# Machine Learning (1)Project

**BUSINESS REPORT**

# Contents

# List of Tables

# List of Figure

# Problem Statement

## Context

AllLife Bank wants to focus on its credit card customer base in the next financial year. They have been advised by their marketing research team, that the penetration in the market can be improved. Based on this input, the Marketing team proposes to run personalized campaigns to target new customers as well as upsell to existing customers. Another insight from the market research was that the customers perceive the support services of the back poorly. Based on this, the Operations team wants to upgrade the service delivery model, to ensure that customer queries are resolved faster. Head of Marketing and Head of Delivery both decide to reach out to the Data Science team for help

## Objective

To identify different segments in the existing customer, based on their spending patterns as well as past interaction with the bank, using clustering algorithms, and provide recommendations to the bank on how to better market to and service these customers.

## Data Description

The data provided is of various customers of a bank and their financial attributes like credit limit, the total number of credit cards the customer has, and different channels through which customers have contacted the bank for any queries (including visiting the bank, online and through a call center).

## Data Dictionary

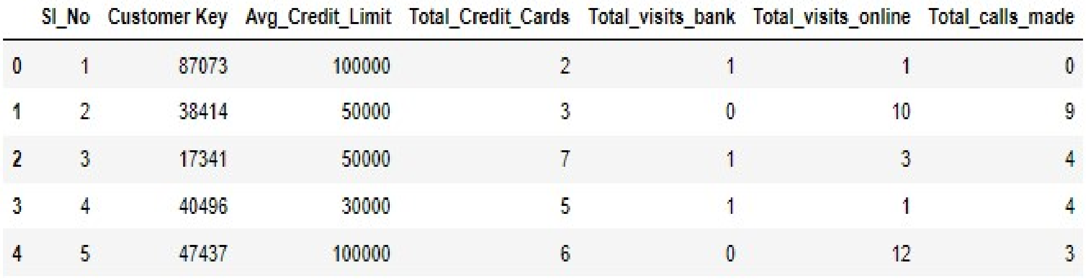
- Sl_No: Primary key of the records
- Customer Key: Customer identification number
- Average Credit Limit: Average credit limit of each customer for all credit cards
- Total credit cards: Total number of credit cards possessed by the customer
- Total visits bank: Total number of visits that customer made (yearly) personally to the bank
- Total visits online: Total number of visits or online logins made by the customer (yearly)
- Total calls made: Total number of calls made by the customer to the bank or its customer service department (yearly)

### 1- Define the problem and perform an Exploratory Data Analysis

- Problem definition, questions to be answered - Data background and contents - Univariate analysis - Bivariate analysis - Insights based on EDA

**>Checking the shape of the dataset**
The dataset has 660 rows and 7 columns.

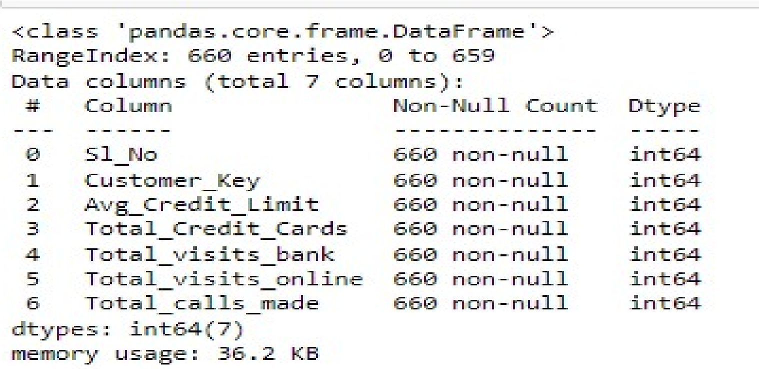
**>Checking few rows of the dataset**



| | Sl_No | Customer Key | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 87073 | 100000 | 2 | 1 | 1 | 0 |
| 1 | 2 | 38414 | 50000 | 3 | 0 | 10 | 9 |
| 2 | 3 | 17341 | 50000 | 7 | 1 | 3 | 4 |
| 3 | 4 | 40496 | 30000 | 5 | 1 | 1 | 4 |
| 4 | 5 | 47437 | 100000 | 6 | 0 | 12 | 3 |

**Table 1- Top 5 rows of the dataset**

**>Checking the data types of the columns for the dataset**

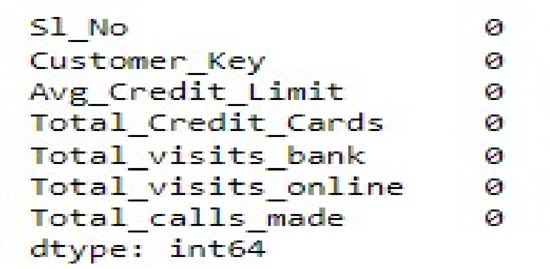


```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 660 entries, 0 to 659
Data columns (total 7 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Sl_No                660 non-null    int64
 1   Customer_Key         660 non-null    int64
 2   Avg_Credit_Limit     660 non-null    int64
 3   Total_Credit_Cards   660 non-null    int64
 4   Total_visits_bank    660 non-null    int64
 5   Total_visits_online  660 non-null    int64
 6   Total_calls_made     660 non-null    int64
dtypes: int64(7)
memory usage: 36.2 KB
```

**Table 2- Basic info of the dataset**

All the rows of the dataset are integer datatypes.

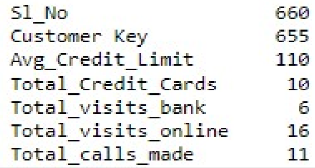
**>Checking missing values of the dataset**



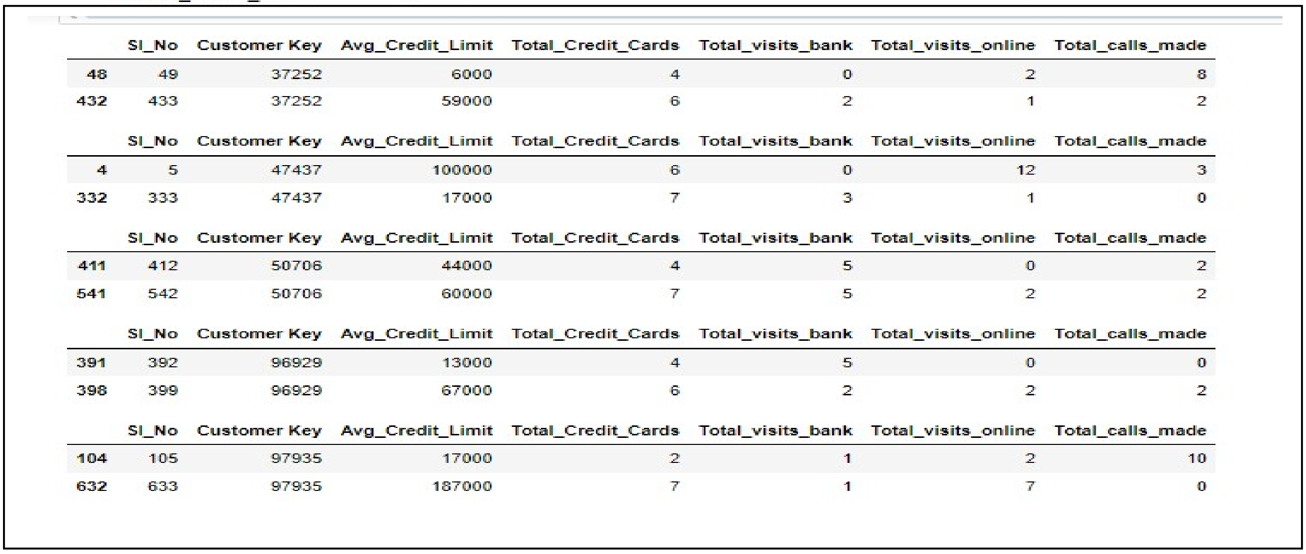
```
Sl_No                  0
Customer_Key           0
Avg_Credit_Limit       0
Total_Credit_Cards     0
Total_visits_bank      0
Total_visits_online    0
Total_calls_made       0
dtype: int64
```

There are no missing values present in the dataset.

