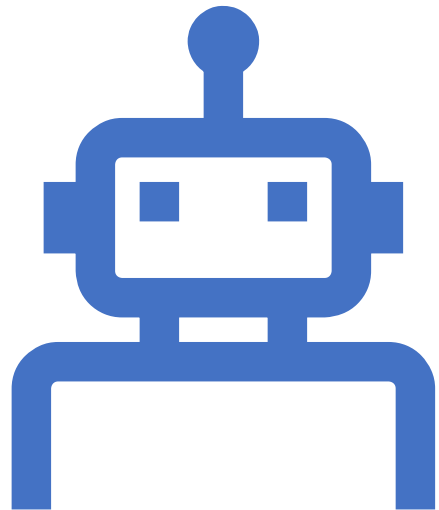


Segmentation de clientèle d'un site e-commerce





Exploration

Problématique, cleaning and feature engineering

Problématique

Aider Olist à comprendre ses différents types de clients

- ❑ Olist connecte des petites entreprises de tout le Brésil avec un seul contrat, leur permettant de vendre leurs produits sur différents marketplaces via le magasin Olist et les expédier directement aux clients en utilisant les partenaires logistiques d'Olist.

❑ La mission

Aider les équipes Marketing d'Olist à comprendre les différents types d'utilisateurs en regroupant les clients de profils similaires

❑ Les données fournies

Informations sur l'historique de commandes, les produits achetés, les commentaires de satisfaction, et la localisation des clients depuis janvier 2017.

Source des données : <https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce>

Le jeu de données

Plusieurs tables sont fournies, avec le schéma suivant., dont la table est centrale est la liste des commandes



On ne conserve que les commandes effectivement livrées.

Valeurs de la variable `order_status`

delivered	96478
shipped	1107
canceled	625
unavailable	609
invoiced	314
processing	301
created	5
approved	2

Name: `order_status`, dtype: int64



On ne conserve que les commandes passées à partir de l'année 2017

	count	unique	top	freq	first	last
order_id	96478	96478	e481f51cbdc54678b7cc49136f2d6af7	1	NaT	NaT
customer_id	96478	96478	9ef432eb6251297304e76186b10a928d	1	NaT	NaT
order_status	96478	1	delivered	96478	NaT	NaT
order_purchase_timestamp	96478	95956	2017-11-20 10:59:08	3	2016-09-15 12:16:38	2018-08-29 15:00:37
order_approved_at	96464	88274	2018-02-27 04:31:10	9	2016-09-15 12:16:38	2018-08-29 15:10:26
order_delivered_carrier_date	96476	80106	2018-05-09 15:48:00	47	2016-10-08 10:34:01	2018-09-11 19:48:28
order_delivered_customer_date	96470	95658	2018-05-08 23:38:46	3	2016-10-11 13:46:32	2018-10-17 13:22:46
order_estimated_delivery_date	96478	445	2017-12-20 00:00:00	507	2016-10-04 00:00:00	2018-10-25 00:00:00

On fusionne ensuite les tables

Après fusion des tables

Création du fichier des commandes

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 115385 entries, 0 to 115384
Data columns (total 29 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   order_id                             115385 non-null object
1   customer_id                           115385 non-null object
2   order_purchase_timestamp              115385 non-null datetime64[ns]
3   order_delivered_carrier_date          115383 non-null datetime64[ns]
4   order_delivered_customer_date         115377 non-null datetime64[ns]
5   order_estimated_delivery_date         115385 non-null datetime64[ns]
6   customer_unique_id                    115385 non-null object
7   customer_zip_code_prefix              115385 non-null int64
8   customer_city                         115385 non-null object
9   customer_state                       115385 non-null object
10  order_item_id                         115385 non-null int64
11  product_id                           115385 non-null object
12  seller_id                            115385 non-null object
13  shipping_limit_date                   115385 non-null datetime64[ns]
14  price                                115385 non-null float64
15  freight_value                        115385 non-null float64
16  payment_sequential                    115385 non-null int64
17  payment_type                          115385 non-null object
18  payment_installments                  115385 non-null int64
19  payment_value                        115385 non-null float64
20  review_id                            114528 non-null object
21  review_creation_date                  114528 non-null datetime64[ns]
22  review_answer_timestamp                114528 non-null datetime64[ns]
23  review_score                          114528 non-null float64
24  product_category_name                 115385 non-null object
25  product_category_name_english         115385 non-null object
26  seller_zip_code_prefix                115385 non-null int64
27  seller_city                           115385 non-null object
28  seller_state                         115385 non-null object
dtypes: datetime64[ns](7), float64(4), int64(5), object(13)
memory usage: 26.4+ MB
```

L'objectif sera de se focaliser sur une segmentation des clients de type **RFM (Récence, Fréquence, Montant)** enrichie d'information concernant la **satisfaction**, et les **délais de livraisons**.

Après exploration et tests, seul un nombre restreint de variables sera d'intérêt.

Les variables dont la valeur est manquante seront imputées :

- Pour les 'order_delivery_customer_date', on remplacera simplement les valeurs manquantes par les review_creation_date.
- Pour les review_scores par la moyenne des scores obtenus sur les mêmes produits ou catégories de produits

Feature Engineering

Création du fichier des données clients pour l'exploitation, avec création des variables pertinentes

0. Identifiant unique du client : on a 93104 clients distincts

1. Récence : nombre de jours écoulés depuis le dernier achat à partir d'une date de référence

2. Montant : montant total dépensé sur la période

3. Fréquence : nombre de commandes effectuées sur la période

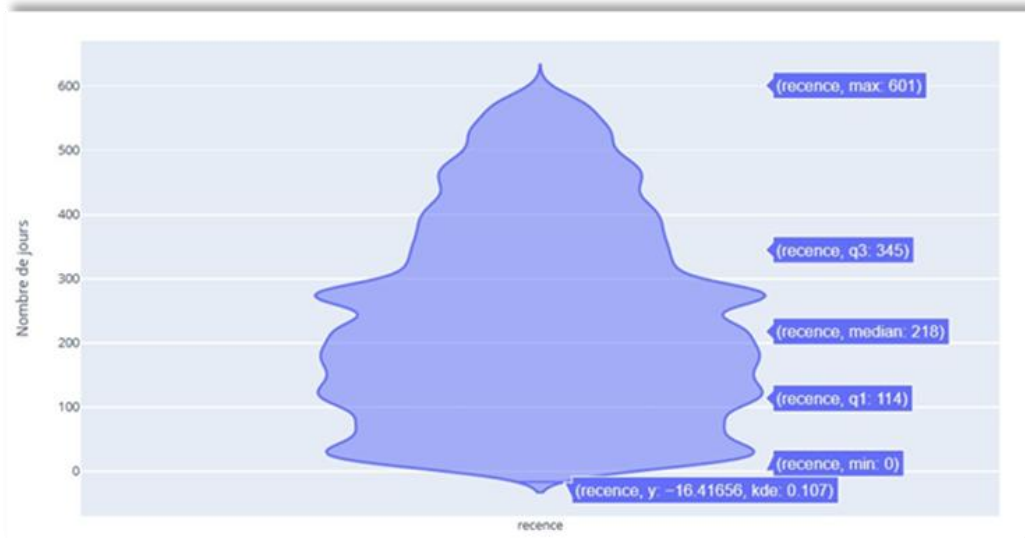
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 93104 entries, 0 to 93103
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype
---  -
0   customer_unique_id    93104 non-null object
1   recence                93104 non-null int64
2   montant               93104 non-null float64
3   frequence             93104 non-null int64
4   nb_days_delivery_delay 93104 non-null float64
5   avg_review_score      93104 non-null float64
dtypes: float64(3), int64(2), object(1)
memory usage: 5.0+ MB
```

5. Satisfaction : moyenne des notes attribuées par les clients

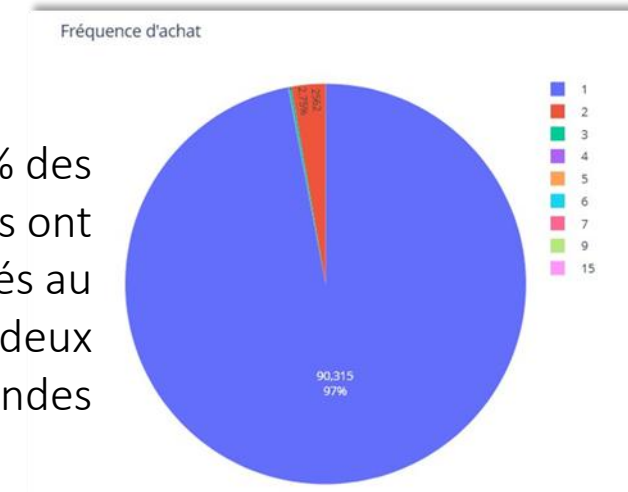
4. Retard de livraison : nombre de jours de retard de livraison cumulés (0 si aucun retard)

