5-Segmentez des clients d'un site e-commerce

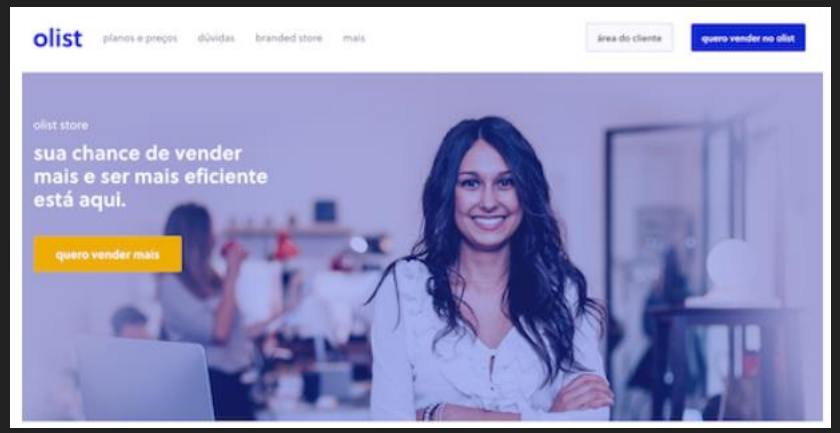
EDA – clustering model testing – simulation over time

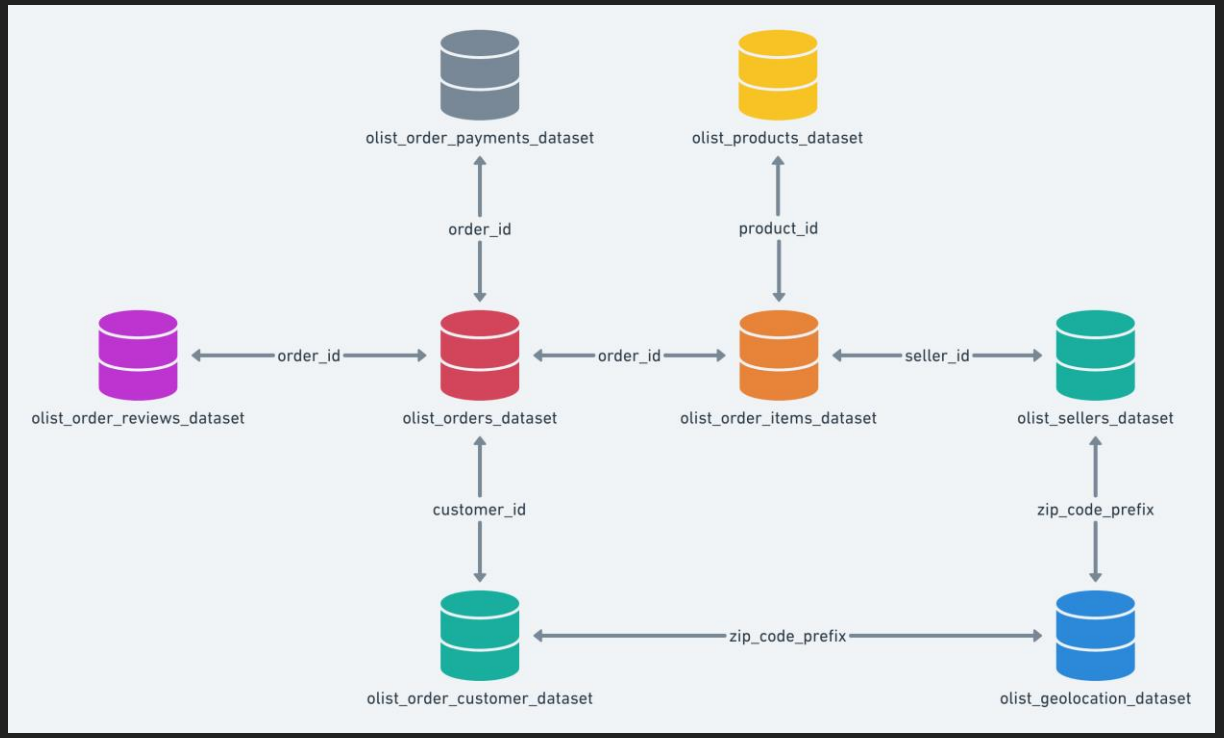
Summary

- Part 1 : Preliminary
- Part 2 : Cleaning and data exploration
- Part 3 : Feature engineering
- Part 4 : Clustering model testing
- Part 5 : Stability of the segmentation over time
- Part 6 : Conclusion

1-Preliminary

- Your mission is to help Olist teams understand the different types of users. You will therefore use unsupervised methods to group customers with similar profiles. These categories can be used by the Marketing team to better communicate.



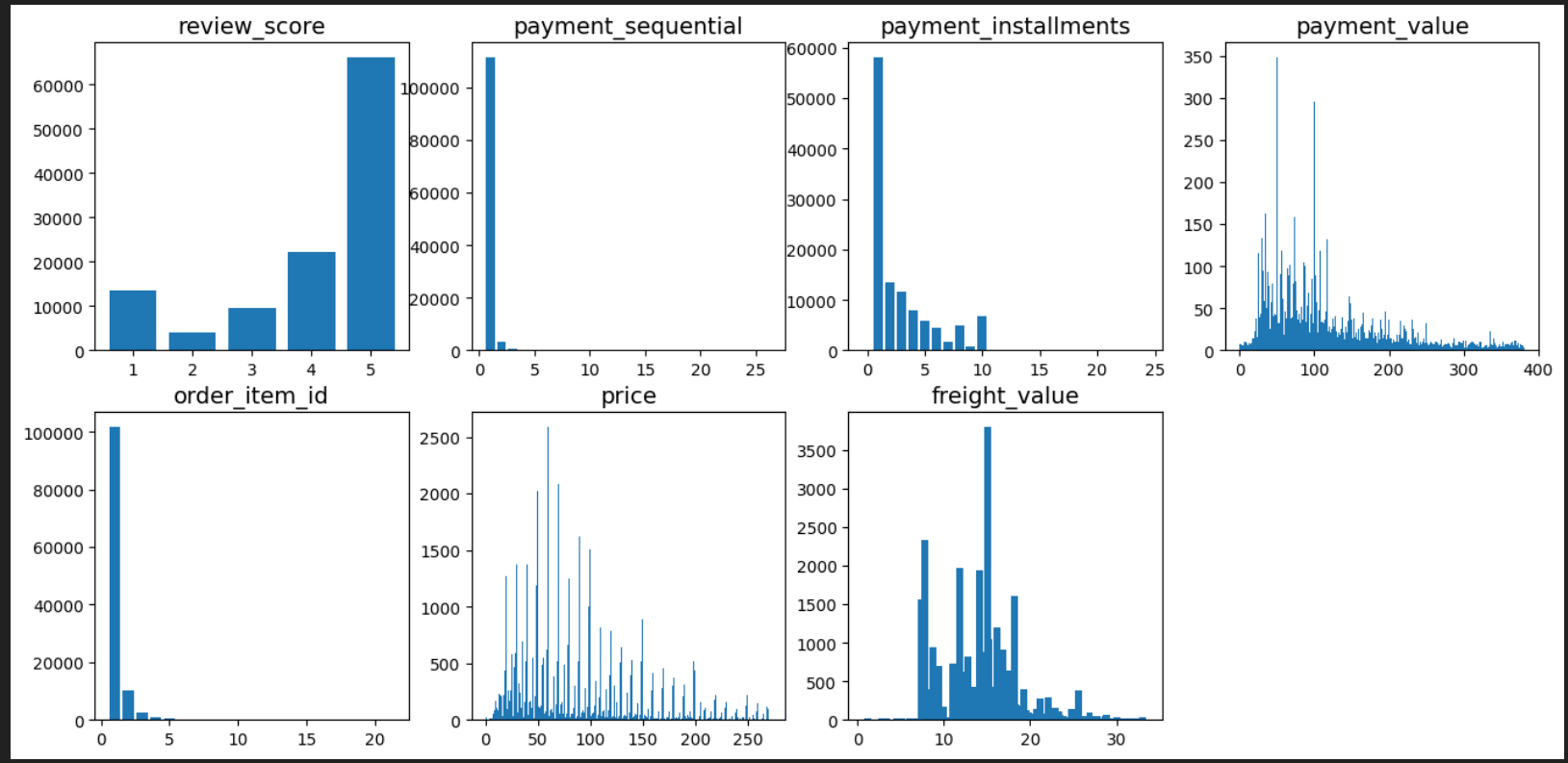


Part 2 : Cleaning and data exploration

- **Cleaning :**
  - Reducing the data to finalized orders only (status : delivered or canceled)
  - Removal of the outliers with the interquartil method

Part 2 : Cleaning and data exploration

- **Data exploration :**
  - Univariate analysis
  - There are only seven numerical fatures
  - The data are not normal



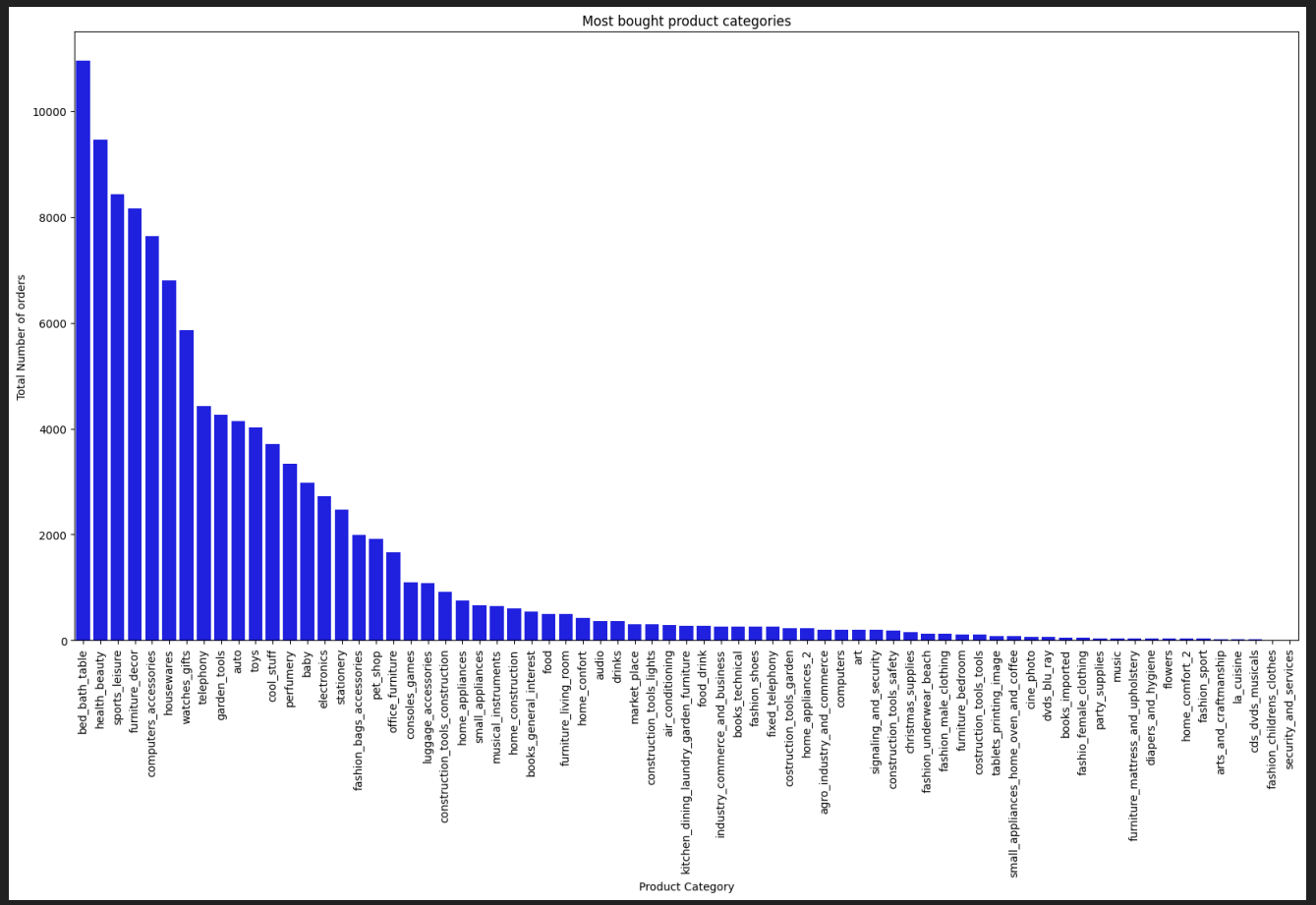
Part 2 : Cleaning and data exploration

- **Data exploration :**
  - Customer localization

  - Seller localization

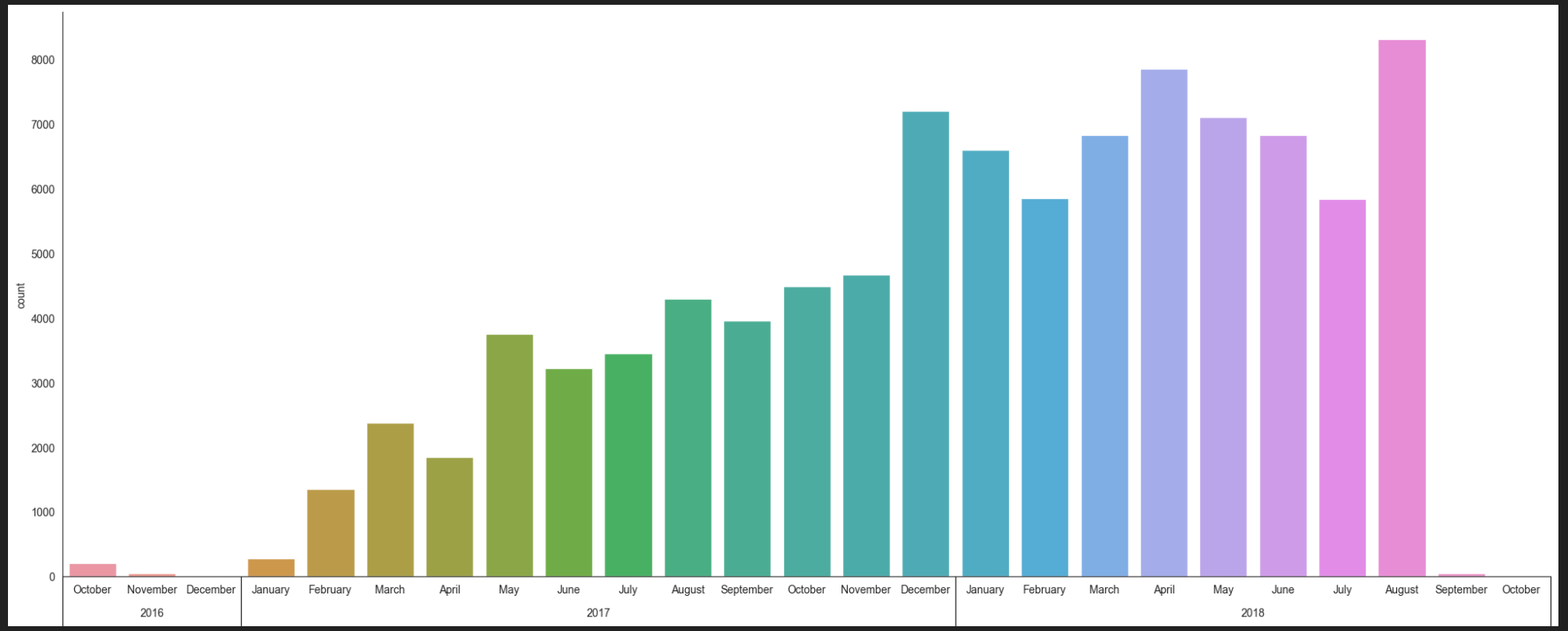
Part 2 : Cleaning and data exploration

- **Data exploration :**
  - Repartition of product category
  - We will group the 71 categories into lesser categories in part-3 for better readibility



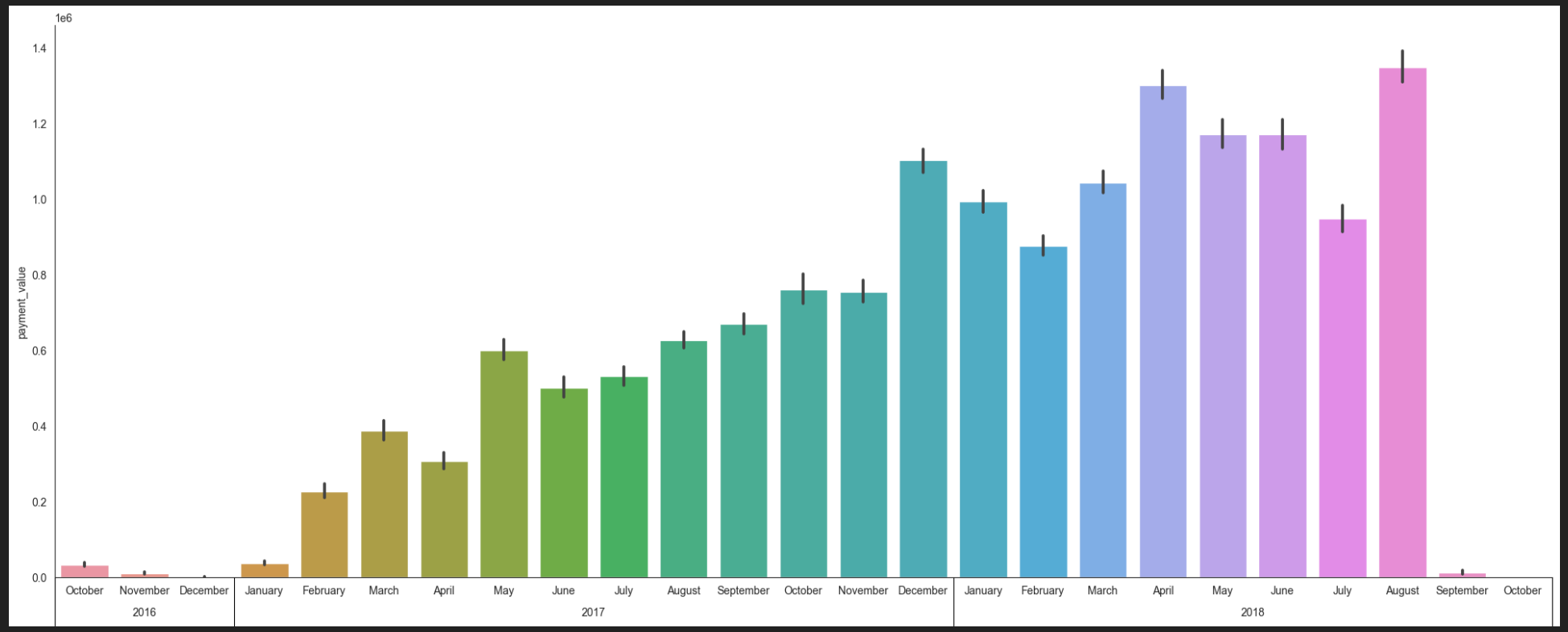
Part 2 : Cleaning and data exploration

- **Data exploration :**
  - Quantity of orders over time



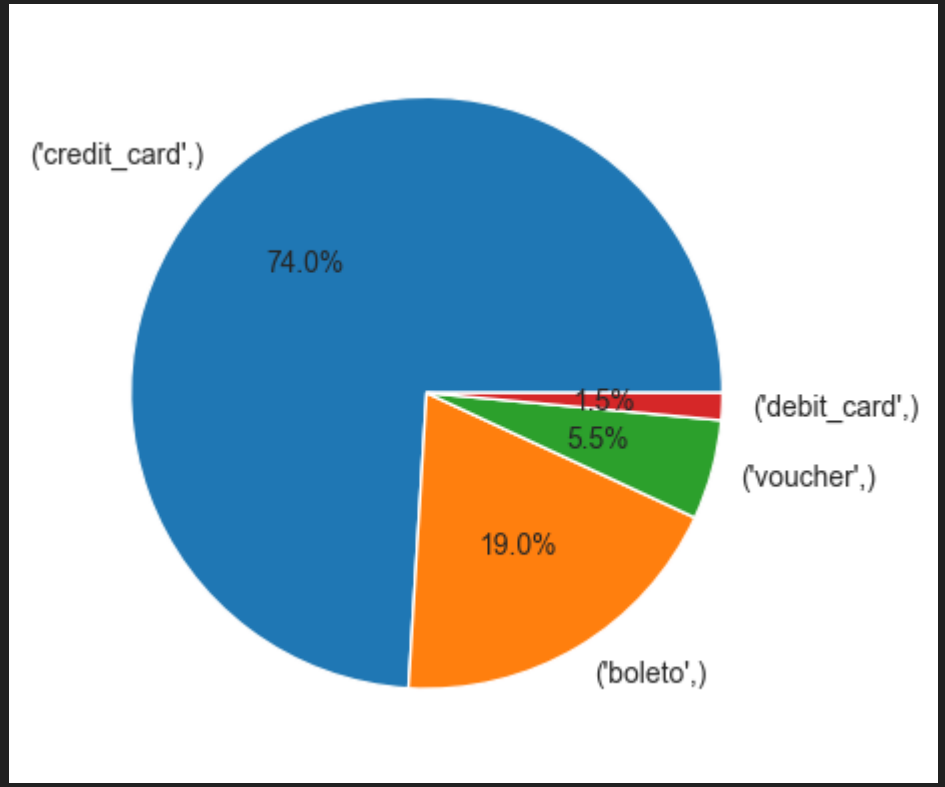
Part 2 : Cleaning and data exploration

- **Data exploration :**
  - Selling over time (Brazilian Real)



