E-commerce platform in Brazil

Azadeh POHIER

Oct. 2021

Contents

Introduction

Olist is the largest department store in Brazilian marketplaces. Olist connects small businesses from all over Brazil to channels without hassle and with a single contract. Those merchants are able to sell their products through the Olist Store and ship them directly to the customers using Olist logistics partners.

After a customer purchases the product from Olist Store a seller gets notified to fulfill that order. Once the customer receives the product, or the estimated delivery date is due, the customer gets a satisfaction survey by email where he can give a note for the purchase experience and write down some comment.

Olist wants to provide their e-commerce teams with customer segmentation that they can use on a daily basis for their communication campaigns.

Our goal is to understand the different types of users through their behavior and their personal data.

We will need to provide the marketing team with an actionable description of your segmentation and its underlying logic for optimal use, as well as a maintenance contract proposal based on an analysis of the stability of the segments over time.

Finally, our client, Olist, specified their request as follows:

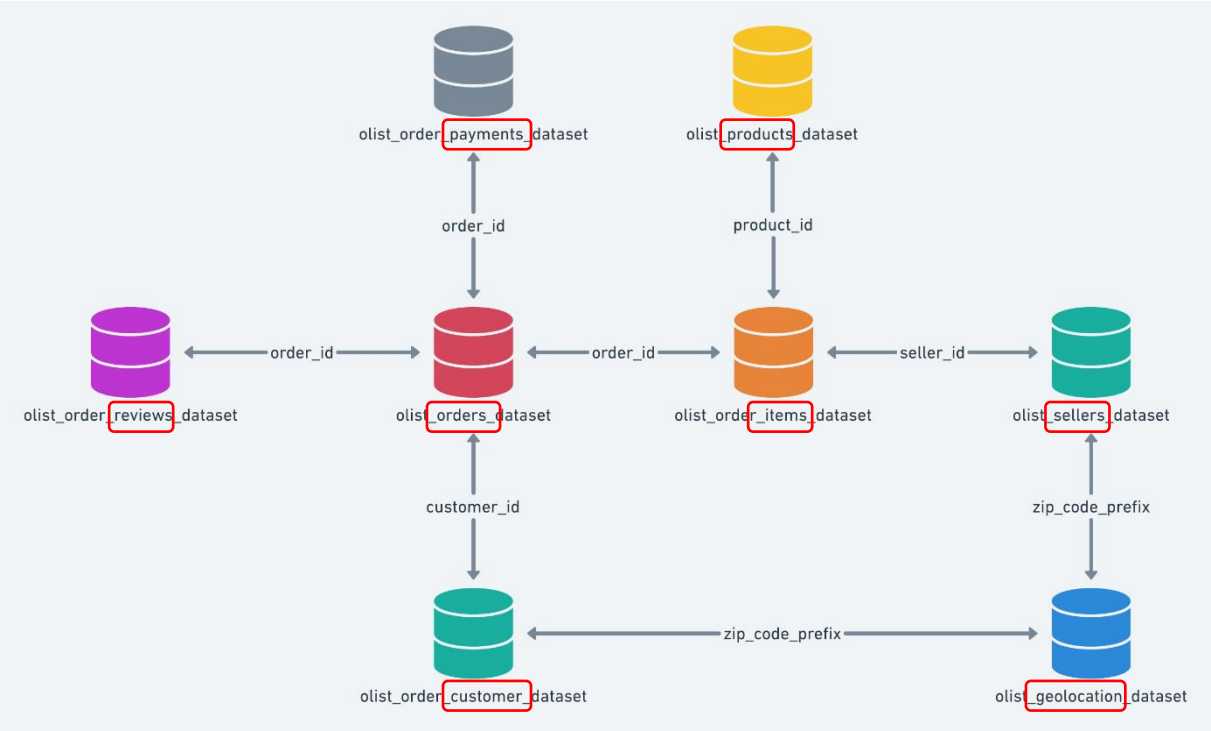
- The proposed segmentation must be actionable and easy to use for the marketing team.
- We will assess how often the segmentation needs to be updated, in order to be able to perform a maintenance contract quote.
- The code provided must respect the PEP8 convention, to be usable by Olist.

Data is taken from kaggle website:

https://www.kaggle.com/olistbr/brazilian-ecommerce

Data discovery

Data Schema



After analysing all csv files, using this schema and the merge() function to merge all the datasets together in one csv file.

Explanation of all data:

Customers Dataset

This dataset has information about the customer and its location. Use it to identify unique customers in the orders dataset and to find the orders delivery location.

At our system each order is assigned to a unique customer*id.* This means that the same customer will get different ids for different orders. The purpose of having a *customer*unique_id on the dataset is to allow you to identify customers that made repurchases at the store. Otherwise you would find that each order had a different customer associated with.

Geolocation Dataset

This dataset has information Brazilian zip codes and its lat/lng coordinates. Use it to plot maps and find distances between sellers and customers.

Order Items Dataset

This dataset includes data about the items purchased within each order.

Example:

The order_id = 00143d0f86d6fbd9f9b38ab440ac16f5 has 3 items (same product). Each item has the freight calculated accordingly to its measures and weight. To get the total freight value for each order you just have to sum.
The total order_item value is: 21.33 * 3 = 63.99
The total freight value is: 15.10 * 3 = 45.30
The total order value (product + freight) is: 45.30 + 63.99 = 109.29

Payments Dataset

This dataset includes data about the orders payment options.

Order Reviews Dataset

This dataset includes data about the reviews made by the customers.

After a customer purchases the product from Olist Store a seller gets notified to fulfill that order. Once the customer receives the product, or the estimated delivery date is due, the customer gets a satisfaction survey by email where he can give a note for the purchase experience and write down some comments.

Order Dataset

This is the core dataset. From each order you might find all other information.

Products Dataset

This dataset includes data about the products sold by Olist.

Sellers Dataset

This dataset includes data about the sellers that fulfilled orders made at Olist. Use it to find the seller location and to identify which seller fulfilled each product.

