Customer Segmentation in E-commerce

W6W7W8 - Python Advanced Assignment

by Annita

RevoU FSDA Section Madrid - Team 5

TABLE OF CONTENTS

01

BUSINESS OVERVIEW

Describing business problem and project objectives

OVERVIEW PROBLEM

An e-commerce startup based in Brazil recently opened an online website to sell their product. They launch the website when the Covid-19 hits and making them grow faster than ever. But, the startup is still not using targeted marketing which hurts their marketing budget as only a fraction of their user comes back to their website.

OBJECTIVES

The goal is to increase their marketing conversion rate by doing customer segmentation analysis to understand the customer's behaviour and planning targeted marketing strategy so that it will not hurt the budget anymore.

02

EDA

Understanding current business performance

DELIVERY TIME

Customer made purchase

It takes around 2 ~ 3 days in general for the e-commerce to process the customer's order before they hand it to carrier.

Carrier is on its way

It takes around one week in general for the carrier to deliver the order to customer's address.

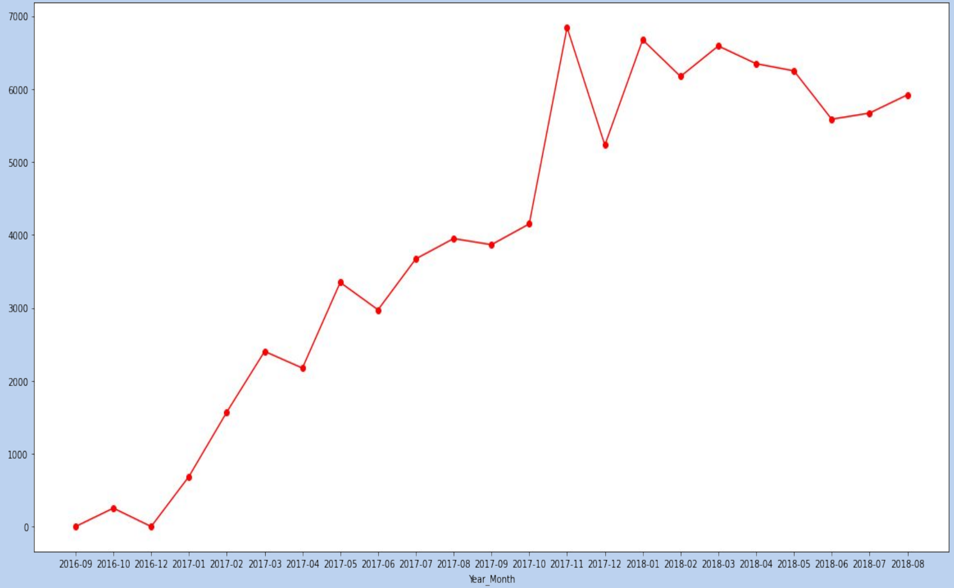
The order is finally delivered

In general, it takes around 9 ~ 10 days since customer's purchase date until the order arrive to the customer's address. Based on estimation time, it takes around 23 days to finish an order.

However, there is still around 7.98% overdue delivered orders.