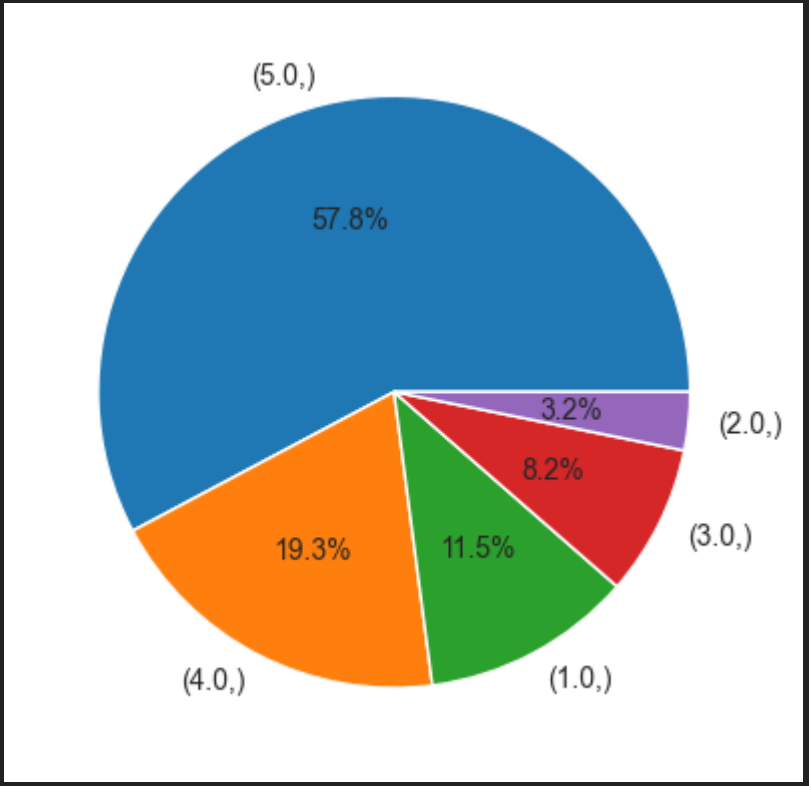
Part 2 : Cleaning and data exploration

- **Data exploration :**
  - Repartition of payment supports



Part 2 : Cleaning and data exploration

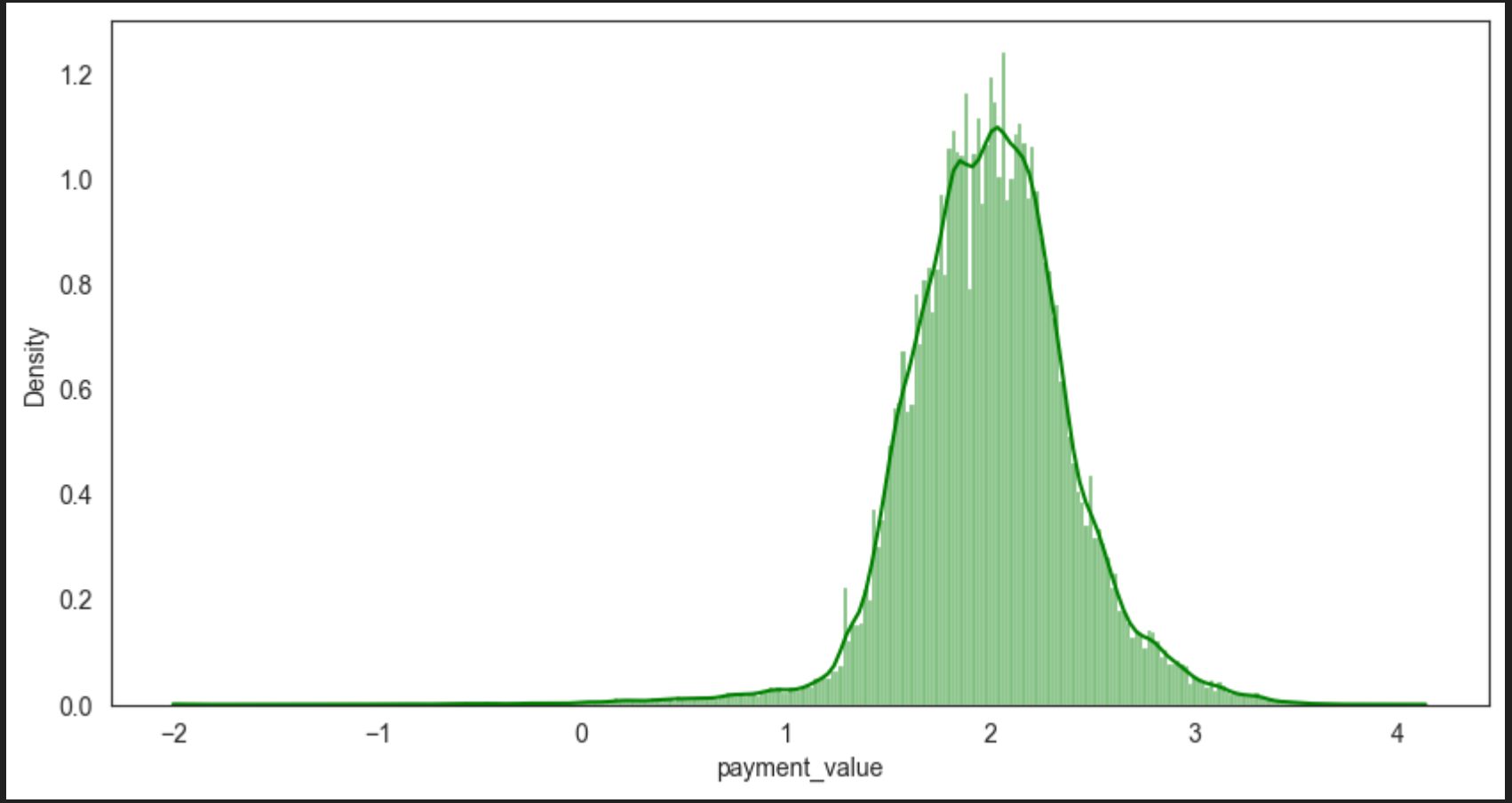
- **Data exploration :**
  - Satisfaction review



Part 2 : Cleaning and data exploration

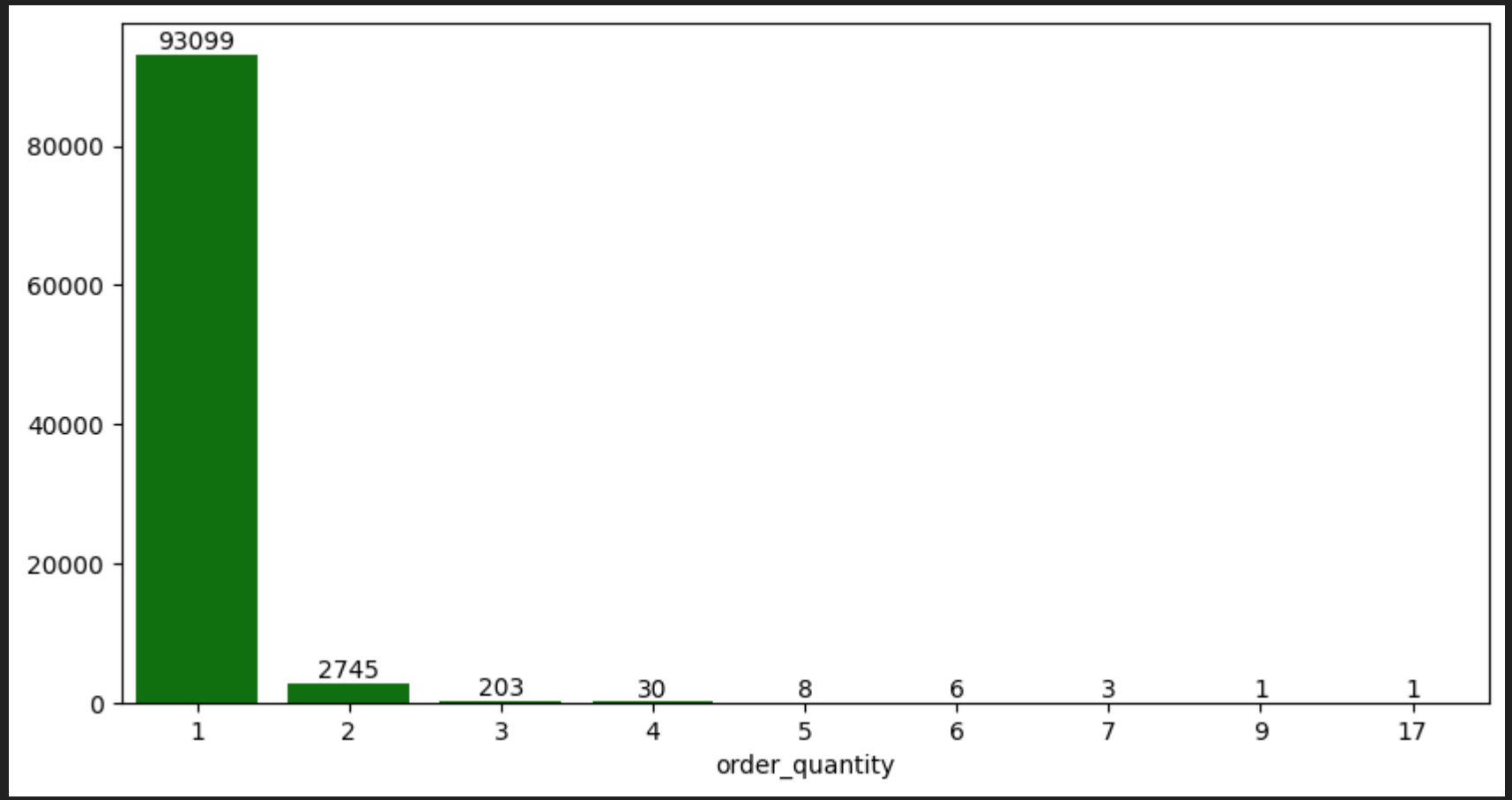
- **Data exploration :**
  - Distribution of payment value (log10 scale - Brazilian Real)

count	103877.000000
mean	154.113732
std	217.498755
min	0.010000
25%	56.820000
50%	100.000000
75%	171.840000
max	13664.080000



Part 2 : Cleaning and data exploration

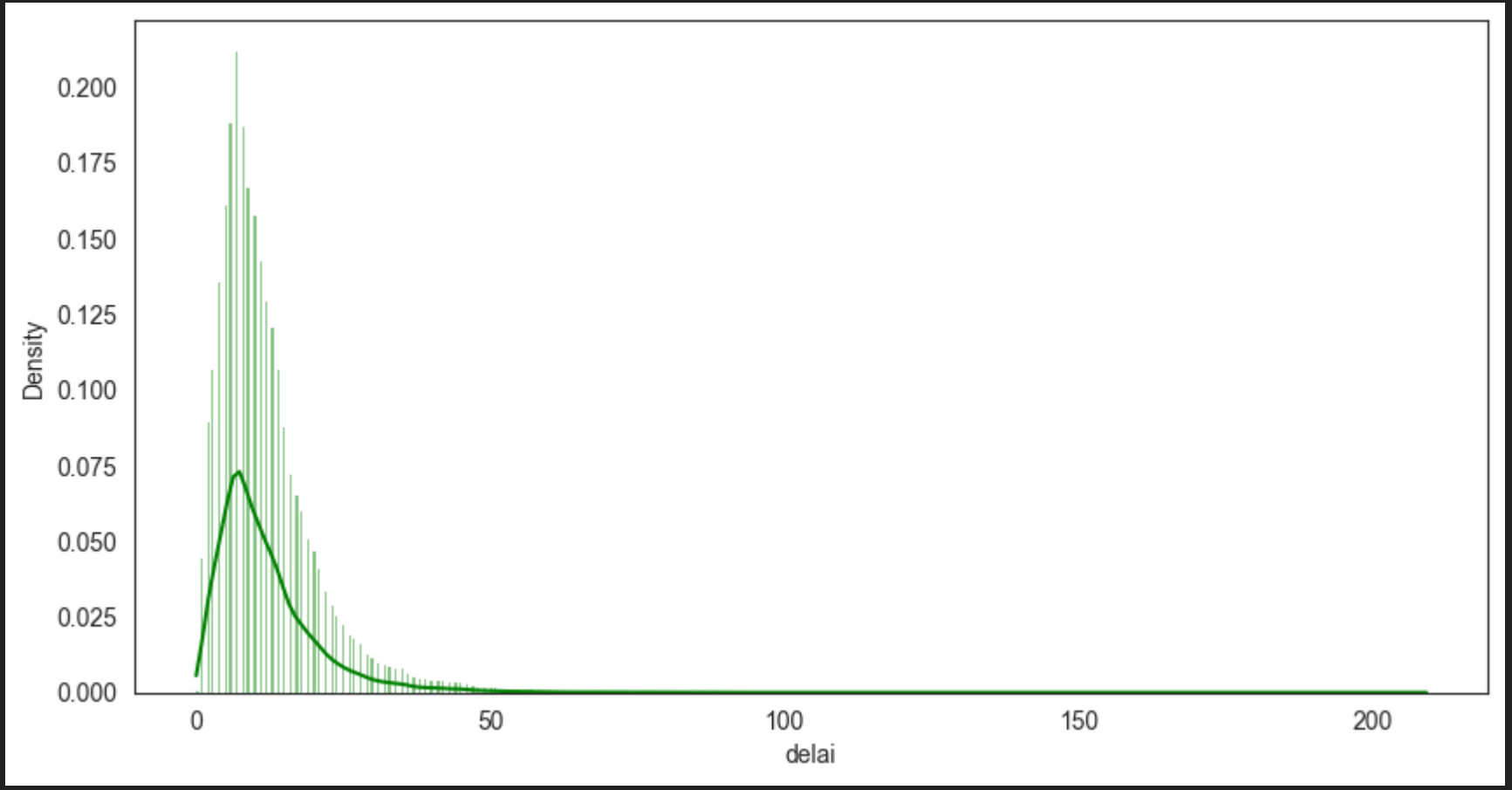
- **Data exploration :**
  - number of orders per customer
  - Only 3% of customers make mora than one order



Part 2 : Cleaning and data exploration

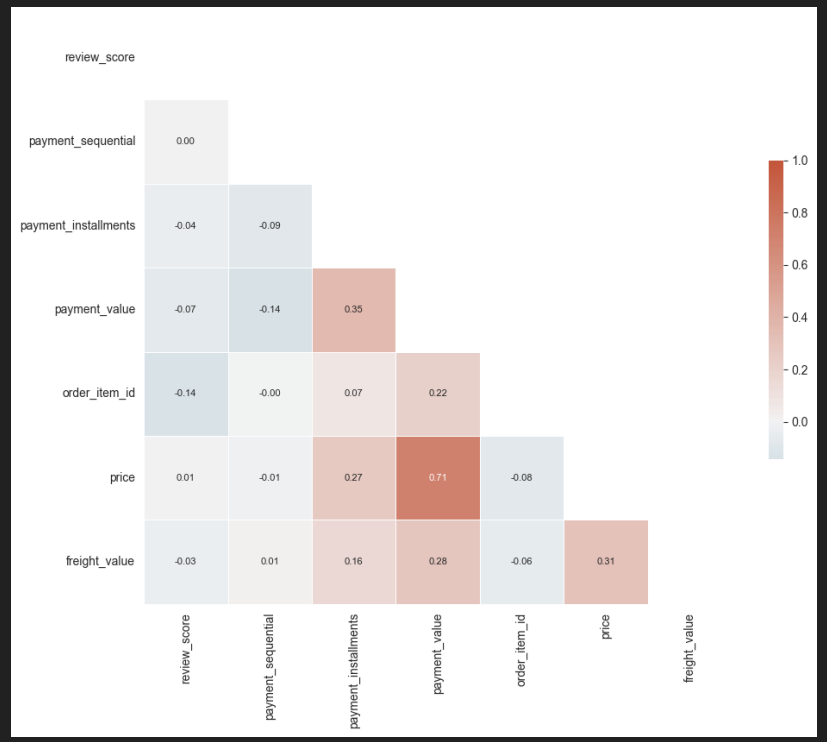
- **Data exploration :**
  - Delivery time (days)