Category Name Translation

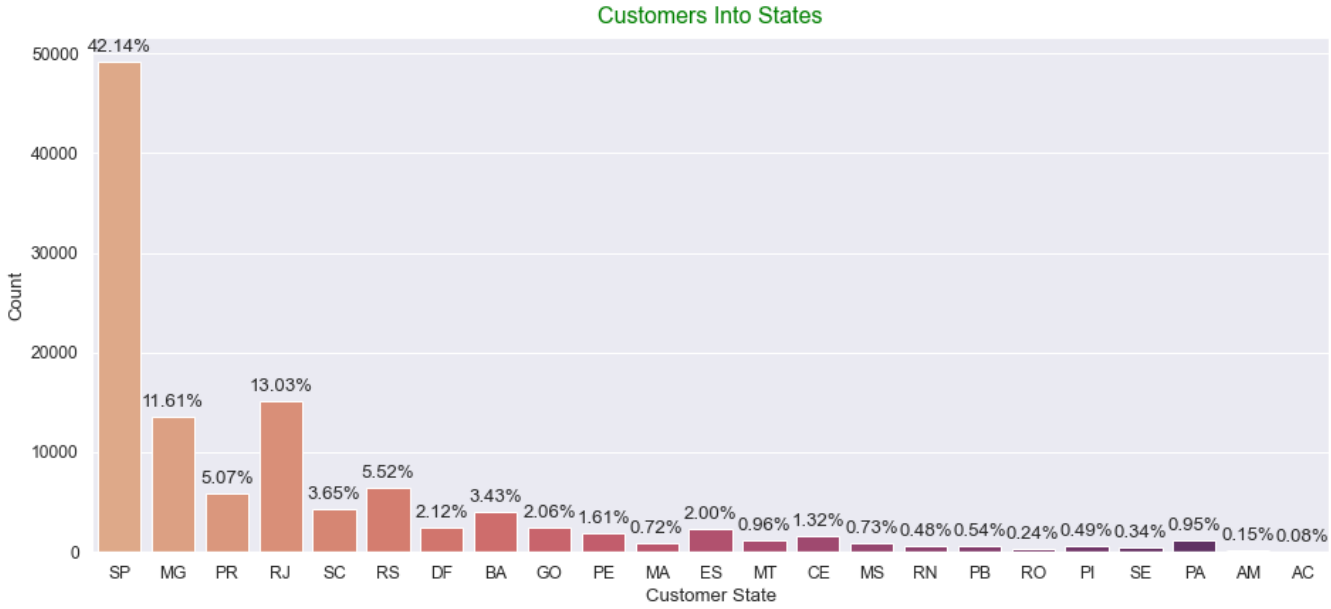
Translates the product*category*name to english.

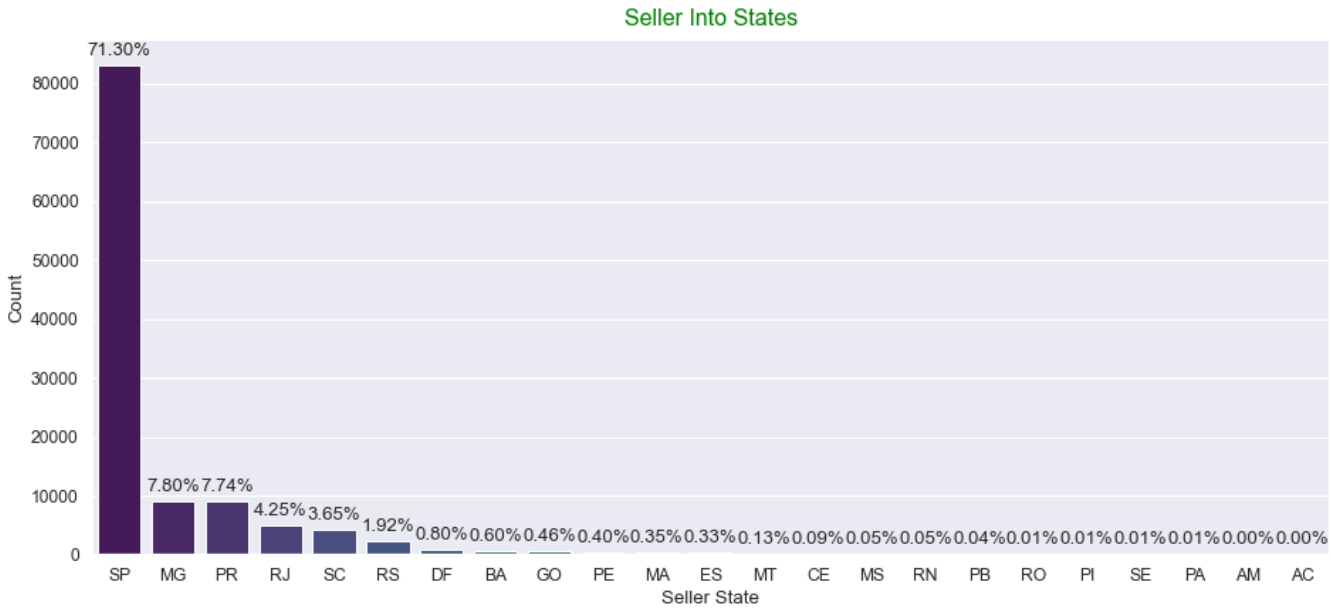
Exploratory Data Analyse

Location map:

From the geolocation dataset we have plotted this map which show us almost all sellers and buyers are in Brazil and some of them are in Portugal & Spain, etc. probably that is why the shipping tax is different.

Ecommerce by Region, State:





The city of Sao Paulo has the highest number of customers & sellers, after that the two cities Minas Gerais and Rio de Janeiro respectively have the highest number in total.