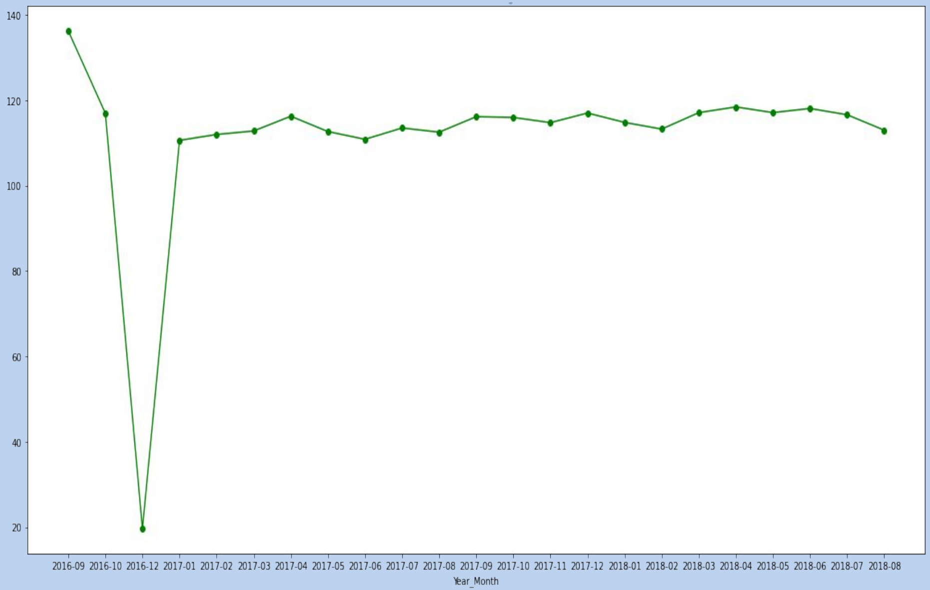
BUSINESS PERFORMANCE

Total Orders in Sept 2016 to Aug 2018



From Sept 2016 to Nov 2017, number of order tend to increase. Largest increment happened from Oct 2017 to Nov 2017 (need investigation), but it decreases in Dec 2017. Start from Jan 2018, number of order increases from last month, but until Aug 2018, they tend to decrease slowly.

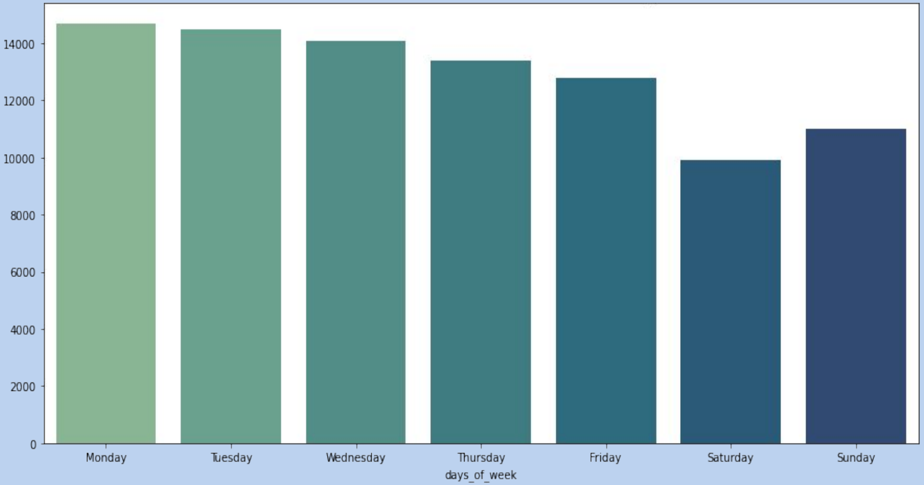
AOV in Sept 2016 to Aug 2018



Average Order Value is quite stable for each month (around 110). AOV in Sept 2016 and Dec 2016 are fluctuate because total non-canceled or unavailable order is only one order.

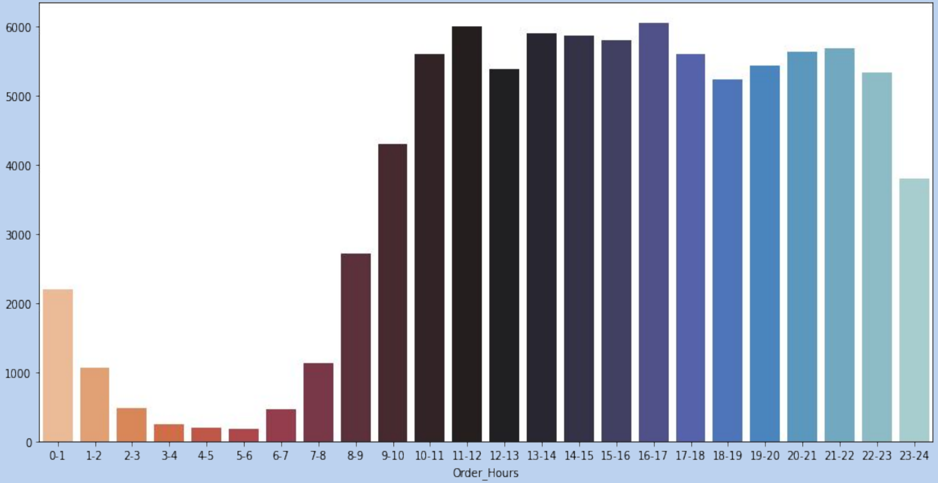
BUSINESS PERFORMANCE

Busiest Day in A Week



From 2016 to 2018, total order in weekday (Monday - Friday) is greater than weekend (Saturday & Sunday). Which means that the customer tend to buy in weekday rather than weekend. Besides, Monday and Tuesday become the busiest days in a week for three years.