Récence, Fréquence d'achat et Montants dépensés

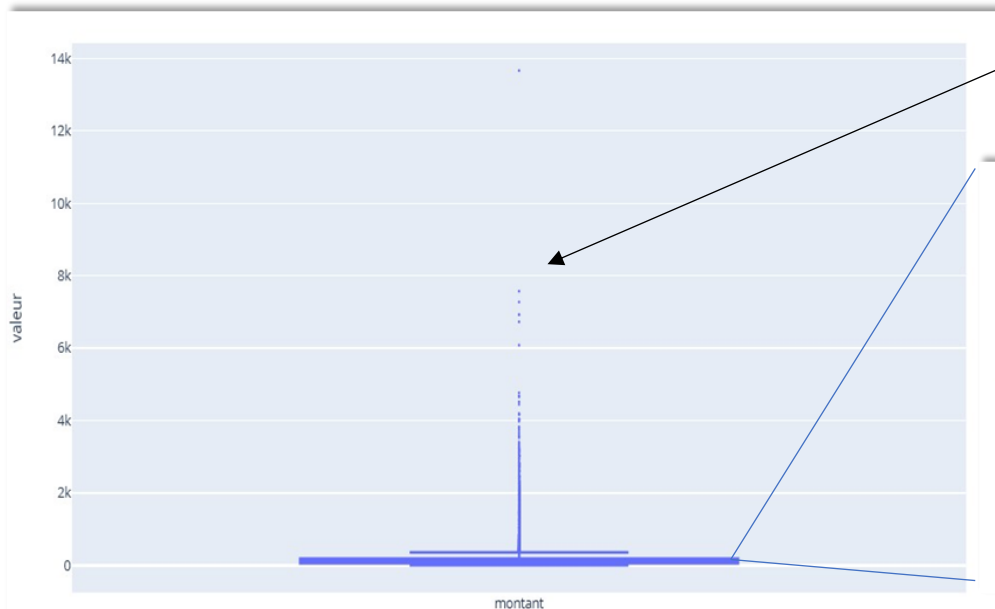
Exploration des données



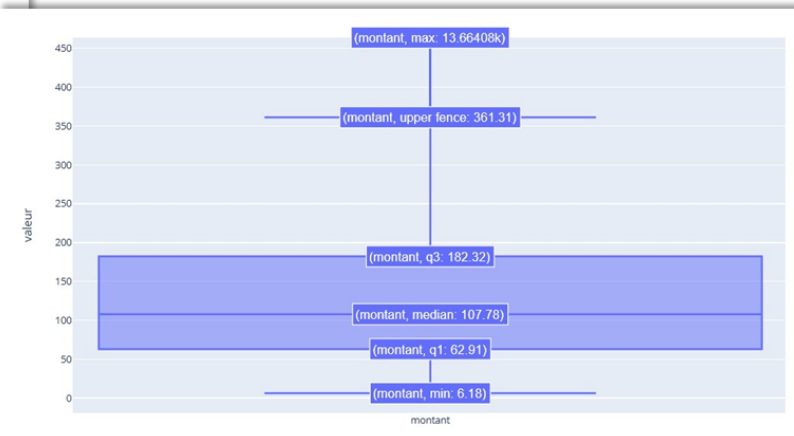
Le nombre de nouveaux clients augmente au cours du temps.



Seuls 3% des clients ont effectués au moins deux commandes



Quelques outliers en terme de montant dépensé...



La stat : 28.8 % des clients ayant commandé au moins deux fois ont passé leurs commandes...le même jour. Cela relativise un peu la notion de « fréquence »

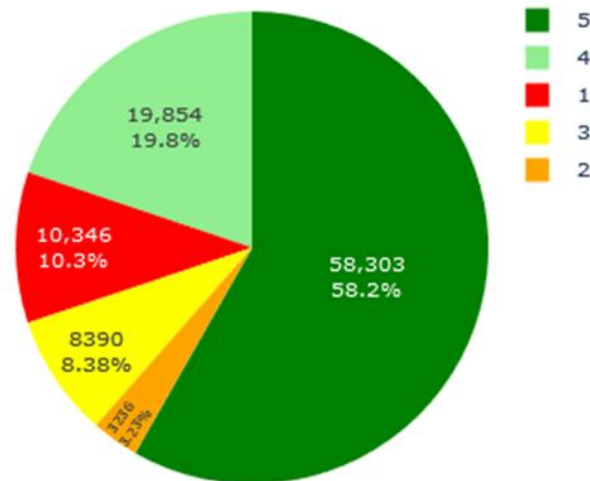
Satisfaction (review scores) et retards de livraisons

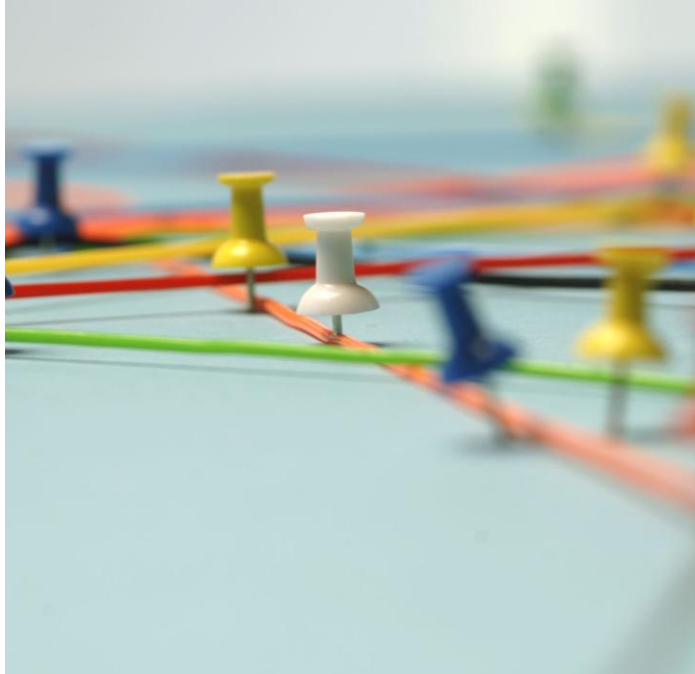
Exploration des données

- 10% des produits achetés ont reçu la note la plus basse.
- 20% sont notés de 1 à 3.

8.13 % des commandes ont été livrées en retard et cela concerne 8.34 % des clients