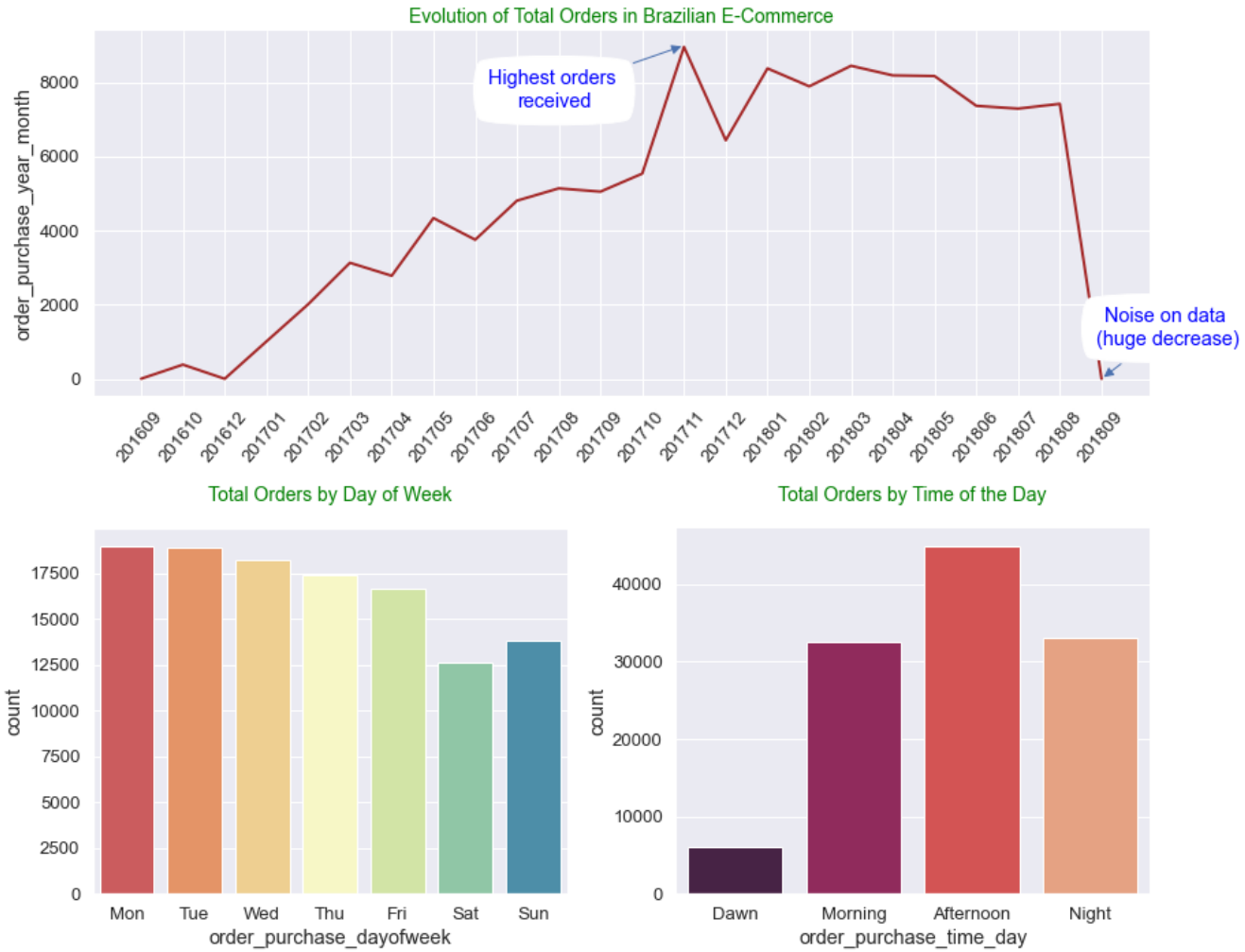
Order Status:

A majority of the orders are delivered while some are in transit.

Order purchase:

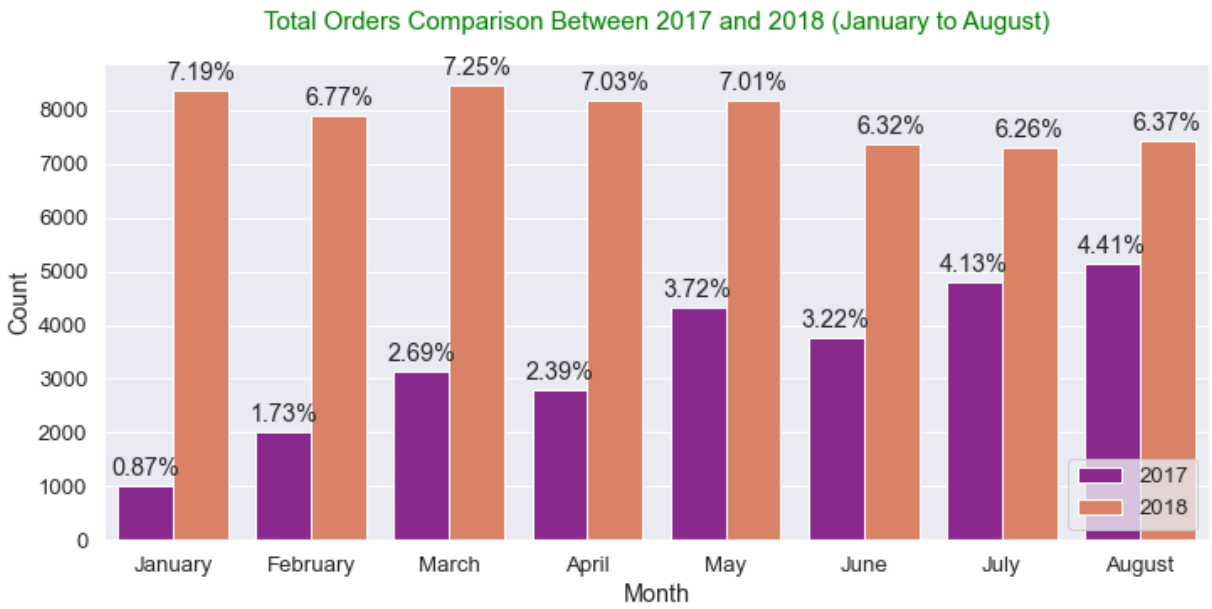
To visualise our data by the date we need to do some feature engineering

```
# Extracting attributes for purchase date - Year and Month
data['order_purchase_year'] = data['order_purchase_timestamp'].apply(lambda x: x.year)
data['order_purchase_month'] = data['order_purchase_timestamp'].apply(lambda x: x.month)
data['order_purchase_month_name'] = data['order_purchase_timestamp'].apply(lambda x: x.strftime('%b'))
data['order_purchase_year_month'] = data['order_purchase_timestamp'].apply(lambda x: x.strftime('%Y%m'))
data['order_purchase_date'] = data['order_purchase_timestamp'].apply(lambda x: x.strftime('%Y%m%d'))
```