**>Checking for duplicates values**



```
Sl_No                  660
Customer Key           655
Avg_Credit_Limit       110
Total_Credit_Cards      10
Total_visits_bank        6
Total_visits_online     16
Total_calls_made        11
```



| | Sl_No | Customer Key | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made |
|---|---|---|---|---|---|---|---|
| 48 | 49 | 37252 | 6000 | 4 | 0 | 2 | 8 |
| 432 | 433 | 37252 | 59000 | 6 | 2 | 1 | 2 |

| | Sl_No | Customer Key | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made |
|---|---|---|---|---|---|---|---|
| 4 | 5 | 47437 | 100000 | 6 | 0 | 12 | 3 |
| 332 | 333 | 47437 | 17000 | 7 | 3 | 1 | 0 |

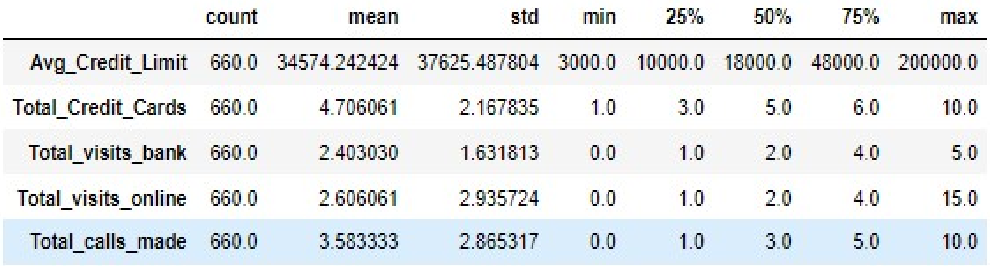
| | Sl_No | Customer Key | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made |
|---|---|---|---|---|---|---|---|
| 411 | 412 | 50706 | 44000 | 4 | 5 | 0 | 2 |
| 541 | 542 | 50706 | 60000 | 7 | 5 | 2 | 2 |

| | Sl_No | Customer Key | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made |
|---|---|---|---|---|---|---|---|
| 391 | 392 | 96929 | 13000 | 4 | 5 | 0 | 0 |
| 398 | 399 | 96929 | 67000 | 6 | 2 | 2 | 2 |

| | Sl_No | Customer Key | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made |
|---|---|---|---|---|---|---|---|
| 104 | 105 | 97935 | 17000 | 2 | 1 | 2 | 10 |
| 632 | 633 | 97935 | 187000 | 7 | 1 | 7 | 0 |

There are less unique values in the Customer Key column than the number of observations in the data. This means that there are duplicate values in the column.

**>Statistical summary of the dataset**



| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| Avg_Credit_Limit | 660.0 | 34574.242424 | 37625.487804 | 3000.0 | 10000.0 | 18000.0 | 48000.0 | 200000.0 |
| Total_Credit_Cards | 660.0 | 4.706061 | 2.167835 | 1.0 | 3.0 | 5.0 | 6.0 | 10.0 |
| Total_visits_bank | 660.0 | 2.403030 | 1.631813 | 0.0 | 1.0 | 2.0 | 4.0 | 5.0 |
| Total_visits_online | 660.0 | 2.606061 | 2.935724 | 0.0 | 1.0 | 2.0 | 4.0 | 15.0 |
| Total_calls_made | 660.0 | 3.583333 | 2.865317 | 0.0 | 1.0 | 3.0 | 5.0 | 10.0 |

**Table 3- Statistical summary of the dataset**

- Avg_Credit_Limit means is 34574.24 which is higher than the other and also maximum 200000.