Répartition des review scores





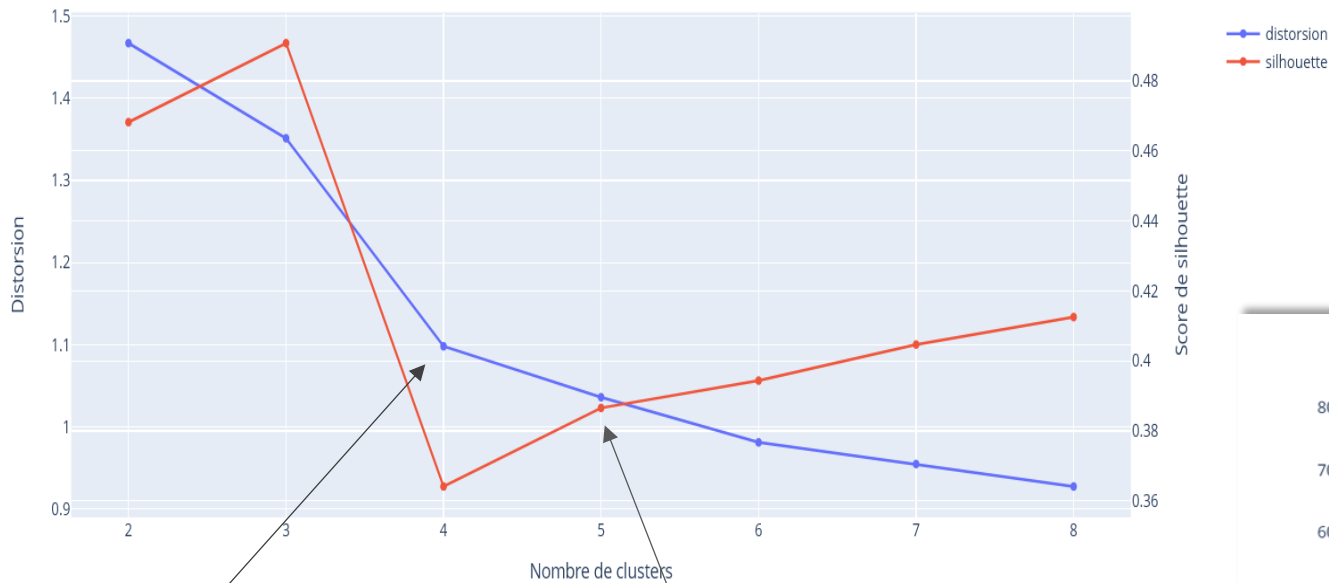
Modélisation

Choix de l'algorithme et
clustering

Algorithme choisi : K-Means

Rapide et pertinent

Méthode du coude et score de silhouette



Coude

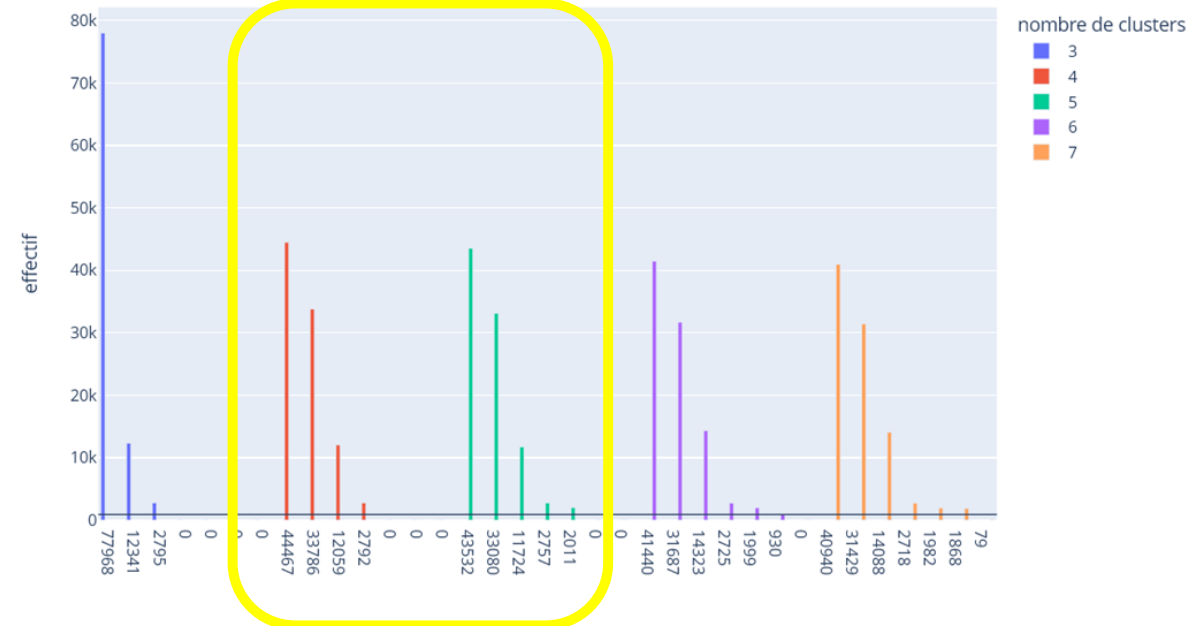
La distorsion est calculée comme la moyenne des distances au carré depuis les centres de cluster des clusters respectifs

Moyenne des scores de silhouette légèrement meilleure

Pour chaque point, son coefficient de silhouette est la différence entre la distance moyenne avec les points du même groupe que lui et la distance moyenne avec les points des autres groupes voisins.

Le choix du nombre de clusters dépend :

- des contraintes métiers : on veut pouvoir décrire les clusters
- de la stabilité des clusters sur plusieurs exécutions de l'apprentissage
- de l'équilibre entre les clusters en terme d'effectifs

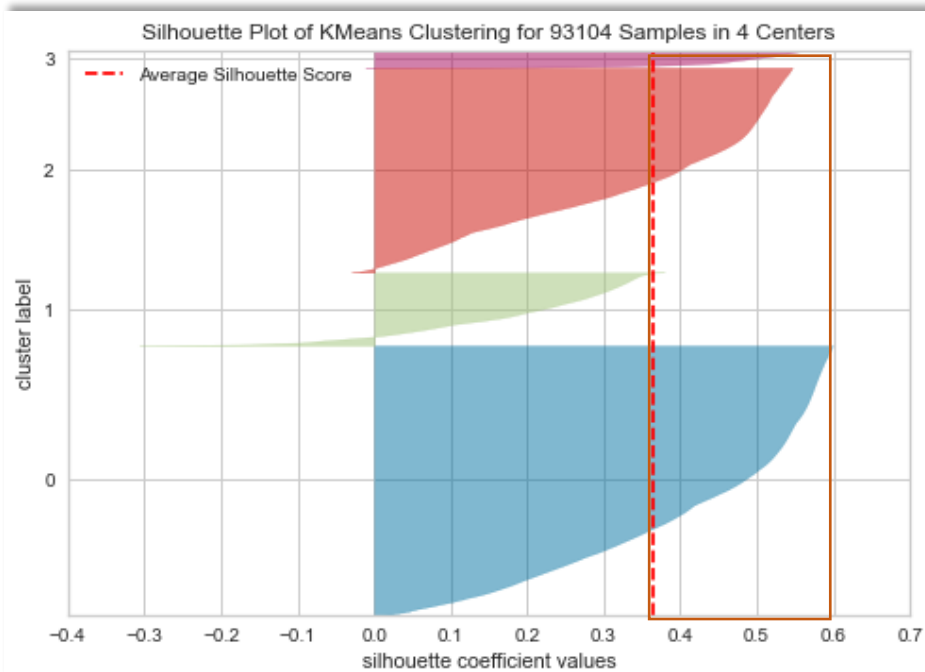


Choix du nombre de clusters

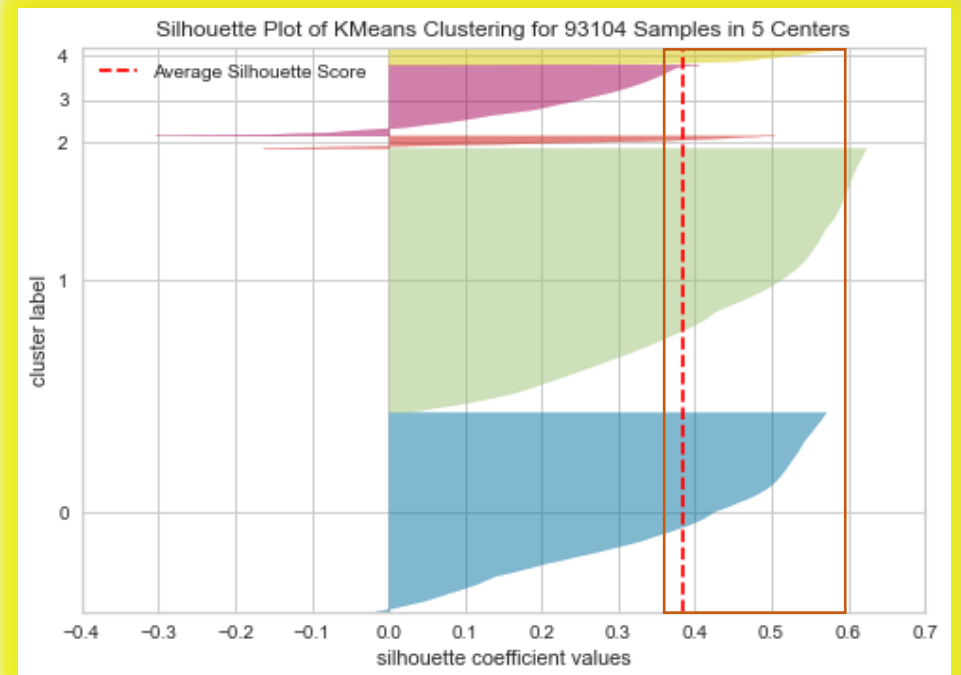
Algorithme K-Means

- Le choix se porte sur 5 clusters :
 - Moyenne des coefficients de silhouettes légèrement meilleurs
 - Meilleur équilibre entre les effectifs des clusters
 - Les tests effectués pour les caractériser ont montré que les descriptions étaient plus lisibles avec 5 clusters

