Capstone Project - 4

Online Retail Customer Segmentation

By:- Milan Ajudiya

Introduction:

Customer segmentation:

As the name suggest, segregating the customers to certain groups based on purchases, frequency of purchases, types of product bought into groups. It is important to analyze and retain the existing customer as well as explore and attract new customers.

It is found that customers retaining leads to more effort than exploring new customers. As existing customer are more likely to spend more on the products. Satisfying these customers will help to build large, and strong reliable customer base.

Content:

- Problem statement
- Data summary
- Exploratory data analysis
- RFM segmentation
- Fitting models
- Conclusion

Problem Statement:

In this dataset we have to identify major customer segments on a transnational data set which contains all the transactions. The data is UK-based and registered non-store online retail. The company mainly sells unique all-occasion gifts. Many customers of the company are wholesalers.

Data Attributes:

InvoiceNo: Invoice number. Nominal, a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation.

StockCode: Product (item) code. Nominal, a 5-digit integral number uniquely assigned to each distinct product.

Description: Product (item) name. Nominal.

Quantity: The quantities of each product (item) per transaction. Numeric.

Data Attributes:

InvoiceDate: Invoice Date and time. Numeric, the day and time when each transaction was generated.

UnitPrice: Unit price. Numeric, Product price per unit in sterling.

CustomerID: Customer number. Nominal, a 5-digit integral number uniquely assigned to each customer.

Country: Country name. Nominal, the name of the country where each customer resides.

Data summary:

transnational data set which contains all the transactions occurring between 1 December 2010 and 0 December 2011 for a UK-based online retail.

```
[173]  1   df.head()
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom

```
[174]  1   df.tail()
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	2011-12-09 12:50:00	4.95	12680.0	France

Sample data:

```
[176]  1   df.info()
```

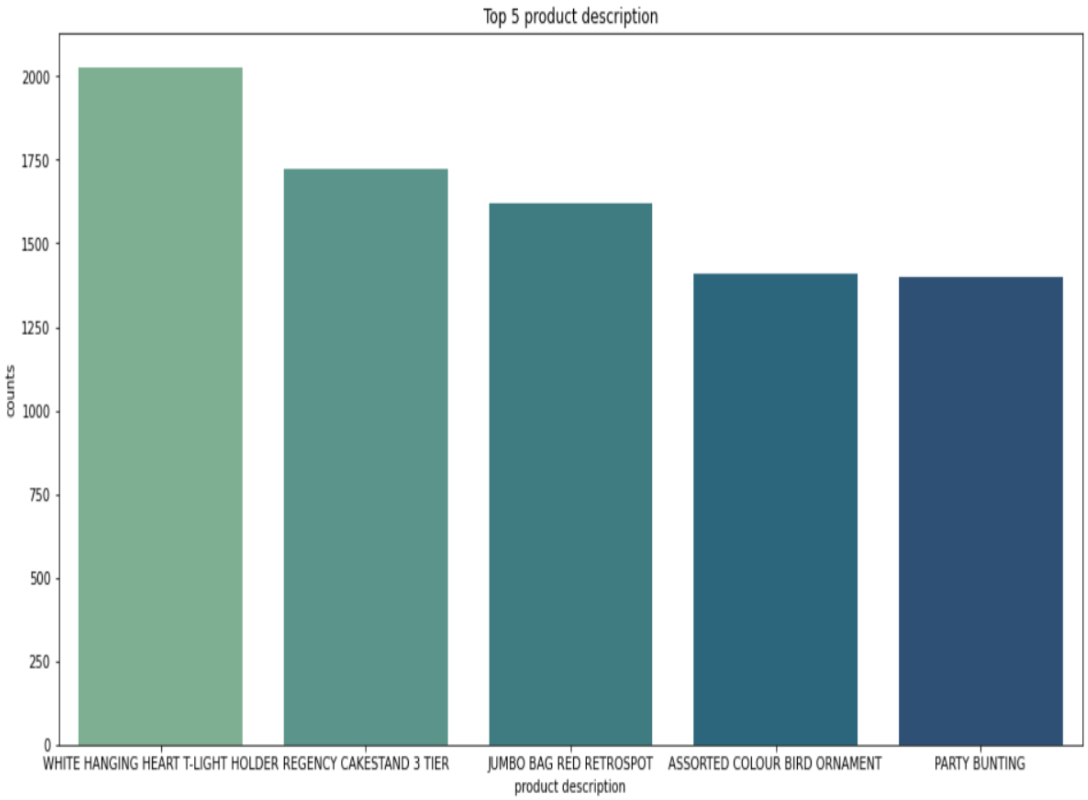
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype         
---  ------       --------------   -----         
 0   InvoiceNo    541909 non-null  object        
 1   StockCode    541909 non-null  object        
 2   Description  540455 non-null  object        
 3   Quantity     541909 non-null  int64         
 4   InvoiceDate  541909 non-null  datetime64[ns]
 5   UnitPrice    541909 non-null  float64       
 6   CustomerID   406829 non-null  float64       
 7   Country      541909 non-null  object        
dtypes: datetime64[ns](1), float64(2), int64(1), object(4)
memory usage: 33.1+ MB
```

As see shape of dataset is 541909 rows and 8 column, some of the columns are having null values.

Exploring dataset:

The dataset contains 541909 rows and 8 column.
CustomerId column has 24% null values.
In dataset InvoiceNo column some observation starts with “c” letter that means the is cancelled transaction.
So, I dropped those row that start with letter “c” , now the data is reduced to 397924 rows and 8 columns.
Now explore the total number of products, transaction, and customer data.

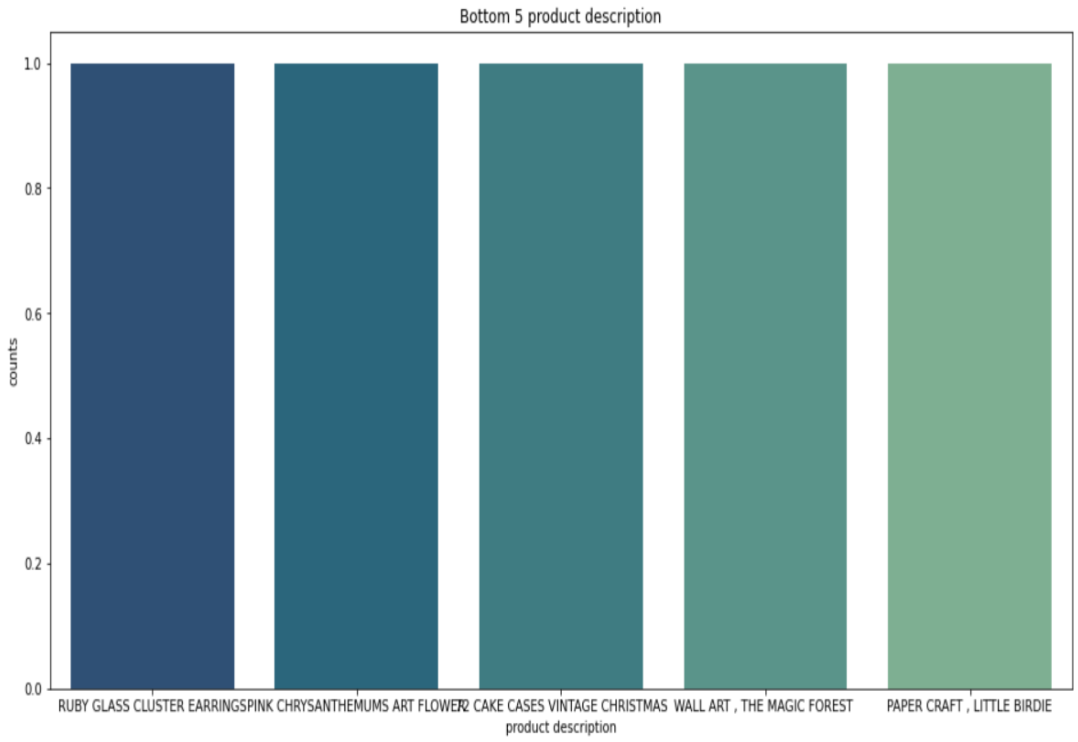
Top 5 product:



Graph Shows Top 5 Product Name:

1 . White Hanging Heart T-light Holder
2 . Regency Cake stand 3 Tier
3 . Jumbo Bag Red Retro spot
4 . Assorted Color Bird Ornament
5 . Party Bunting

Bottom 5 product:



Graph Shows The Bottom 5 Products:

1. Ruby Glass Cluster Earrings
2. Pink Chrysanthemums Art Flower
3. 72 Cake Cases Vintage Christmas
4. Wall Art , The Magic Forest
5. Paper Craft , Little Birdie

Customer belong to countries:

	Country name	counts
0	United Kingdom	354345
1	Germany	9042
2	France	8342
3	EIRE	7238
4	Spain	2485

Most of the transaction are done in united kingdom.