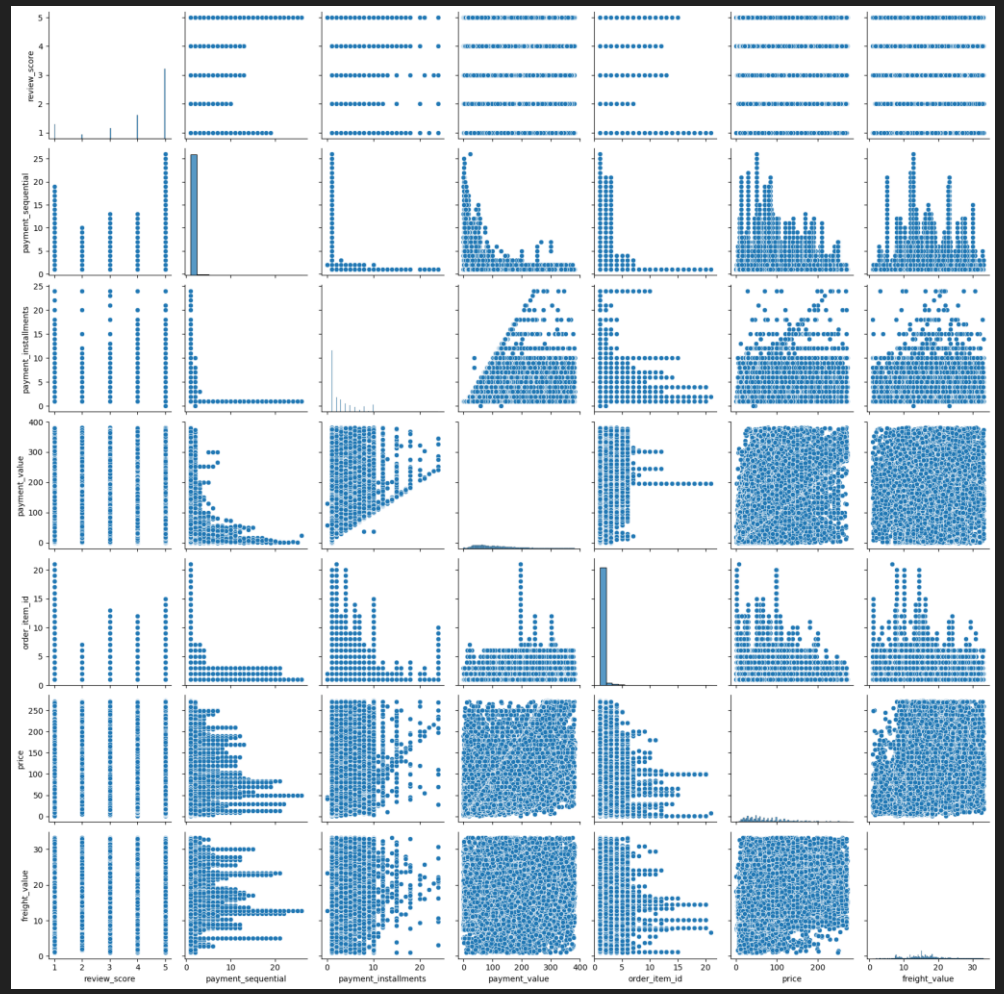
count	110189.0
mean	12.007342
std	9.451153
min	0.0
25%	6.0
50%	10.0
75%	15.0
max	209.0



Part 2 : Cleaning and data exploration

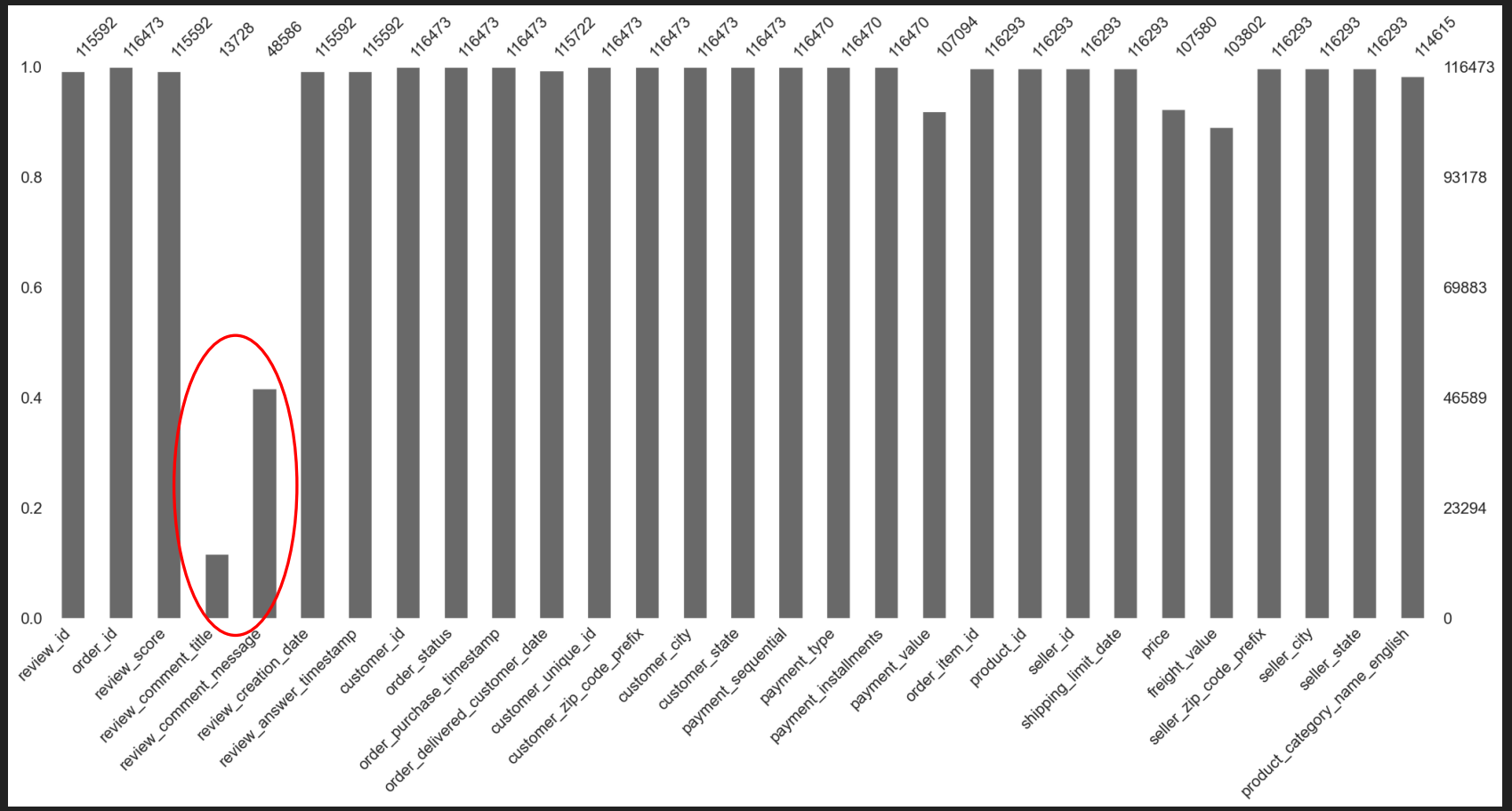
- **Data exploration :**
  - Bivariate analysis





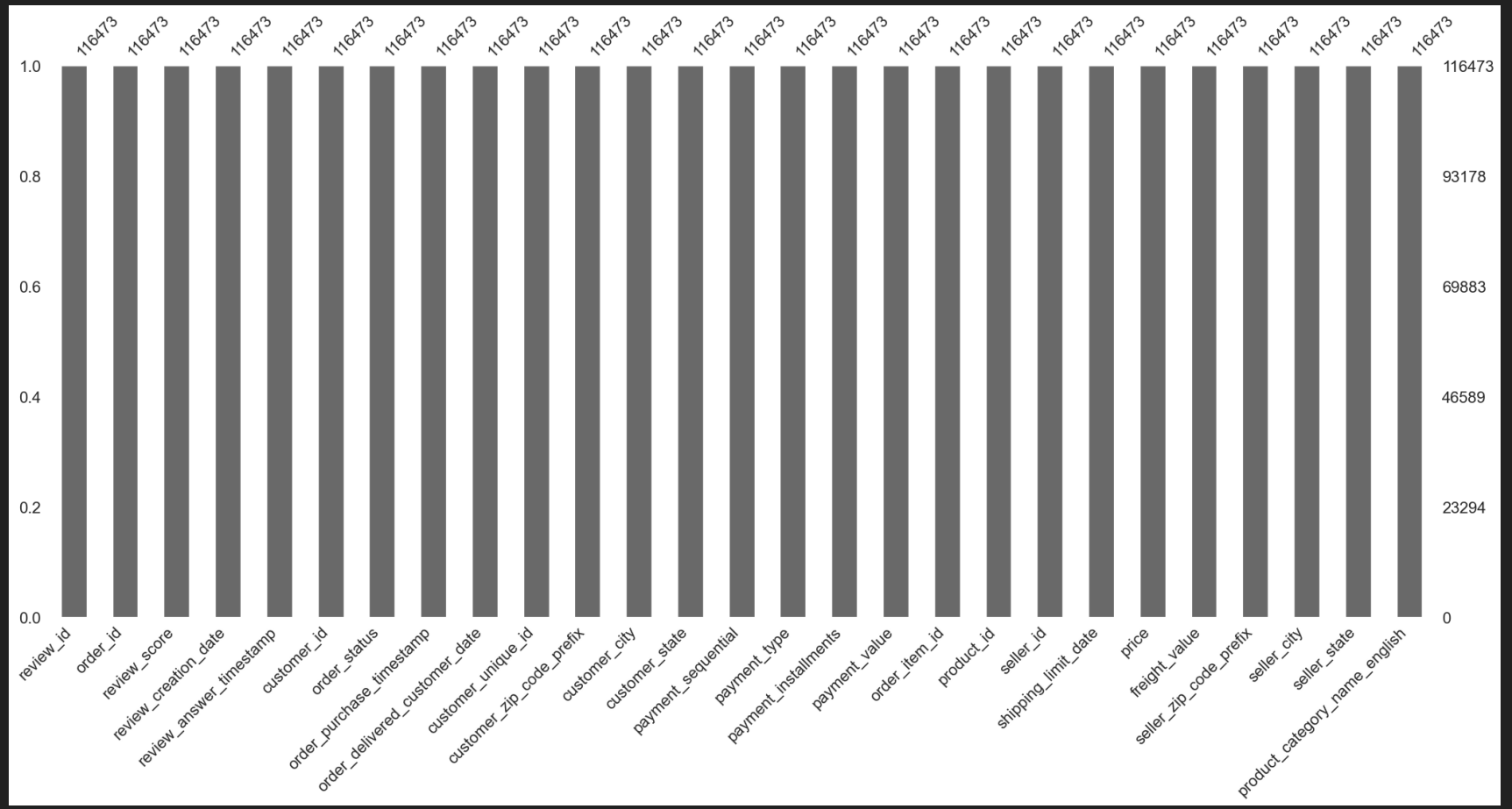
Part 3 : Feature engineering

- **Discarding columns:**
  - 2 review_comment columns are discarded.
  - We keep only the review score columns



Part 3 : Feature engineering

- **Filling NaN:**
  - The NaN are replaced by the mode



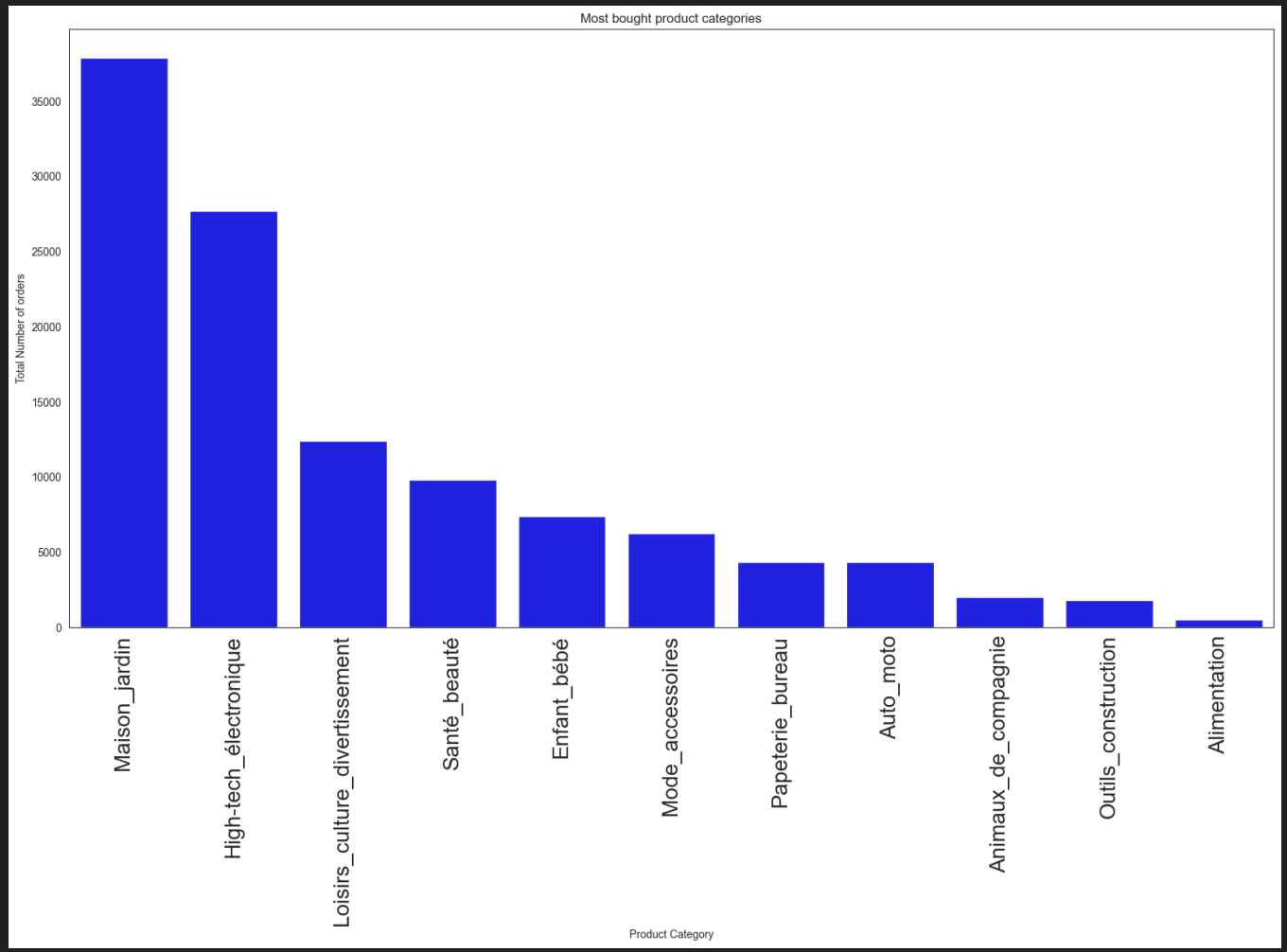
Part 3 : Feature engineering

- **New features**
  - The 71 product categories are merged into 11 categories

- **New features**
  - **'Maison_jardin'** : ['home_confort', 'furniture_bedroom', 'furniture_living_room','home_appliances_2','housewares', 'furniture_decor', 'garden_tools', 'bed_bath_table', 'air_conditioning', 'small_appliances', 'luggage_accessories', 'home_appliances', 'kitchen_dining_laundry_garden_furniture', 'home_construction', 'home_comfort_2', 'small_appliances_home_oven_and_coffee', 'furniture_mattress_and_upholstery', 'signaling_and_security', 'flowers'],
  - **'Mode_accessoires'** : ['perfumery', 'fashion_underwear_beach', 'fashion_bags_accessories', 'fashion_shoes', 'fashion_sport', 'fashio_female_clothing', 'fashion_male_clothing', 'fashion_childrens_clothes'],
  - **'Auto_moto'** : ['auto'],
  - **'Animaux_de_compagnie'** : ['pet_shop'],
  - **'Papeterie_bureau'** : ['stationery', 'office_furniture'],
  - **'High-tech_électronique'** : ['security_and_services','computers_accessories', 'electronics', 'cool_stuff', 'watches_gifts', 'telephony', 'fixed_telephony', 'consoles_games', 'tablets_printing_image', 'audio', 'cine_photo', 'computers', 'dvds_blu_ray'],
  - **'Santé_beauté'** : ['health_beauty'],
  - **'Enfant_bébé'** : ['toys', 'baby', 'diapers_and_hygiene'],
  - **'Outils_construction'** : ['costruction_tools_tools','construction_tools_construction', 'construction_tools_lights', 'costruction_tools_garden', 'construction_tools_safety'],
  - **'Loisirs_culture_divertissement'** : ['arts_and_craftmanship','sports_leisure', 'books_general_interest', 'books_technical', 'cds_dvds_musicals', 'music', 'arts_and_craftsmanship', 'books_imported', 'christmas_supplies', 'drinks', 'food_drink', 'la_cuisine', 'industry_commerce_and_business', 'market_place', 'agro_industry_and_commerce', 'party_supplies', 'art', 'musical_instruments'],
  - **'Alimentation'** : ['food']

Part 3 : Feature engineering

- Data exploration :



Part 3 : Feature engineering

- **New features**
  - **3 columns RFM (Recency, Frequency and Monetary) are added**
  - Recency : - (update date of the base – purchasing date)
  - Frequency : number of orders per unique customer over (update date of the base – first purchasing date of unique customer)
  - Monetary : sum of payment values of unique customer over (update date of the base – first purchasing date of unique customer)

Part 3 : Feature engineering

- **Encoding categorial features**
  - From all categorial features only **'payment_type'** and **'product_category_name_english'** are kept and encoded with the method **get_dummies**

Part 3 : Feature engineering

- Database info at this stage :
  - 24 columns
  - 93895 lines

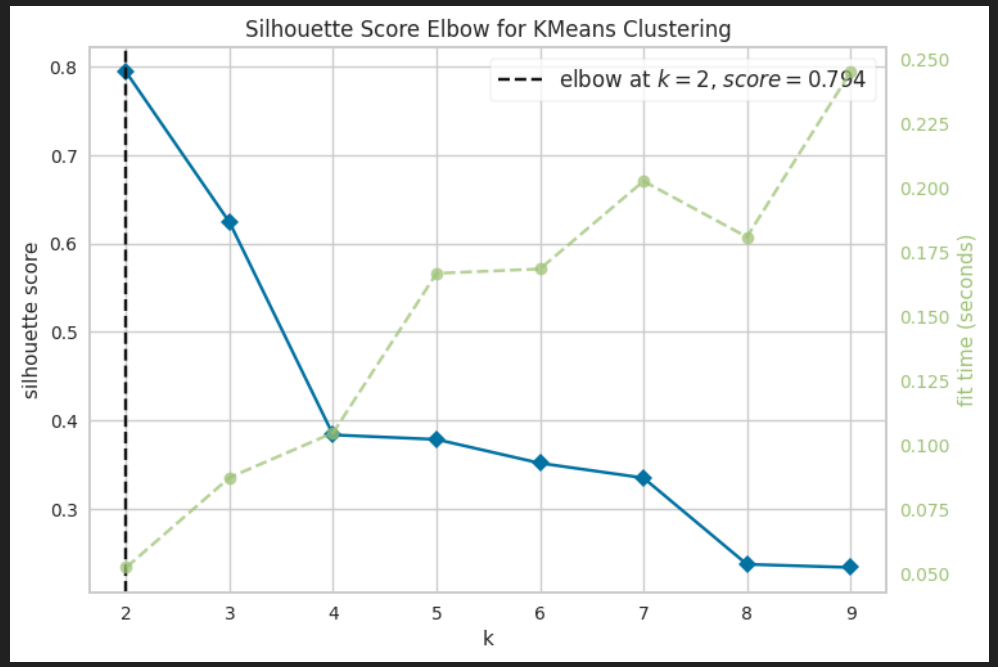
```
RangeIndex: 93895 entries, 0 to 93894
Data columns (total 24 columns):
 #   Column                                                Non-Null Count  Dtype
---  ------                                                --------------  -----
 0   review_score                                          93895 non-null  float64
 1   payment_sequential                                    93895 non-null  float64
 2   payment_installments                                  93895 non-null  float64
 3   payment_value                                         93895 non-null  float64
 4   price                                                 93895 non-null  float64
 5   freight_value                                         93895 non-null  float64
 6   Recence                                               93895 non-null  int64
 7   Frequence                                             93895 non-null  float64
 8   Montant                                               93895 non-null  float64
 9   payment_type_boleto                                   93895 non-null  float64
 10  payment_type_credit_card                              93895 non-null  float64
 11  payment_type_debit_card                               93895 non-null  float64
 12  payment_type_voucher                                  93895 non-null  float64
 13  product_category_name_english_Alimentation            93895 non-null  float64
 14  product_category_name_english_Animaux_de_compagnie    93895 non-null  float64
 15  product_category_name_english_Auto_moto               93895 non-null  float64
 16  product_category_name_english_Enfant_bébé             93895 non-null  float64
 17  product_category_name_english_High-tech_électronique  93895 non-null  float64
 18  product_category_name_english_Loisirs_culture_divertissement  93895 non-null  float64
 19  product_category_name_english_Maison_jardin           93895 non-null  float64
 20  product_category_name_english_Mode_accessoires        93895 non-null  float64
 21  product_category_name_english_Outils_construction     93895 non-null  float64
 22  product_category_name_english_Papeterie_bureau        93895 non-null  float64
 23  product_category_name_english_Santé_beauté            93895 non-null  float64
dtypes: float64(23), int64(1)
```