4 clusters

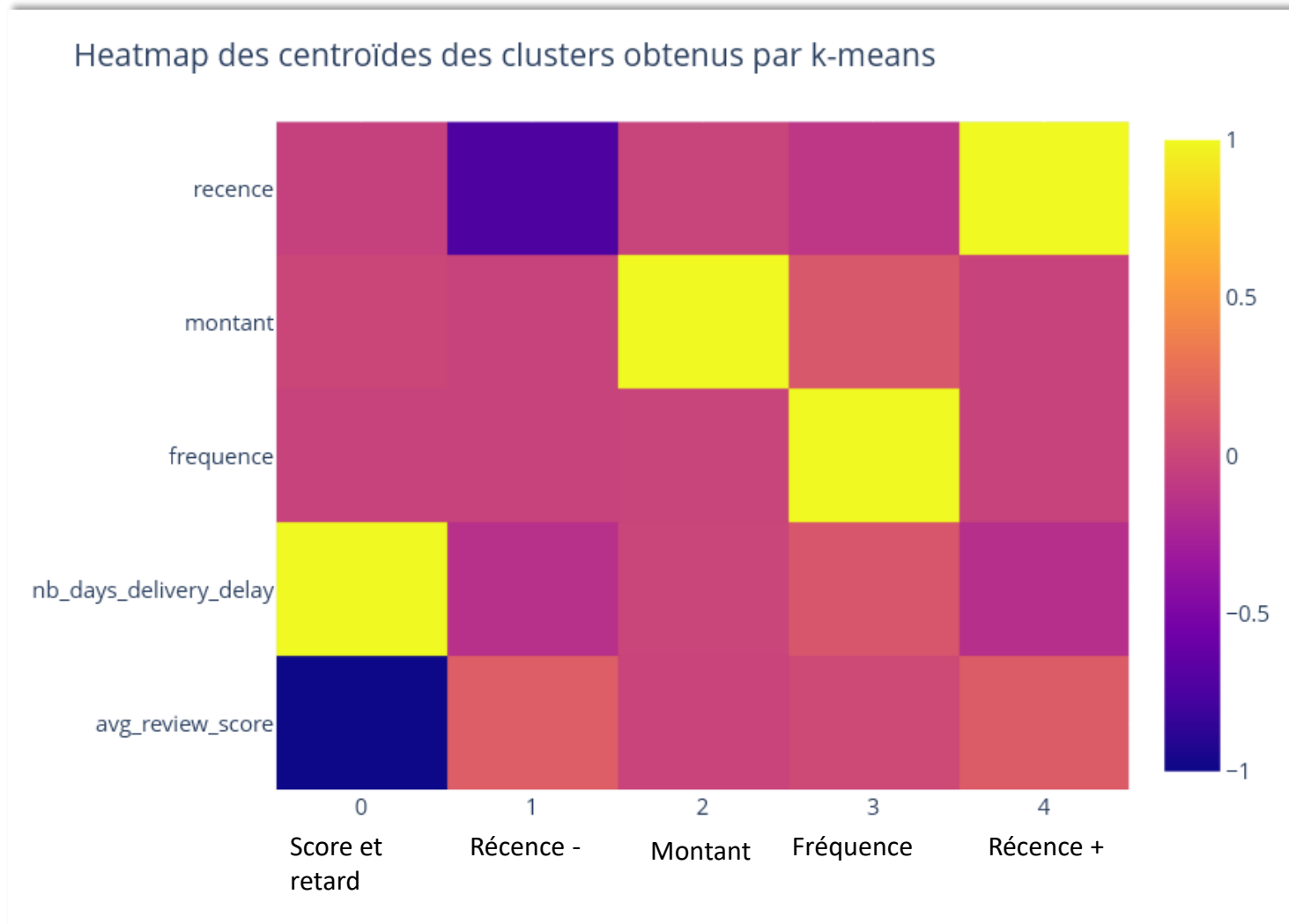


5 clusters



Centroïdes des clusters

Dominantes des clusters

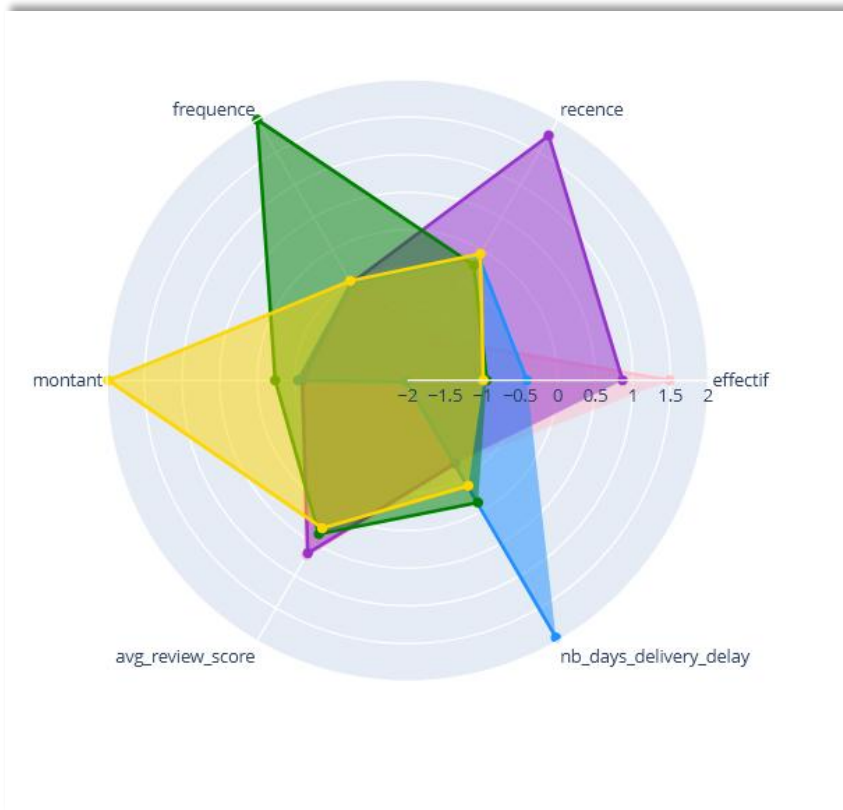


L'observation des coordonnées des centroïdes sur les différentes variables permet de déterminer les caractéristiques dominantes pour chaque cluster.

Retour au K-means

Visualisation des moyennes des clusters et répartition des effectifs

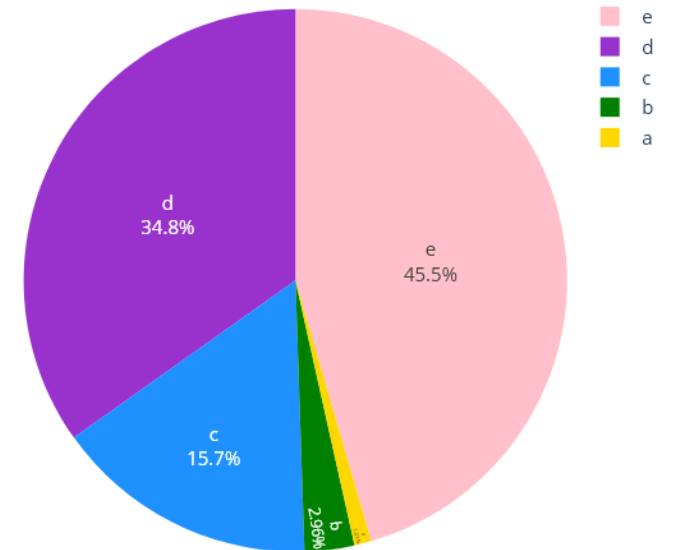
Les lettres sont attribuées aux clusters par effectif croissant de a à e.



L'observation sous forme de radar des moyennes des différentes variables pour chaque cluster correspond bien à l'observation faite sur les centroïdes précédemment.

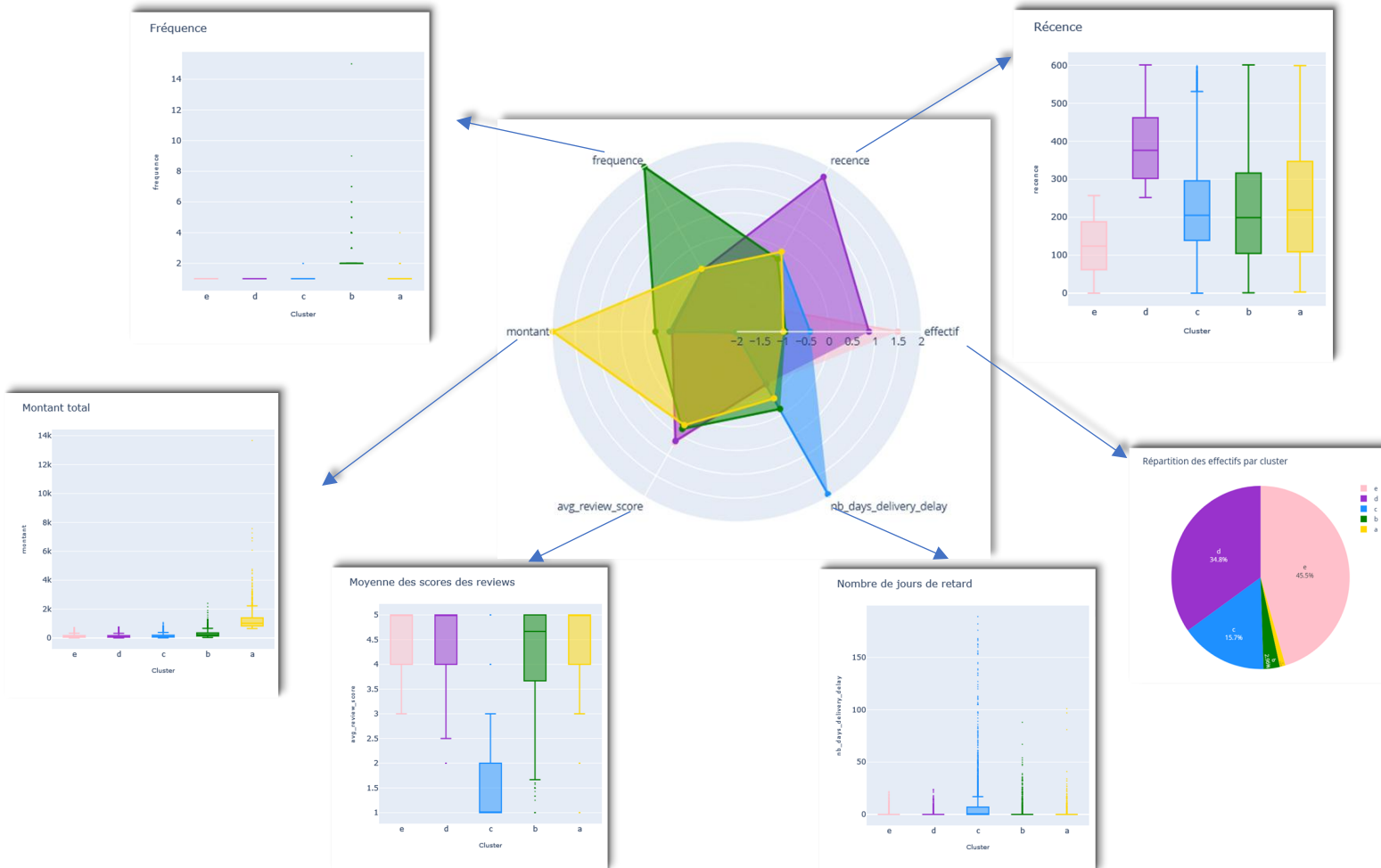
Les clusters les plus gros se distinguent essentiellement par la récence (clients lointains/clients plus récents)