```
[ ]  1  df_country.tail()
```

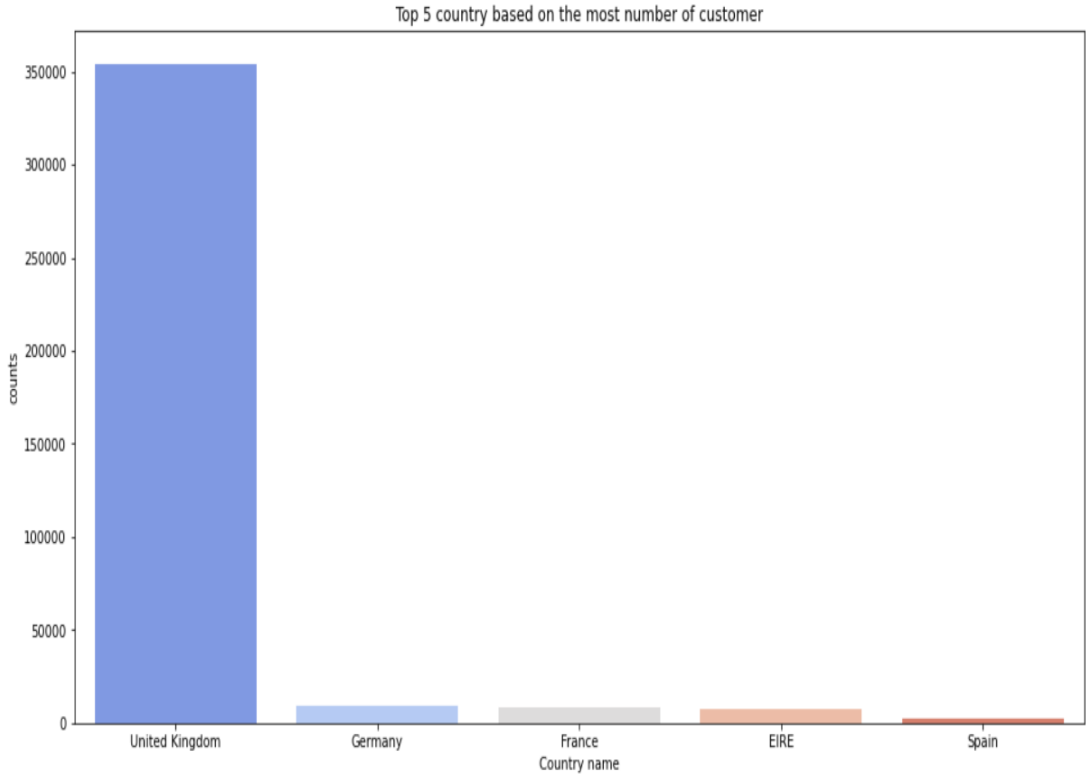
	Country name	counts
32	Lithuania	35
33	Brazil	32
34	Czech Republic	25
35	Bahrain	17
36	Saudi Arabia	9

Analyze the customer from which country belongs to:

From first: Uk, Germany, France, EIRE, Spain

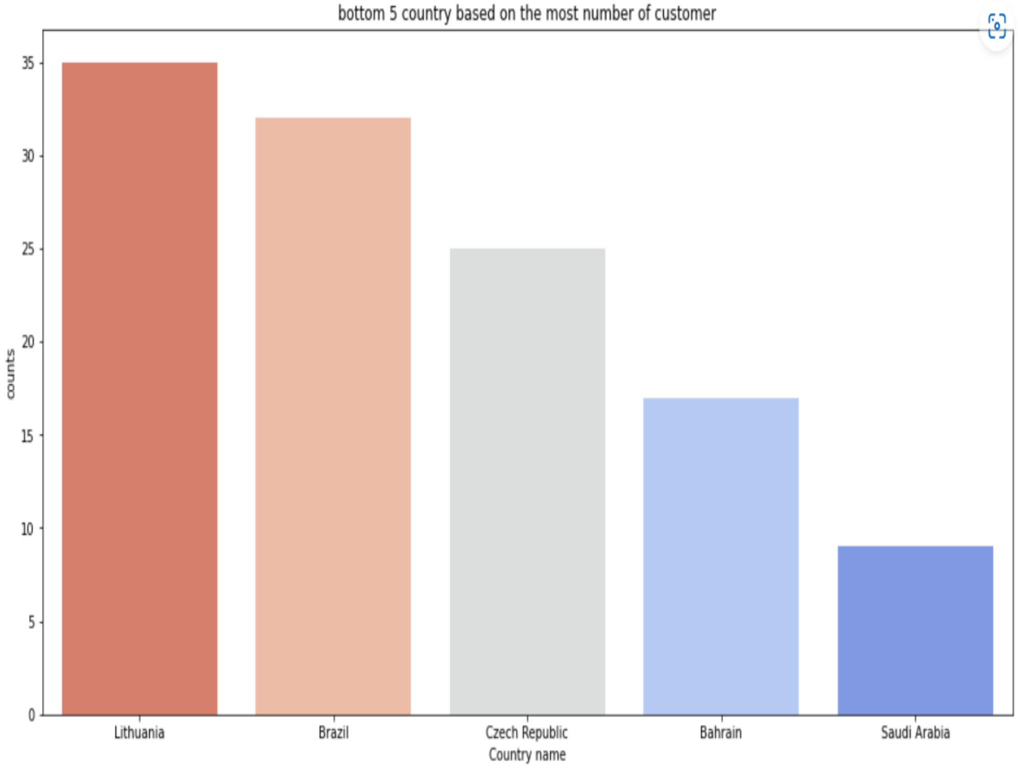
From last : Lithuania, Brazil, Czech Republic, Bahrain, Saudi Arabia

Top 5 Country:



From graph we can see that most of the customer is from united kingdom, then Germany, France, EIER, Spain.

Bottom 5 Country:

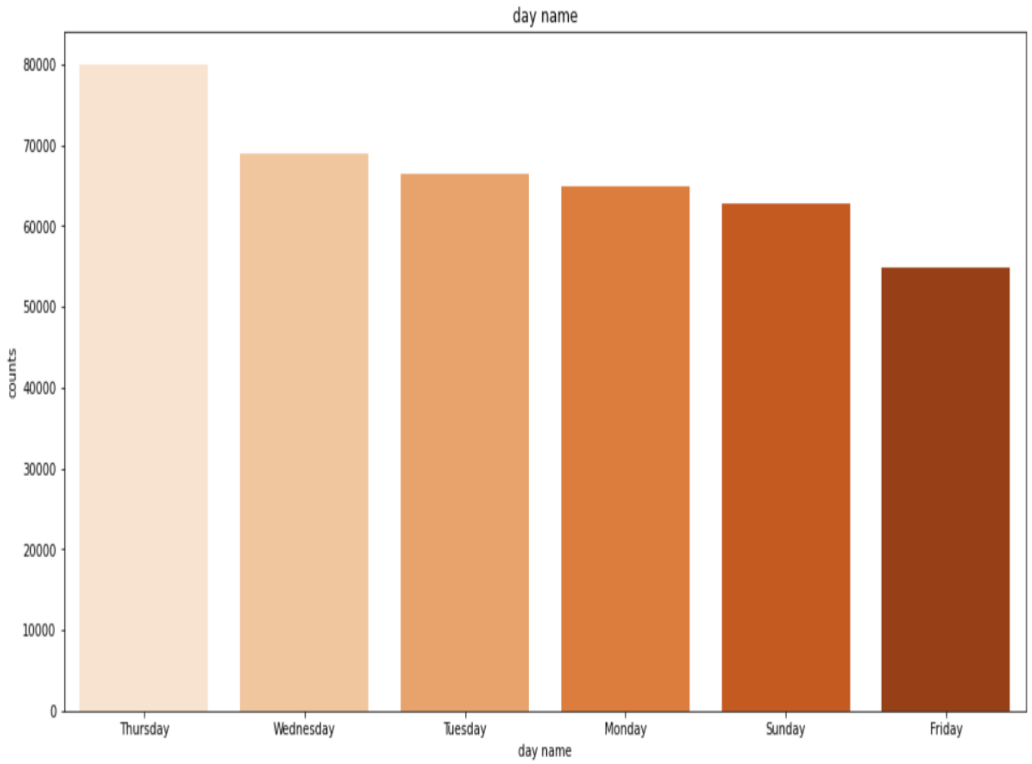


From graph we can see that bottom 5 countries from the customer is Lithuania, Brazil, Czech Republic, Bahrain, Saudi Arabia.

Feature Engineering:

- Convert invoice date column into datetime column.
- Creating new feature from invoice date.
- Creating new columns by extracting days, year, month, hour column.
- Preparing data to run on RFM model by creating a new column