**>Univariate Analysis**

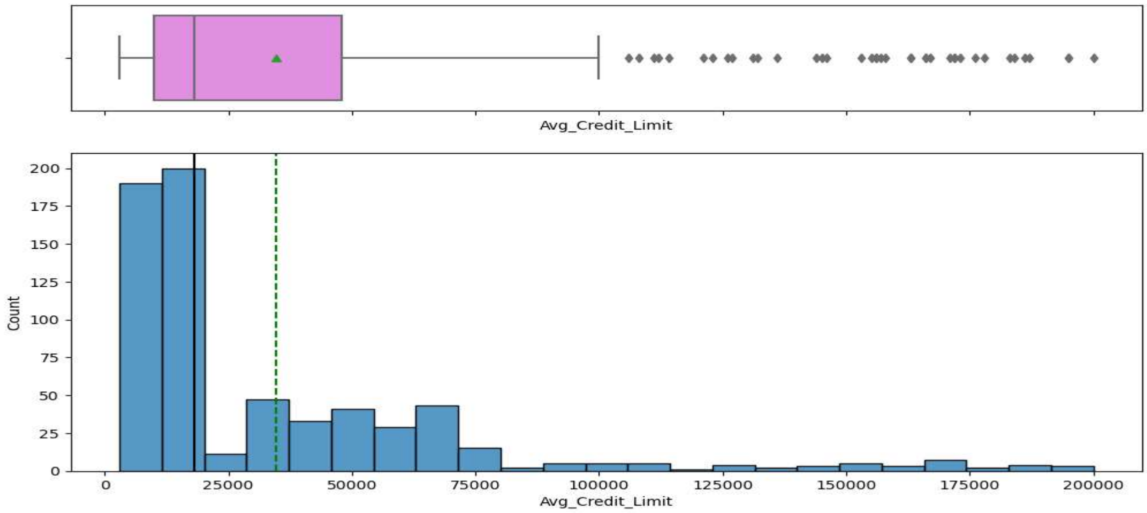


**Fig 1- Avg Credit Limit**

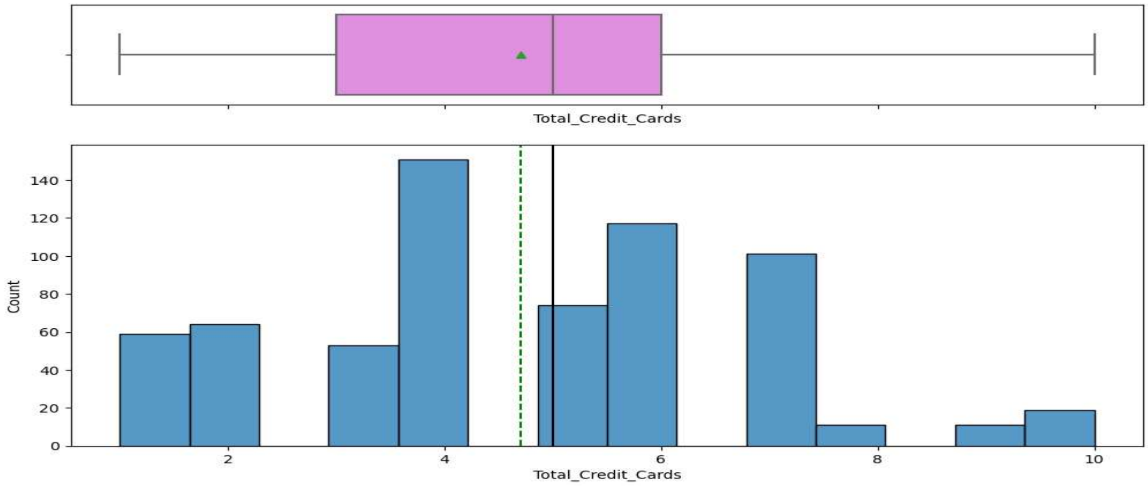


**Fig 2- Total Credit Cards**

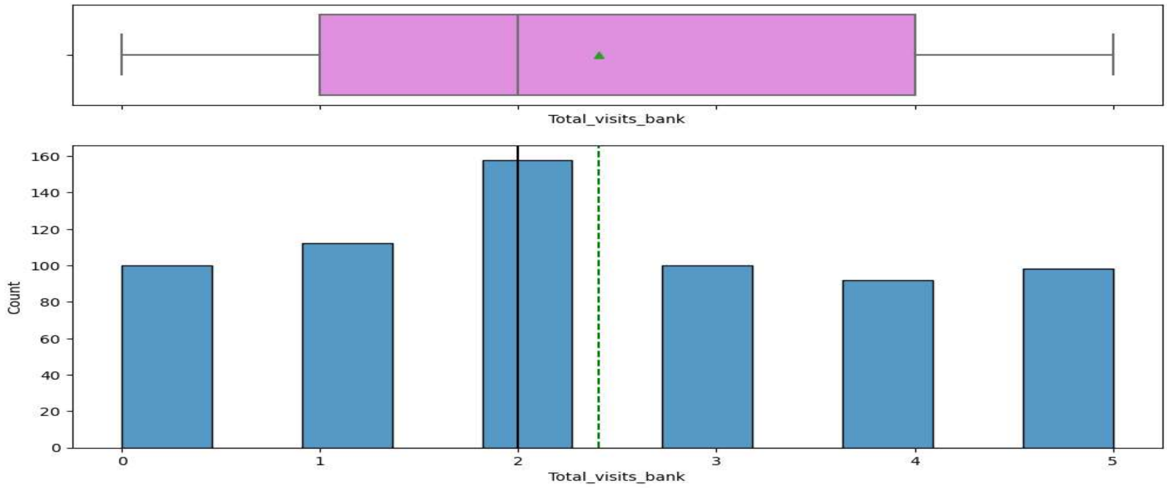


**Fig 3- Total Visits bank**

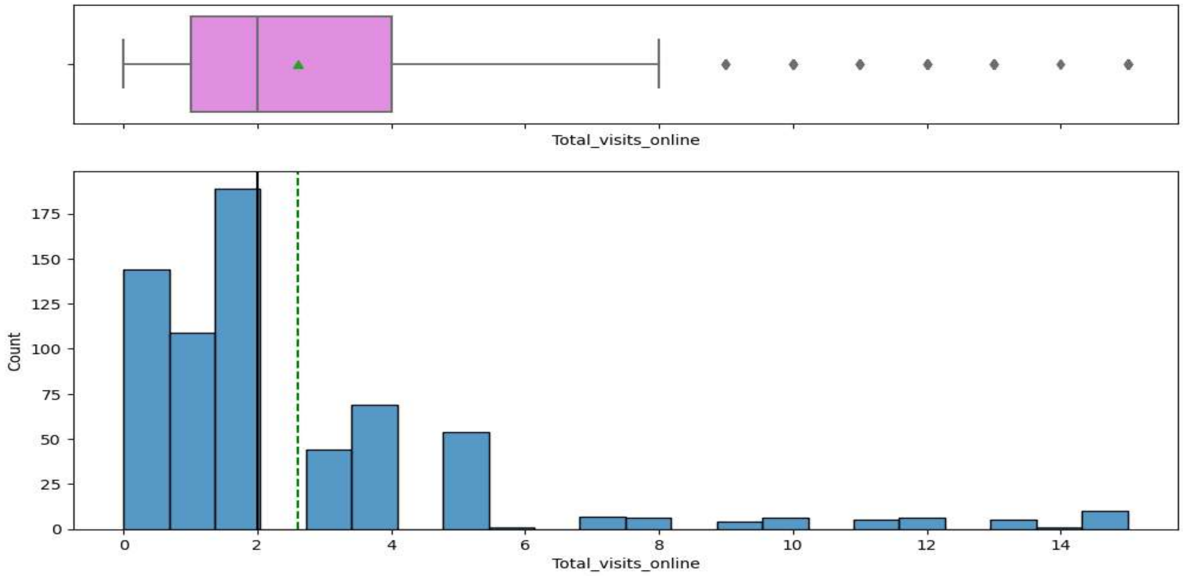


**Fig 4- Total visits online**

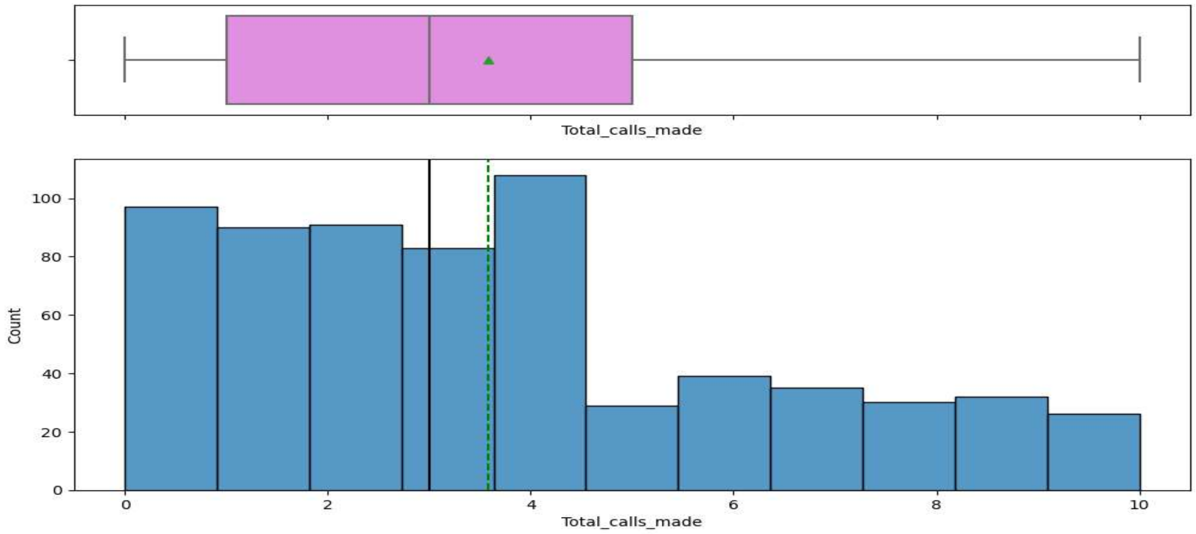


**Fig 5- Total calls made**

Observations: From all the above boxplot

- Avg_credit_limits has right s skew.
- Avg_credit_limits have outlier but no require to treat the outlier.
- Some customers have more then 7 credit cards.
- Most of the customer like to visits bank physically.
- Some customers prefer to visits online for their requirment.
- There are small amount of customer have called bank 10 times.