Busiest Hour in A Day



From 2016 to 2018, most of the customers do purchase in 10 AM to 11 PM. Highest total orders happened in 11-12 and 16-17.

**Busiest Day and Hour are based on number of order for three years.*

OUR CUSTOMER

Top 10 States with Most Customer

São Paulo
(42.56% of customer, ARPC = 107)

Rio de Janeiro
(12.91% of customer, ARPC = 120.52)

Minas Gerais
(11.76% of customer, ARPC = 121.06)

Rio Grande do Sul
(5.5% of customer, ARPC = 121.88)

Paraná
(5.06% of customer, ARPC = 116.6)

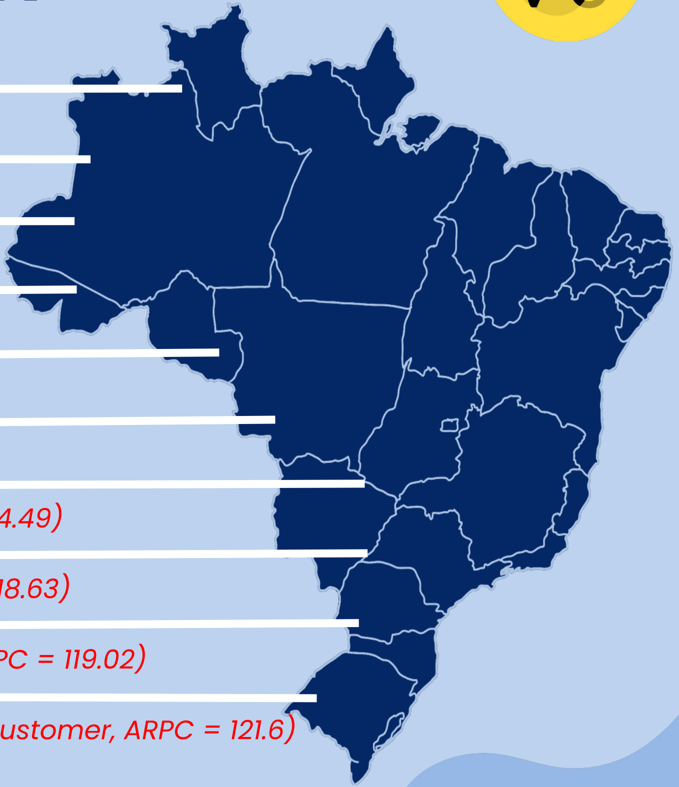
Santa Catarina
(3.64% of customer, ARPC = 124.64)

Bahia
(3.34% of customer, ARPC = 124.49)

Distrito Federal
(2.16% of customer, ARPC = 118.63)

Espírito Santo
(2.06% of customer, ARPC = 119.02)

Goiás
(2.01% of customer, ARPC = 121.6)



Our customers come from **27** different states, but **91%** of our customers come from **Top 10 States** above. Although São Paulo has the biggest percentage of customers, the Average Revenue per Country is the lowest among the other 26 states.
There are 4 payment methods, but most of our customers love to pay with **credit card** and **boleto**. (The others are voucher and debit card)

RECOMMENDATIONS

Based on results of Exploratory Data Analysis, there are several recommendations that can be given :

- In order to increase Average Order Value, we can try cross sell complementary product and upsell our products or even provide bundle deals, this will encourage customers to buy more complete products (and more expensive of course), thus increasing our average order value.
- For efficiency cost and human resources, we can create an effective working schedule to allocate more workers on weekdays, especially Monday and Tuesday, and allocate less workers on weekend.
- We can run limited-time offers to boost sales in several busiest hours.
- For increasing Average Revenue per Country, we can create order minimums for free shipping. This will encourage customers to add more products to their carts, and if their amount of order surpasses certain amount, they will get free shipping coupon.
- Most of our customers are credit card or boleto users, we can reward our existing customers by giving discount or cashback for selected payment method. Not only that, we should pay more attention to our security and customer service in order to minimize payment failure and payment fraud.