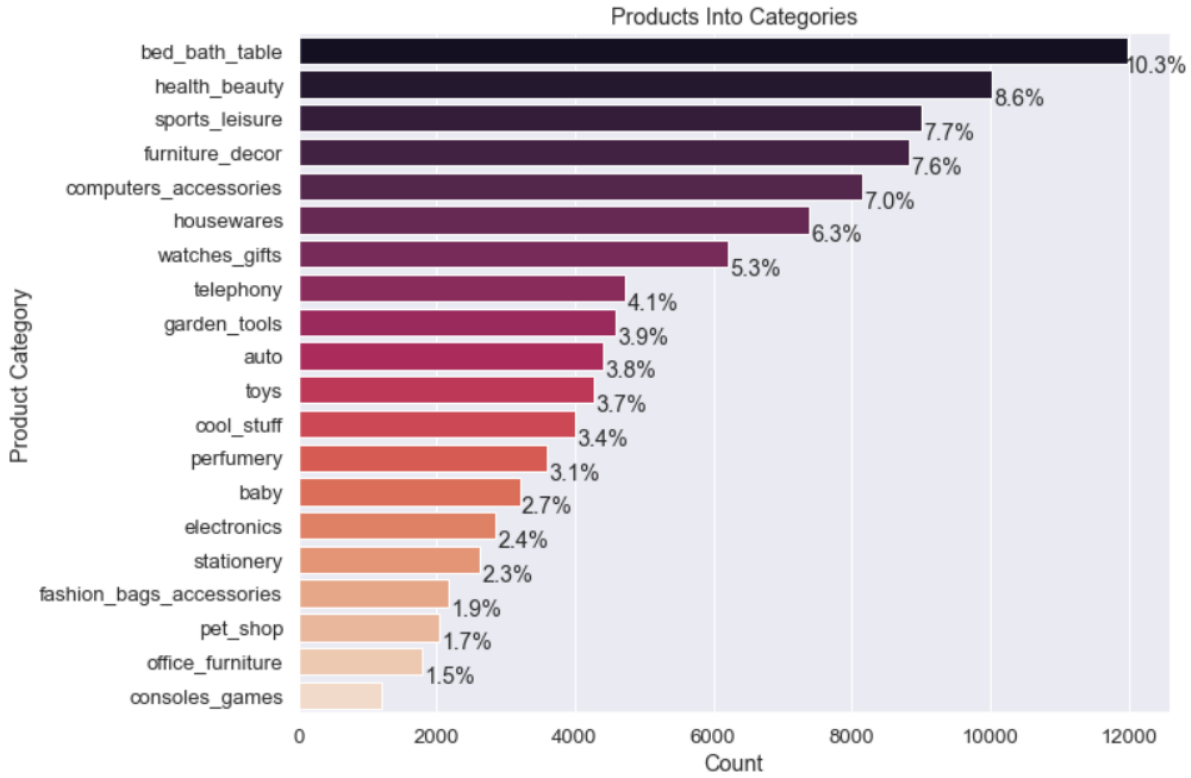
- E-commerce on Brazil has a growing trend along the time.
- We can see a peak on November due to Black Friday.
- Sales are weak on Dec (holidays), but in general customers are buying more than before.
- Monday's and Tuesday's are the most preferred days for customers and they buy more in the afternoons.
- As we can see a sharp decrease between August and September 2018, that maybe because of noise in data.

Comparison of orders between 2017 & 2018:



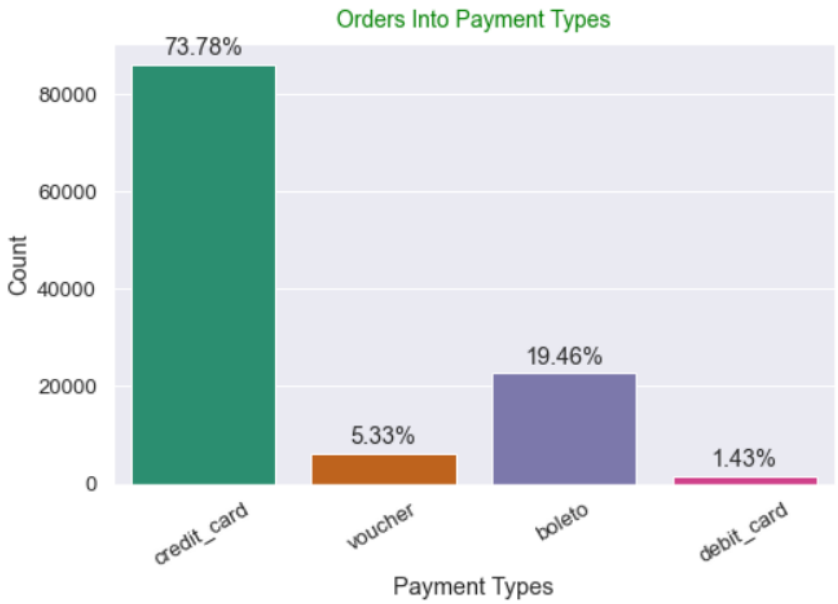
To compare 2017 and 2018, from Jan to Aug, we had more order in 2018 (around 143% more) but from Jan to Aug 2018 the number of order is going to decrease.

Product:



Here we see the first 20 interesting products for our customers.

Payment:



Most of the customers are paid by credit card and around 20% by boleto.

Review score:

Most of the client are happy and gave the positive review score (review score >= 3). Note that, generally the customer who are not happy give their comment and score surely.

Feature engineering

In this part we calculated from geolocation dataset:

- The distance between seller and customers

And for RFM segmentation:

- Number of days since the last order
- The sum of price
- The sum of delivery cost
- Total purchase

Customer segmentation (RFM)

Customer segmentation is important for multiple reasons. We get a deeper knowledge of our customers and can tailor targeted marketing campaigns.

The **RFM** method was introduced by Bult and Wansbeek in 1995 and has been successfully used by marketers since. It analyzes customers' behavior on three parameters:

- **Recency**: How recent is the last purchase of the customer.
- **Frequency**: How often the customer makes a purchase.
- **Monetary**: How much money does the customer spends.

The advantages of RFM is that it is easy to implement and it can be used for different types of business. It helps craft better marketing campaigns and improves CRM and customer's loyalty.

The disadvantages are that it may not apply in industries where customers are usually one time buyers. It is based on historical data and won't give much insight about prospects.

Methodology

To get the RFM score of a customer, we need to first calculate the R, F and M scores on a scale from 1 (worst) to 4 (best).

1. Calculate Recency = number of days since last purchase
2. Calculate Frequency = number of purchases during the studied period (usually one year)
3. Calculate Monetary = total amount of purchases made during the studied period
4. Find quintiles for each of these dimensions
5. Give a grade to each dimension depending in which quintiles it stands
6. Combine R, F and M scores to get the RFM score or class
7. Map RF scores to segments