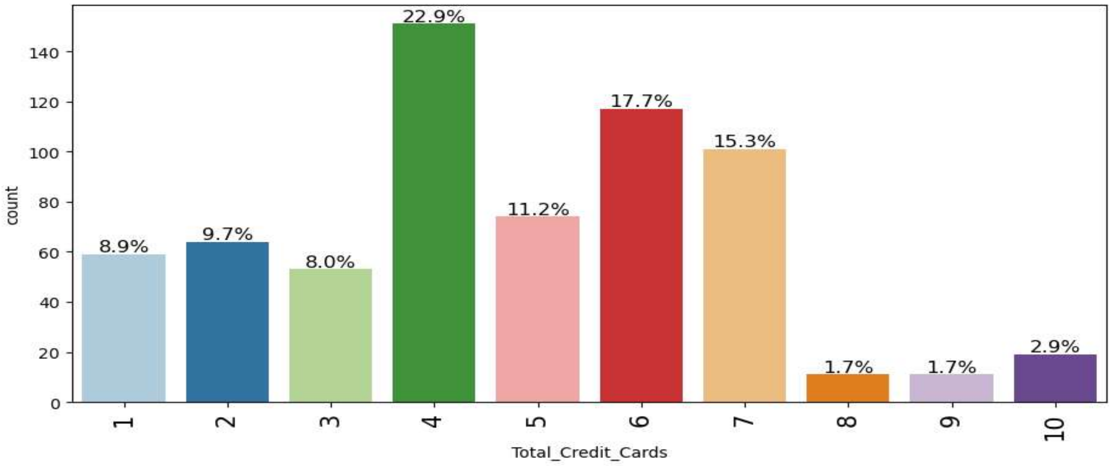


**Fig 6- Bar plot of total credit cards.**

Observations: Around 22.9% customer have 4 credit cards

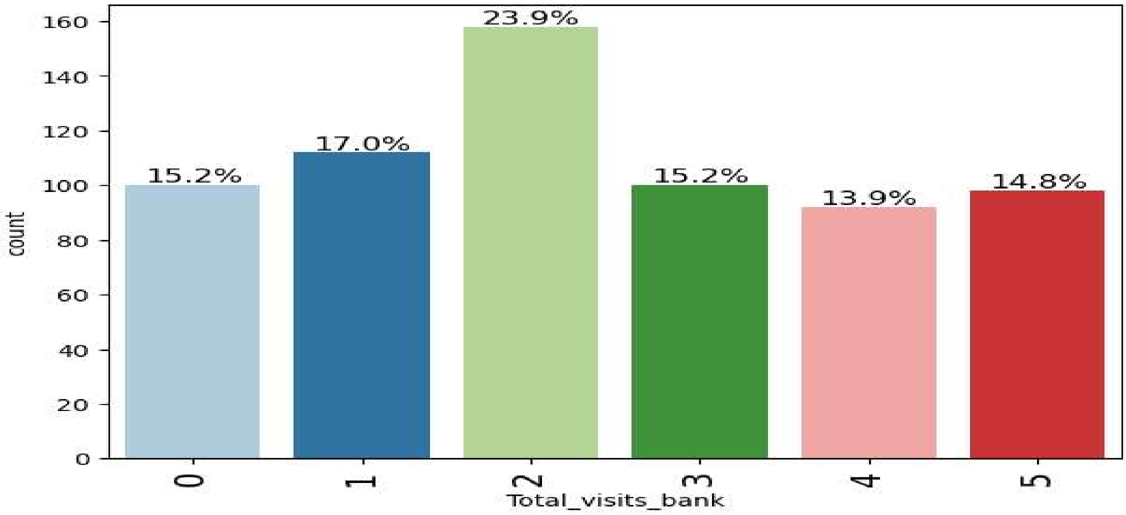


**Fig 7- Bar plot total visits bank**

Observations: Around 23.9% or 160 customer have visited bank 2 times.

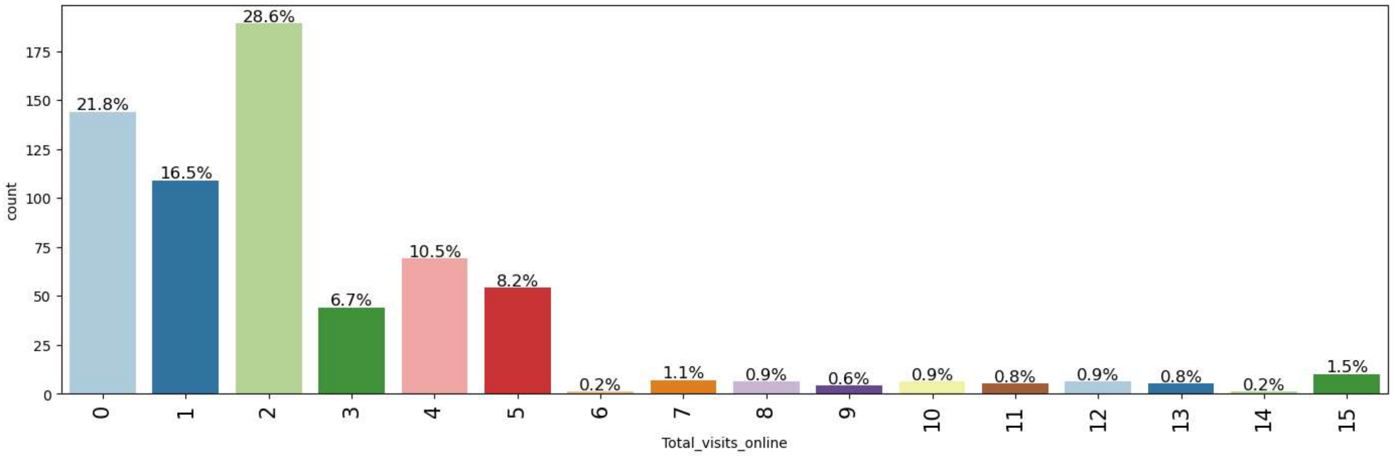


**Fig 8- Bar plot of total visits online**

Observations: 28.6% customers visits online 2 times. And around 21.8% customers never visited online portal.