Répartition des effectifs par cluster



Description des clusters

Résumé des caractéristiques



a : Montant total dépensé le plus important

b : Fréquence d'achat supérieure ou égale à deux

c : **Insatisfaction et grand nombre de jours de retard**



d : Clients ayant commandé le plus récemment

e : Clients dont les commandes remontent le plus loin

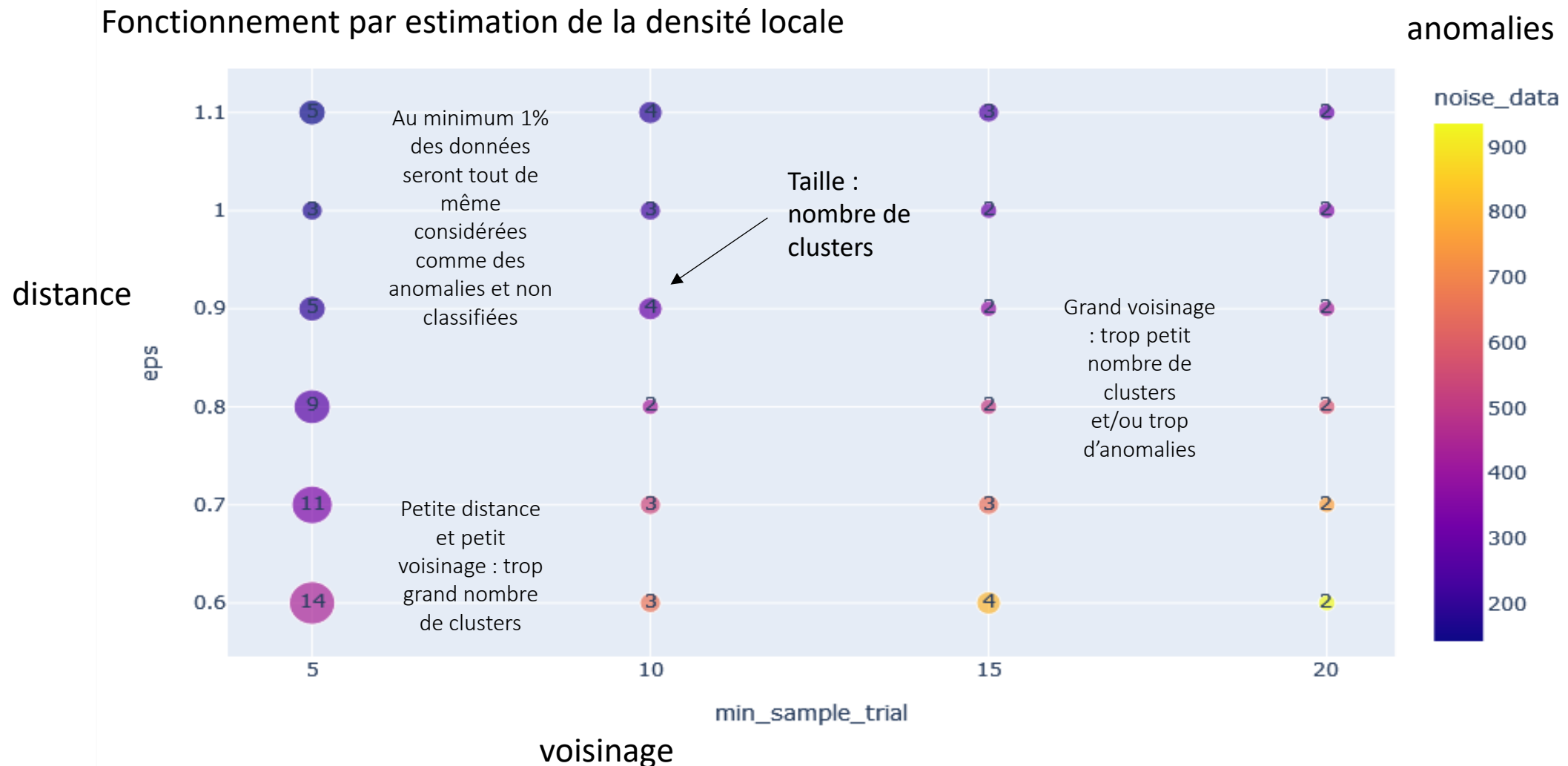
Un mot sur les algorithmes non retenus...

Et les explications

Choix écarté : DBScan

Longue durée d'exécution et non-adéquation à la résolution du problème (anomalies)

Les tests ont donc été effectués sur un sample de 20% des données uniquement.



Choix écarté : Agglomerative Clustering

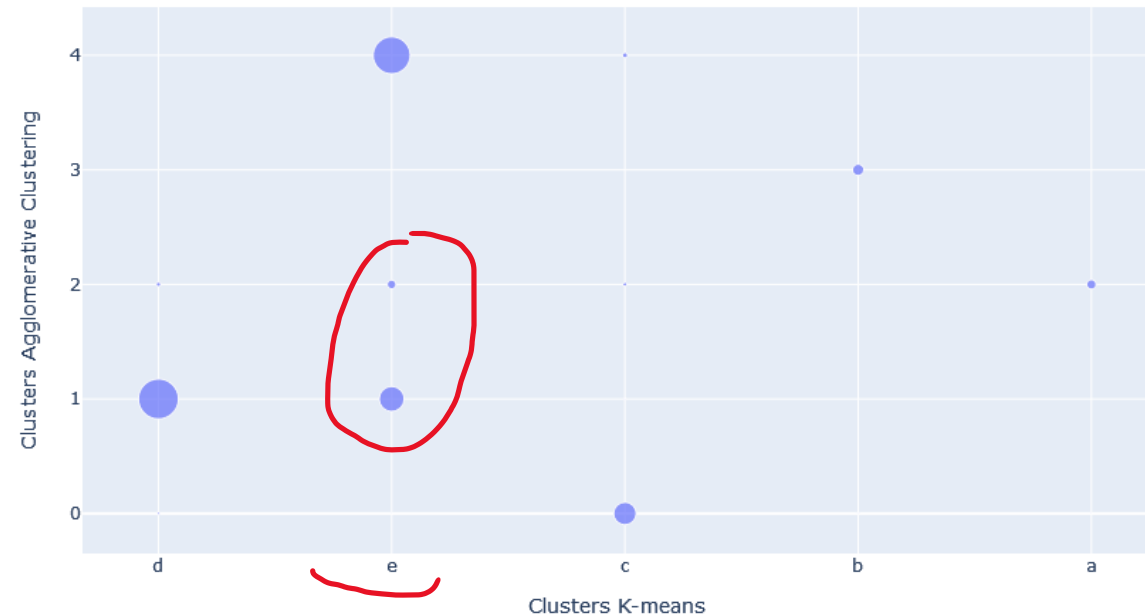
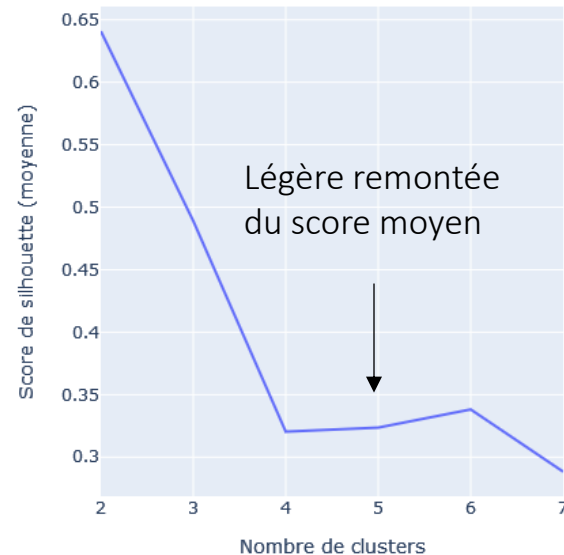
Longue durée d'exécution et gros usage mémoire

MemoryError: Unable to allocate 32.3 GiB for an array with shape (4334130856,) and data type float64

Les tests ont donc été effectués sur un sample de 20% des données uniquement.