- Total amount = quantity * unit price

Customer shop on days:



As see in graph that most of the customers are shopped at Thursday, Wednesday, and Tuesday then it is decreases.

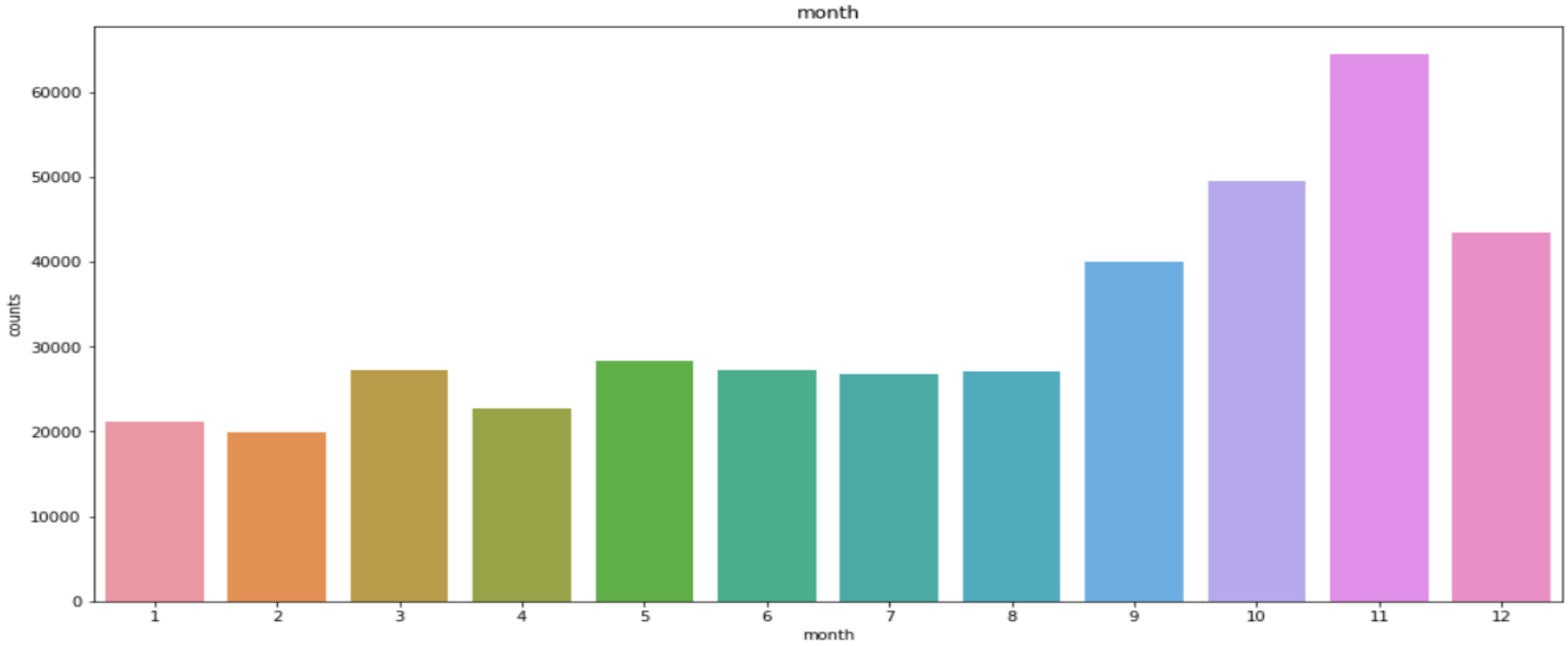
Maximum sale on month:

	month	counts
0	11	64545
1	10	49557
2	12	43464
3	9	40030
4	5	28322
5	6	27185
6	3	27177
7	8	27013
8	7	26827
9	4	22644
10	1	21232
11	2	19928

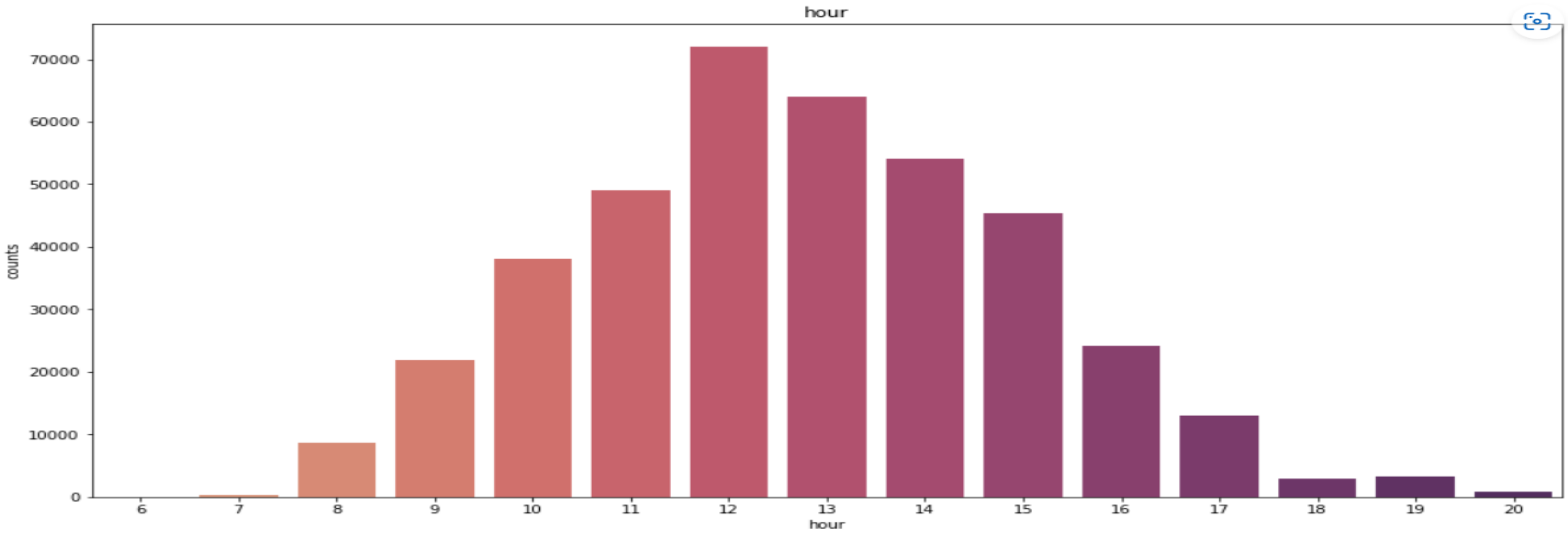
Most number of customer prefers to shop in month of November, October, December, September.

November ➔ 64545
October ➔ 49557
December ➔ 43464
September ➔ 40030

Month wise sell:

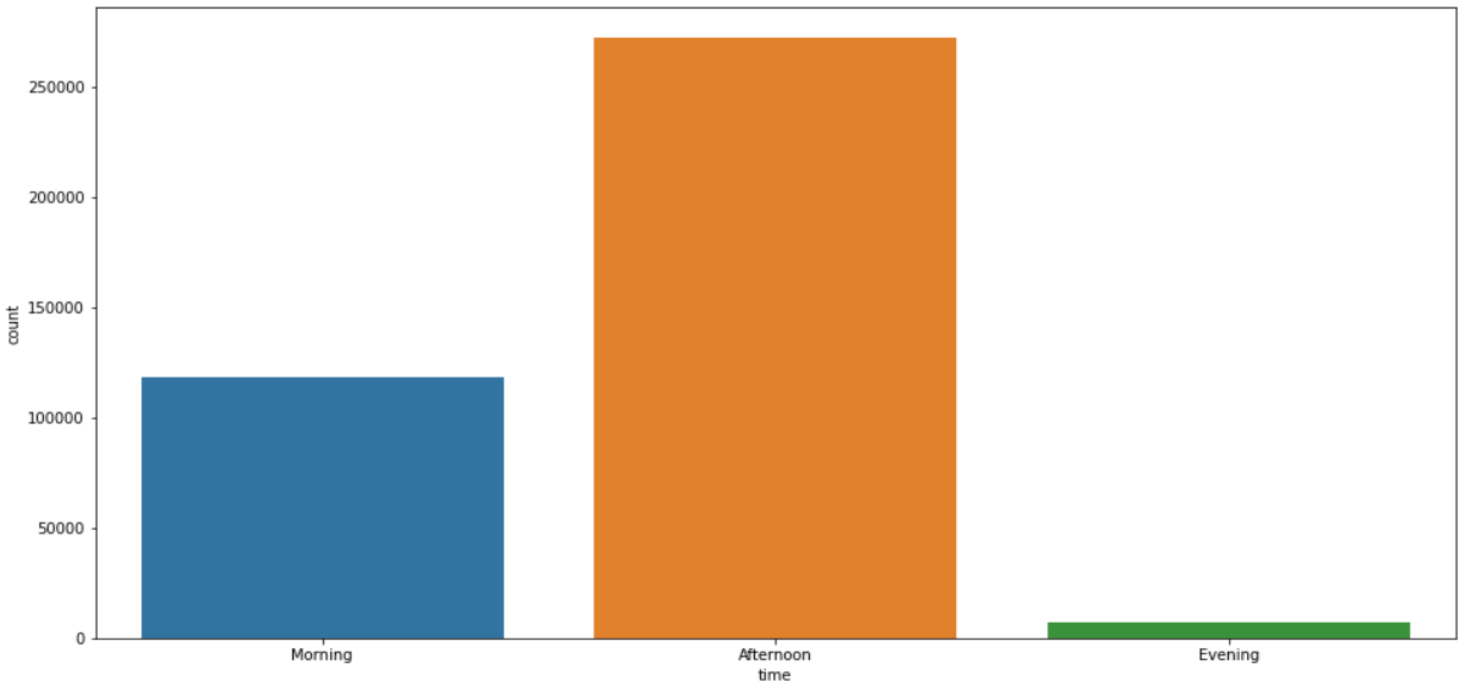


Hour wise sell:

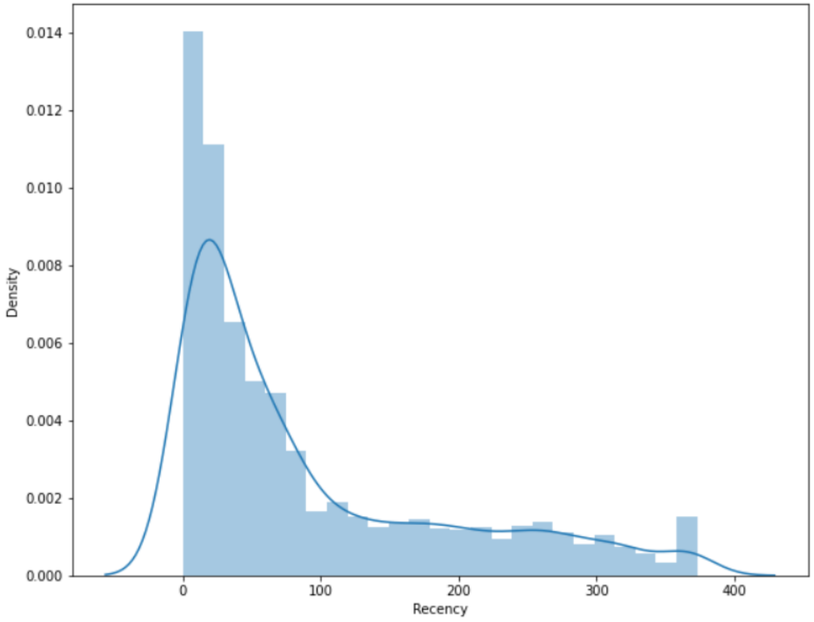


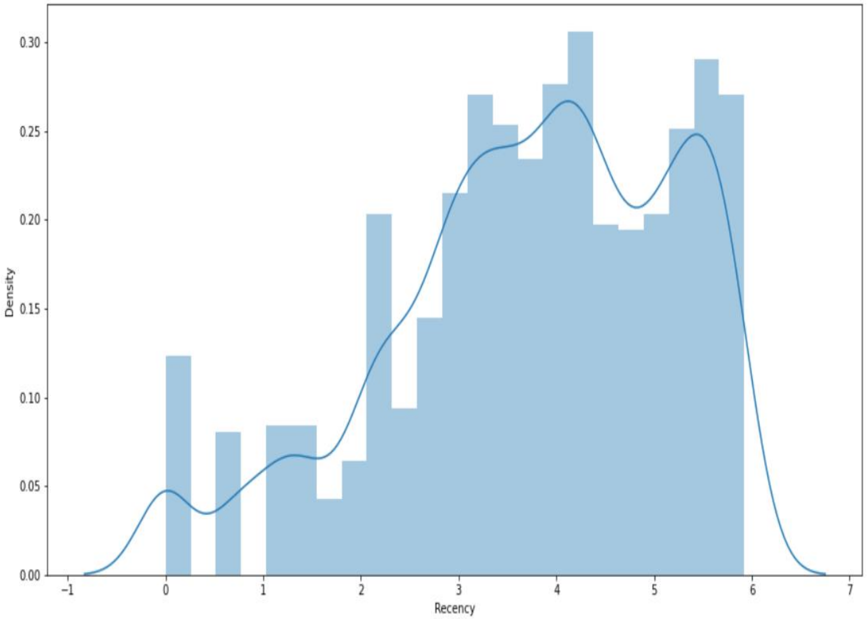
From the graph we can say that Afternoon time most of the customer prefere to purchase items.

Three groups Morning, Afternoon and Evening:

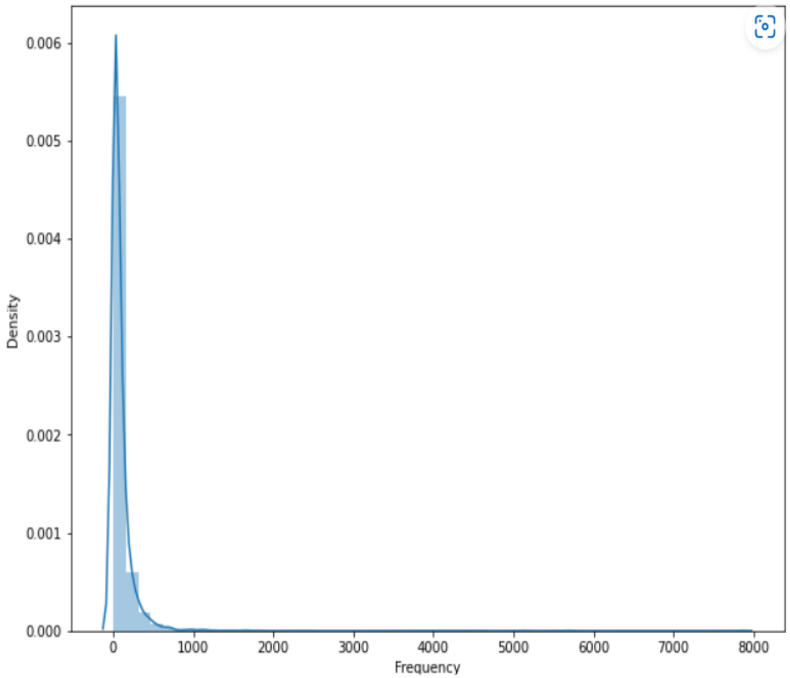