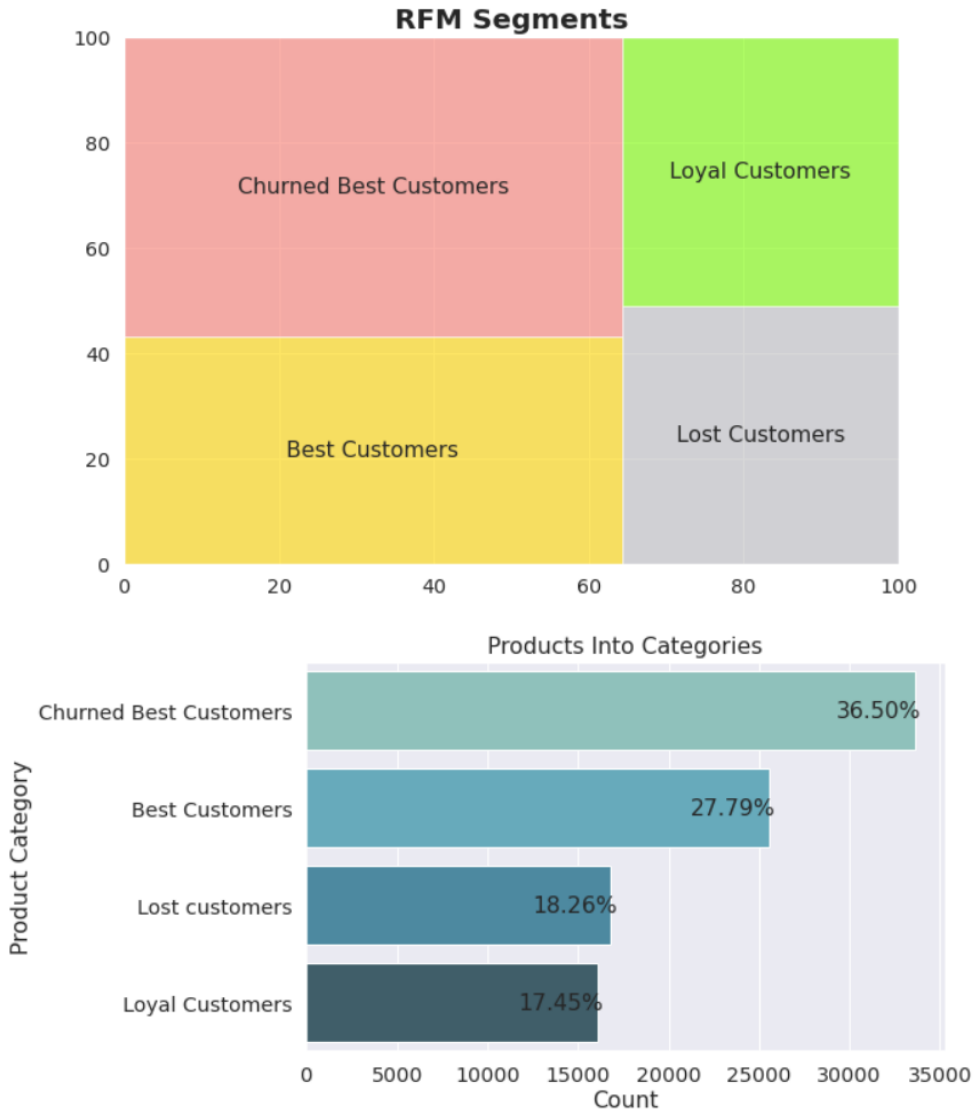
	customer_unique_id	Recency	Frequency	Monetary	R_Quartile	F_Quartile	M_Quartile	RFMClass
0	0000366f3b9a7992bf8c76cfdf3221e2	112	1	141.90	1	4	2	142
1	0000b849f77a49e4a4ce2b2a4ca5be3f	115	1	27.19	2	4	4	244
2	0000f46a3911fa3c0805444483337064	538	1	86.22	4	4	3	443
3	0000f6ccb0745a6a4b88665a16c9f078	322	1	43.62	3	4	4	344
4	0004aac84e0df4da2b147fca70cf8255	289	1	196.89	3	4	1	341

It is helpful to assign names to segments of interest. Here are just a few examples to illustrate:

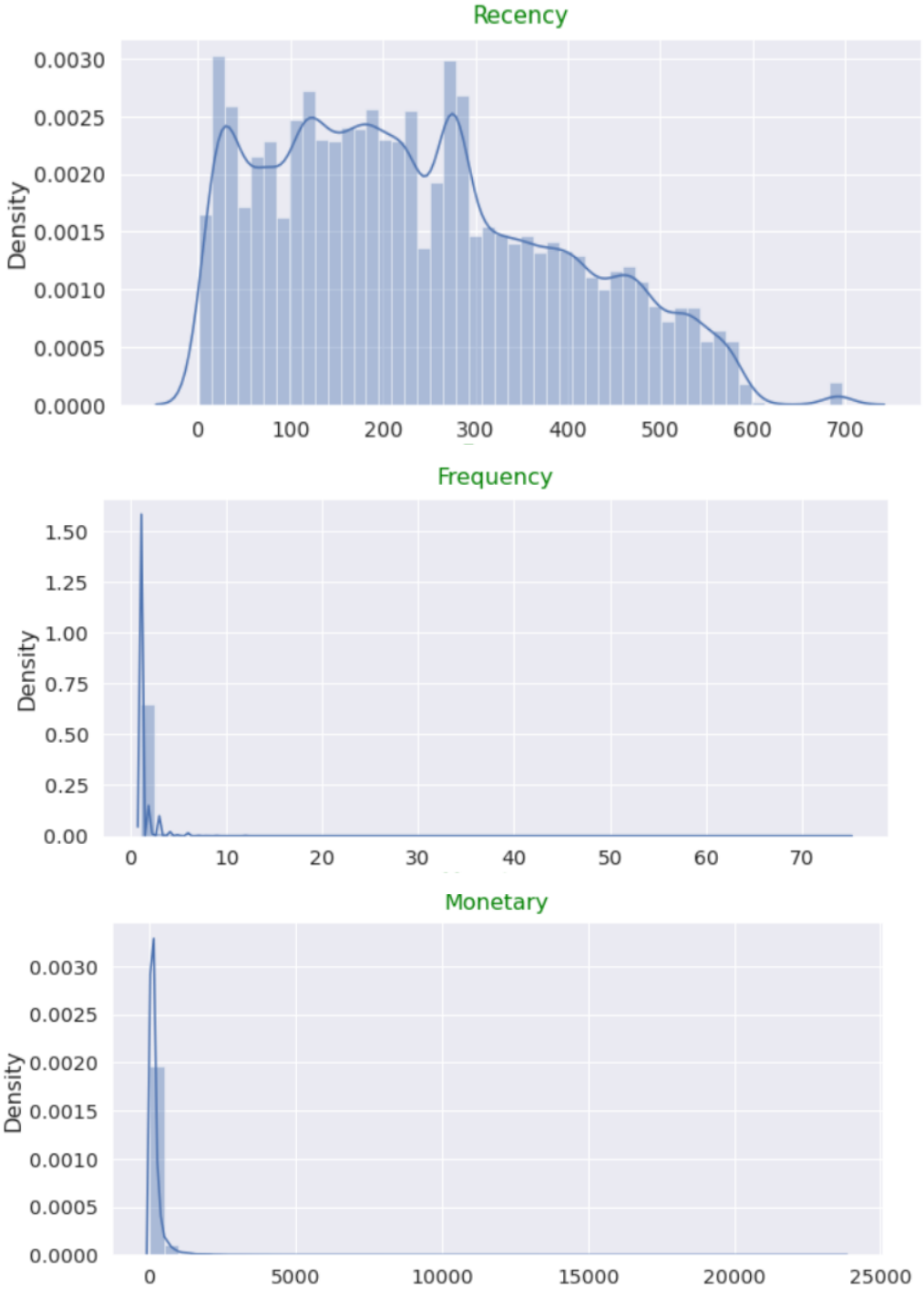
- **Best Customers** – This group consists of those customers who are found in **R-Tier-1, F-Tier-1 and M-Tier-1**, meaning that they transacted recently, do so often and spend more than other customers. A shortened notation for this segment is 1-1-1; we'll use this notation going forward.
- **Loyal Customers** – This group consists of those customers in segments **1-1-3 and 1-1-4** (they transacted recently and do so often, but spend the least).
- **Churned Best Customers** – This segment consists of those customers in groups **4-1-1, 4-1-2, 4-2-1 and 4-2-2** (they transacted frequently and spent a lot, but it's been a long time since they've transacted).
- **Lost customers** –The customers who were not spending so much and were absent for a long time **4-4-4, 4-3-4**.