Le choix s'étant porté sur 5 clusters, permettant de comparer avec la classification obtenue par K-means sur le dataset complet. Excepté pour un cluster, on observe une classification similaire.

Le cluster k-Means **e** est celui dont les clients dont les commandes remontent le plus loin. Sur un sample des données, il n'est pas surprenant que cette classification diffère. Toutefois on peut supposer que sur le dataset complet, on obtiendrait un clustering similaire.





Délai de maintenance

Etude par simulation et
proposition

Objectif de la simulation

Et méthode employée

Afin de pouvoir proposer un délai de maintenance adéquat, on va **estimer le temps au bout duquel un modèle perd de sa pertinence** pour la classification des données.

Pour ce faire, nous allons **classifier l'ensemble complet des données à disposition en utilisant des modèles entraînés sur des ensembles de données « antérieurs »** (donc plus petits), c'est-à-dire dont la dernière date est antérieure à celle de l'ensemble complet.

Il s'agira ensuite de **comparer les classifications obtenues** pour l'ensemble des données

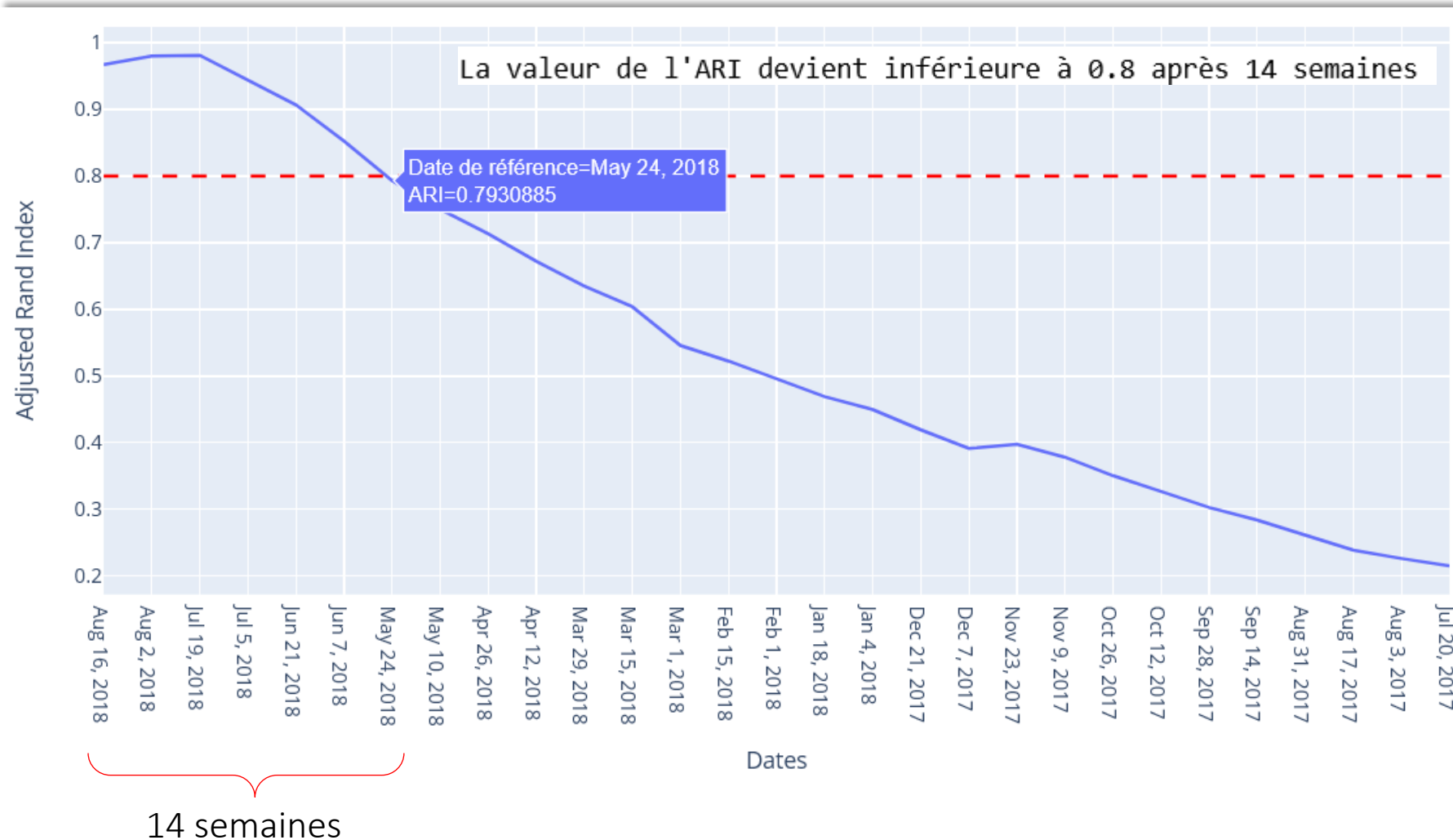
- avec le modèle entraîné sur l'ensemble complet
- avec les modèles plus anciens.

Cette comparaison peut être effectuée en calculant **l'indice de Rand ajusté** et en observant **son évolution** à mesure qu'on remonte dans le temps.

* L'indice de Rand ajusté permet de mesurer la similarité entre deux classifications, y compris lorsqu'elles ont un nombre de classes différent.

Simulation

Courbe des valeurs d'indice de Rand ajusté (Adjusted Rand Index, ARI)



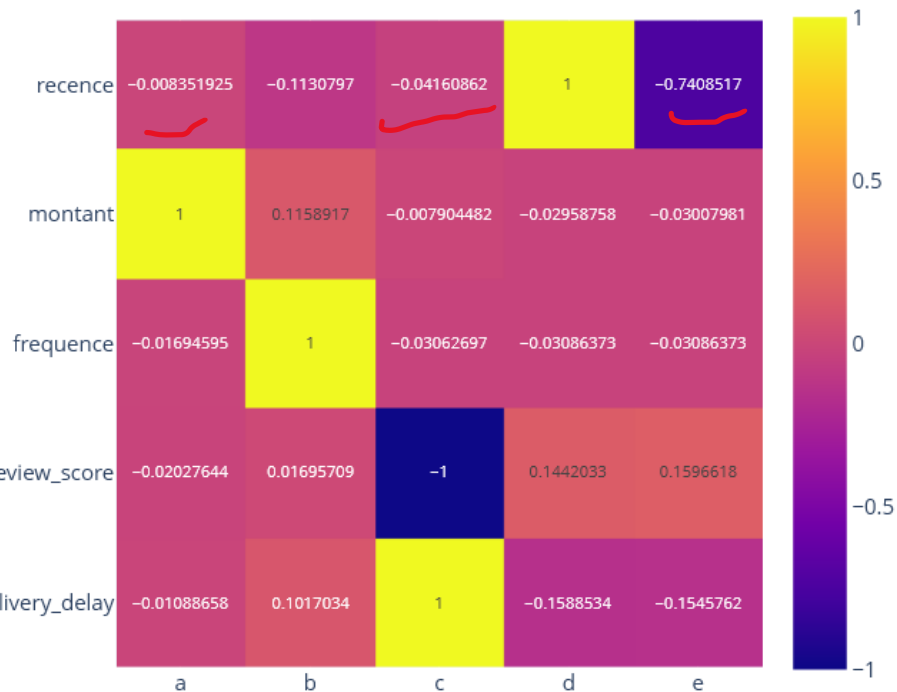
On considère en général qu'il est préférable de réentraîner le modèle si la valeur de l'ARI passe en dessous de 0.8

Le délai de maintenance proposé est donc de **14 semaines** (98 jours)

Variation des modèles

Coordonnées des centroïdes des clusters

Heatmap des centroïdes des clusters - modèle le plus récent



La segmentation est similaire dans ses dominantes, mais on peut observer que les coordonnées des centroïdes varient de façon notable sur la recence

Heatmap des centroïdes des clusters : 2018-05-24



Différences de classification

Explications

Clustering obtenu avec le modèle le plus récent

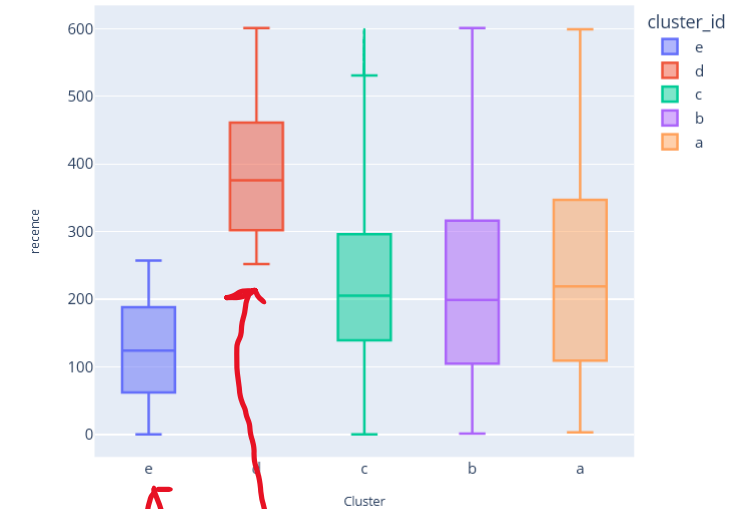
2018-05-24	a	b	c	d	e
Last model					
a	2041	0	0	12	0
b	0	2757	0	0	0
c	28	1	11081	605	1
d	8	0	0	32974	0
e	71	0	13	5389	38123

Tableau croisé : avec le clustering obtenu avec le modèle entraîné 14 semaines auparavant.

