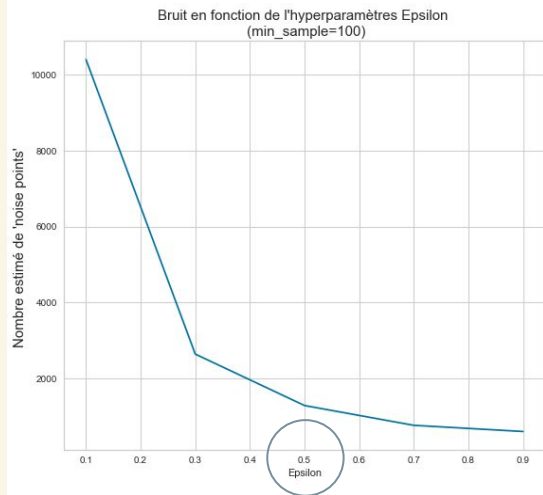
epsilon = 0.5 \Rightarrow k =6 clusters

Segmentation par DBSCAN des clients Olist
eps=0.5, min_sample=10, k =6



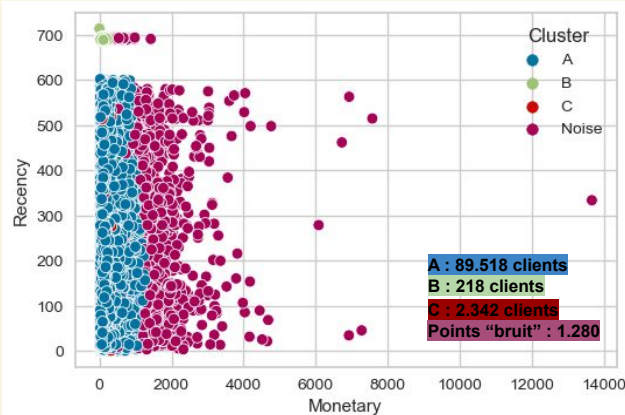
Le cluster A contient presque tous les clients.
 \Rightarrow Le DBSCAN est moins satisfaisant que le K-Means.

Résultat du DBSCAN (min_sample=100)



epsilon = 0.5 \Rightarrow k =3 clusters

Segmentation par DBSCAN des clients Olist
eps=0.5, min_sample=100, k =3



Le cluster A contient presque tous les clients.
 \Rightarrow Le DBSCAN est moins satisfaisant que le K-Means.

III. Contrat de maintenance

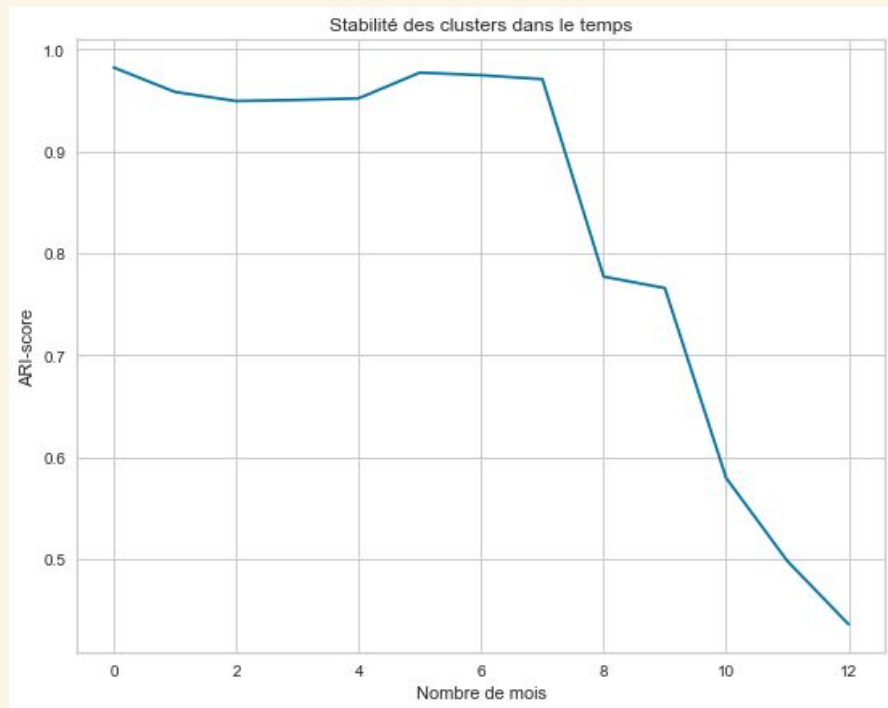
Contrat de maintenance : objectif et méthodologie

- **Objectif** : Analyser la stabilité des segments au cours du temps et **déterminer la fréquence nécessaire de mise à jour** du modèle de segmentation, dans le but de pouvoir établir un devis de contrat de maintenance.

Méthodologie :

- L'historique de commandes du jeu de données fourni par Olist s'étale sur près de 24 mois. On fixe une **période initiale de 12 mois**, on itère sur les 12 mois suivants, **par période de 1 mois**.
- On va déterminer au **bout de combien de temps la prédiction obtenue avec le modèle de clustering initial, devient obsolète** (comparaison *.fit()* et *.predict()*). Cela signifie que les clients changent de segment et qu'il est donc nécessaire d'entraîner un nouveau modèle de clustering.
- Pour comparer 2 clusterings, **utilisation du score ARI** (Adjusted Rand Index), score compris entre -0.5 et 1, mesurant la similarité entre 2 clusterings.

Stabilité dans le temps de la segmentation



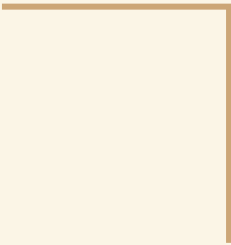
- Après 7 mois, le score ARI diminue nettement.

⇒ **Recommandation de fréquence de mise à jour de la segmentation : 2 fois par an.**

Conclusion

Ce projet a été une opportunité de :

- Mettre en œuvre des techniques de **feature engineering**, notamment liées à la **segmentation RFM**.
- Mettre en œuvre les **algorithmes non supervisés KMeans et DBSCAN** et adapter leur hyperparamètres.
- Rechercher un **nombre optimal de clusters** et leur donner une **interprétation métier**.
- Réaliser une analyse de la **stabilité temporelle des clusters**.



Merci pour votre attention !
Des questions ?