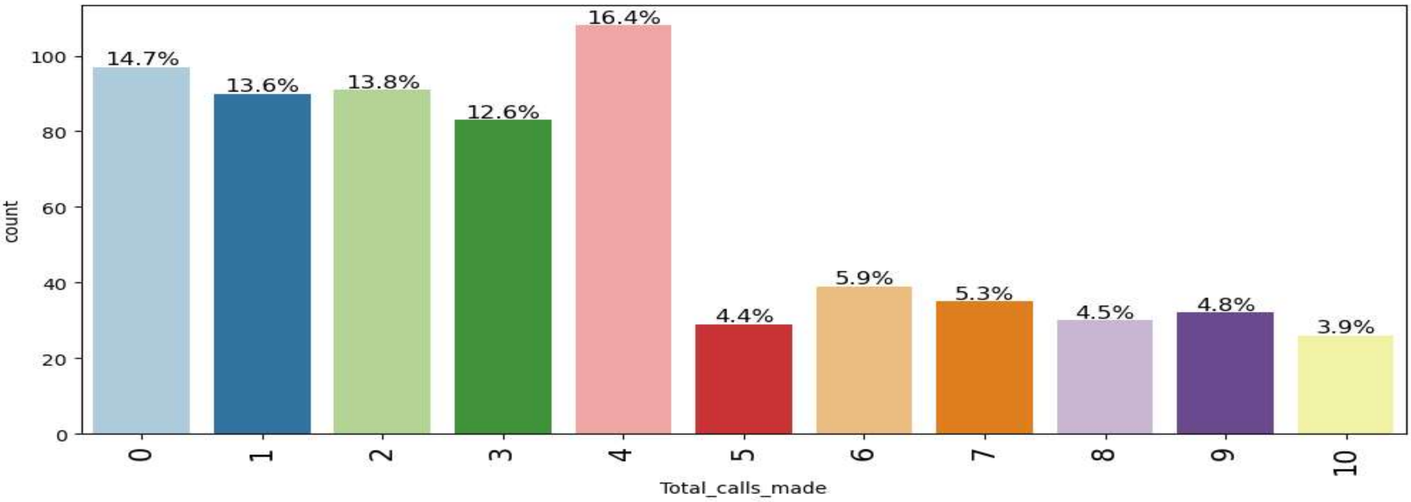


**Fig 9- Bar plot of total calls made**

Observations: Around 16.4% customers called the bank for their requirement 4 times. And 14.7% customers never called the bank.

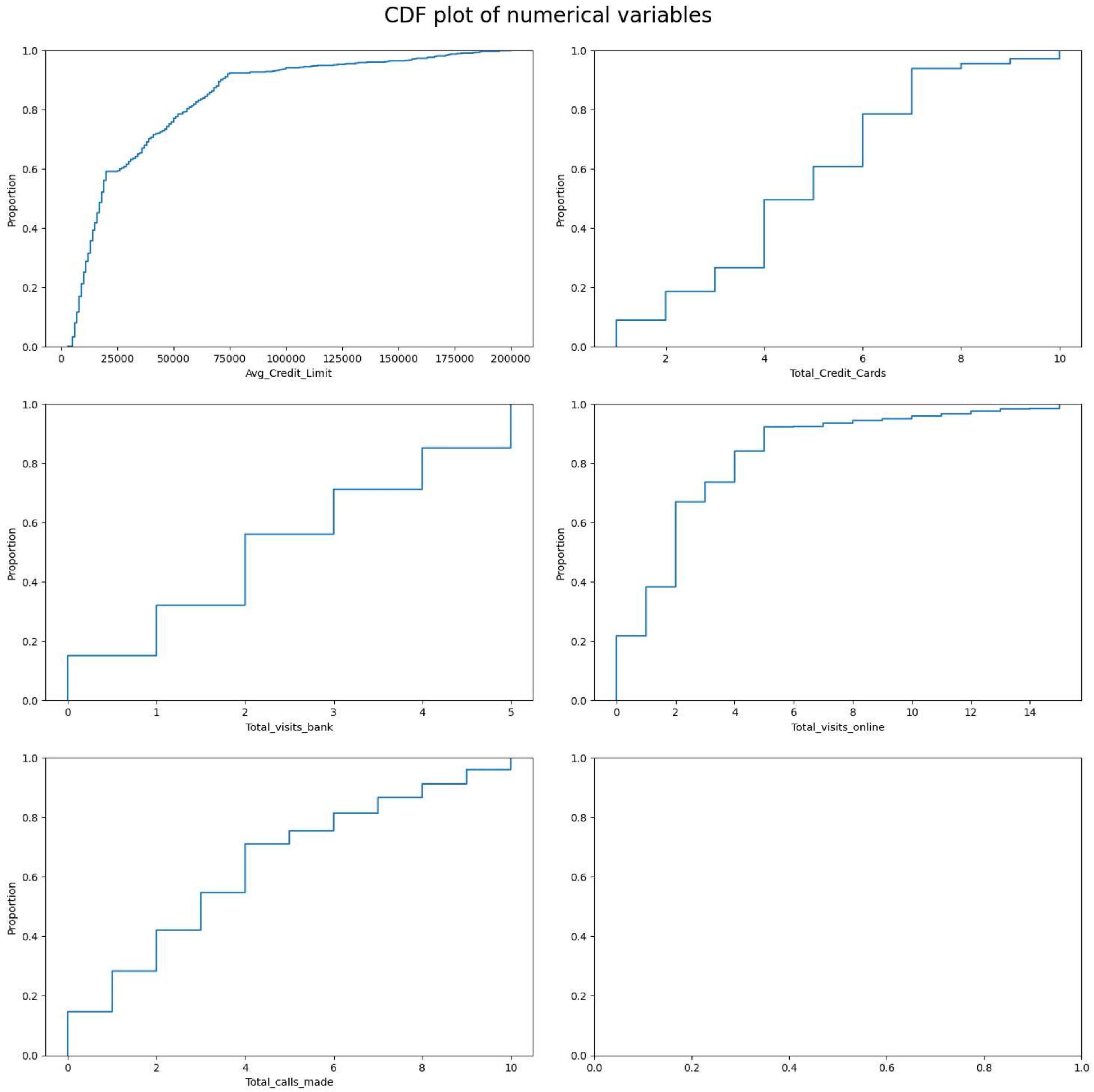


**Fig 10- CDF plot for numerical variables**

Observations:

1. 90% customers have credit limit of $75k or less.
2. 95% customers have 7 or less credit cards.
3. 90% customers have visited bank upto 4 times.
4. 95% customers have made 8 or less calls to bank.