03

CLUSTER ANALYSIS

Do cluster analysis for better targeting customer

CLUSTERING PROCESS

Data Preparation
Prepare the datasets for cluster analysis

Data Preprocessing
Checking outliers and scaling the numbers

Cluster Analysis
Determine cluster number and fit the data to model

Interpreting Results
See the behaviour for each cluster

Business Recommendations
Give recommendations for improve business

DATA PREPARATION

Import Datasets

Import three datasets that are going to be used in this project.

Clean & Merge

Handling unlogical, missing, duplicates, typos and outliers values for each datasets and merged them become one dataset.

Removing Data

Removing unused rows and columns.

Create Table

Create RFM Table for doing Cluster Analysis

CREATE RFM TABLE

Recency

It shows time since last order from customer.

	customer_unique_id	recency
0	0000366f3b9a7992bf8c76cfdf3221e2	160
1	0000b849f77a49e4a4ce2b2a4ca5be3f	163
2	0000f46a3911fa3c0805444483337064	585
3	0000f6ccb0745a6a4b88665a16c9f078	369
4	0004aac84e0df4da2b147fca70cf8255	336
...	...	...
87417	fffbf87b7a1a6fa8b03f081c5f51a201	293
87418	fffea47cd6d3cc0a88bd621562a9d061	310
87419	ffff371b4d645b6ecea244b27531430a	617
87420	ffff5962728ec6157033ef9805bacc48	168
87421	ffffd2657e2aad2907e67c3e9daecbeb	532

First, find the last date purchase for each customers. Then, calculate recency since last date purchase made in the e-commerce.

Frequency

It shows total number of transactions purchased by customer.

	customer_unique_id	order_id
0	0000366f3b9a7992bf8c76cfdf3221e2	1
1	0000b849f77a49e4a4ce2b2a4ca5be3f	1
2	0000f46a3911fa3c0805444483337064	1
3	0000f6ccb0745a6a4b88665a16c9f078	1
4	0004aac84e0df4da2b147fca70cf8255	1
...	...	...
87417	fffbf87b7a1a6fa8b03f081c5f51a201	1
87418	fffea47cd6d3cc0a88bd621562a9d061	1
87419	ffff371b4d645b6ecea244b27531430a	1
87420	ffff5962728ec6157033ef9805bacc48	1
87421	ffffd2657e2aad2907e67c3e9daecbeb	1

Find total number of transactions purchased by each customer.

Monetary

It shows transaction value that customer spends in total.

	customer_unique_id	payment_value
0	0000366f3b9a7992bf8c76cfdf3221e2	141.90
1	0000b849f77a49e4a4ce2b2a4ca5be3f	27.19
2	0000f46a3911fa3c0805444483337064	86.22
3	0000f6ccb0745a6a4b88665a16c9f078	43.62
4	0004aac84e0df4da2b147fca70cf8255	196.89
...	...	...
87417	fffbf87b7a1a6fa8b03f081c5f51a201	167.32
87418	fffea47cd6d3cc0a88bd621562a9d061	84.58
87419	ffff371b4d645b6ecea244b27531430a	112.46
87420	ffff5962728ec6157033ef9805bacc48	133.69
87421	ffffd2657e2aad2907e67c3e9daecbeb	71.56

Calculate total transaction value that customer spends in the e-commerce.

MERGE RFM TABLE

Recency

It shows time since last order from customer.

Frequency

It shows total number of transactions purchased by customer.

Monetary

It shows transaction value that customer spends in total.