Part 4 : Clustering model testing

- **We normalize first the data with RobustScaler :**
  - RobustScaler is is more robust than StandardScaler to outliers and is less likely to be affected by them. It uses interquartile range (IQR) instead of standard deviation to calculate the scale, which is more robust to outliers as it is based on percentiles.
- **We will select the best model according to several indicators :**
  - **Davies Bouldin** for cluster separation (the lower the greater separation)
    - *The Davies-Bouldin index is calculated as the average similarity between each cluster and its closest neighbor cluster, where similarity is defined as the maximum value of the ratio of the within-cluster sum of squares to the between-cluster distance between the two clusters.*
  - **Intracluster inertia** for cluster compacity (the lower the biggest compacity)
    - *It measures the sum of the squared distances between each data point and the centroid of its assigned cluster.*
  - **Silhouette score** mixing compacity and separation of the clusters (between -1 and 1 and 1 is the best score)
    - *The silhouette score is calculated for each data point by computing the average distance between that point and all other points within the same cluster, and dividing this value by the average distance between that point and all other points in the next closest cluster. The silhouette score ranges from -1 to 1, where a higher score indicates a better clustering solution.*

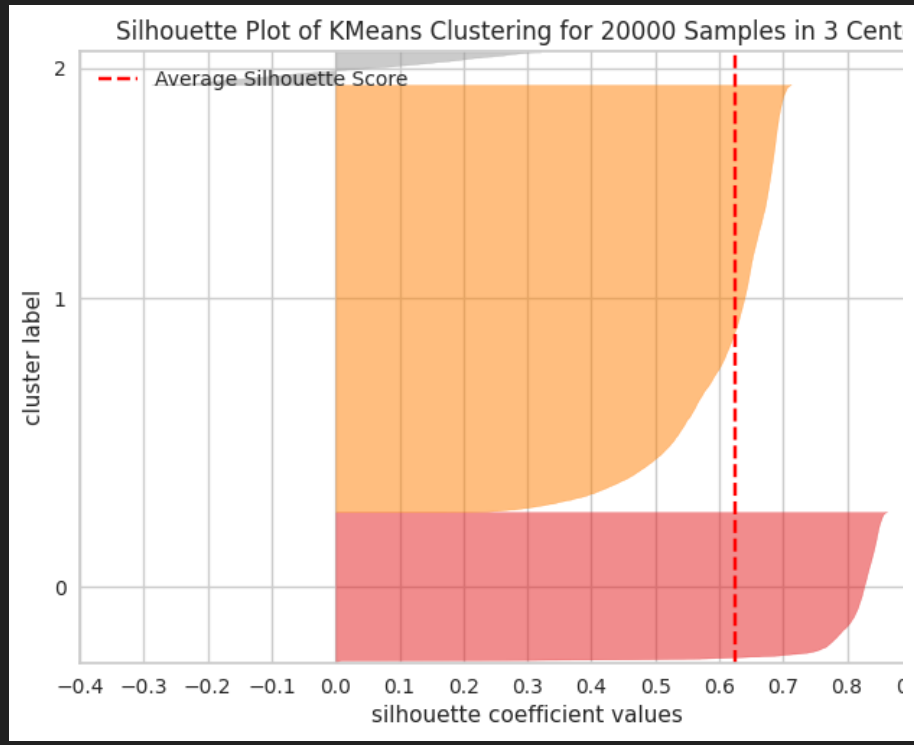
Part 4 : Clustering model testing

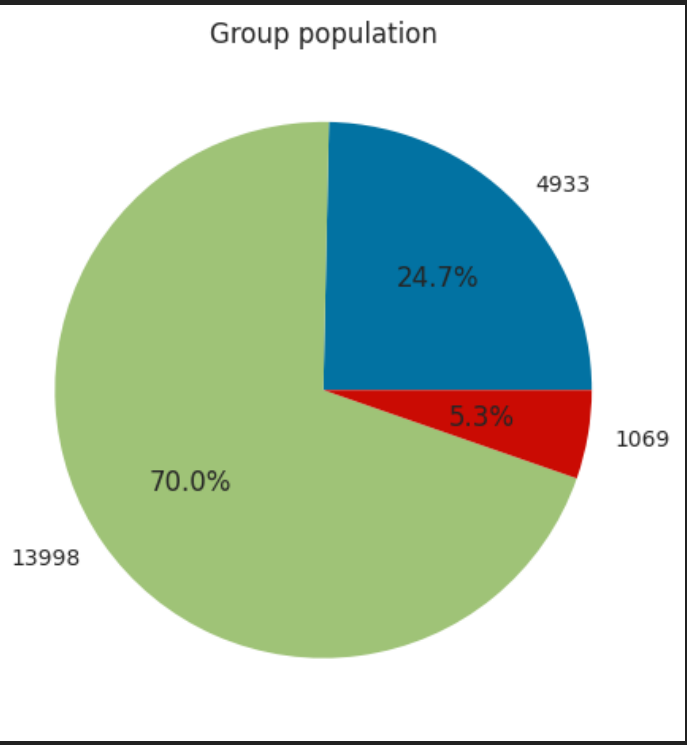
- **KMeans**
  - Selection of best parameter K and Silhouette score. We take K = 3 (tradeoff between score and coherent K number for segmentation)



Part 4 : Clustering model testing

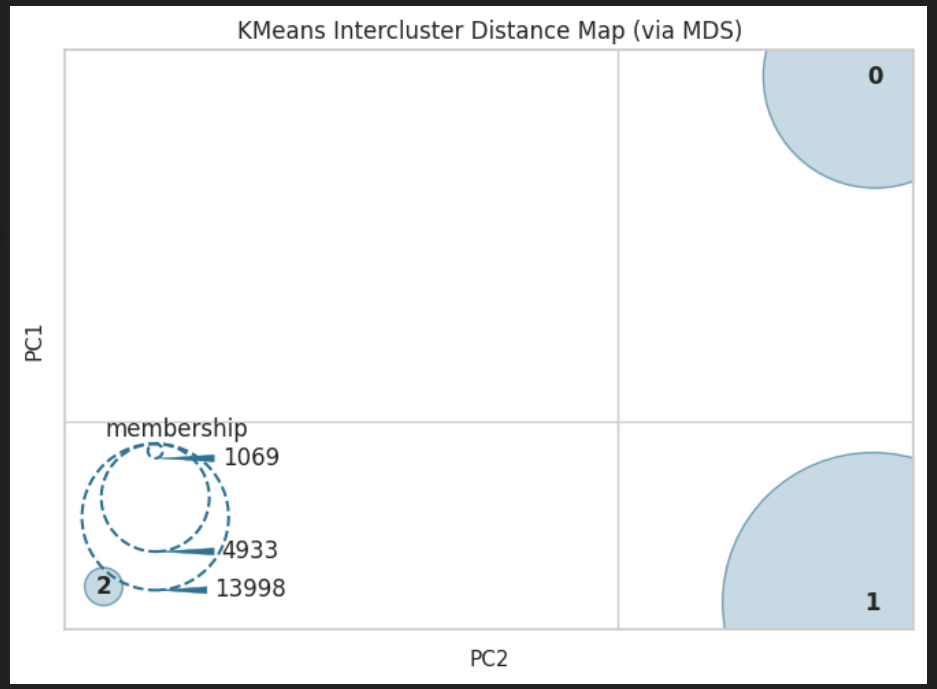
- KMeans
  - K=3 => Silhouette = 0.62
  - This average score between 0 and 1 means that the clusters are compact and there is a good separation between clusters
  - Nevertheless, on the smallest cluster one third of the points have a score below 0 which means that these points are not well assigned

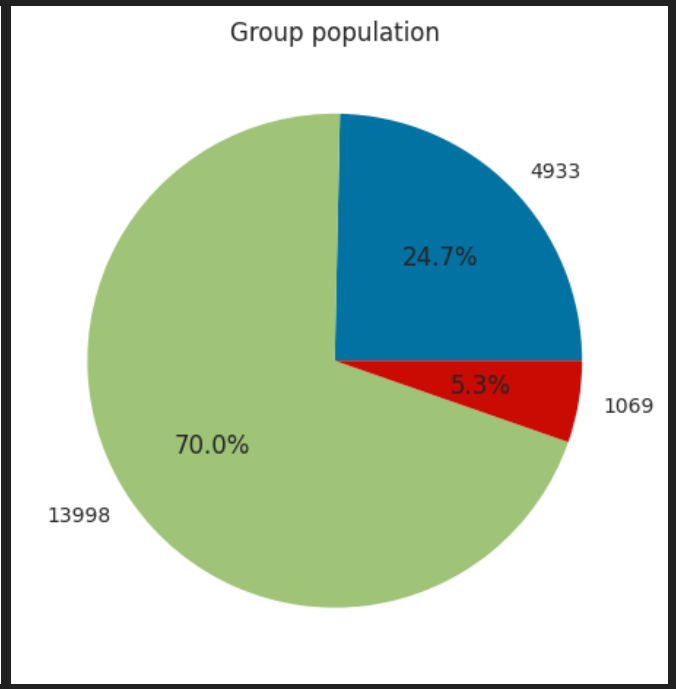




Part 4 : Clustering model testing

- KMeans
  - K=3 => Davies Bouldin = 0.91
    - This value under 1 suggests that the clusters are well-separated from each other.
    - We can see on the visualization on the left that the clusters are well separated according to the two first principal components.



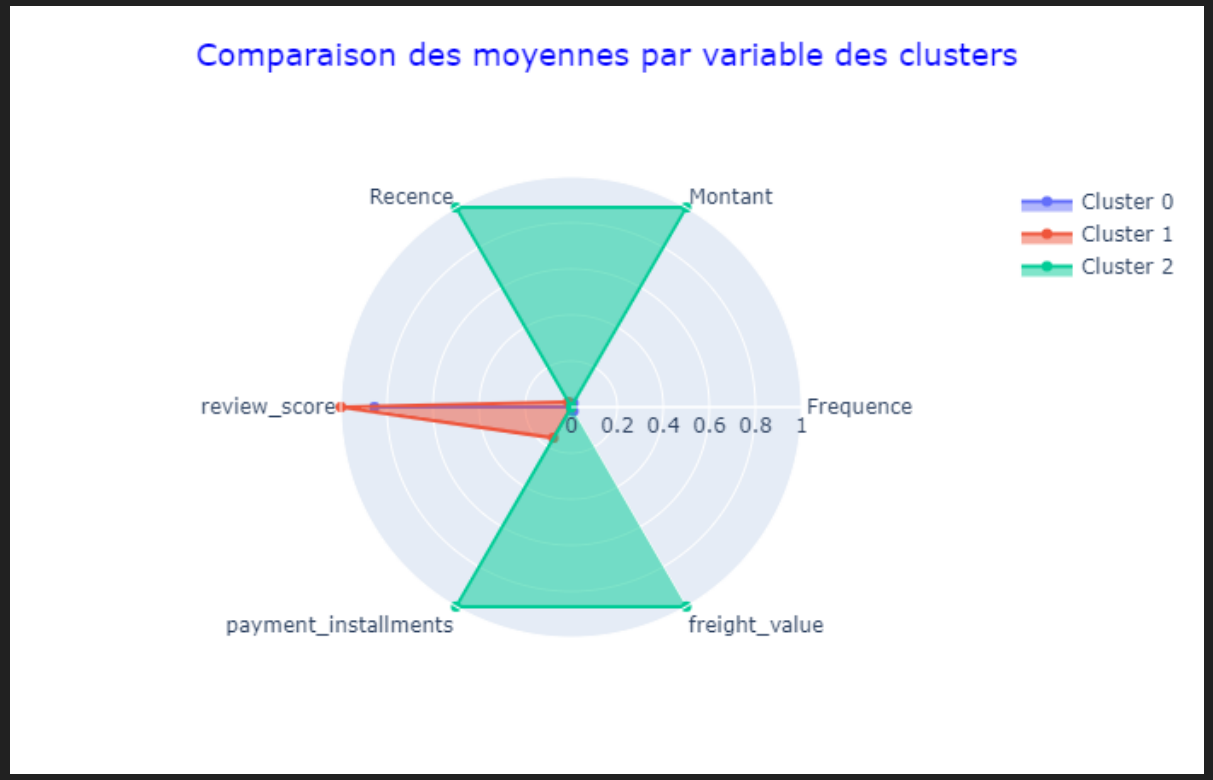


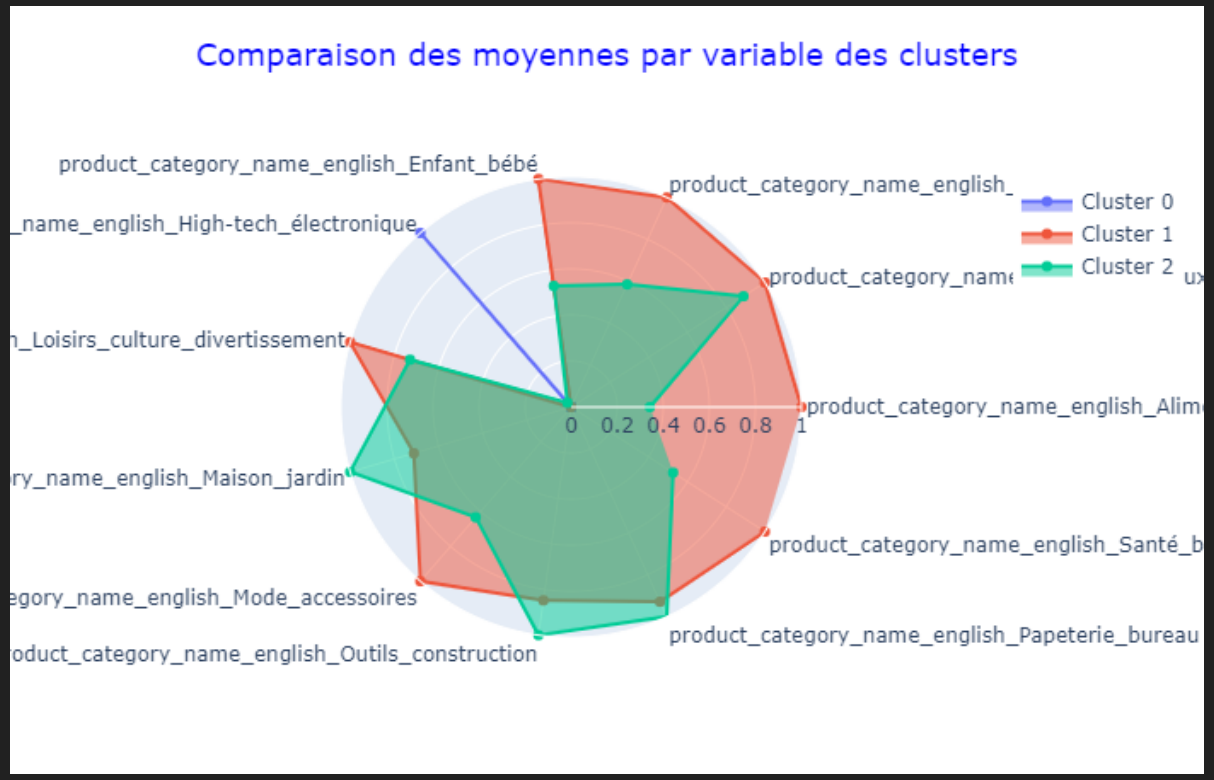
Part 4 : Clustering model testing

- KMeans
  - K=3 => Intracluster Inertia = 204765
    - This value gives an indication of the compacity of the clusters. It should be compared with the next segmentations.

Part 4 : Clustering model testing

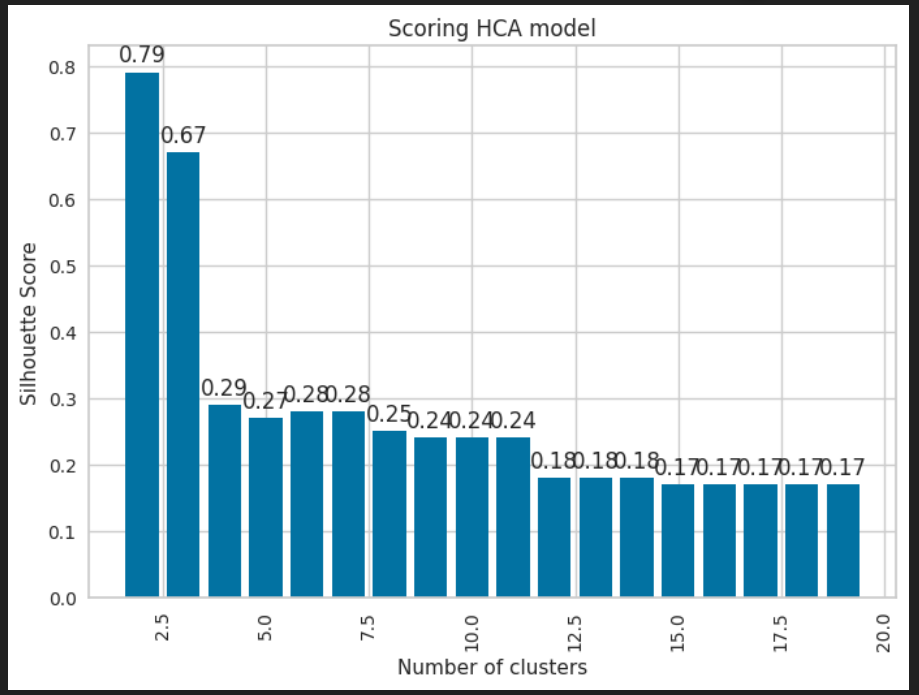
- KMeans





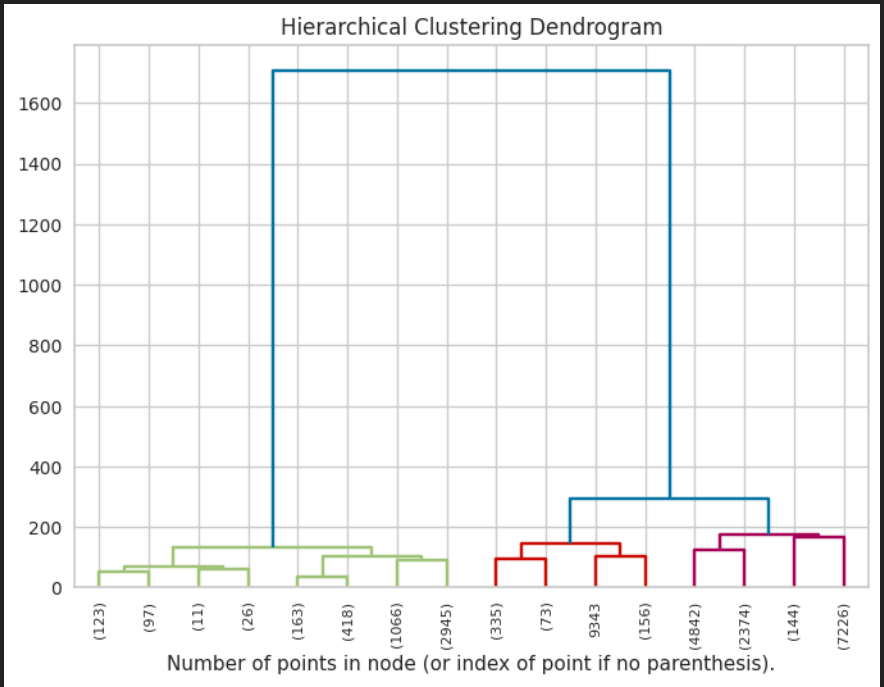
Part 4 : Clustering model testing

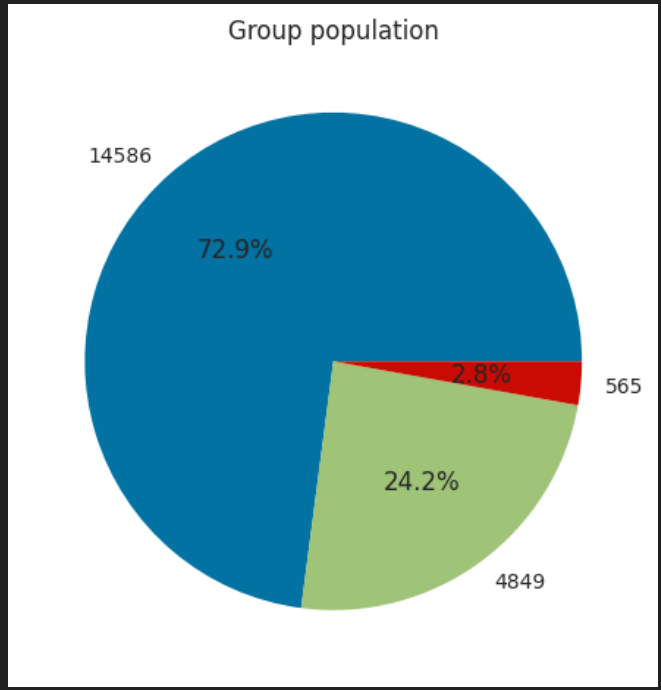
- **HCA**
  - Selection of best parameter K and Silhouette score. We take K = 3 (tradeoff between score and coherent K number for segmentation)