Marketers should assemble groups of customers most relevant for their particular business objectives and retention goals.

RFM map:



RFM distribution:

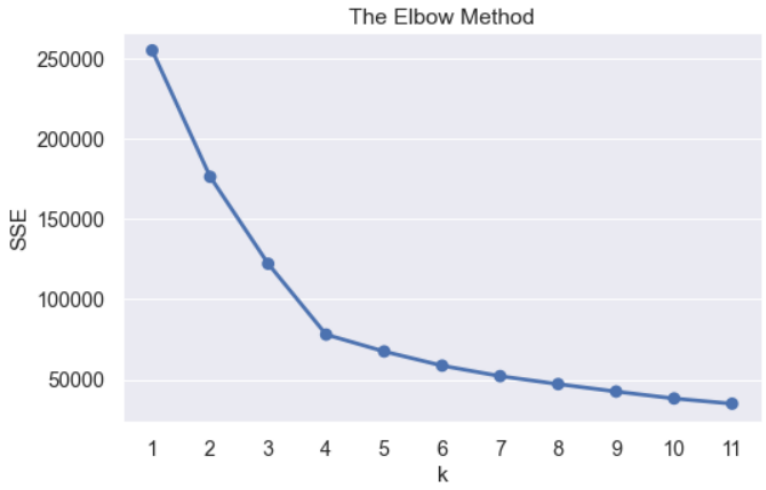


Clustering Algorithms

Before clustering we need to normalized our dataset. To do that we are using StandardScaler() function. Then we use two different non supervised machine learning models “K-Mean & DBSCAN” to classify our client.

1. K-mean

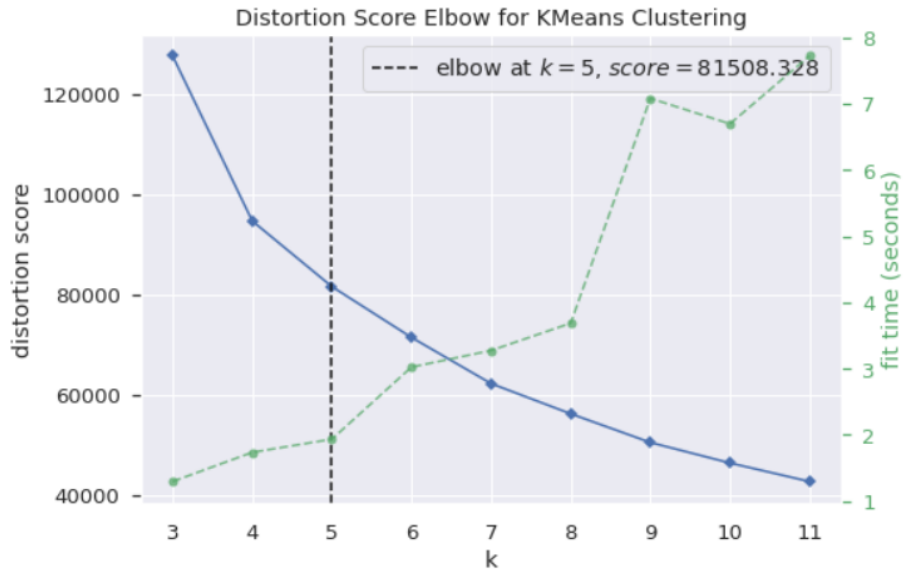
K-means clustering is one of the simplest and popular unsupervised machine learning algorithms. The K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible. The *‘means’* in the K-means refers to averaging of the data; that is, finding the centroid.



Evaluation clustering algorithms

To evaluating the number of cluster for our model we need to use Yellowbrick — Clustering Evaluation method.

- K-Elbow plot

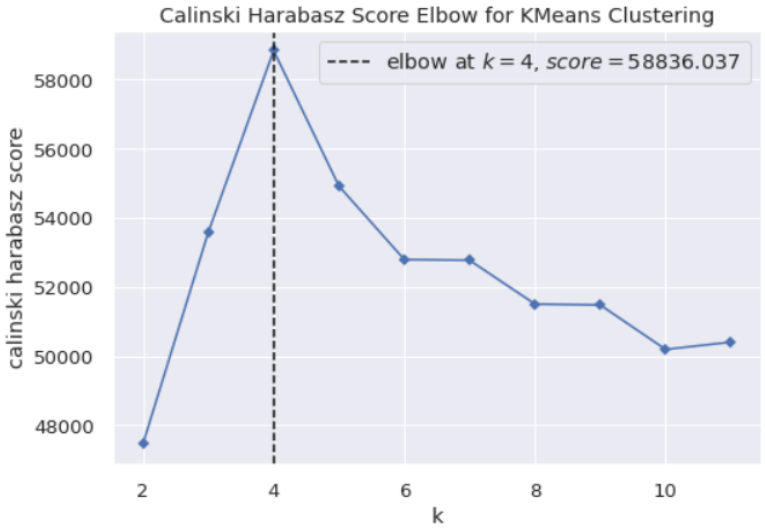


- Calinski harabasz plot

CalinskiHarabaszEvaluation is an object consisting of sample data, clustering data, and Calinski-Harabasz criterion values used to evaluate the optimal number of clusters.

Performance based on average intra and inter-cluster SSE (Tr):

$$s(k) = \frac{\mathrm{Tr}(B_k)}{\mathrm{Tr}(W_k)} \times \frac{N-k}{k-1}$$



From Elbow method the optimal cluster is equal to 5 and from the method Calinski Harabasz , the optimal cluster is equal to 4. Now try to check them with silhouette method.

- Silhouette Visualizer

Silhouette: Validates performance based on intra and inter-cluster distances:

$$S = \frac{1}{N}\sum_{i=0}^{N}\frac{b_i - a_i}{max(a_i, b_i)}$$

Silhouette analysis can be used to evaluate the density and separation between clusters. The score is calculated by averaging the silhouette coefficient for each sample, which is computed as the difference between the average intra-cluster distance and the mean nearest-cluster distance for each sample, normalized by the maximum value. This produces a score between -1 and +1, where scores near +1 indicate high separation and scores near -1 indicate that the samples may have been assigned to the wrong cluster.