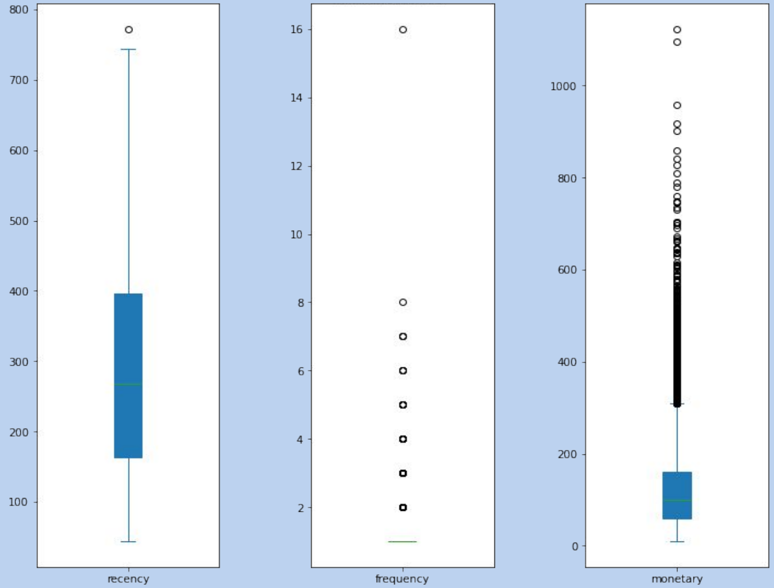
	customer_unique_id	recency	frequency	monetary
0	0000366f3b9a7992bf8c76cfdf3221e2	160	1	141.90
1	0000b849f77a49e4a4ce2b2a4ca5be3f	163	1	27.19
2	0000f46a3911fa3c0805444483337064	585	1	86.22
3	0000f6ccb0745a6a4b88665a16c9f078	369	1	43.62
4	0004aac84e0df4da2b147fca70cf8255	336	1	196.89
...	...	...	...	...
87417	fffbf87b7a1a6fa8b03f081c5f51a201	293	1	167.32
87418	fffea47cd6d3cc0a88bd621562a9d061	310	1	84.58
87419	ffff371b4d645b6ecea244b27531430a	617	1	112.46
87420	ffff5962728ec6157033ef9805bacc48	168	1	133.69
87421	ffffd2657e2aad2907e67c3e9daecbeb	532	1	71.56

```
# Join all the tables
RFM_table = pd.merge(pd.merge(df_recency,
df_frequency, how='inner'), df_monetary,
how='inner')
# Rename the column name
RFM_table.rename(columns={'order_id':'freq
uency',
'payment_value':'monetary'},inplace =
True)
RFM_table
```

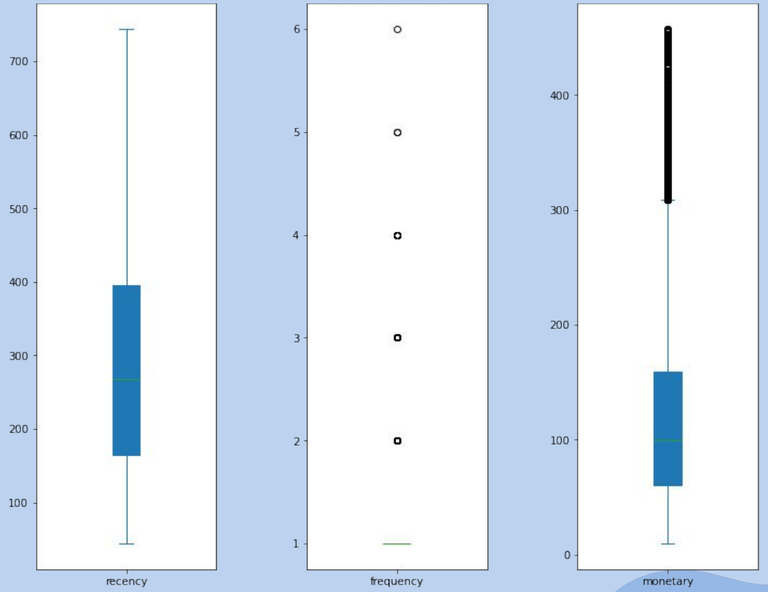
DATA PREPROCESSING

Checking Outliers

Before Treatment



After Treatment



For recency column, the outliers will be removed by Lower & Upper Inner Bound method (1.5*IQR).
For frequency column, the outliers are simply removed for frequency greater than 7.
For monetary column, the outliers will be removed by Lower & Upper Outer Bound method (3*IQR).