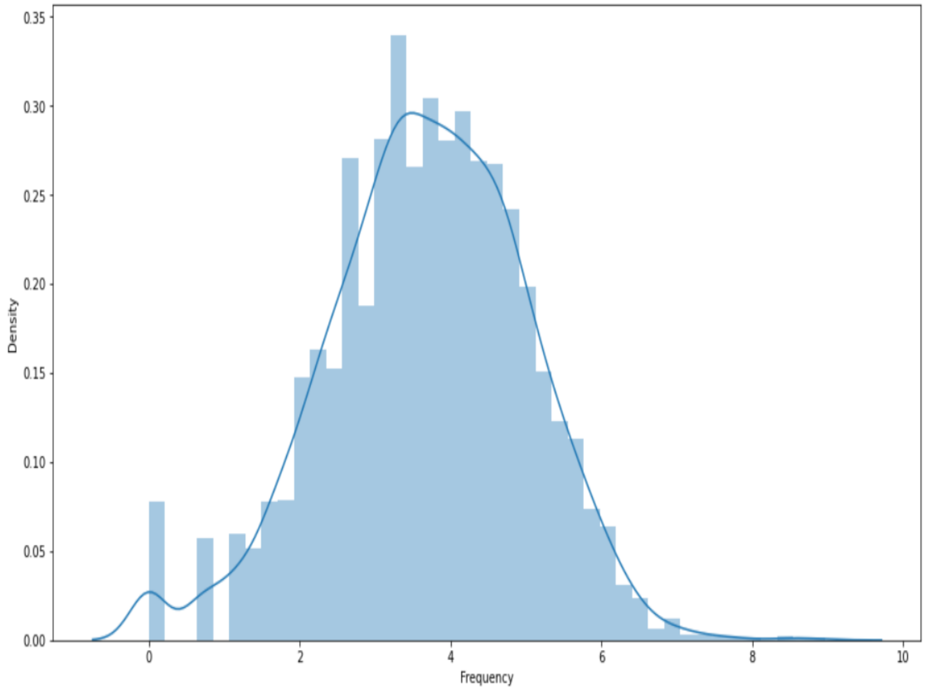
Recency:


0.014
0.012
0.010
0.008
0.006
0.004
0.002
0.000
Density
0
100
200
300
400
Recency

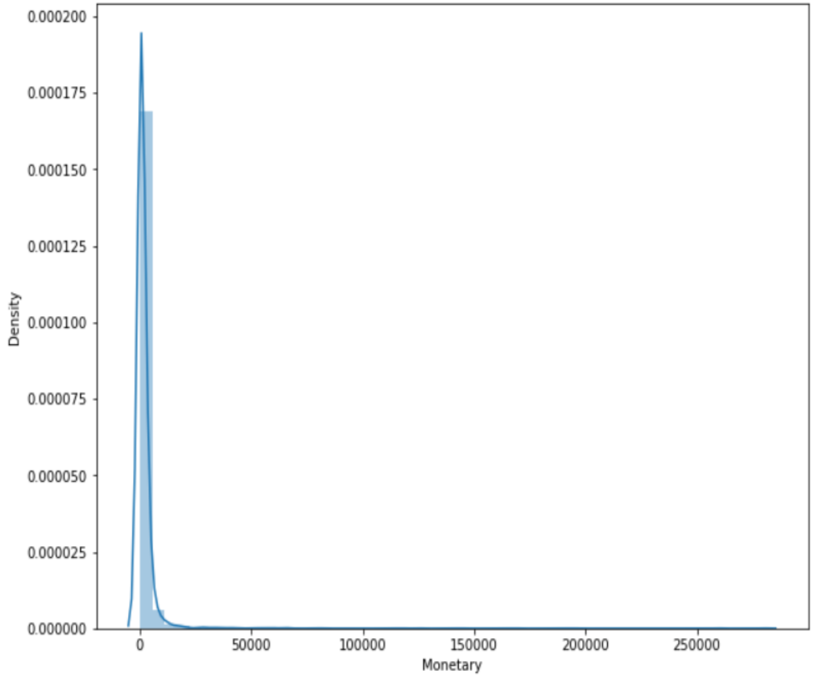

0.30
0.25
0.20
0.15
0.10
0.05
0.00
Density
−1
0
1
2
3
4
5
6
7
Recency

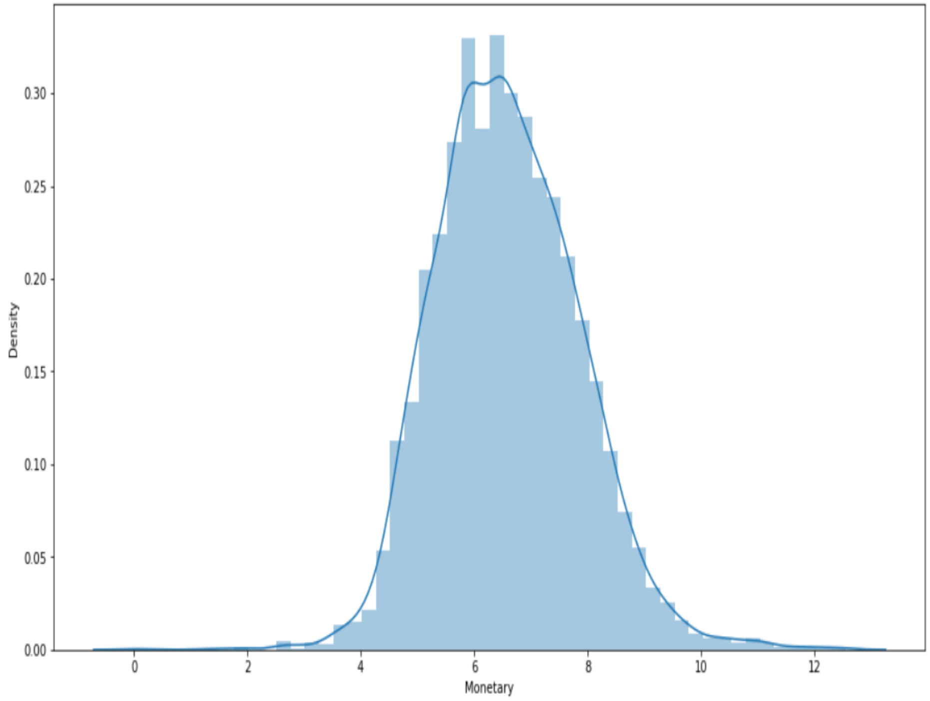
Frequency:


0.006
0.005
0.004
0.003
0.002
0.001
0.000
Density
0
1000
2000
3000
4000
5000
6000
7000
8000
Frequency


0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
Density
0
2
4
6
8
10
Frequency

Monetary:


0.000200
0.000175
0.000150
0.000125
0.000100
0.000075
0.000050
0.000025
0.000000
Density
0
50000
100000
150000
200000
250000
Monetary


0.30
0.25
0.20
0.15
0.10
0.05
0.00
Density
0
2
4
6
8
10
12
Monetary

RFM Metrics

RECENCY

The freshness of the customer activity, be it purchases or visits

E.g. Time since last order or last engaged with the product

FREQUENCY

The frequency of the customer transactions or visits