## >Bivariate Analysis

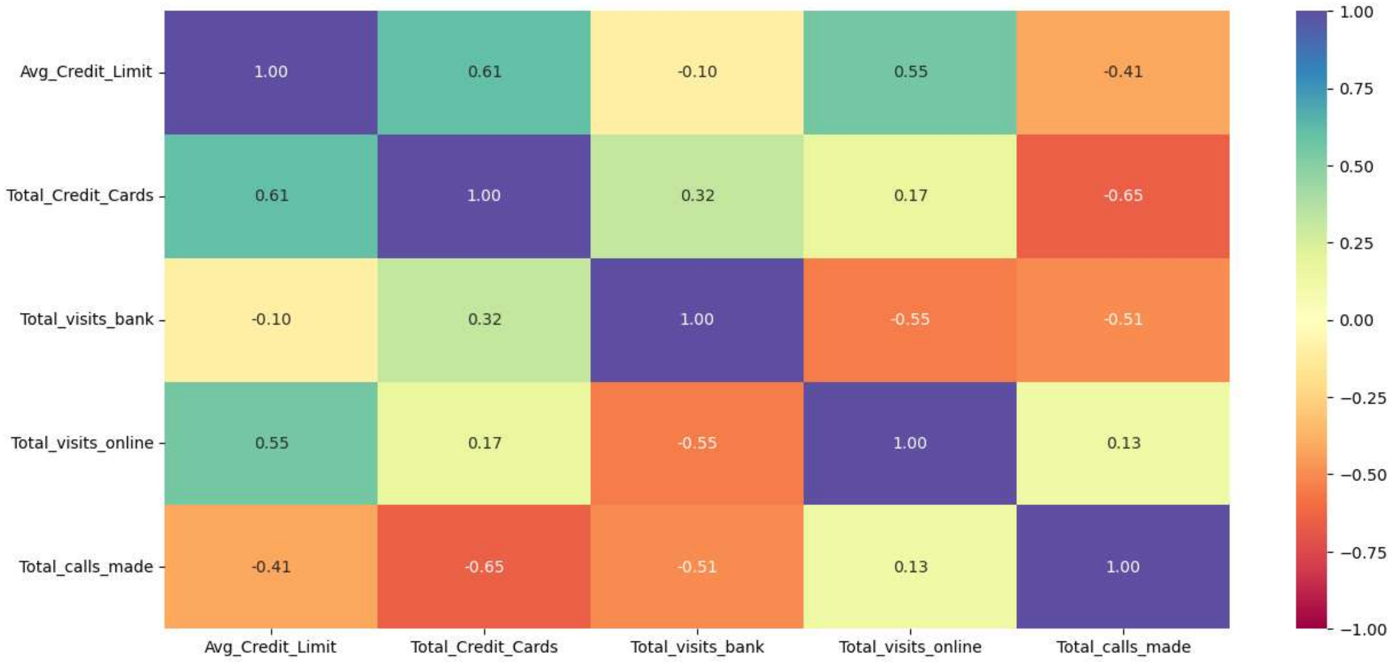


**Fig 11- Heatmap for correlations**

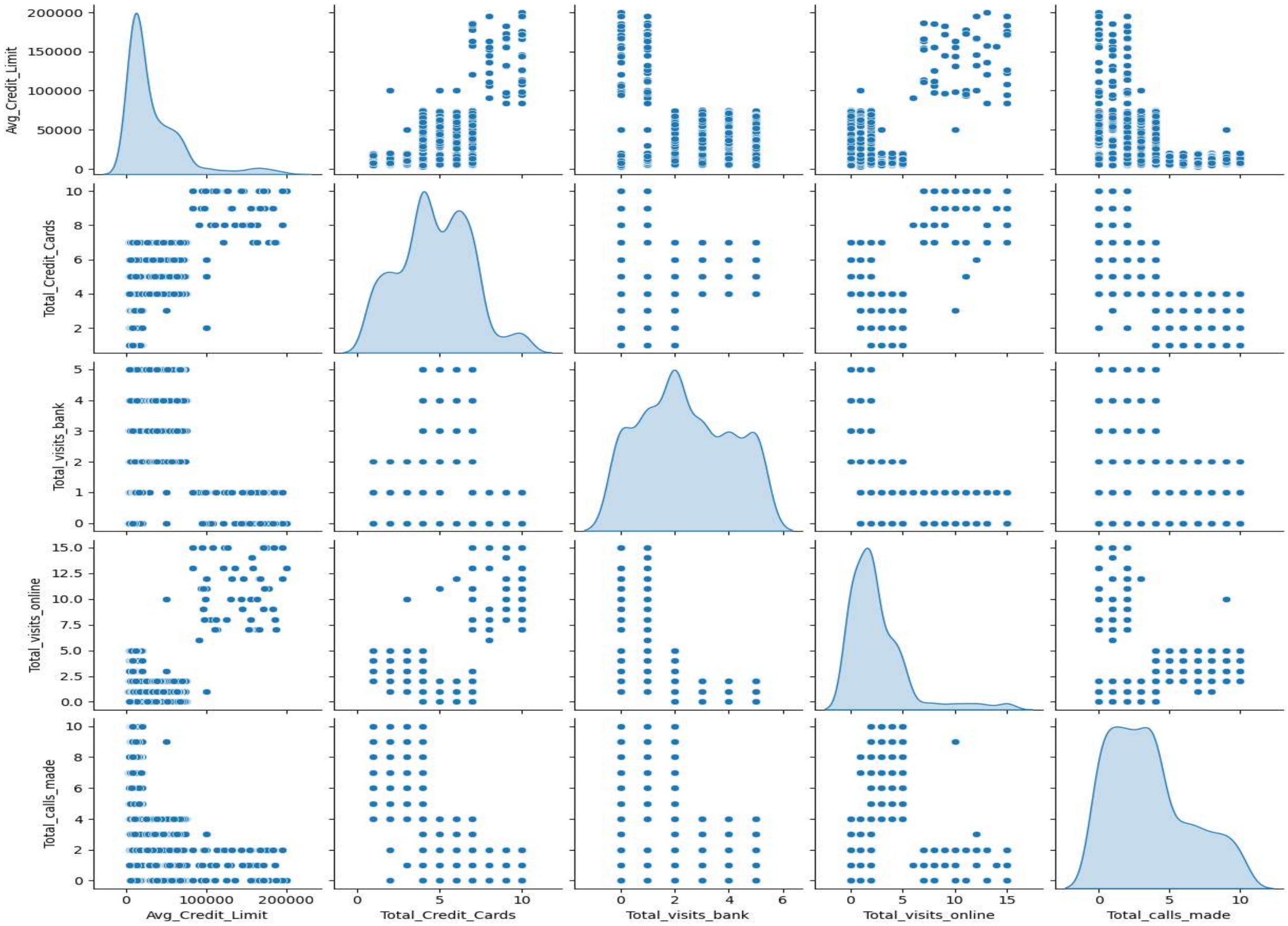


**Fig 12- Pairplot for bivariate analysis**

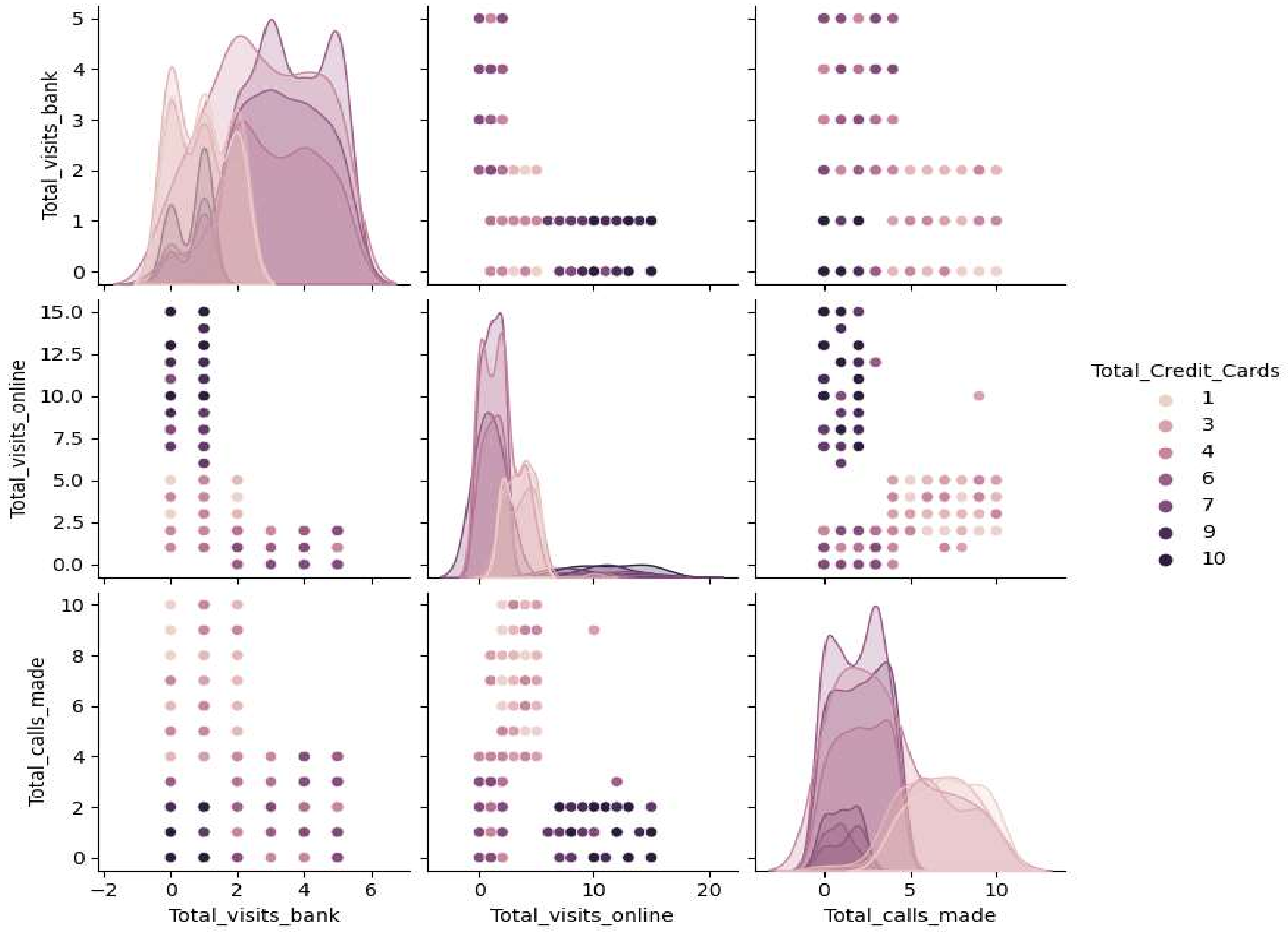


Fig 13- Pairplot to see relationship between all the features

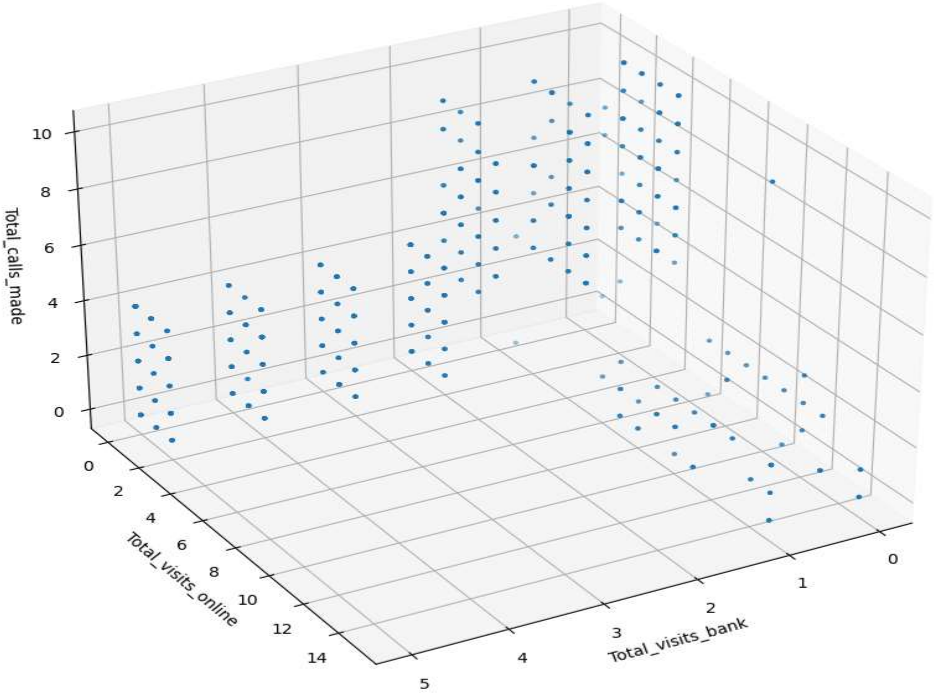


Fig 14- visualize the modes of contacting the bank in a 3D plot.

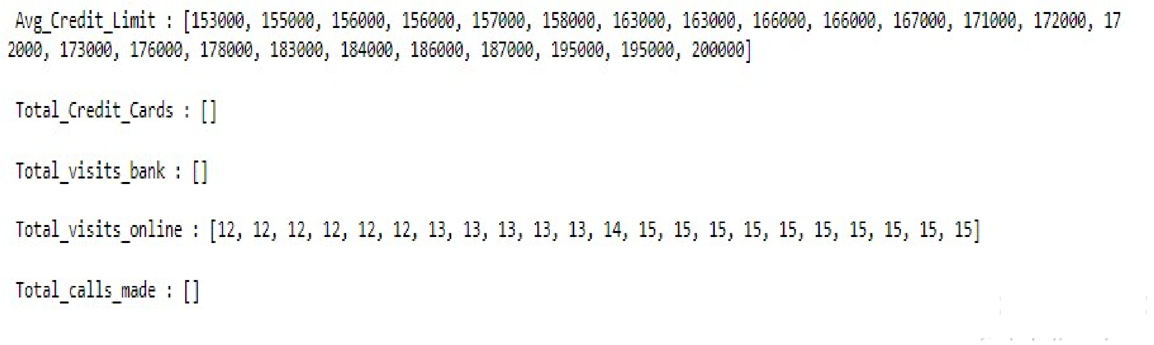
Overall observations from all the above plot:

- From the above heatmap plot there is a negative correlation between the three Customer Contact variables.
- Total credit cards and total visit online has medium positive correlation.
- Total credit cards and total visit bank has medium negative correlation.
- Total visit online and medium negative correlation with total visit bank.
- Lower the credit limit, higher the phone calls and higher the visits to the bank.
- From the 3D plot we can observe three segments of the customers by their preferred mode of contacting the bank.

## 2- Data preprocessing

### >Outlier Detection

The following are the outlier for the dataset;



```
 Avg_Credit_Limit : [153000, 155000, 156000, 156000, 157000, 158000, 163000, 163000, 166000, 166000, 167000, 171000, 172000, 17
2000, 173000, 176000, 178000, 183000, 184000, 186000, 187000, 195000, 195000, 200000]

 Total_Credit_Cards : []

 Total_visits_bank : []

 Total_visits_online : [12, 12, 12, 12, 12, 12, 13, 13, 13, 13, 13, 14, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15]

 Total_calls_made : []
```

Observations: The avg_credit_limit and total_visists_online shows outliers in the dataset.