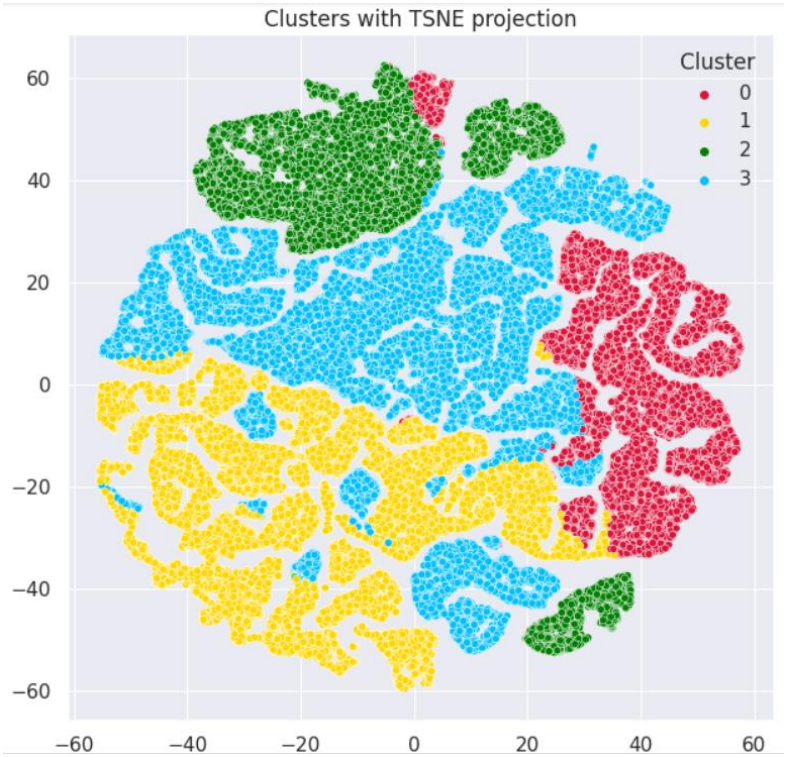
The Elbow Method is more of a decision rule, while the Silhouette is a metric used for validation while clustering. Thus, it can be used in combination with the Elbow Method. Therefore, the Elbow Method and the Silhouette Method are not alternatives to each other for finding the optimal K. Rather they are tools to be used together for a more confident decision.

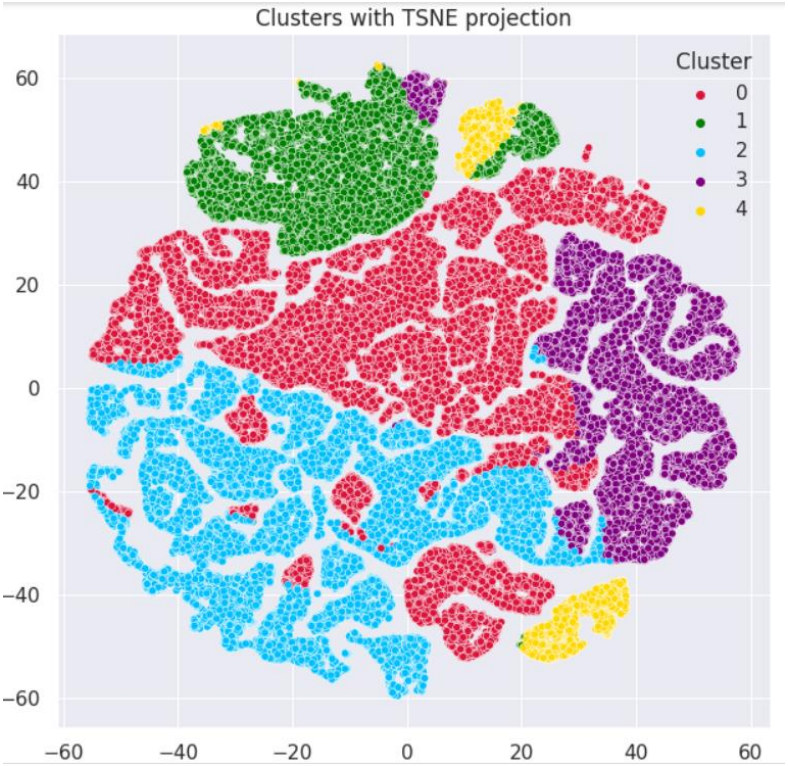
In both graphs, silhouette coefficient are equal to 0.37 and as you can see the width of one cluster is too tiny in second plot. Going to try with TSNE method to find our best classification number.

TSNE

T-Distributed Stochastic Neighbor Embedding (t-SNE) is an unsupervised, nonlinear technique primarily used for data exploration and visualizing high-dimensional data.

Here we have visualization of our dataset in 4 and 5 cluster. After comparing both of them we continue with 4 cluster.





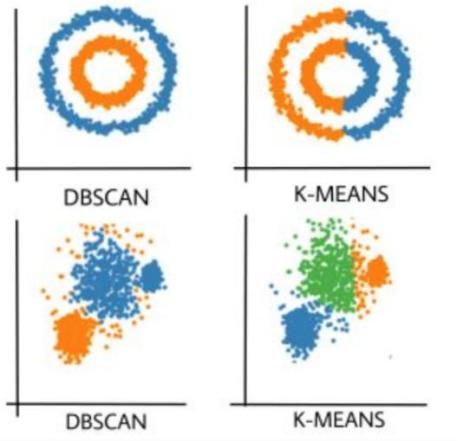
2. DBSCAN

DBSCAN stands for density-based spatial clustering of applications with noise. It is able to find arbitrary shaped clusters and clusters with noise (i.e. outliers). The main idea behind DBSCAN is that a point belongs to a cluster if it is close to many points from that cluster.

After applying DBSCAN algorithm in our dataset we received the result below,

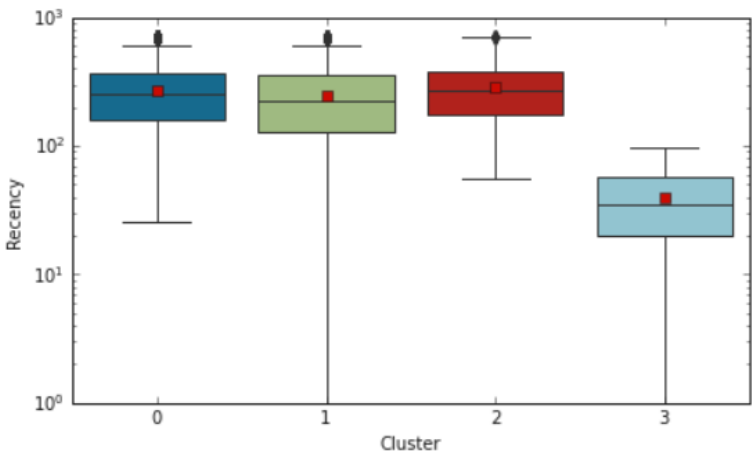
```
Estimated no. of clusters: 21
Estimated no. of noise points: 445
```