Part 4 : Clustering model testing

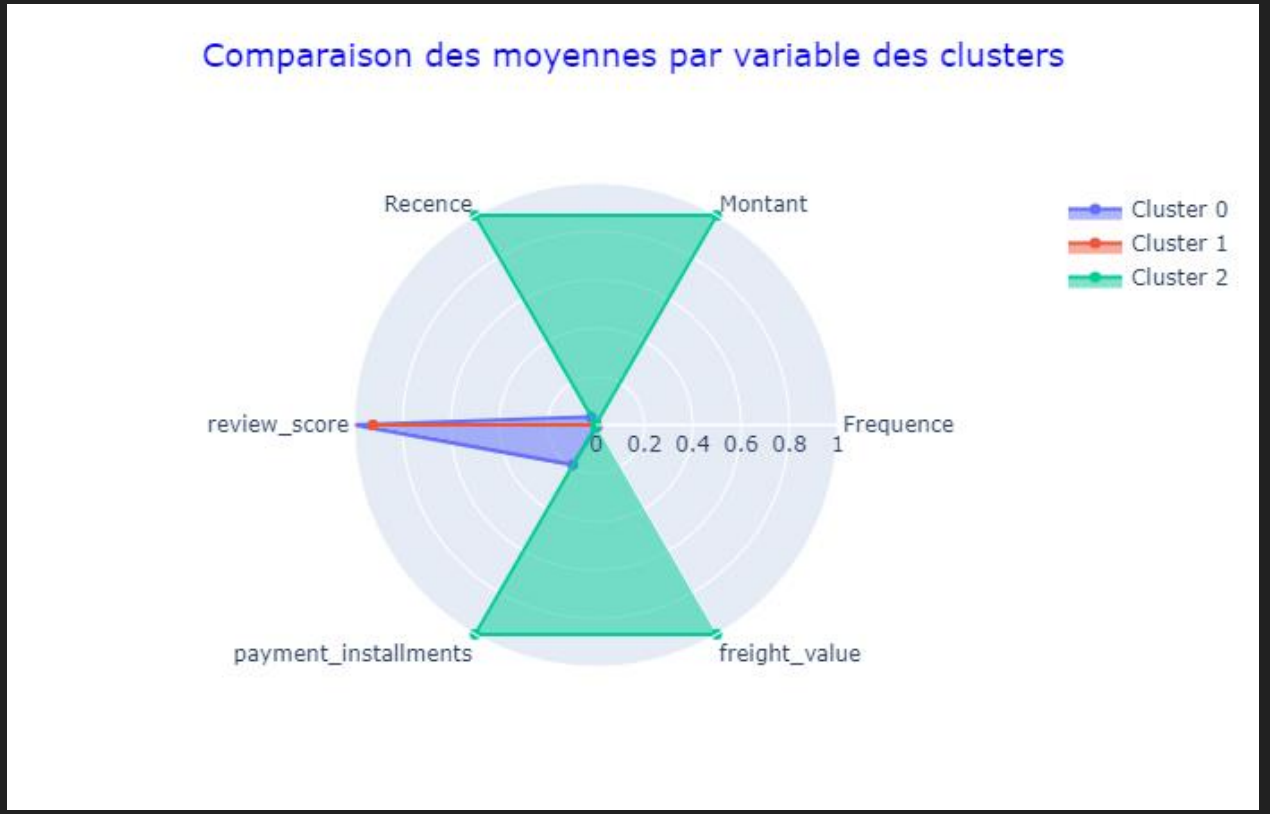
- **HCA**
  - **K=3 => Davies Bouldin = 0.83 – Intracluster Inertia = 55332 – Silhouette = 0.67**
  - *For Kmeans we had : K=3 => Davies Bouldin = 0.91 – Intracluster Inertia = 204765 – Silhouette = 0.62*

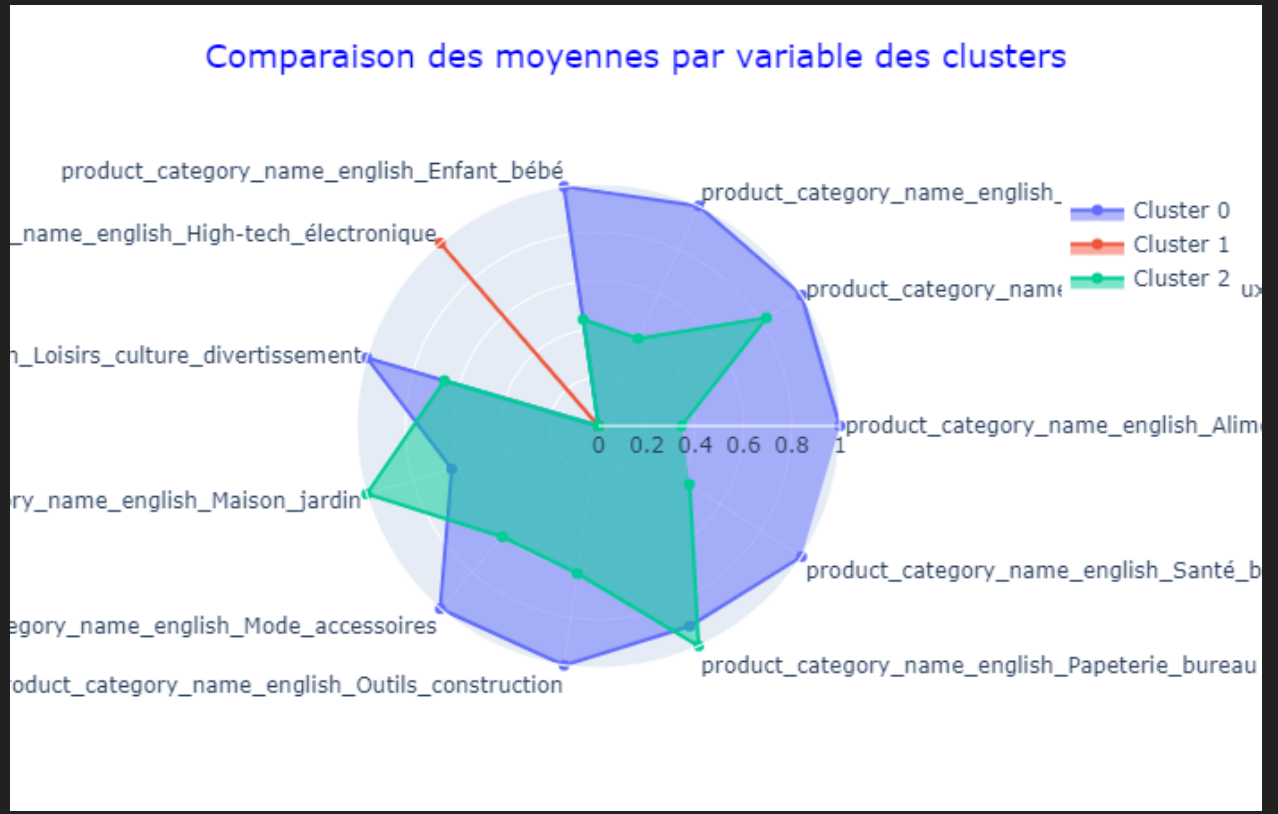




Part 4 : Clustering model testing

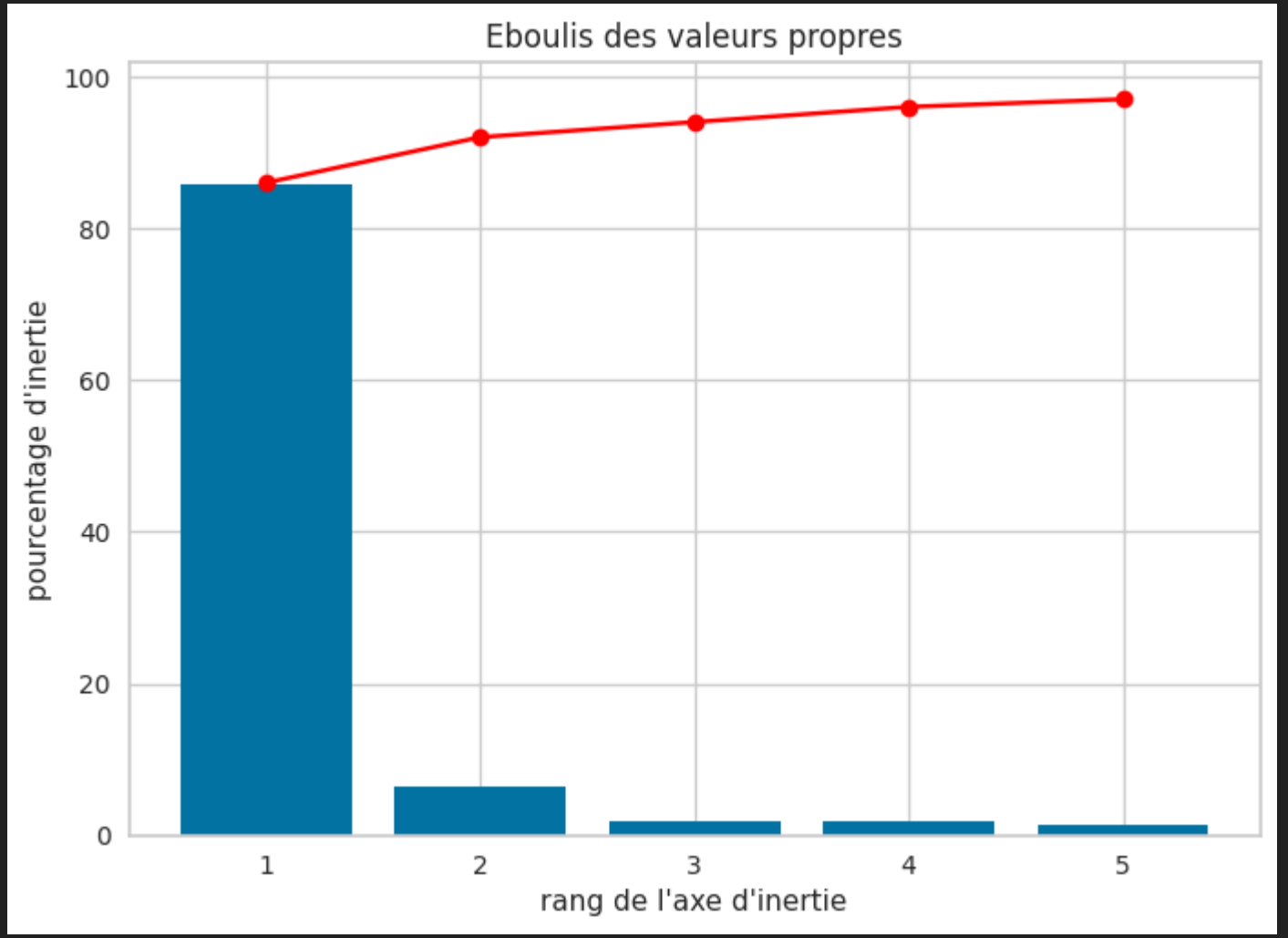
- HCA



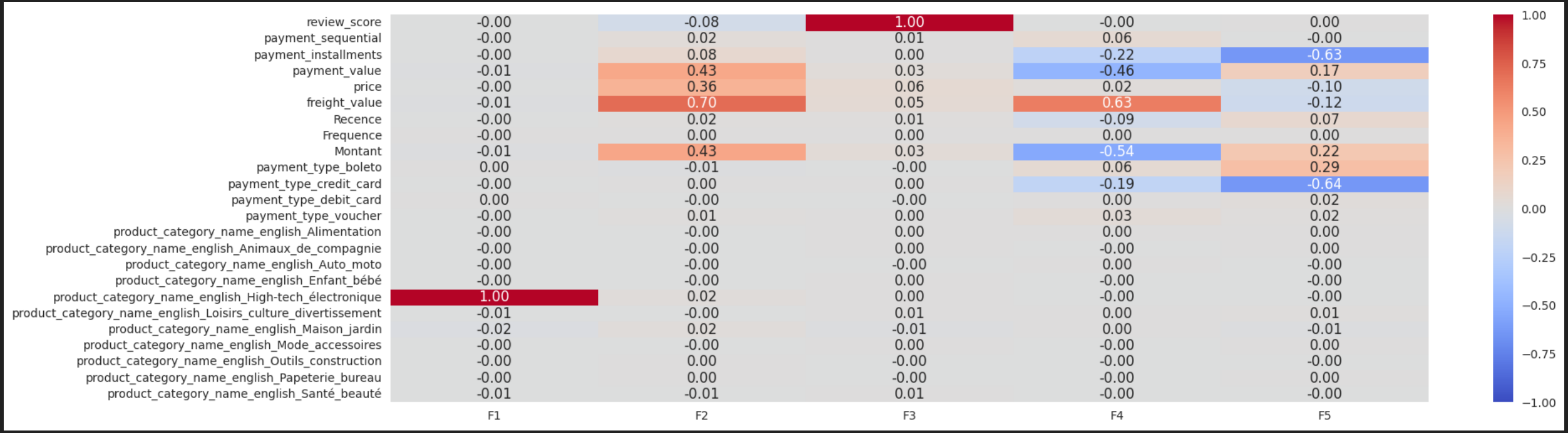


Part 4 : Clustering model testing

- **PCA**
- We now implement a PCA and feed the models with the data reduced to their most important components
- We take the 5 first components covering >90% of the variance

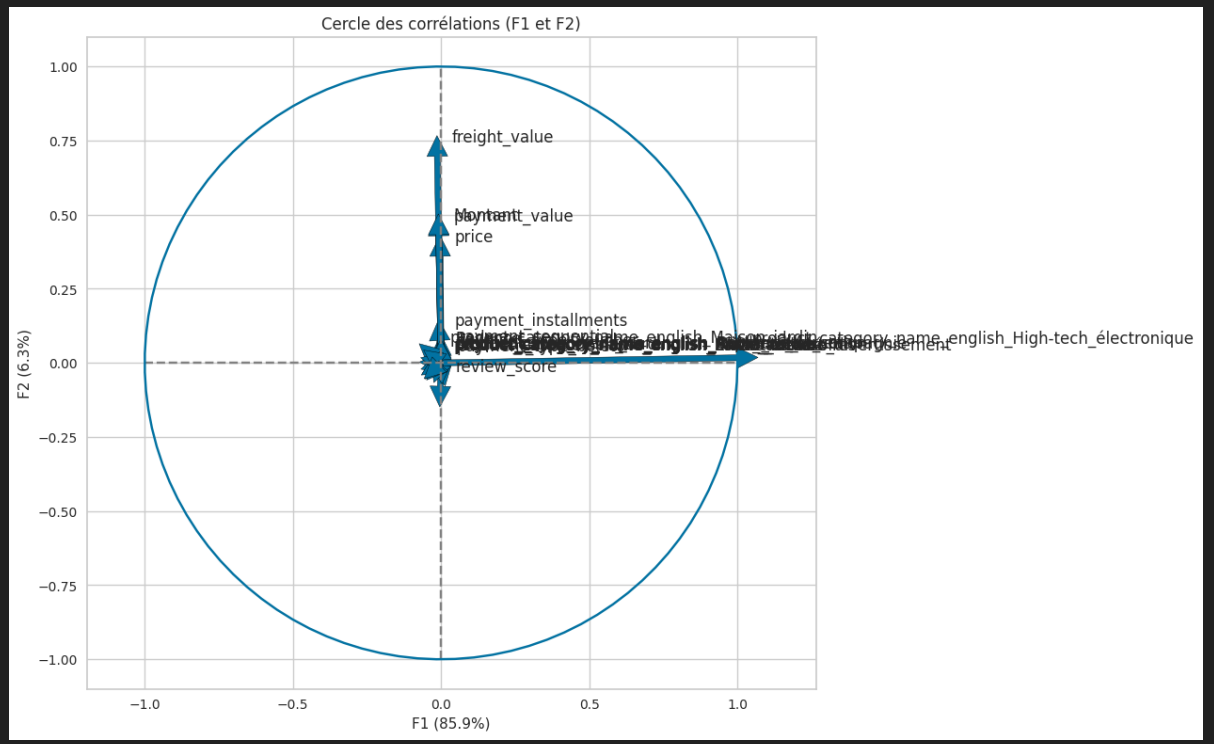


Part 4 : Clustering model testing



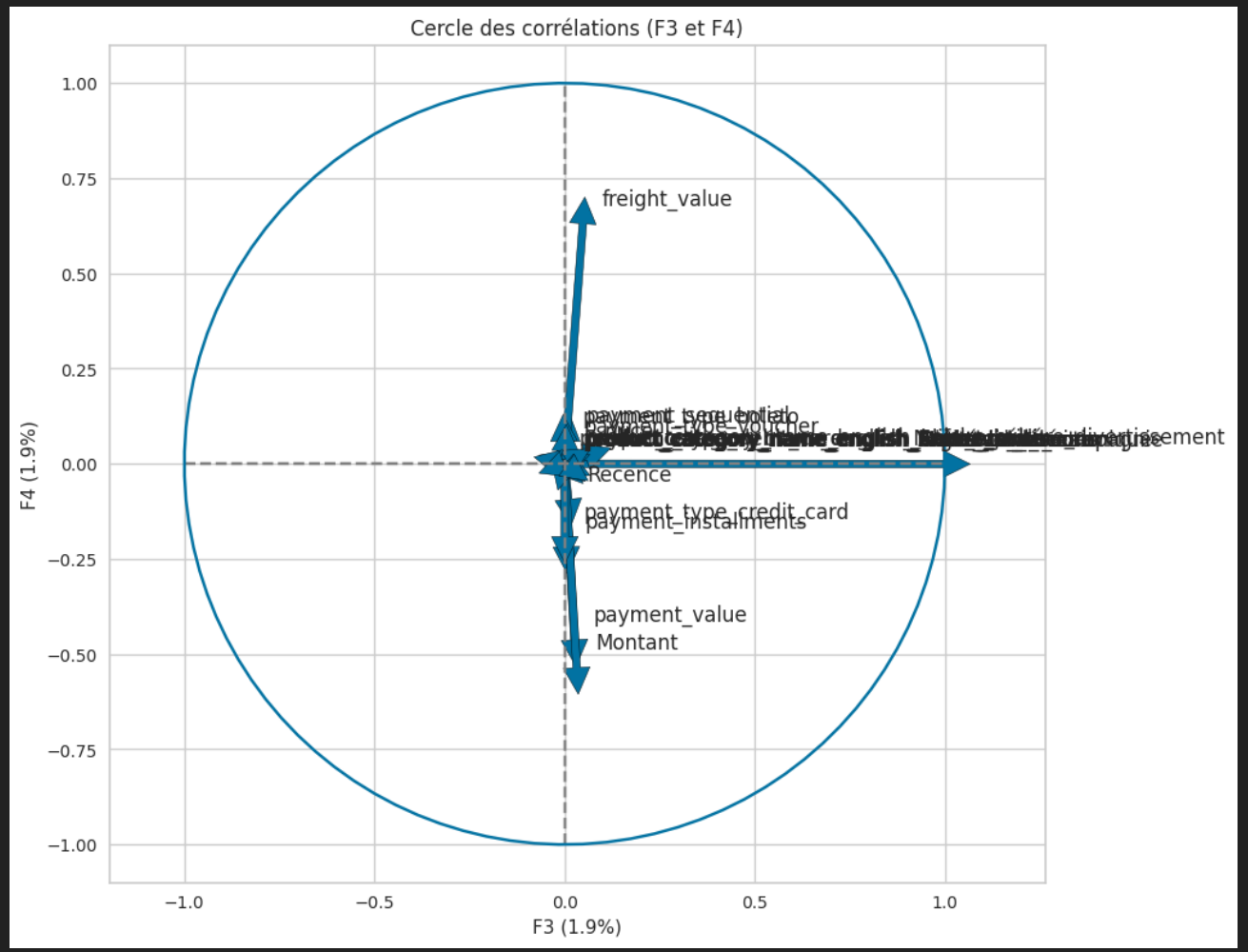
Part 4 : Clustering model testing

- **PCA**
- **F1-F2 factorial plane**
- Left to Right => strongly correlated to ‘High_tech_électronique’ product category
- Lower to Upper side is correlated to the overall cost of the purchase