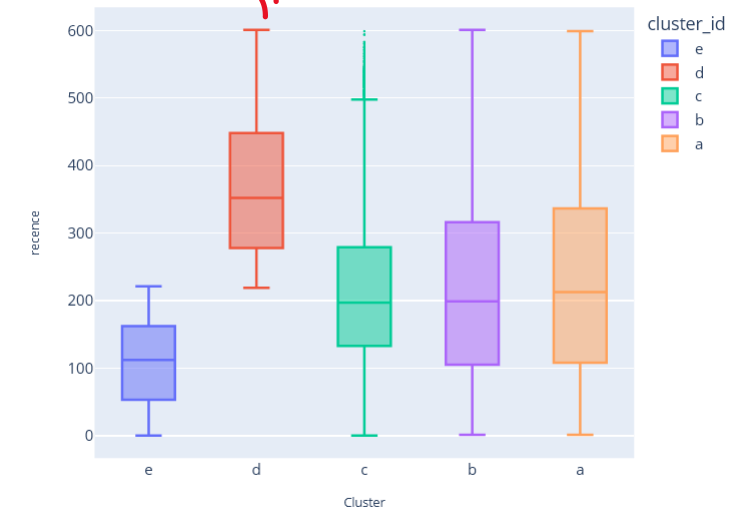
Toutes variables égales par ailleurs, un point classé dans le cluster 'd' (récence importante) avec le modèle précédent peut être maintenant classé différemment ('e' ou 'c')

Comme la grande majorité des clients ne sont apparus qu'une seule fois, la variable récence prend des valeurs qui augmentent de...98 jours entre deux entraînements, ceci affectant bien sûr le clustering (exception faite des nouveaux clients)

Récence - dernier modèle



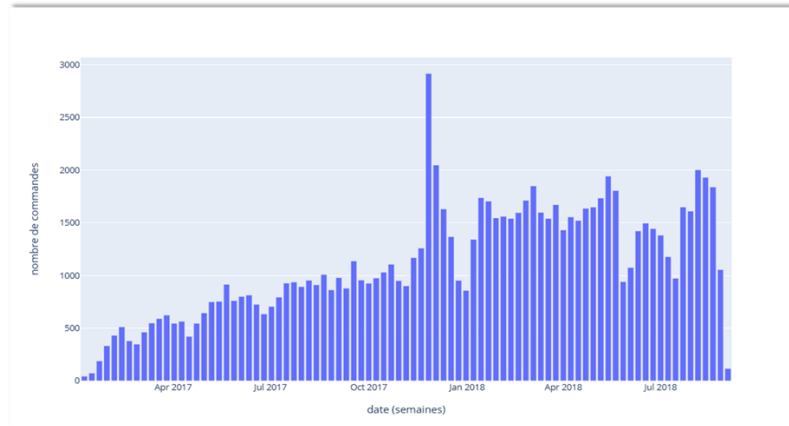
Récence - modèle précédent



Conclusion

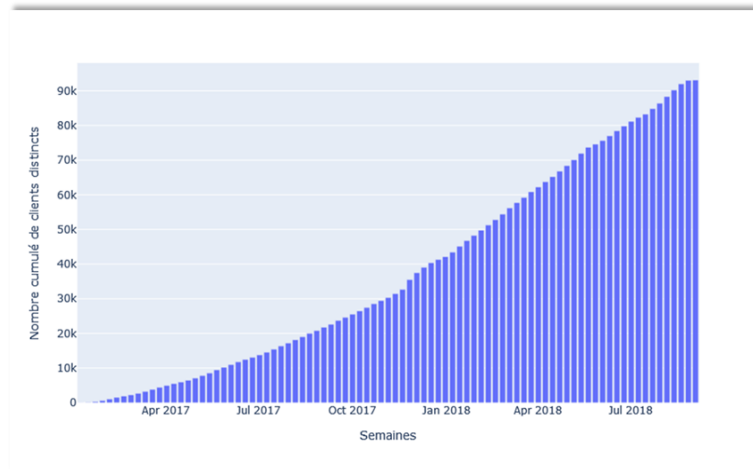
Et recommandations

Si la fréquence d'achat est faible, la récence aura un impact certain sur le clustering, d'où la nécessité de réentraîner le modèle régulièrement



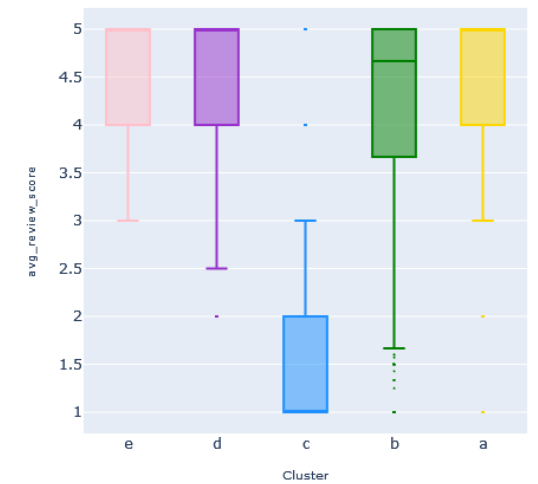
Pour les autres critères, il convient de

- suivre l'évolution du nombre de commandes au cours du temps
- suivre l'évolution du nombre de clients distincts

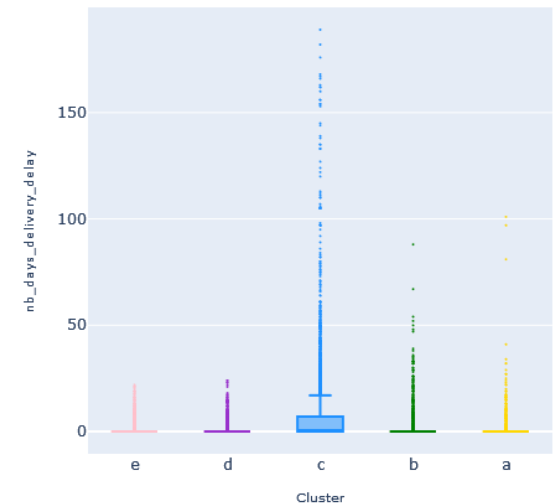


D'un point de vue marketing, l'analyse des motifs potentiels d'insatisfaction, en particulier celle probablement liée au retard de livraison peut s'avérer très utile.