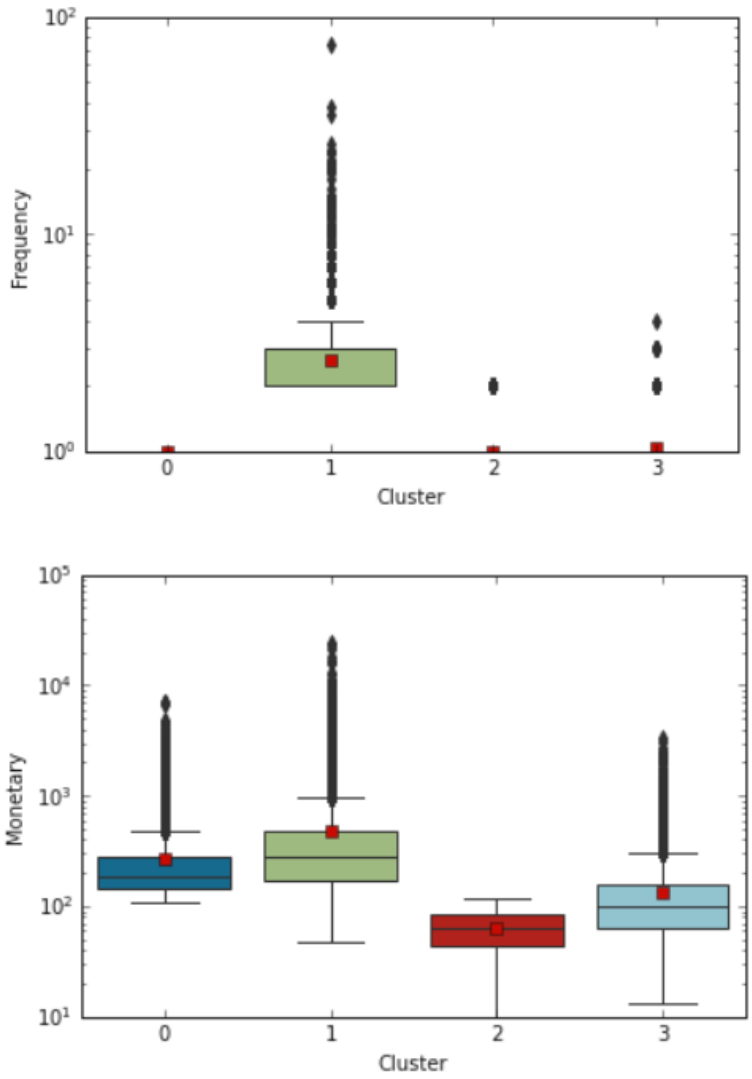
Actually the optimal number of cluster with this model is 21 which is a lot for our marketing and the information of 445 of our customers are incorrect in this dataset.



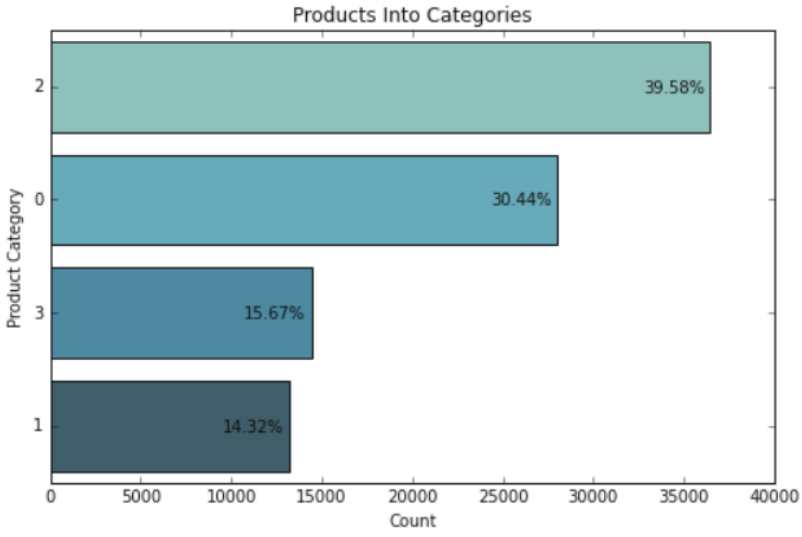
Cluster of customers

By comparison of RFM's mean we give the name to our each cluster.





- cluster 0: churned best customers(high monetary & recency)
- cluster 1: best customers (High monetary & frequency)
- cluster 2: lost customers (less monetary & high recency)
- cluster 3: loyal customers (less monetary & recency)



Temporal stability

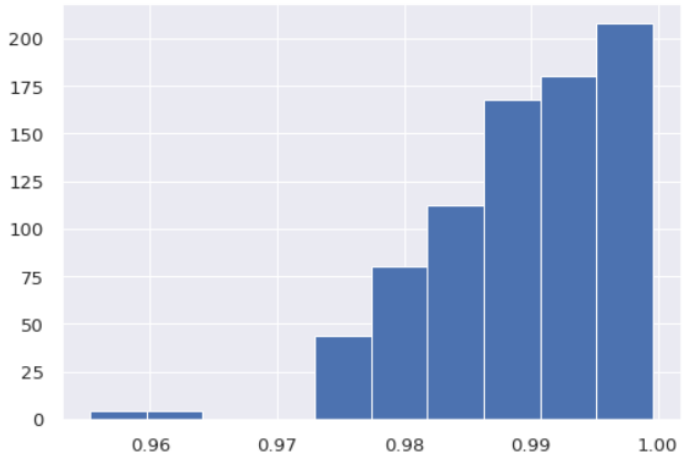
Temporal stability refers to the stability of the content of reports across the time. It is conceptually similar to test-retest reliability, but is primarily a function of intervening events rather than of random cognitive errors.

ARI

The Adjusted Rand score is introduced to determine whether two cluster results are similar to each other. In the formula, the “RI” stands for the rand index, which calculates a similarity between two cluster results by taking all points identified within the same cluster.

```
ARI = (RI - Expected_RI) / (max(RI) - Expected_RI)
```

The adjusted Rand index have a value close 1.0 so the clustering is identical and recovery is excellent.



For the stability of cluster we used ARI (calculate the model vs predict) from Jun 2017 to Jul 2018 for each two months period.

As you can see, the AR score is going down in 4th month and 12th month so we need to update our model every 8 months.

Conclusion:

- **RFM** method helped us to segment our customers in four different groups.
- **K-mean** algorithm worked well for us in this dataset with 4 clutters.
- For **temporal stability,** we must update our model every 8 months.

Possible to improve:

- Repeat the study while keeping the outliers
- Added new features (gender, age)