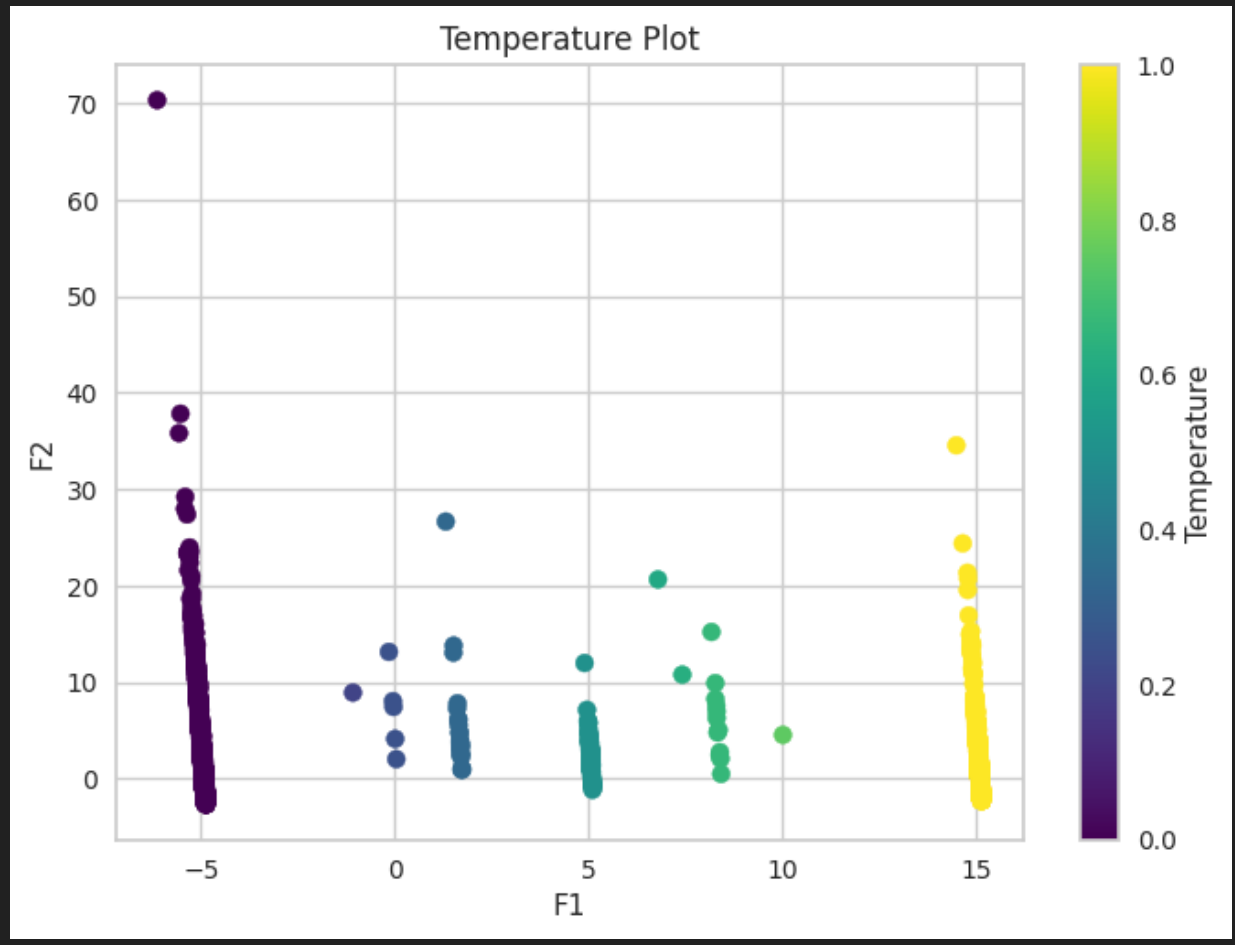
Part 4 : Clustering model testing

- **PCA**
- **Factorial plane F3-F4**
- Left to right : strong correlation with the review_score.



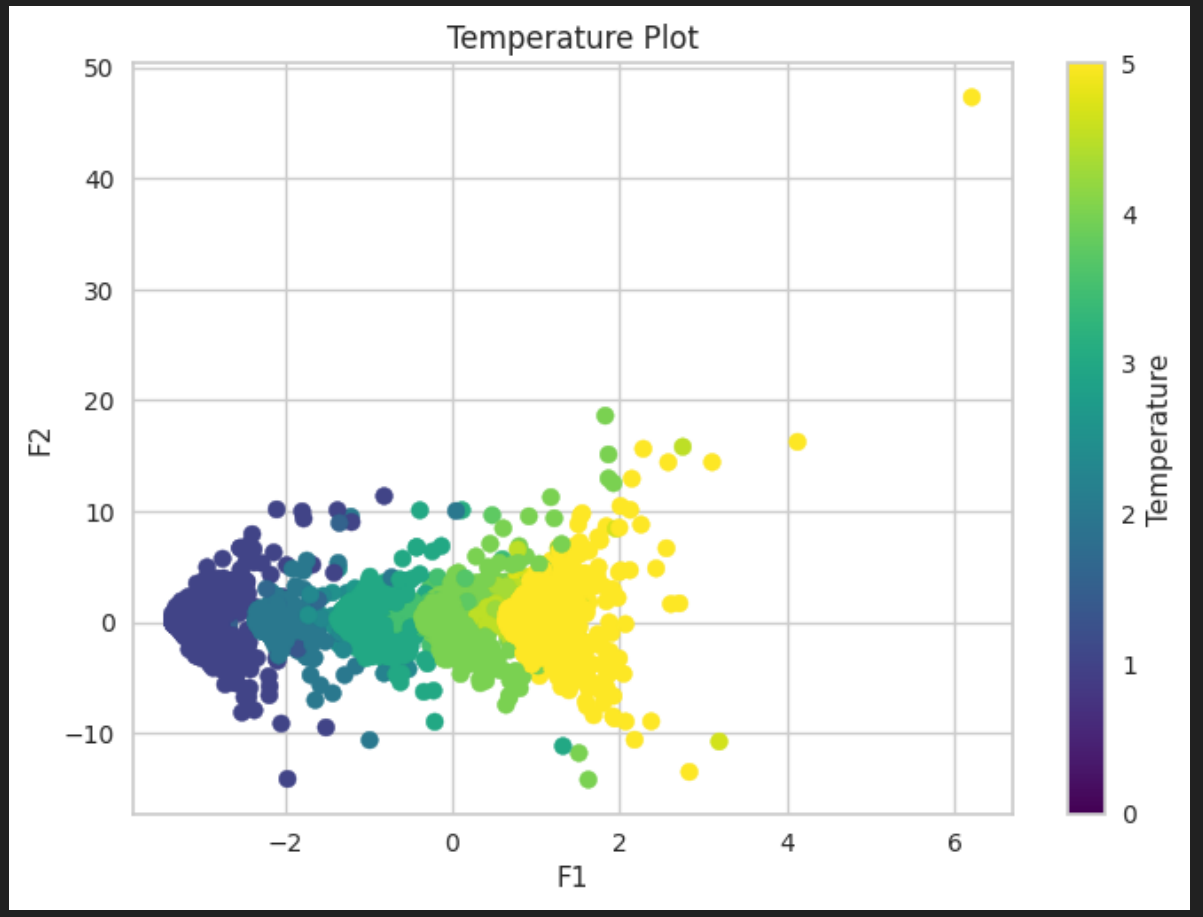
Part 4 : Clustering model testing

- **PCA**
- Data displayed on the F1-F2 factorial plane : see the clear trend with product category ‘High_tech_électronique’



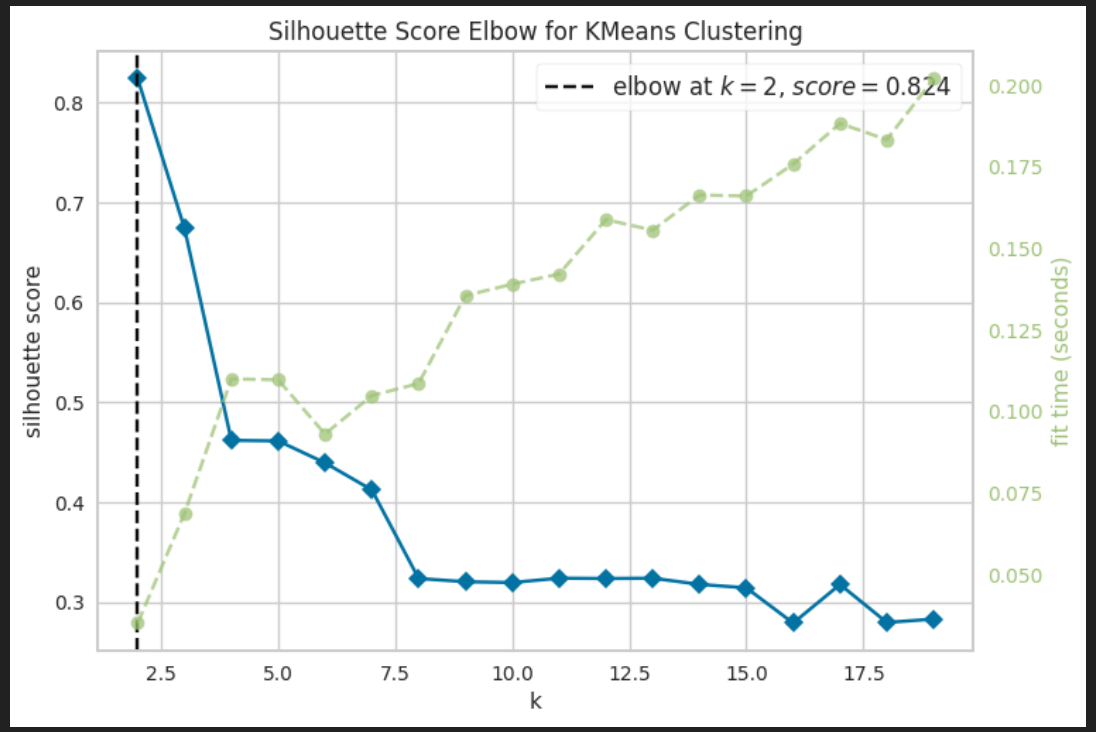
Part 4 : Clustering model testing

- **PCA**
- Data displayed on the F3-F4 factorial plane : see the clear trend with ‘review_score’



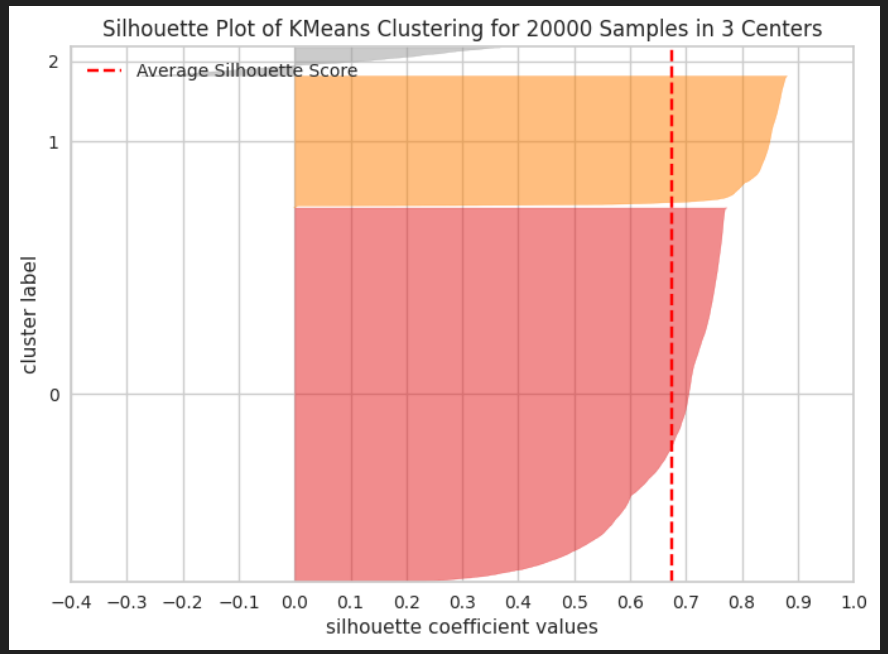
Part 4 : Clustering model testing

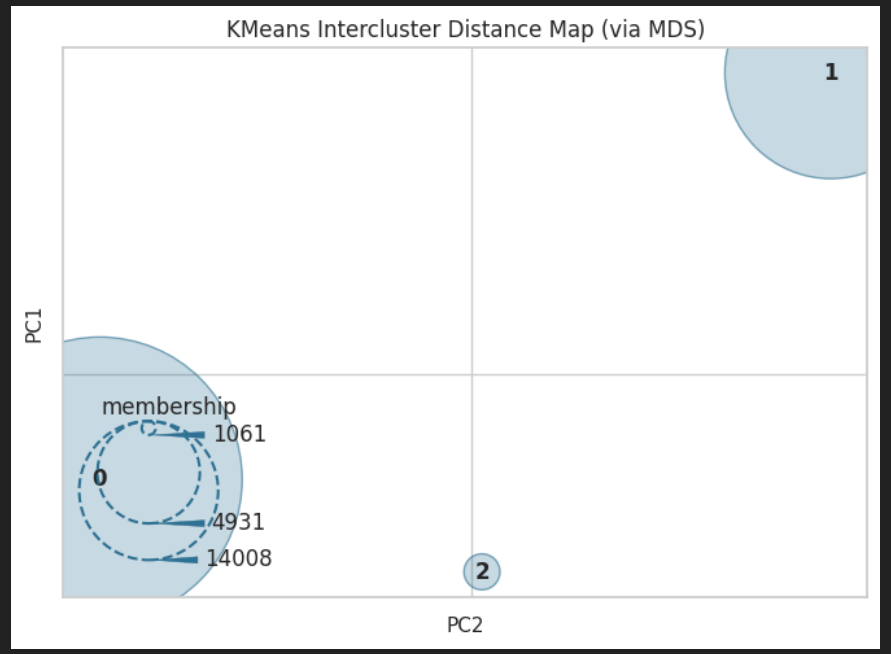
- **PCA + KMeans**
  - Selection of best parameter K and Silhouette score. We take K = 3 (tradeoff between score and coherent K number for segmentation)

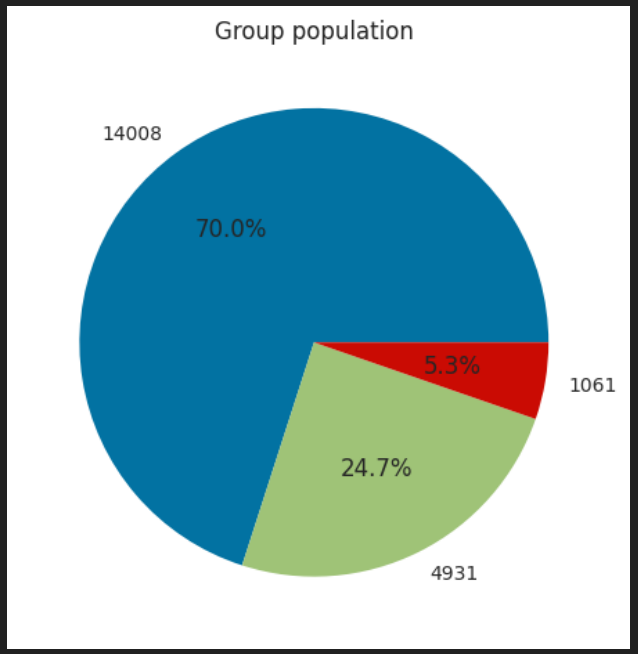


Part 4 : Clustering model testing

- **PCA + KMeans**
  - **K=3 => Davies Bouldin = 0.80 – Intracluster Inertia = 159826 – Silhouette = 0.67**
  - *Without PCA it was : K=3 => Davies Bouldin = 0.91 – Intracluster Inertia = 204765 – Silhouette = 0.62*