Total rows :
87422 rows → 87257 rows

DATA PREPROCESSING

Scaling The Numbers

Since each of numerical columns have different scale, so all of numerical values will be scaled by using MinMaxScaler method.

	recency	frequency	monetary
customer_unique_id			
0000366f3b9a7992bf8c76cfdf3221e2	0.165714	0.0	0.295361
0000b849f77a49e4a4ce2b2a4ca5be3f	0.170000	0.0	0.039289
0000f46a3911fa3c0805444483337064	0.772857	0.0	0.171064
0000f6ccb0745a6a4b88665a16c9f078	0.464286	0.0	0.075967
0004aac84e0df4da2b147fca70cf8255	0.417143	0.0	0.418118
...	...	...	...
fffbf87b7a1a6fa8b03f081c5f51a201	0.355714	0.0	0.352107
fffea47cd6d3cc0a88bd621562a9d061	0.380000	0.0	0.167403
ffff371b4d645b6ecea244b27531430a	0.818571	0.0	0.229641
ffff5962728ec6157033ef9805bacc48	0.177143	0.0	0.277034
ffffd2657e2aad2907e67c3e9daecbeb	0.697143	0.0	0.138338

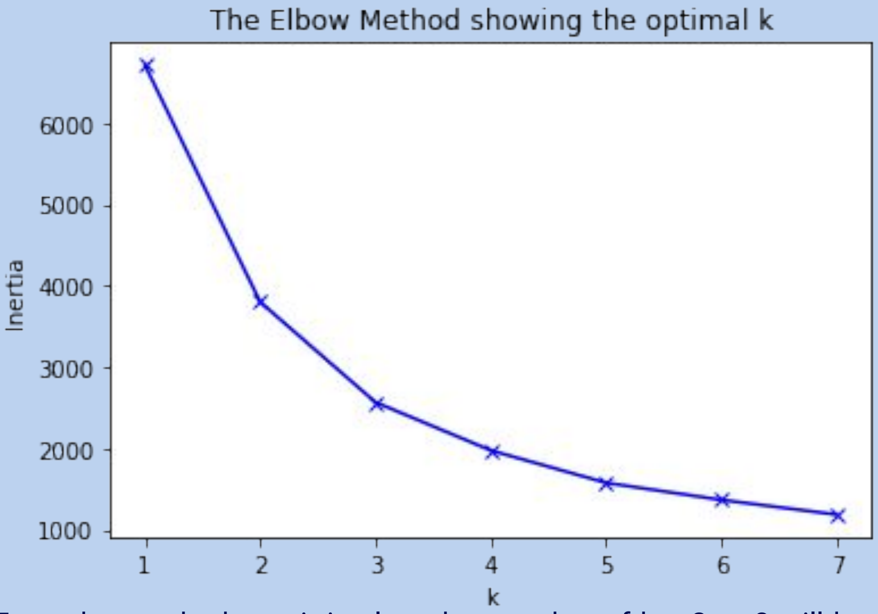
```python
# Import Library
from sklearn.preprocessing import MinMaxScaler

numerical_column = ['recency', 'frequency',
'monetary']
# Scale DataFrame by using MinMaxScaler
scaler = MinMaxScaler()
RFM_scale[numerical_column] =
scaler.fit_transform(RFM_scale[numerical_column])
RFM_scale
```

MinMax Scaler will shrinks the data within the given range, in this project, range from 0 to 1 will be used.

CLUSTER ANALYSIS

Determine Number of Cluster

Elbow Method and Silhouette Analysis will be used for determining optimal number of cluster.



Elbow Method

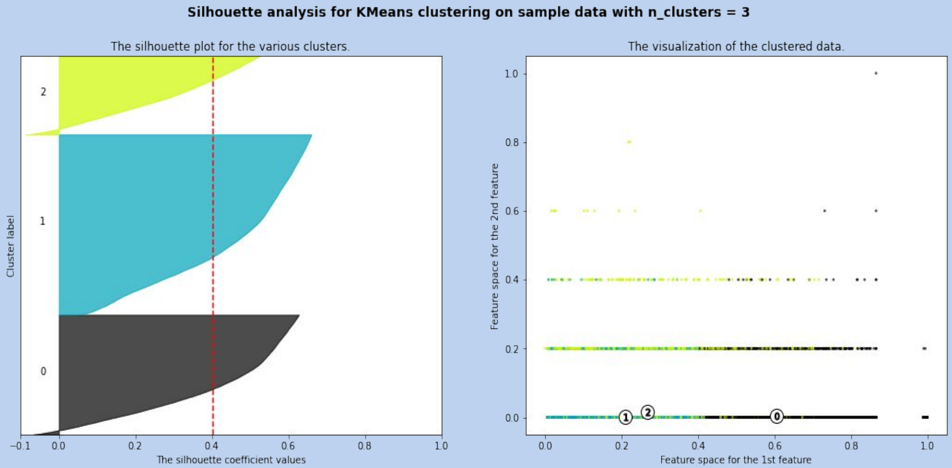
```
# Create Elbow Plot for Determining Number of Cluster
distortions = []
K = range(1,8)
for k in K:
    kmeanModel = cluster.KMeans(n_clusters=k)
    kmeanModel.fit(RFM_scale)
    distortions.append(kmeanModel.inertia_)

plt.figure(figsize=(15,10))
plt.figure()
plt.plot(K, distortions,'bx-')
plt.xlabel('k')
plt.ylabel('Inertia')
plt.title('The Elbow Method showing the optimal k')
plt.show()
```