E.g. Total number of transactions or average time between transactions/ engaged visits

MONETARY

The intention of customer to spend or purchasing power of customer

E.g. Total or average transactions value

Create the RFM model:

RFM stands for Recency, Frequency, Monetary. RFM is a method used to analyse customer value.

Recency: It stores the number of days the customer has done his last purchase with respect to last date in the dataset. it is just to find the customer is last purchased from store.

Frequency: It is the number of times each customer has made a purchase by counting unique invoice date by each customer while making a purchase.

Monetary: It is the total amount spent by the customer.

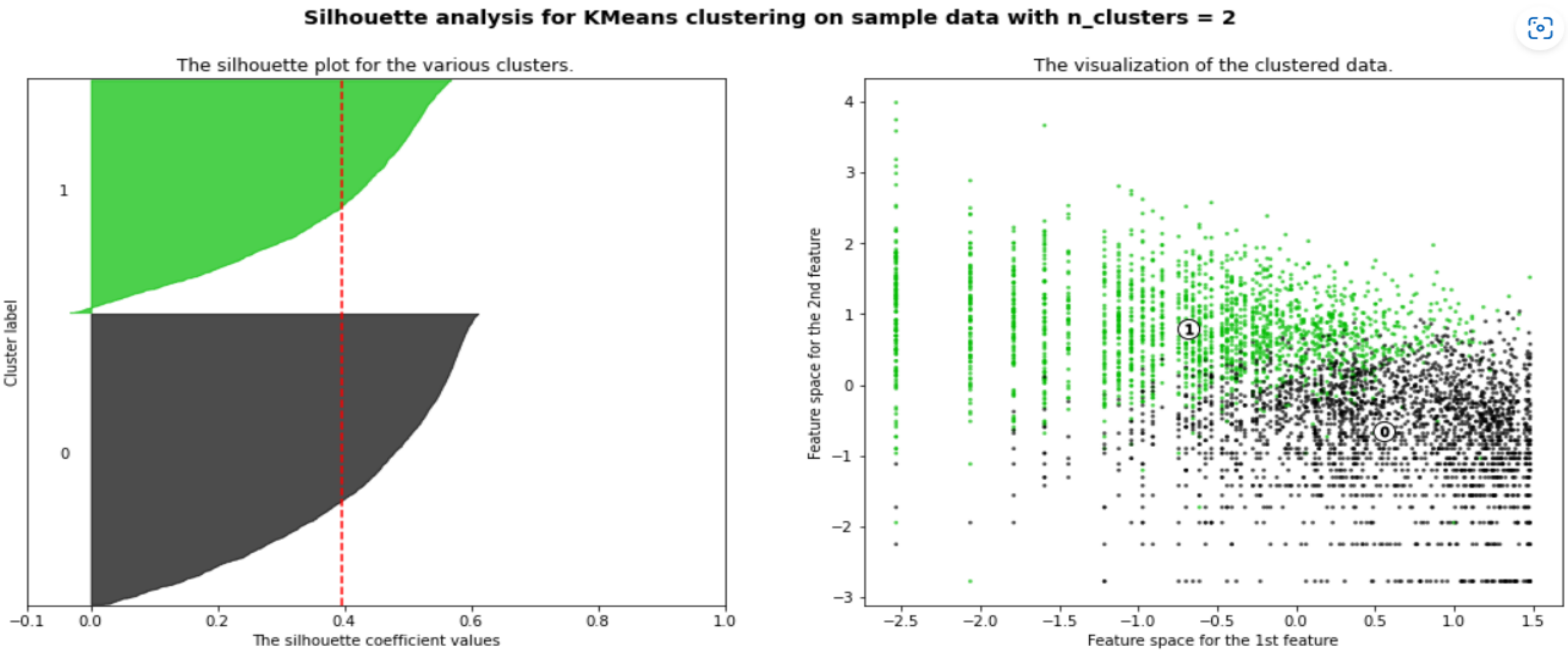
Create the RFM model:

Performing RFM model:

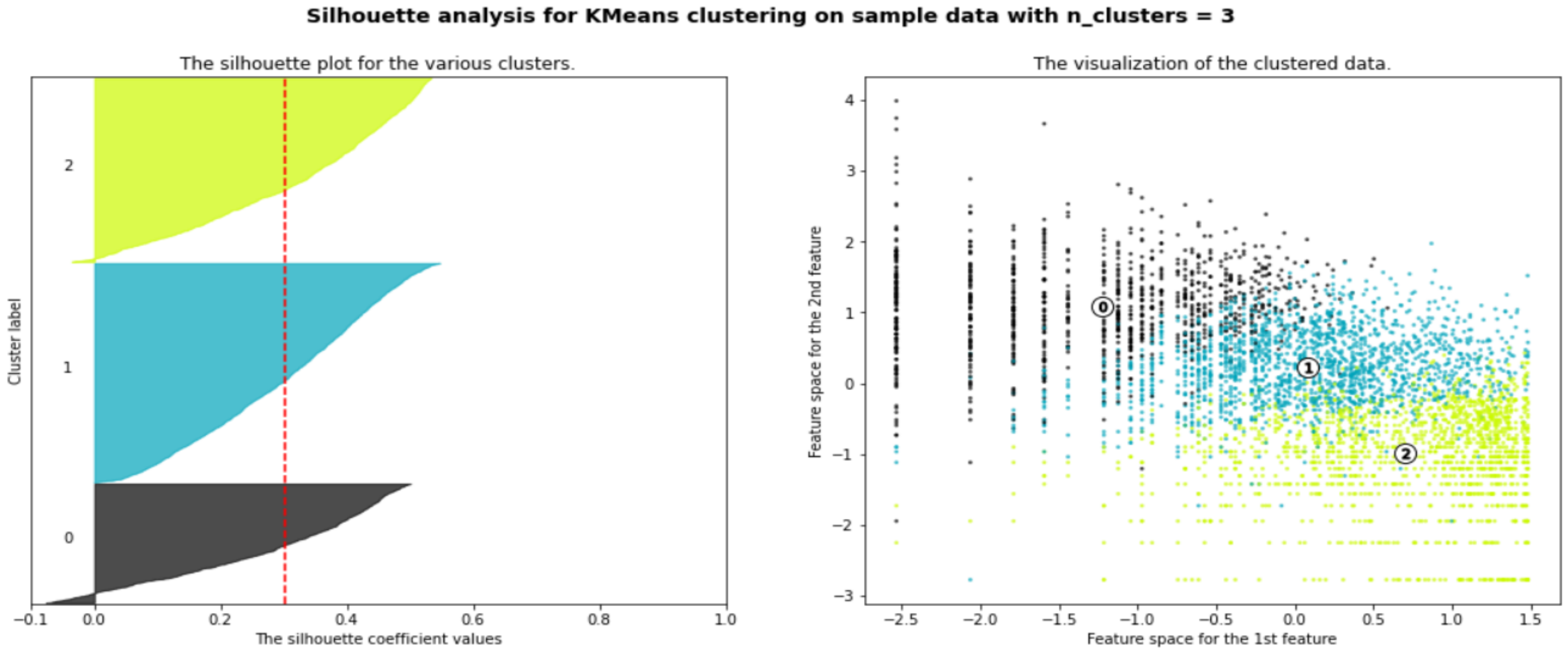
- The first step is to assign the Recency, Frequency, Monetary value to each customer.