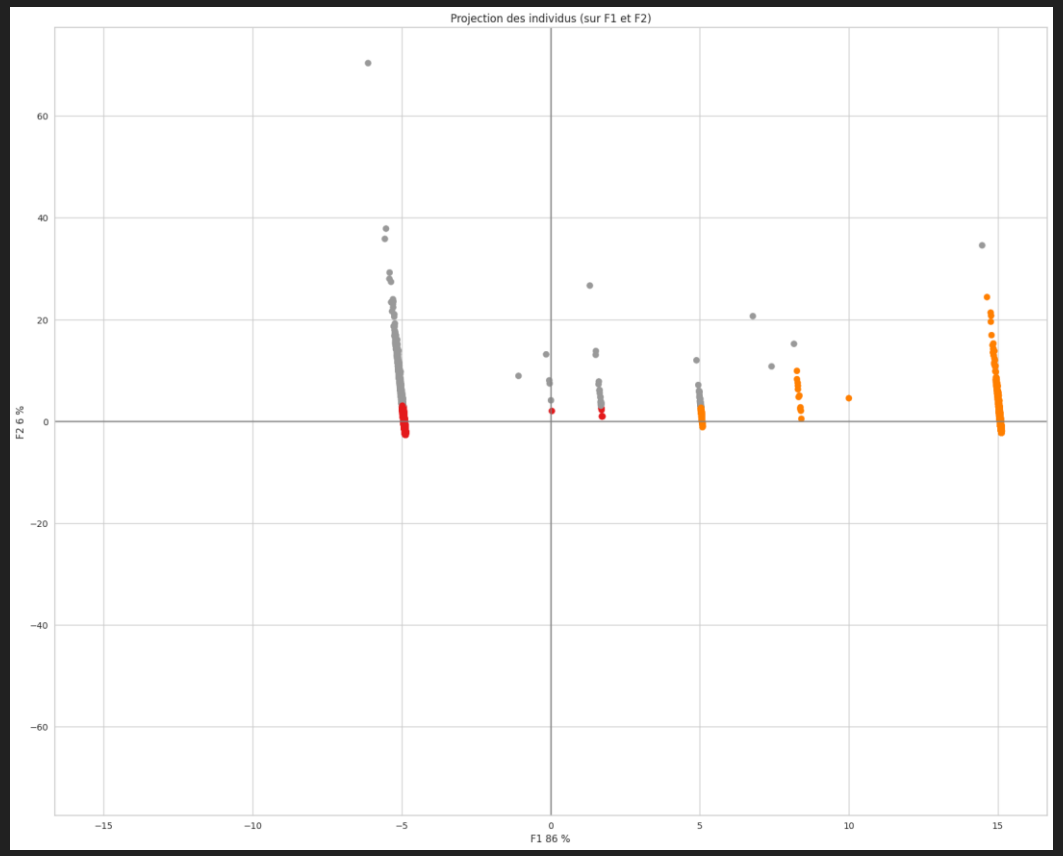


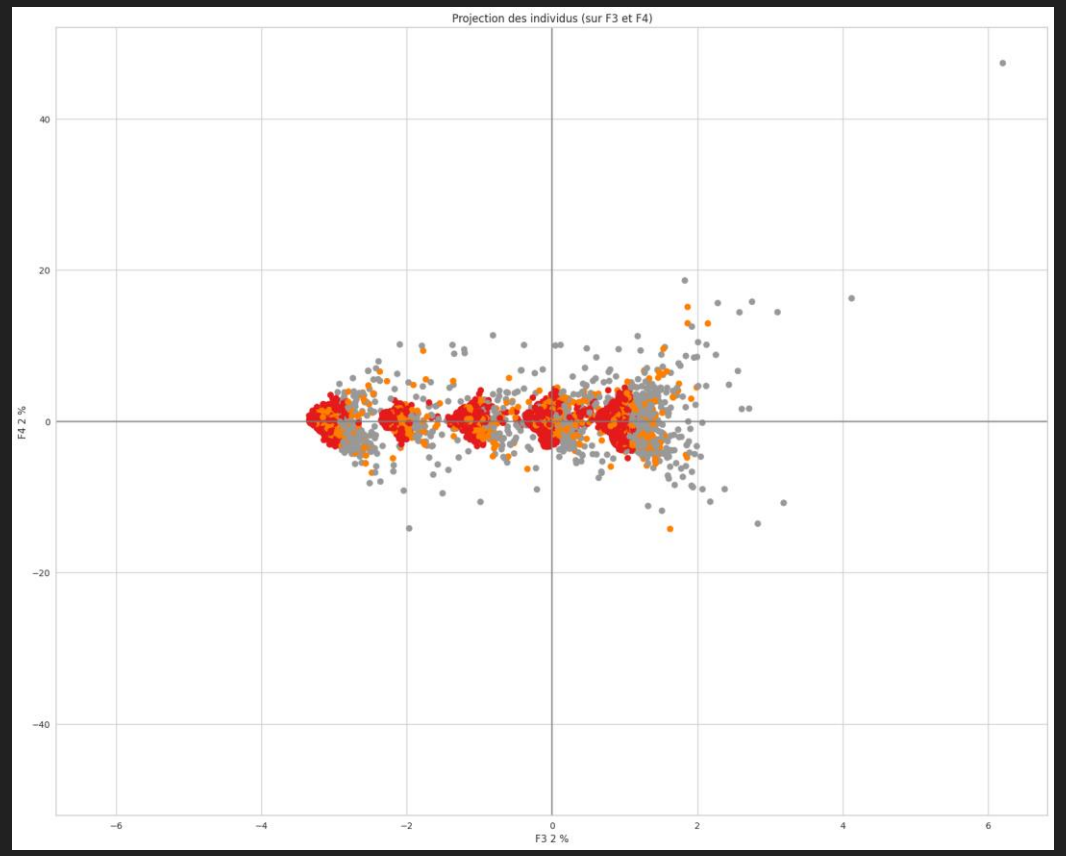




Part 4 : Clustering model testing

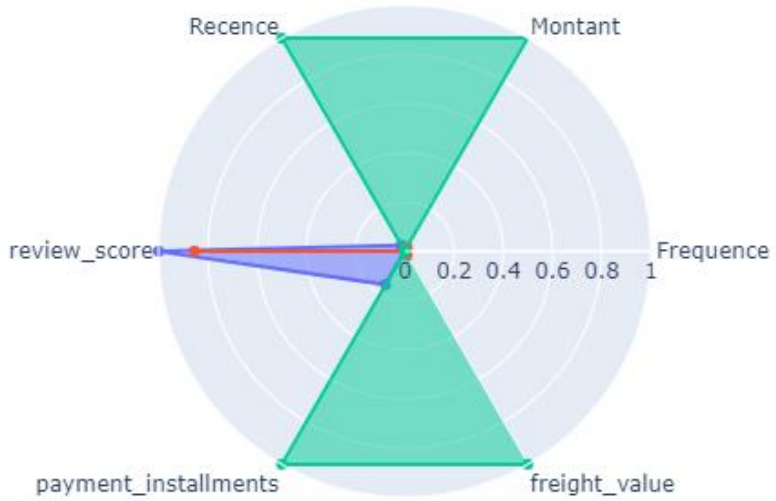
- PCA + KMeans

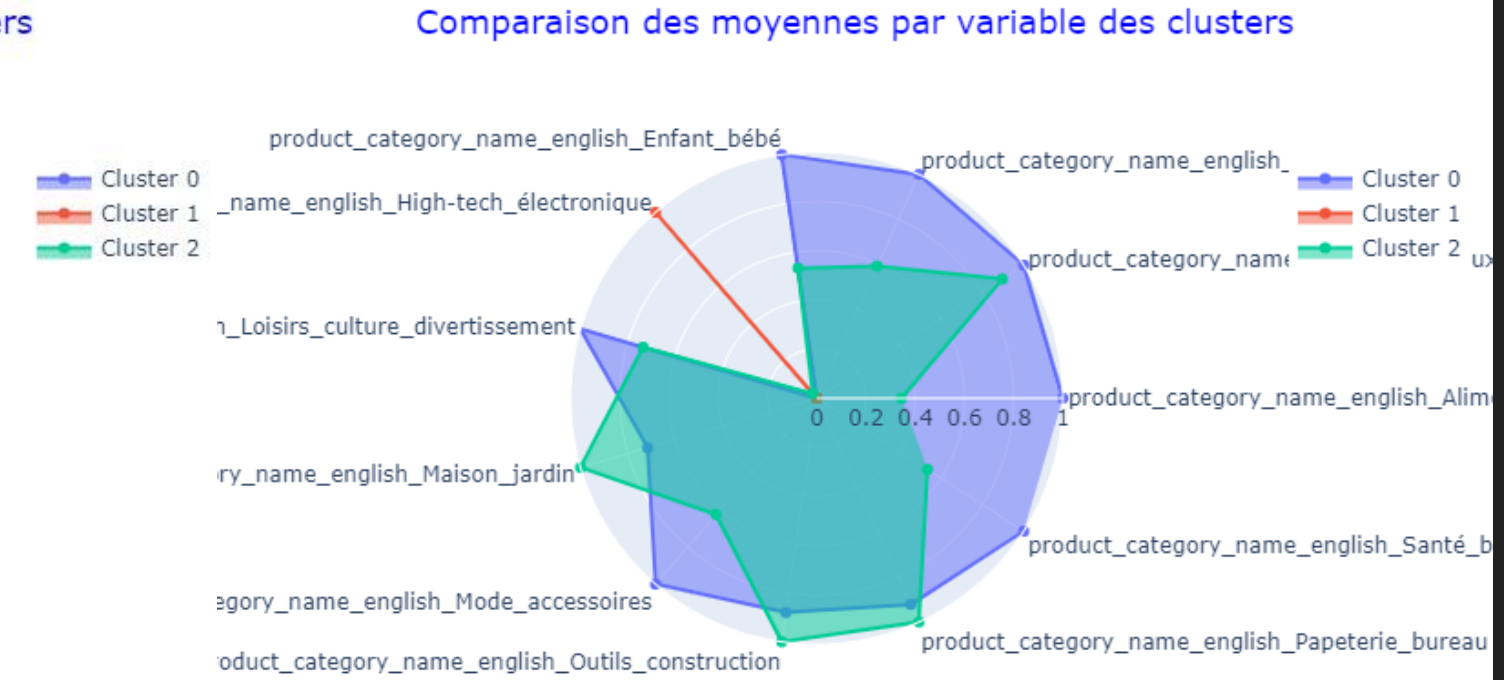




Part 4 : Clustering model testing

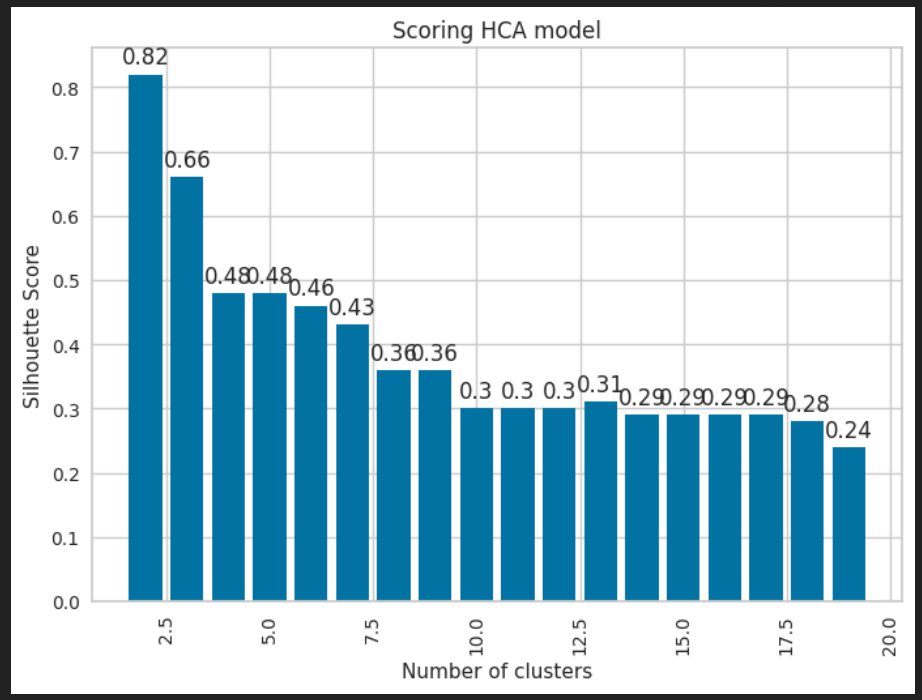
- PCA + KMeans





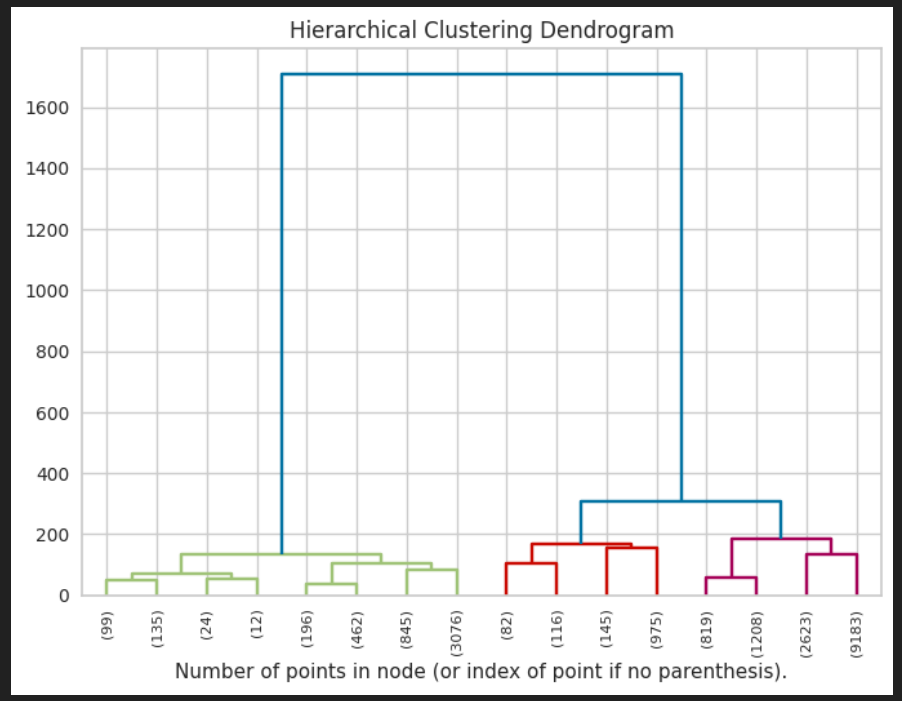
Part 4 : Clustering model testing

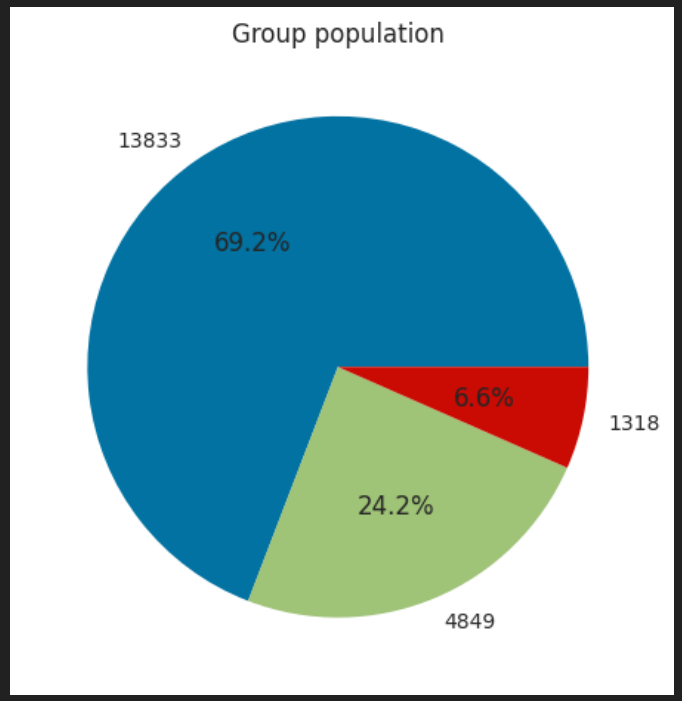
- **PCA + HCA**
  - Selection of best parameter K and Silhouette score. We take K = 3 (tradeoff between score and coherent K number for segmentation)



Part 4 : Clustering model testing

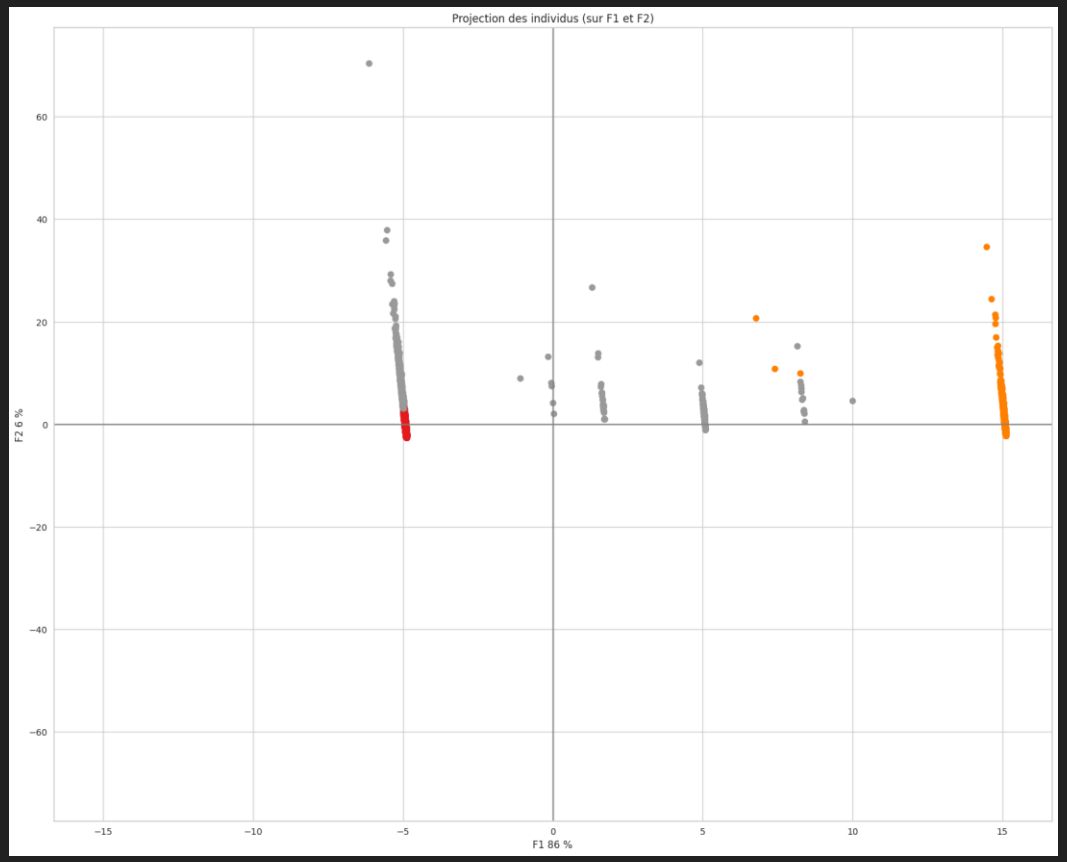
- **PCA + HCA**
  - **K=3 => Davies Bouldin = 0.91 – Intracluster Inertia = 45329 – Silhouette = 0.66**
  - *Without PCA it was : K=3 => Davies Bouldin = 0.83 – Intracluster Inertia = 55332 – Silhouette = 0.67*