From the graph above, it is clear that number of k = 2 or 3 will be the optimal for number of cluster. But for more accurate analysis, Silhouette Analysis will be performed.

CLUSTER ANALYSIS

Determine Number of Cluster

Elbow Method and Silhouette Analysis will be used for determining optimal number of cluster.




```
For n_clusters = 2 The average silhouette_score is : 0.3967991279610512
For n_clusters = 3 The average silhouette_score is : 0.404201834304504
For n_clusters = 4 The average silhouette_score is : 0.36158892503002943
For n_clusters = 5 The average silhouette_score is : 0.3733258747477178
```

Silhouette Analysis

```
# Import Library
from silhoutte import silhoutte_analysis

# Perform silhouette analysis for
determine the number of cluster
silhoutte_analysis(RFM_scale,[2,3,4,5])
```

From the Silhouette Score and Silhouette plot, k = 3 will be chosen as number of cluster because average silhouette score reach the highest when number of cluster is 3.

CLUSTER ANALYSIS

Fit Data Into Model

After determining the number of cluster, the next step is fit the data into cluster model.

customer_unique_id	recency	frequency	monetary	cluster
0000366f3b9a7992bf8c76cfdf3221e2	0.165714	0.0	0.295361	1
0000b849f77a49e4a4ce2b2a4ca5be3f	0.170000	0.0	0.039289	1
0000f46a3911fa3c0805444483337064	0.772857	0.0	0.171064	2
0000f6ccb0745a6a4b88665a16c9f078	0.464286	0.0	0.075967	2
0004aac84e0df4da2b147fca70cf8255	0.417143	0.0	0.418118	0
...	...	...	...	...
fffbf87b7a1a6fa8b03f081c5f51a201	0.355714	0.0	0.352107	0
fffea47cd6d3cc0a88bd621562a9d061	0.380000	0.0	0.167403	1
ffff371b4d645b6ecea244b27531430a	0.818571	0.0	0.229641	2
ffff5962728ec6157033ef9805bacc48	0.177143	0.0	0.277034	1
ffffd2657e2aad2907e67c3e9daecbeb	0.697143	0.0	0.138338	2

```python
# Import Library
from sklearn import cluster

cluster_model =
cluster.KMeans(n_clusters=3,random_state=2)
cluster_model.fit(RFM_fitmodel)
cluster_label = cluster_model.labels_
RFM_fitmodel['cluster'] = cluster_label
RFM_fitmodel
```

In this project, K-Means Clustering Method will be used.

CLUSTER ANALYSIS

Bring The Cluster To Data

After determining the number of cluster and fit them into the model, finally bring the cluster to the original data.

	customer_unique_id	recency	frequency	monetary	cluster
0	0000366f3b9a7992bf8c76cfdf3221e2	160	1	141.90	1
1	0000b849f77a49e4a4ce2b2a4ca5be3f	163	1	27.19	1
2	0000f46a3911fa3c0805444483337064	585	1	86.22	2
3	0000f6ccb0745a6a4b88665a16c9f078	369	1	43.62	2
4	0004aac84e0df4da2b147fca70cf8255	336	1	196.89	0
...	...	...	...	...	...
87252	fffbf87b7a1a6fa8b03f081c5f51a201	293	1	167.32	0
87253	fffea47cd6d3cc0a88bd621562a9d061	310	1	84.58	1
87254	ffff371b4d645b6ecea244b27531430a	617	1	112.46	2
87255	ffff5962728ec6157033ef9805bacc48	168	1	133.69	1
87256	ffffd2657e2aad2907e67c3e9daecbeb	532	1	71.56	2

By having the cluster information, the next step is interpreting the descriptive statistics for each columns to understand the clusters' behaviour.