- The second step is to divide the customer list into groups for each of the three dimension.

After that calculate the RFM score.

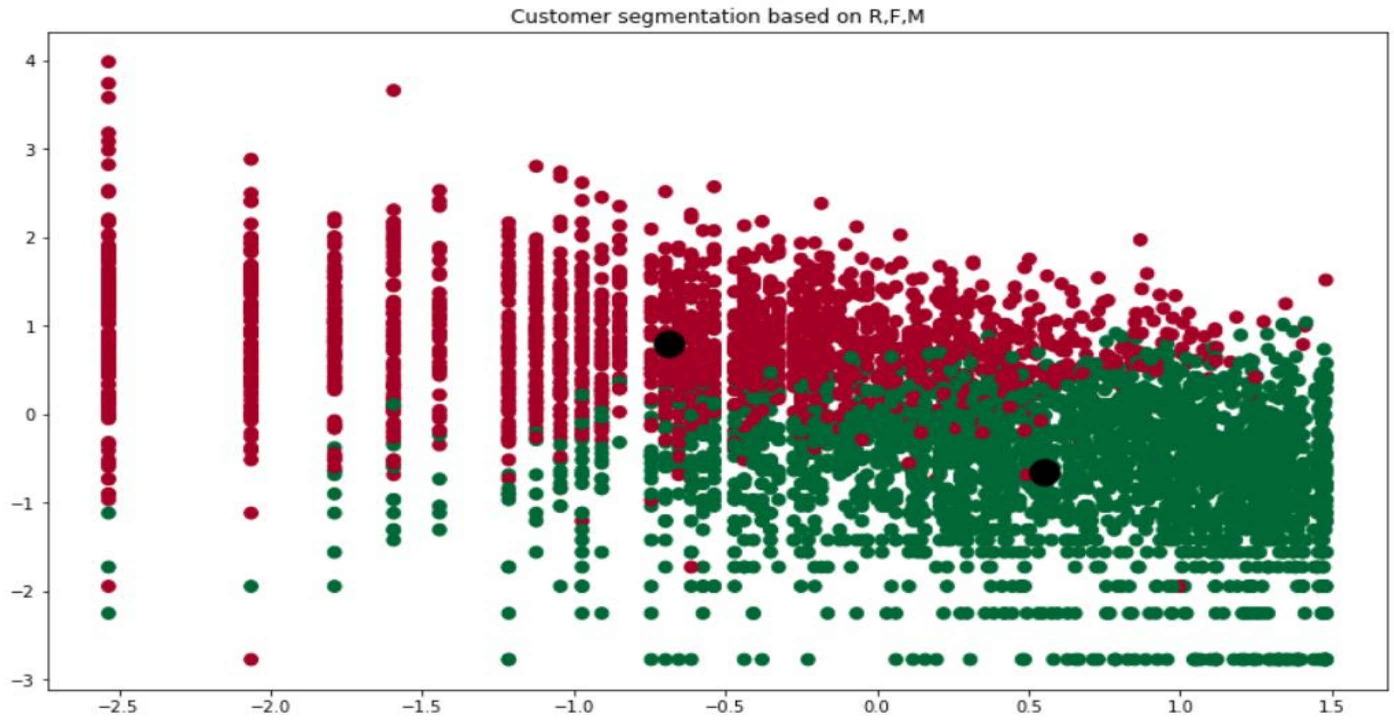
Kmeans clustering with 2 clusters:



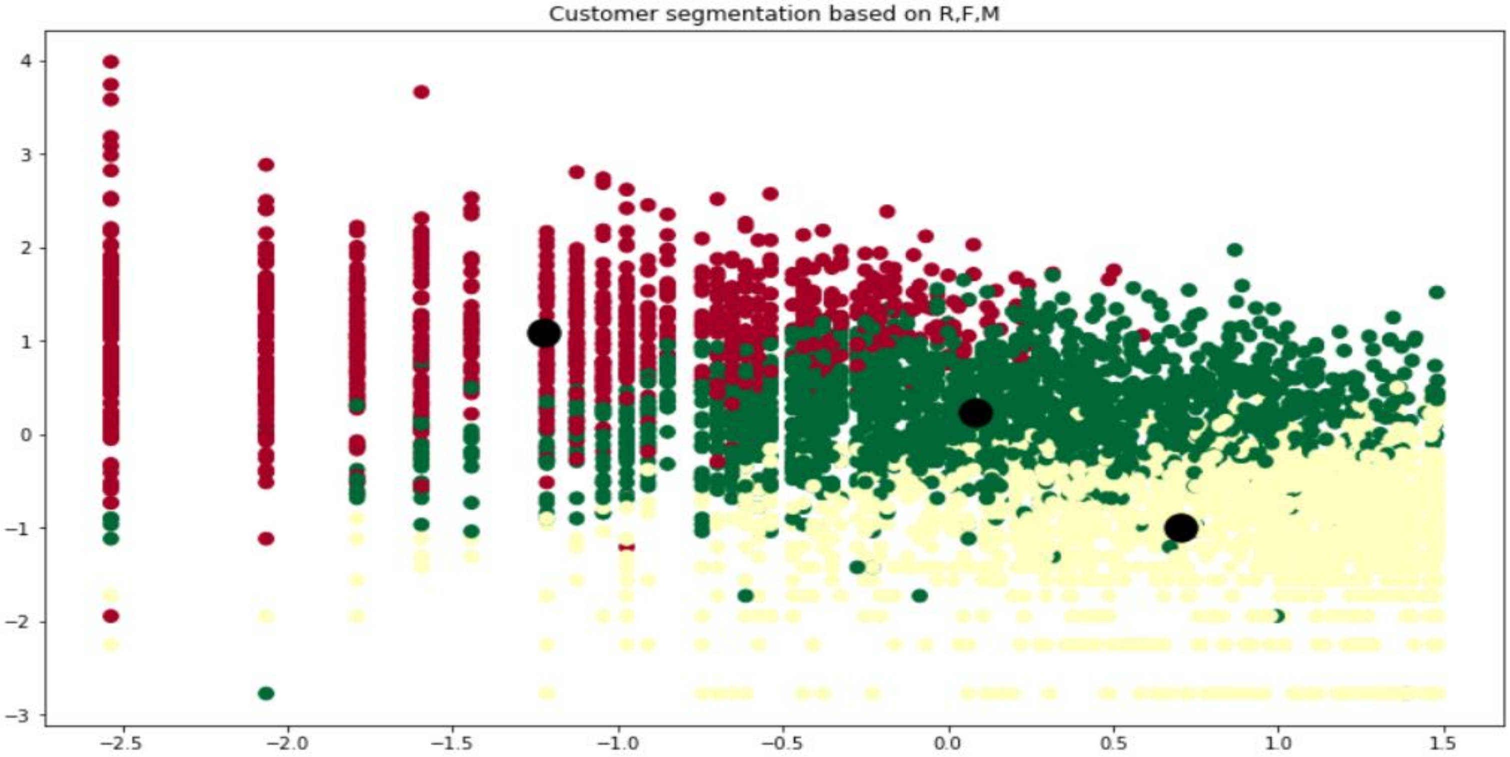
Kmeans clustering with 3 clusters:



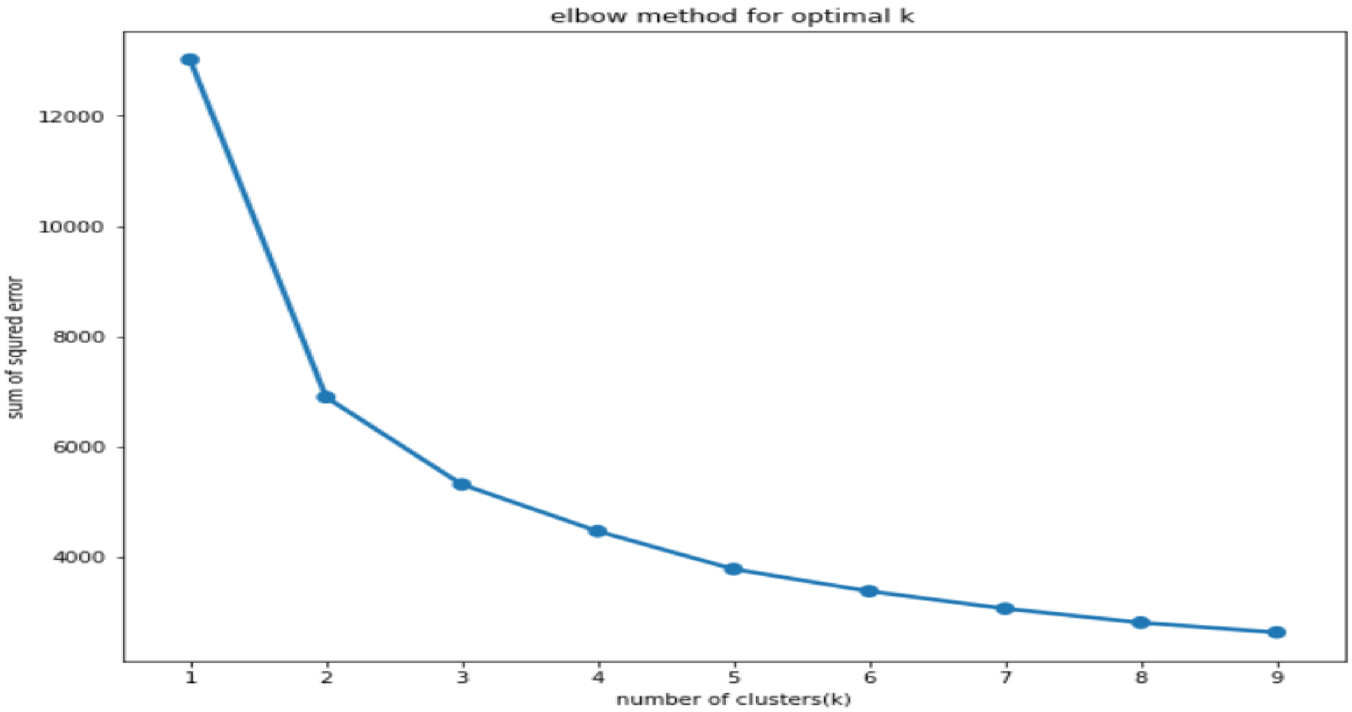
Segmentation based on R,F,M with 2 clusters:



Segmentation based on R,F,M with 3 clusters:



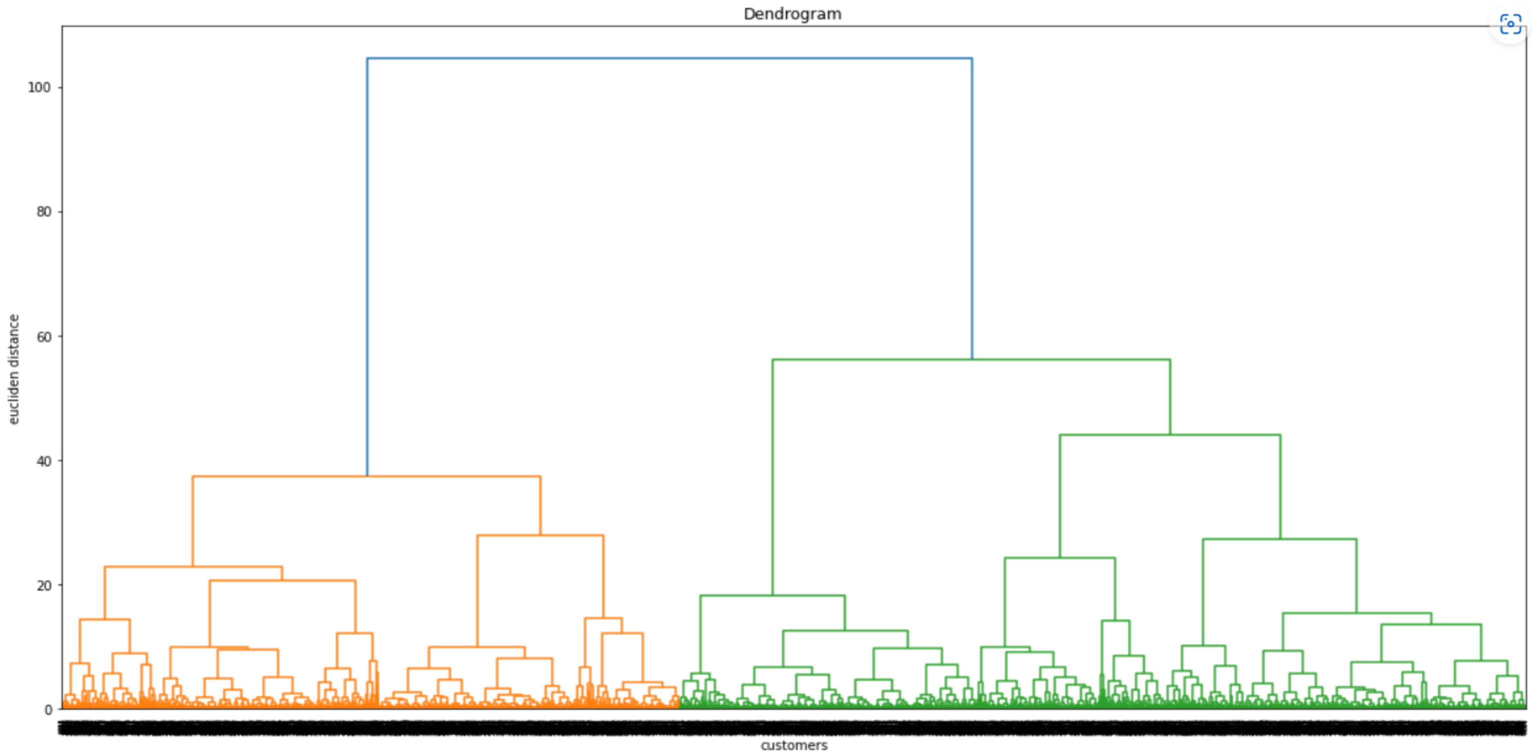
Elbow method for clustering k= 1-9



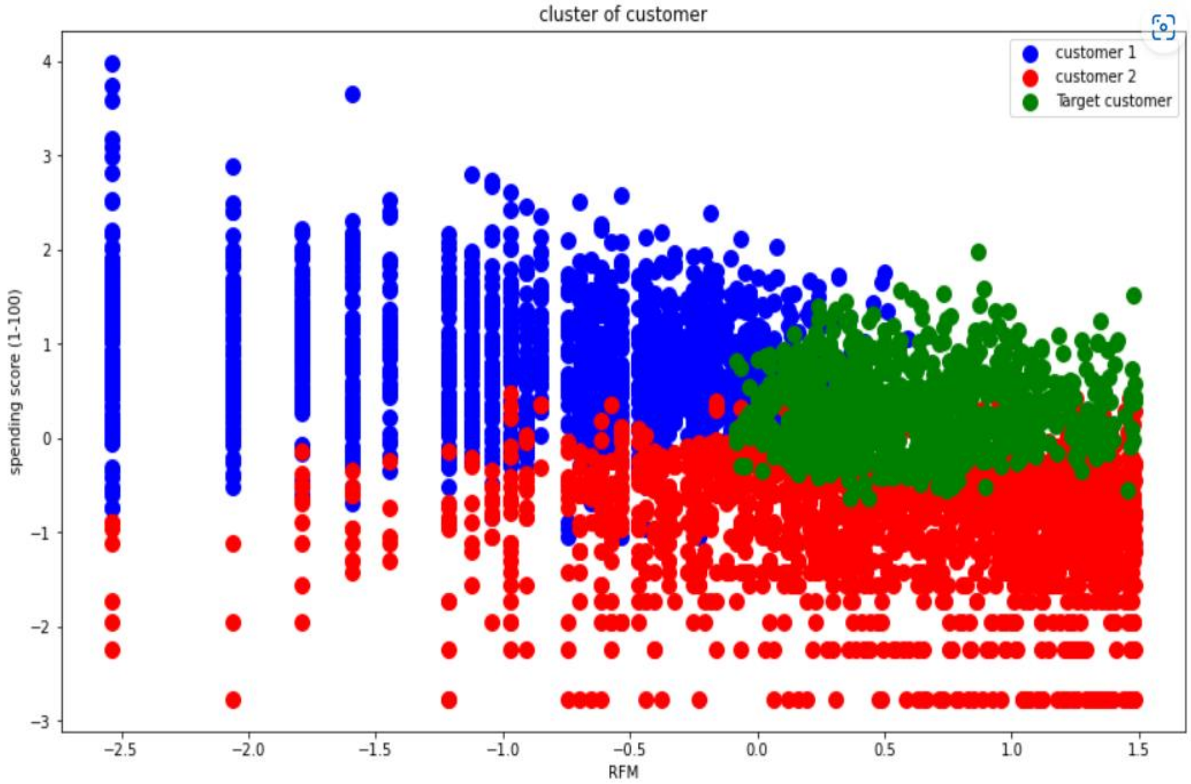
Perform Kmean clustering:

	Recency	Frequency	Monetary	R	F	M	RFMgroup	RFMscore	Recency_log	Frequency_log	Monetary_log	clusters
CustomerID												
12346.0	325	1	77183.60	4	4	1	441	9	5.783825	0.000000	11.253942	0
12347.0	2	182	4310.00	1	1	1	111	3	0.693147	5.204007	8.368693	2
12348.0	75	31	1797.24	3	3	1	331	7	4.317488	3.433987	7.494007	0
12349.0	18	73	1757.55	2	2	1	221	5	2.890372	4.290459	7.471676	0