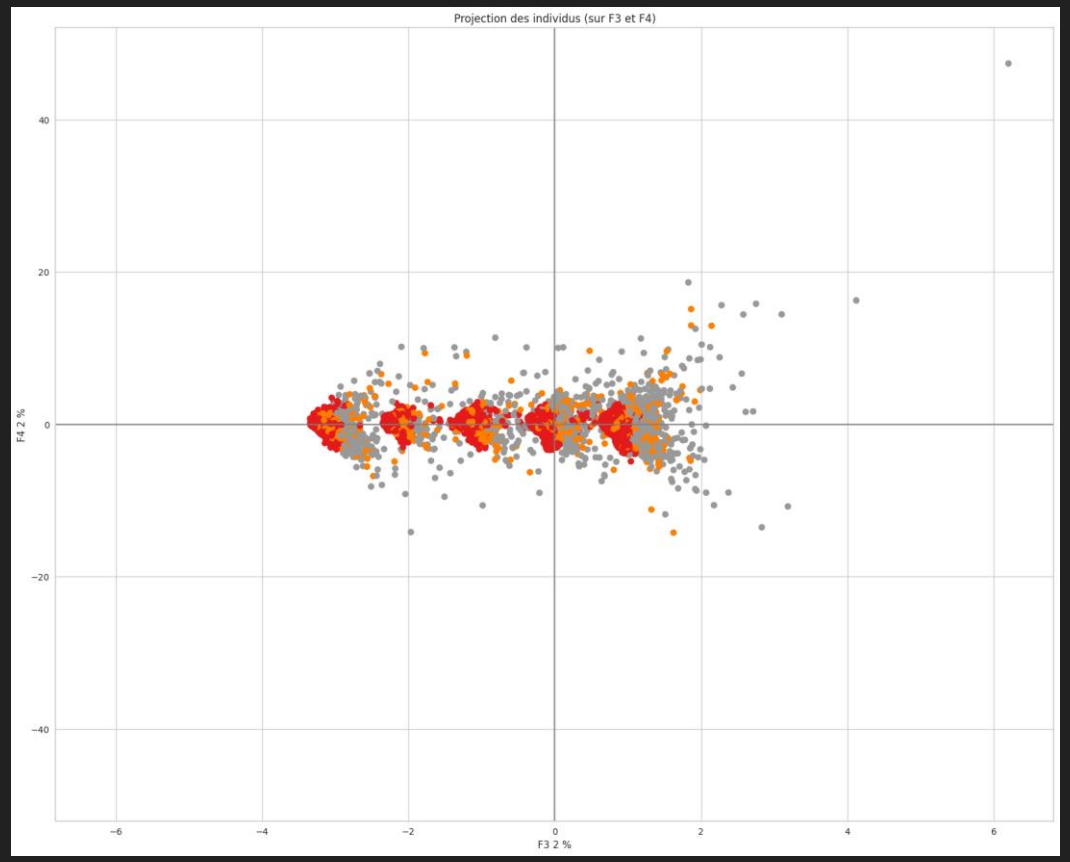




Part 4 : Clustering model testing

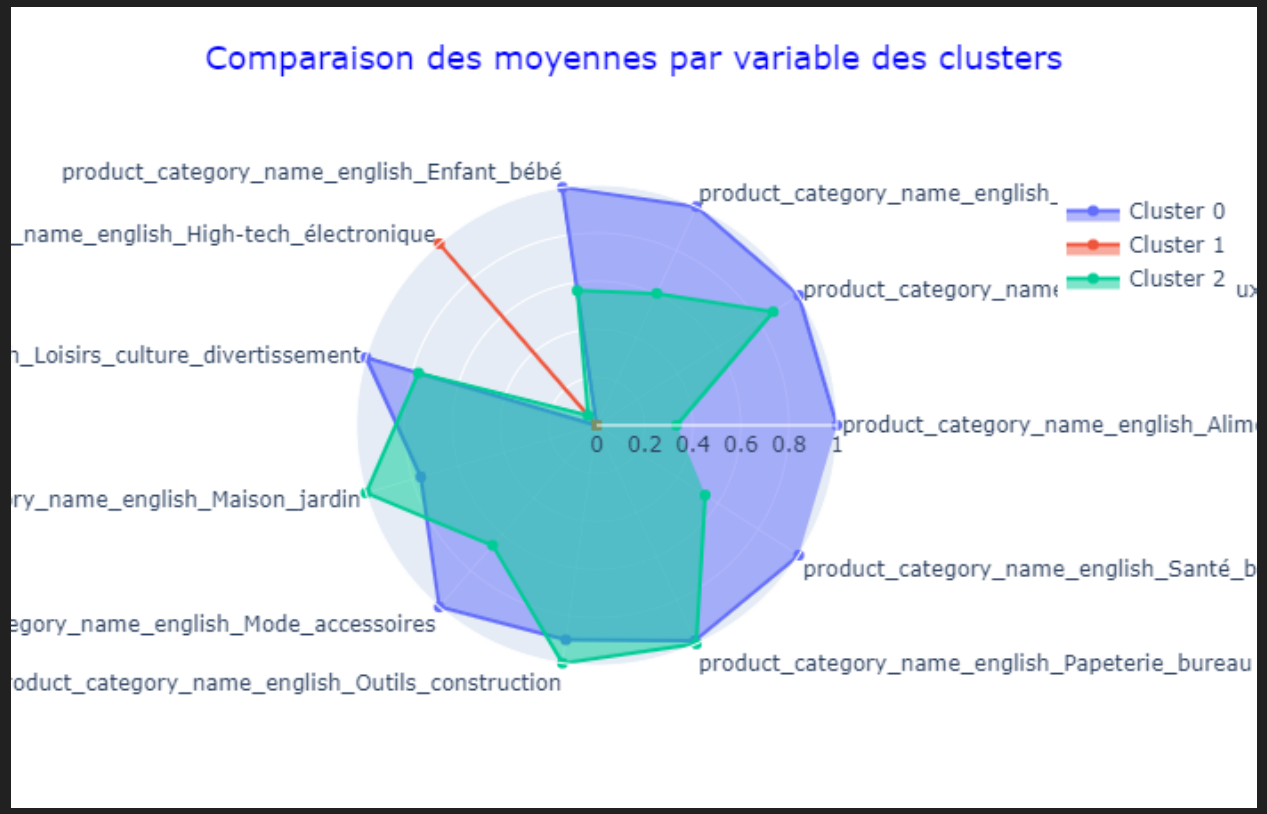
- PCA + HCA

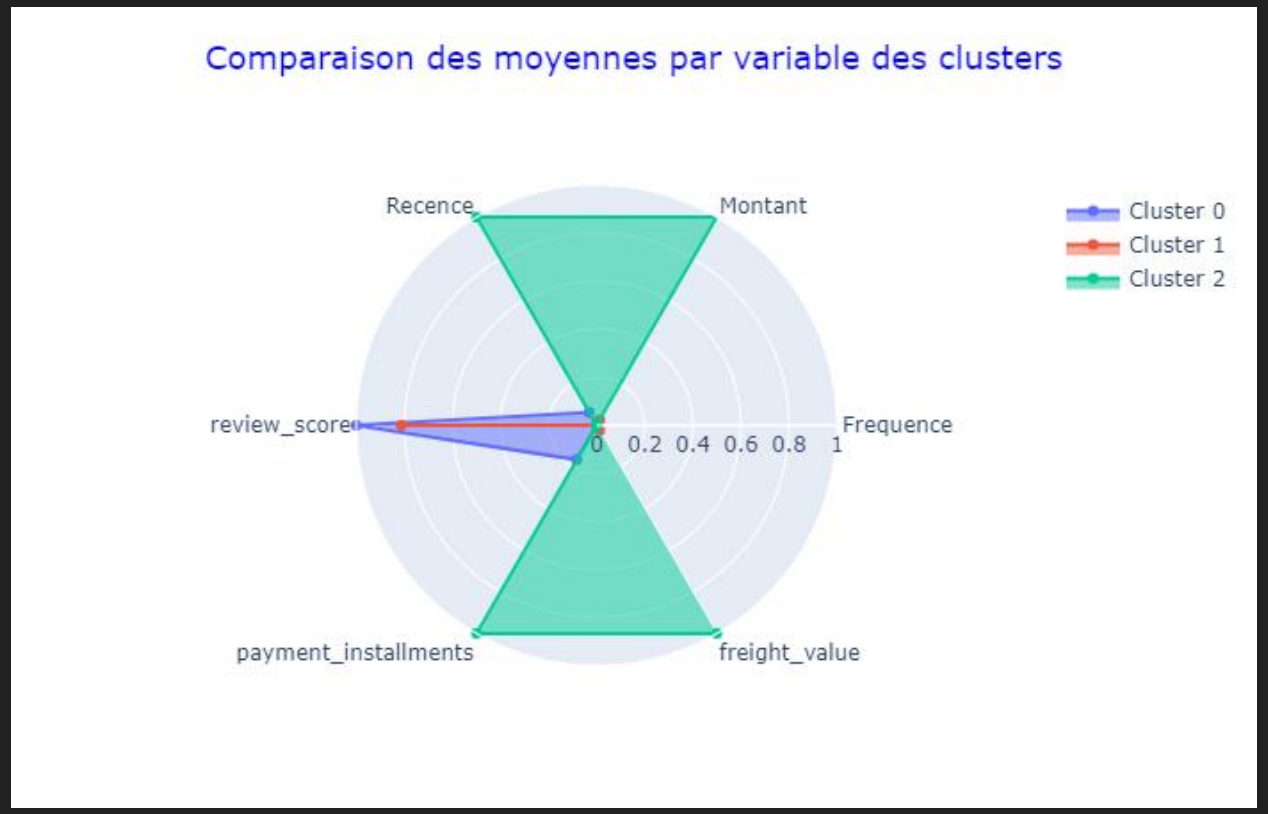




Part 4 : Clustering model testing

- PCA + HCA





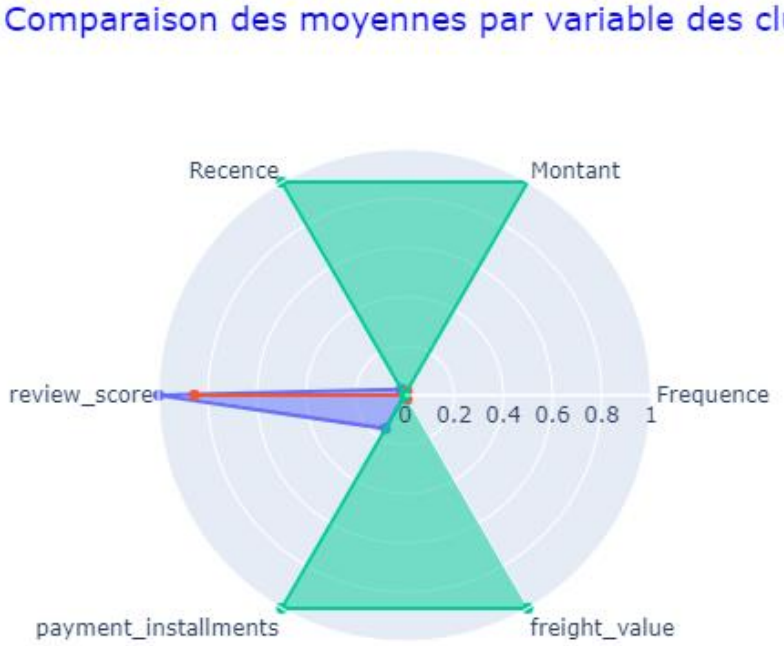
Part 4 : Clustering model testing

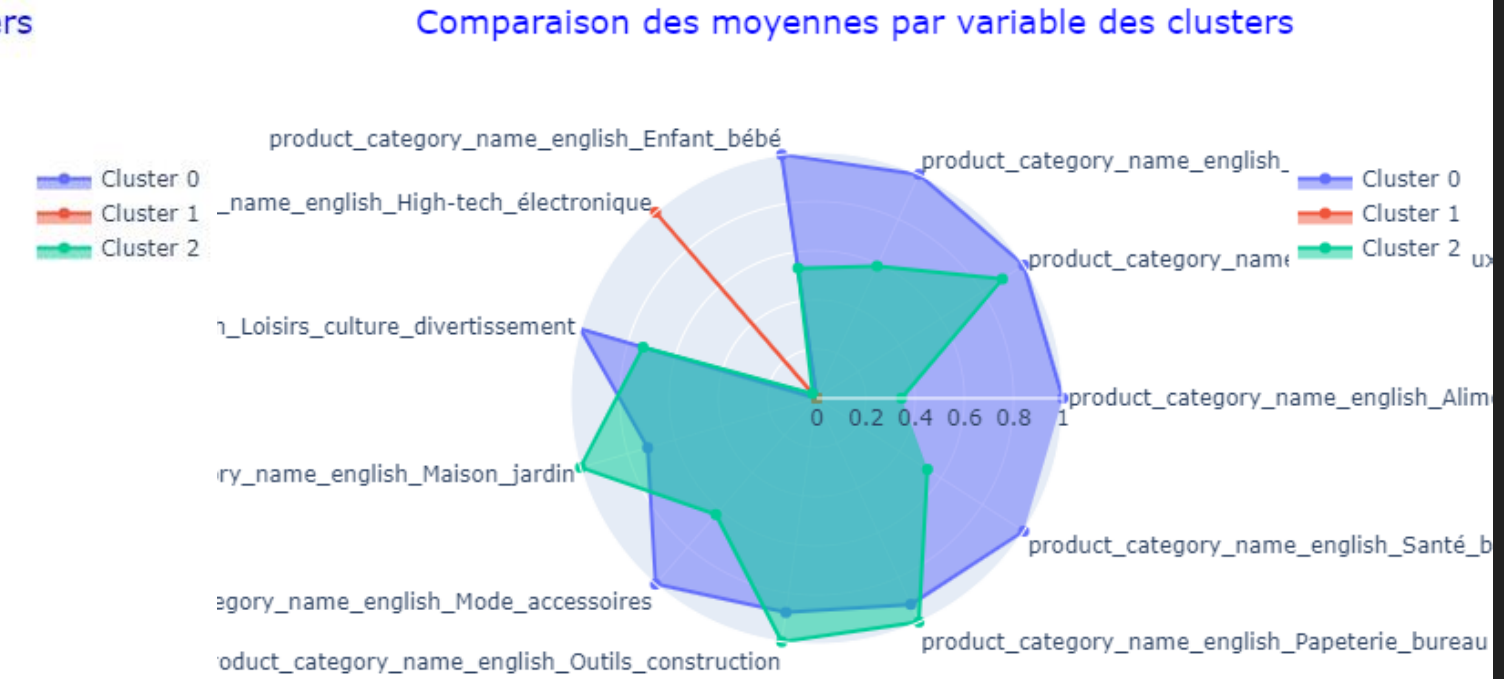
- **Result sum-up :**

model	nb of clusters	silhouette_score	Davies Bouldin	Intracluster Varaiance	Model fitting time
KMeans(n_clusters=3, random_state=42)	3	0.623916	0.91202	204765.306198	0.090427
AgglomerativeClustering(affinity='euclidean', ...	3	0.667645	0.833372	55332.515778	12.88574
PCA+KMeans(n_clusters=3, random_state=42)	3	0.674009	0.797791	159826.585848	0.067263
PCA+AgglomerativeClustering(affinity='euclidea..	3	0.657249	0.911396	45329.963358	10.100905

Part 4 : Clustering model testing

- PCA + Kmeans is the best model



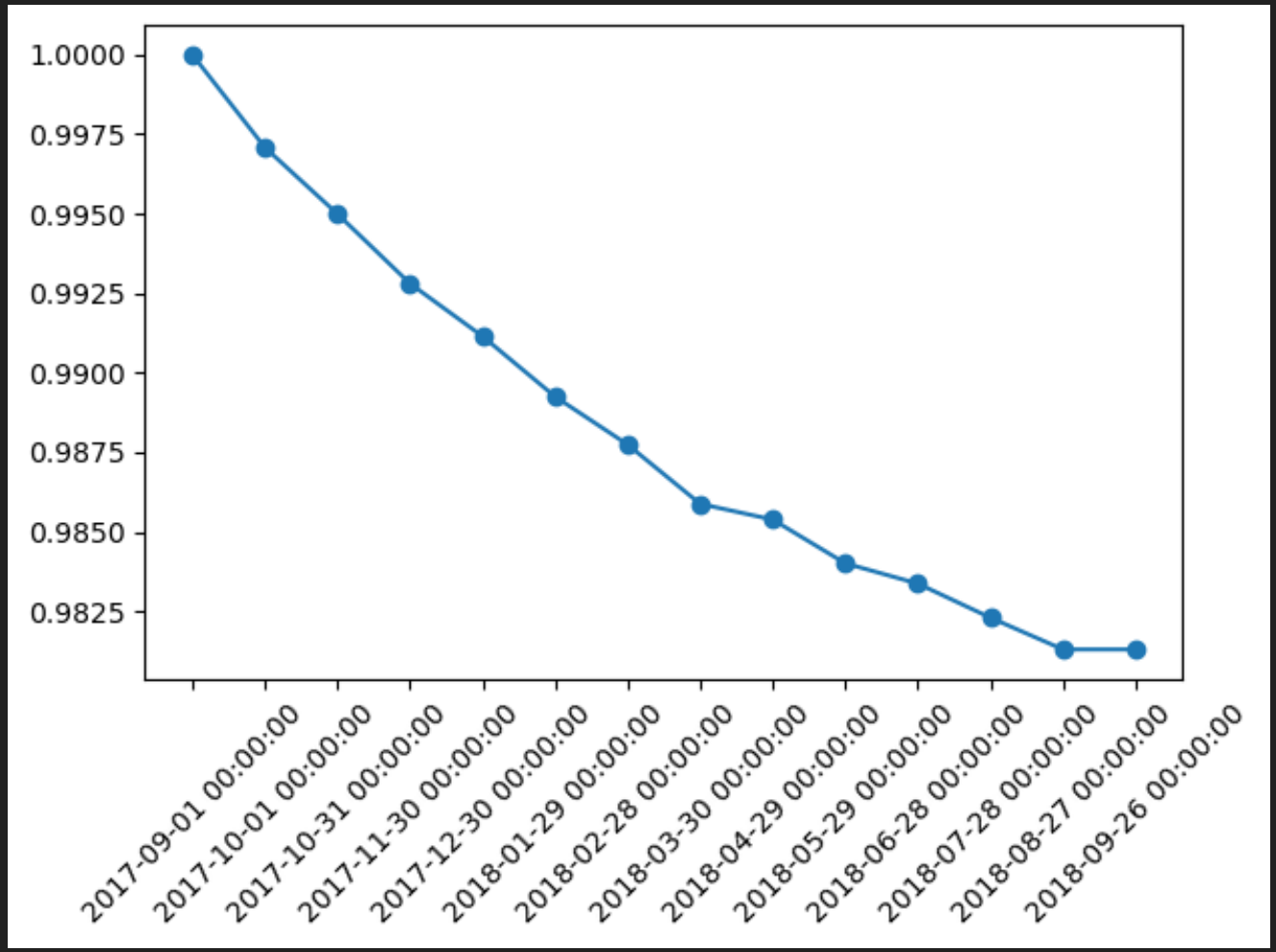


Part 4 : Clustering model testing

- **All clusters are represented by one purchase only customers**
- **Cluster 0: *70% of customers***
  - The customers from this group give very good review_score but they don't spend much and use small fractional payment. They are ancient.
  - They buy in all categories except high tech and electronics.
- **Cluster 1: *24.7% of customers***
  - The customers from this group give good review_score but they don't spend much and no fractional payment. They are ancient.
  - They buy only high tech and electronics.
- **Cluster2: 5.3% of customers**
  - The customers from this group give bad review_score but they spend a high amount of money and use fractional payment. They are recent.
  - They buy in all categories except high tech and electronics. They buy more products in the home, garden, construction, stationery categories than the other customers.

Part 5 : Stability of the segmentation over time

- We take the database at date = 01/09/2017 and follow the segmentation over time
- The model used is PCA(n_components = 5) + KMeans(K = 3)



Part 6 : Conclusion

- The best model is PCA+Kmeans for three clusters with a silhouette score of 0.67.
- The segmentation consists in 3 groups. We note that in all groups there are only one-purchase only customers :
  - Cluster 0 : 70% of the customers. They are ancient customers buying cheap products using small fractional payment and buying in all categories of products except for high tech and electronics.
  - Cluster 1 : 24.7% of the customers. They are like in the first cluster but they only buy high tech and electronics.
  - Cluster 2 : 5.3% of the customers. They make expensive purchases but give bad reviews. They buy in all categories mostly products for the house in general and don't buy electronics and high tech products
- The segmentation doesn't change much over time. After one year only 2% have moved to another segment. This is due to the very high proportion of one purchase only customers (97%)
- We recommend a maintenance of the segmentation only once a year. A « marketing » focus should be made on the cluster 2 where the dissatisfaction is big and the amount spent is big as well.