INTERPRETING RESULTS

Descriptive Statistics

By performing descriptive analysis, each clusters' behaviour will be interpreted in order to give suitable recommendations.

	Recency								Frequency							Monetary					
	count	mean	std	min	25%	50%	75%	max	mean	std	min	25%	50%	75%	max	mean	std	median	sum	min	max
cluster																					
0	18025.0	234.694591	104.137087	44.0	156.0	232.0	314.0	615.0	1.085936	0.306742	1.0	1.0	1.0	1.0	5.0	228.022478	57.616154	214.13	4110105.16	138.16	457.55
1	41703.0	190.403568	82.566276	49.0	119.0	190.0	259.0	343.0	1.012517	0.113948	1.0	1.0	1.0	1.0	3.0	80.930355	37.119192	75.25	3375038.60	9.59	173.70
2	27529.0	468.119002	87.369095	320.0	396.0	460.0	532.0	744.0	1.018962	0.145416	1.0	1.0	1.0	1.0	6.0	102.708697	58.071210	90.28	2827467.73	10.07	350.97

From the table above, it can be concluded that :

- Cluster 0 is customer's cluster who made purchase a quite long time ago, they only come one time but they spent highest amount of money among the others. Although their population is the smallest, they contribute the most sales in e-commerce.
- Cluster 1 is customer's cluster who spend lowest amount of money but they are most recent purchasers but also one-time buyers. They dominate most of our customer population.
- Cluster 2 is customer's cluster who made only one purchase and hasn't been back for very long time, but they spent moderate amount of money.

Most of our customers are one-time purchasers.

NAMING THE CLUSTERS

Loyalist Squarepants!

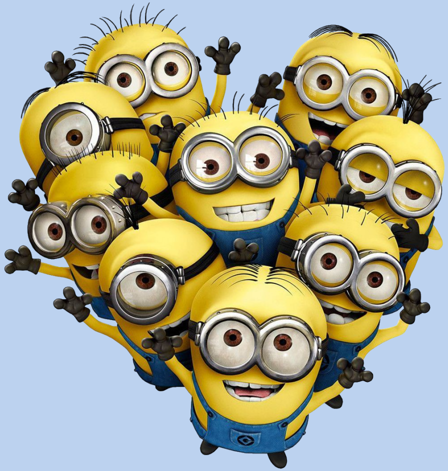
20.66% of customers

They only come one time and haven't been back for a quite long time, but they spend much amount of money as long as they happy. They spend the highest amount among the others.

Sleeping Snorlax ~

31.55% of customers

They spend moderate amount of money, also come only one time but they made their purchase long long time ago.



Minions on Shopping :D

47.79% of customers

They are recent purchasers who dominate the customer population, but they spend less money and also come only one time. They spend the lowest among the others.

BUSINESS RECOMMENDATIONS

Loyalist Squarepants

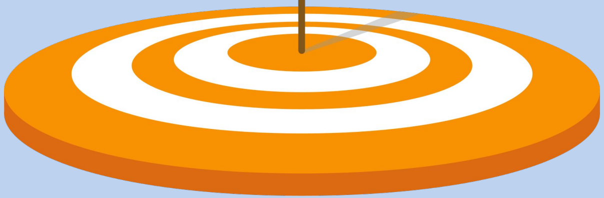
Giving special new product introductions based on their purchase product history. We can also try add cross/up-sells strategy such as bundling in order to increase their AOV. Sending special voucher on special day will make them feel special too!

(20.66% of customers)

Sleeping Snorlax

Bring them back with Reactivation campaign or promotions (not too often), and run e-mail surveys to find out the reasons why customers didn’t come back. If possible, we can try giving discounts, but we need to consider our marketing budget.

(31.55% of customers)



(47.79% of customers)

Minions on Shopping

Giving welcome discount with small rate or amount will make them feel welcome to our e-commerce. Build a promote referrals/review program is also recommended, so we can turn them into our advocates while acquiring new customers and gain positive image to our customers.

Overall Customers

Build a membership programs where customers get certain points for every purchase they make that can be encashed during the next purchase. Also create VIP programs with exclusive offers specifically for high-contribute customers. This will encourage new customers to shop/spend more and join the group.

Customer Segmentation in E-commerce

W6W7W8 - Python Advanced Assignment

by Annita

RevoU FSDA Section Madrid - Team 5