12350.0	310	17	334.40	4	4	3	443	11	5.736572	2.833213	5.812338	1
...	...	...	...	...	...	...	...	...	...	...	...	...
18280.0	277	10	180.60	4	4	4	444	12	5.624018	2.302585	5.196285	1
18281.0	180	7	80.82	4	4	4	444	12	5.192957	1.945910	4.392224	1
18282.0	7	12	178.05	1	4	4	144	9	1.945910	2.484907	5.182064	1
18283.0	3	756	2094.88	1	1	1	111	3	1.098612	6.628041	7.647252	2
18287.0	42	70	1837.28	2	2	1	221	5	3.737670	4.248495	7.516041	0

Dendogram method for clustering k=2 -15



Fitting Hierarchical clustering:



As see in hierarchical clustering we get the number of clusters is 3.

By apply different algorithm we get the number of clusters is 3.

Summay:

```python
from prettytable import PrettyTable

# specify the column name
table = PrettyTable(["No.","Model Name","Data","Optimal number of clusters"])

# add tyhe number of rows
table.add_row(["1","KMeans with elbow method","RFM","3"])
table.add_row(["2","KMeans with silhouette method","RFM","3"])
table.add_row(["3","Hierarchical clustering","RFM","3"])
print(table)

```

```
+-----+-------------------------------+------+----------------------------+
| No. |           Model Name          | Data | Optimal number of clusters |
+-----+-------------------------------+------+----------------------------+
|  1  |    KMeans with elbow method   | RFM  |             3              |
|  2  | KMeans with silhouette method | RFM  |             3              |
|  3  |    Hierarchical clustering    | RFM  |             3              |
+-----+-------------------------------+------+----------------------------+
```

Conclusion:

cluster 0 is loyal customer they are frequent and heavy spending customers.

cluster 1 is new customer they are recently visited to store with minimum frequency and spending.

cluster 2 is Risk of leaving type of customer they are Average spenders and moderately visited to store.

Thank you