Moyenne des scores des reviews



Nombre de jours de retard



Merci pour votre
attention

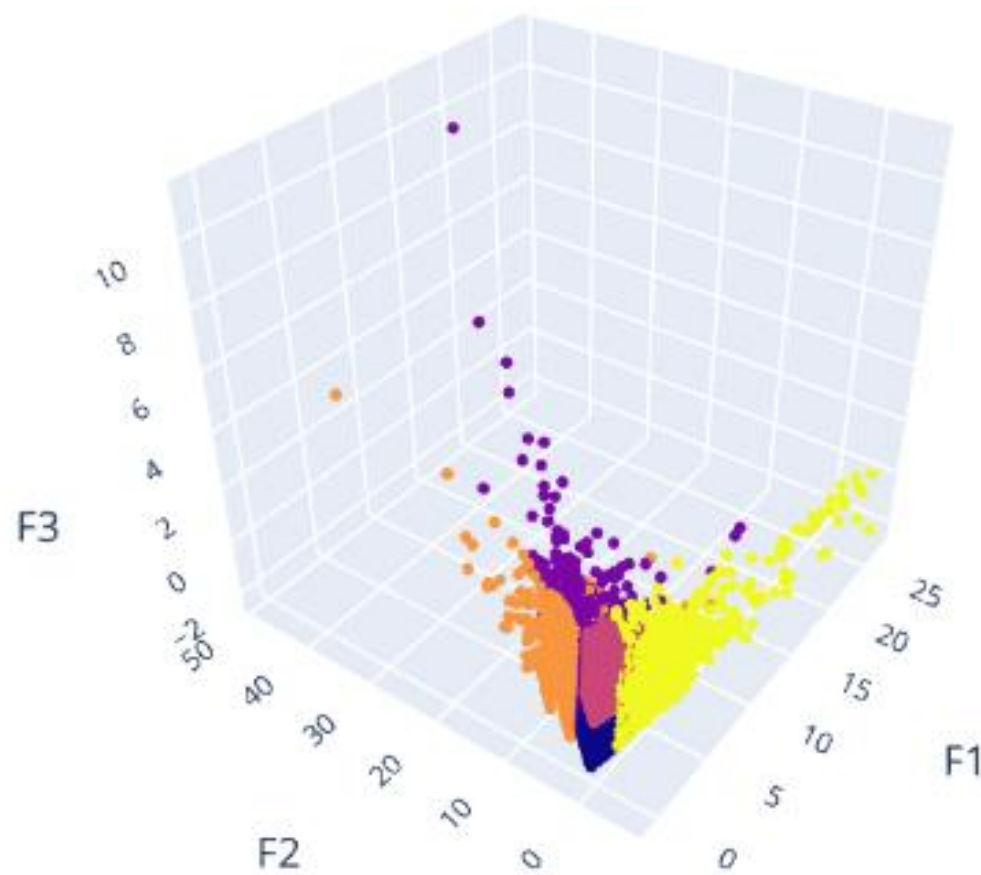


BACK-UP SLIDES

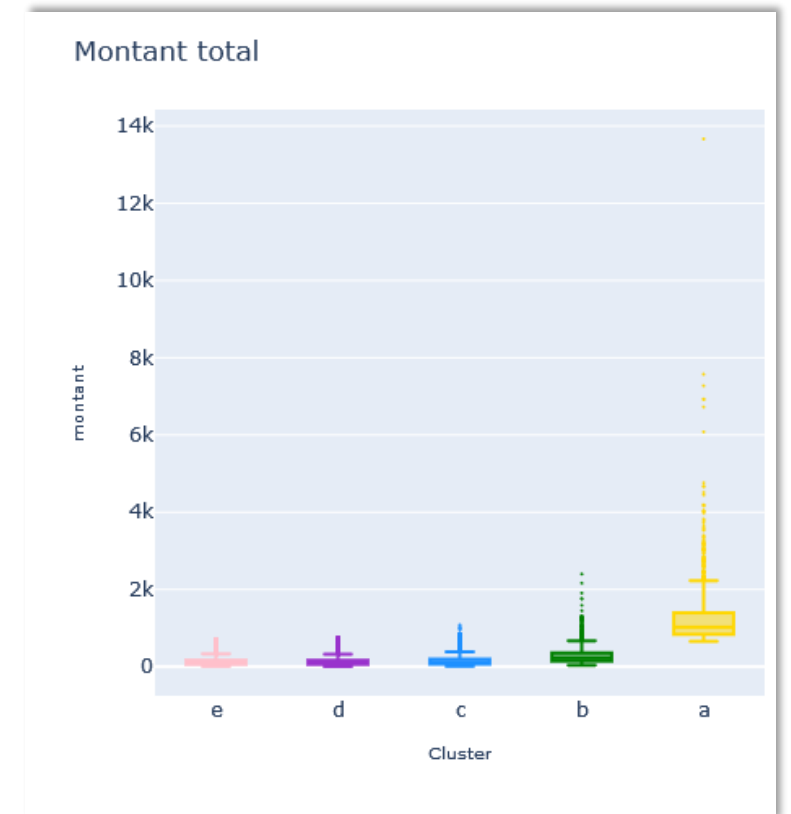
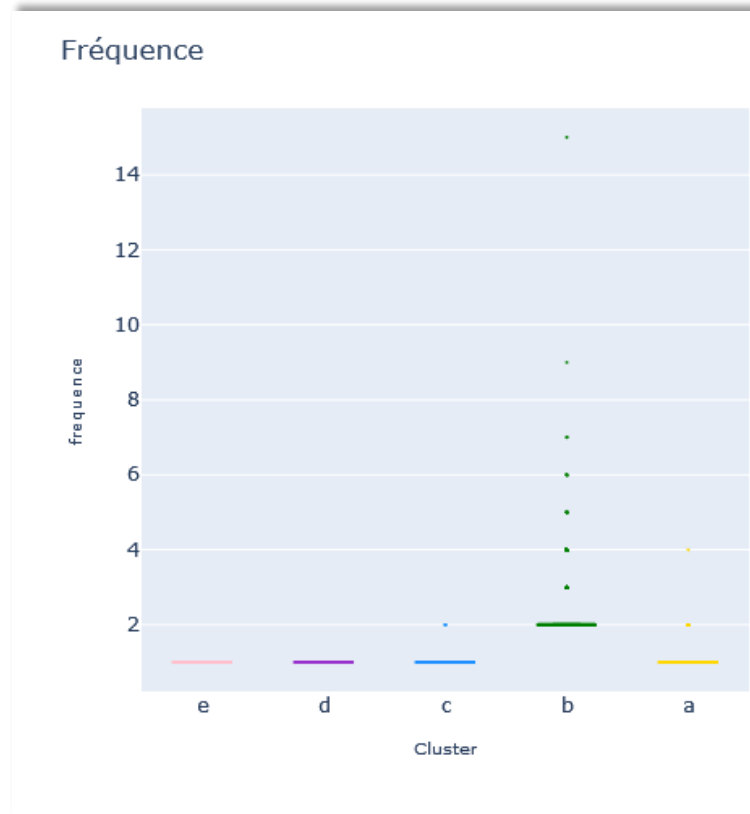
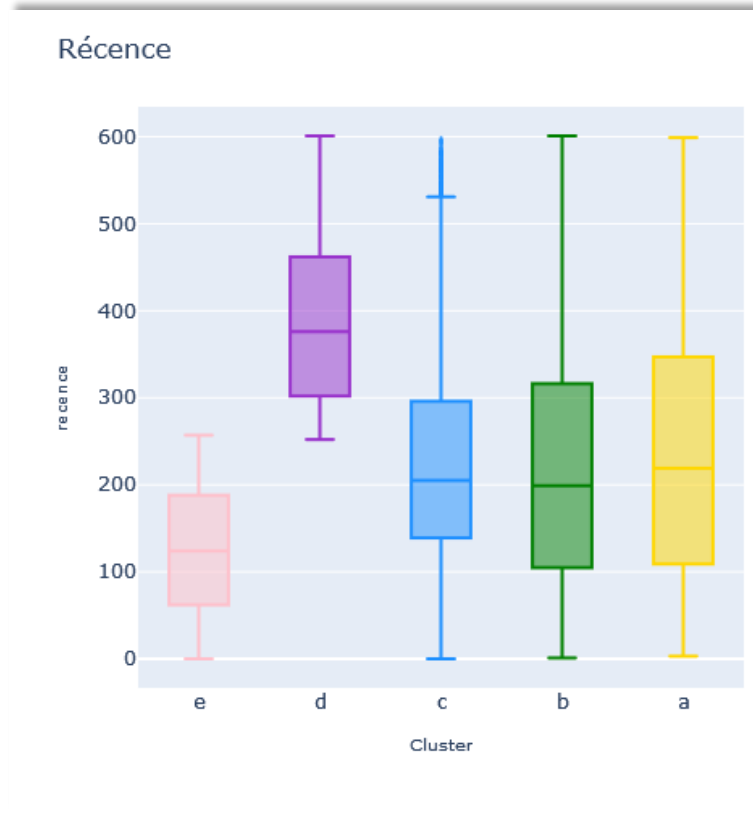


Visualisation des clusters

Projection sur les 3 composantes principales



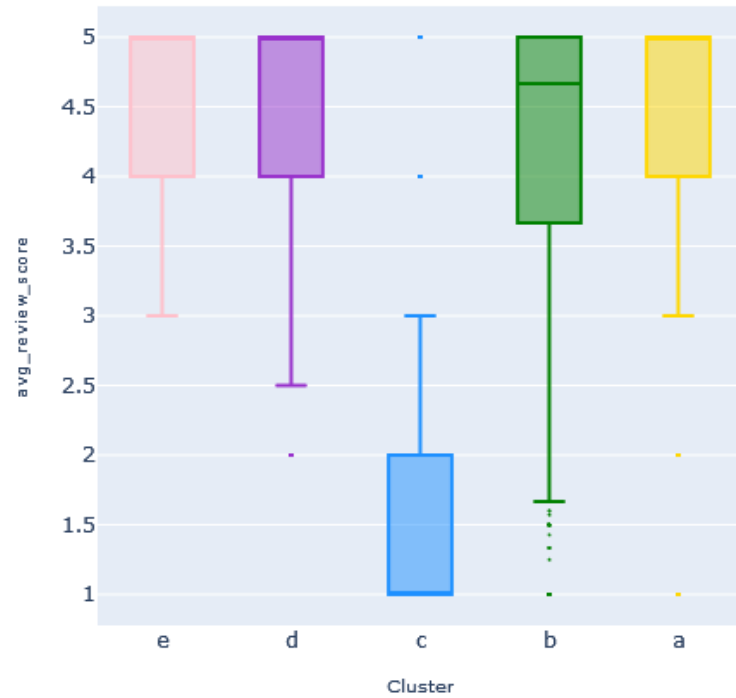
Récence, Fréquence et Montant



Satisfaction et retard de livraison

Les causes potentielles d'insatisfaction

Moyenne des scores des reviews



Nombre de jours de retard