## 3- Applying K-Means clustering

### >Checking elbow plot

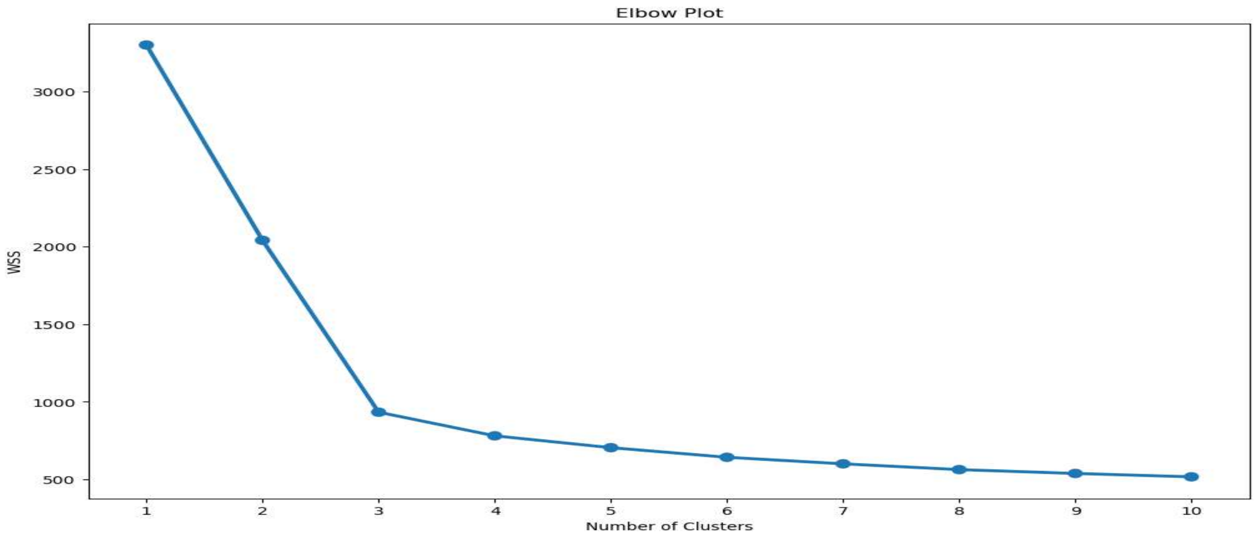


**Fig 15- Elbow plot for k-means clustering**

**>Checking Silhouette score**

- The Average Silhouette Score for 2 clusters is 0.41842
- The Average Silhouette Score for 3 clusters is 0.51572
- The Average Silhouette Score for 4 clusters is 0.35567
- The Average Silhouette Score for 5 clusters is 0.27175
- The Average Silhouette Score for 6 clusters is 0.25591
- The Average Silhouette Score for 7 clusters is 0.24791
- The Average Silhouette Score for 8 clusters is 0.22705
- The Average Silhouette Score for 9 clusters is 0.21339
- The Average Silhouette Score for 10 clusters is 0.21381

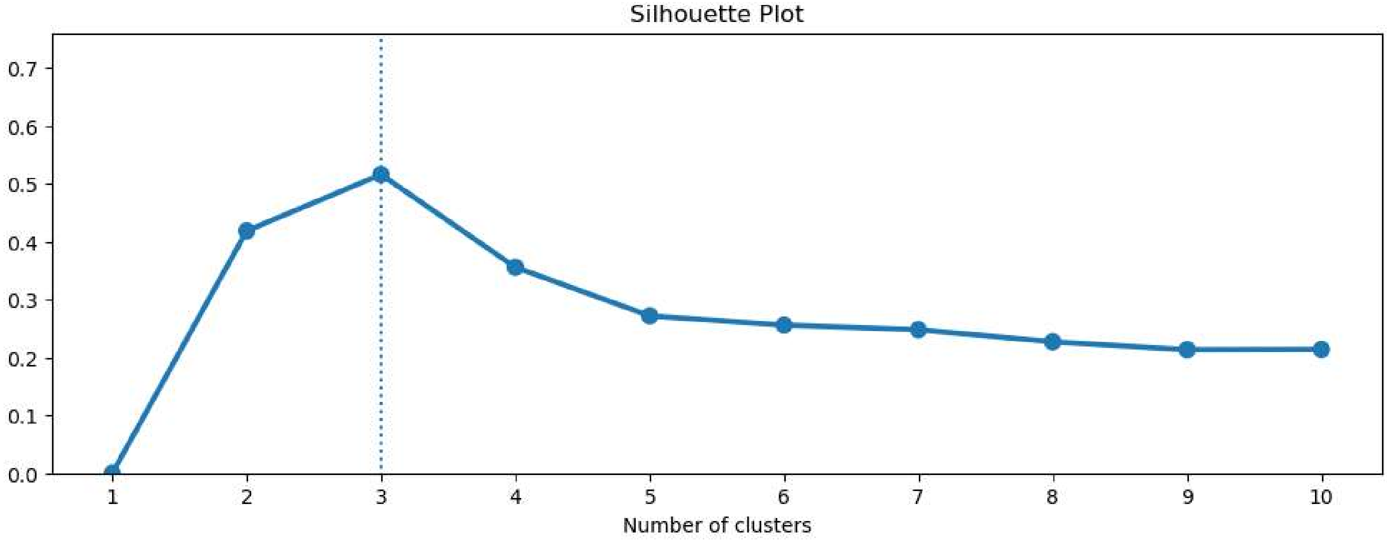
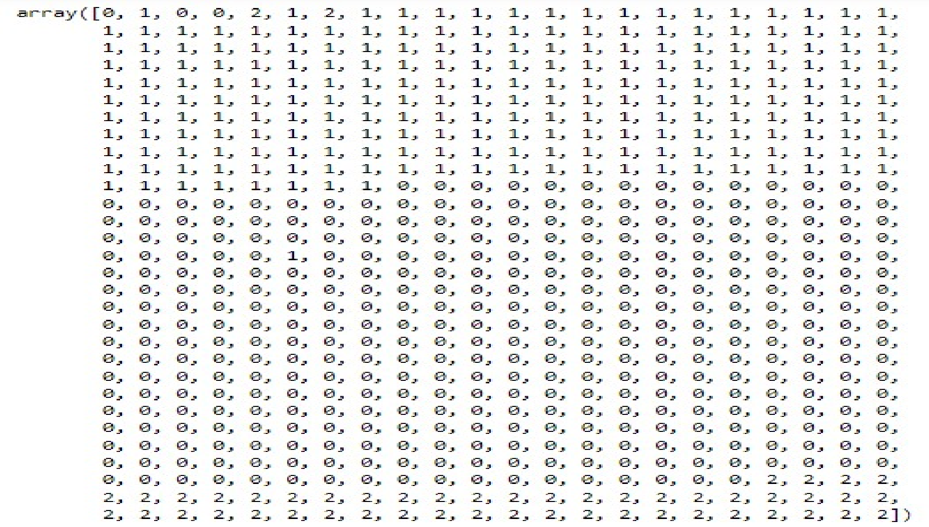


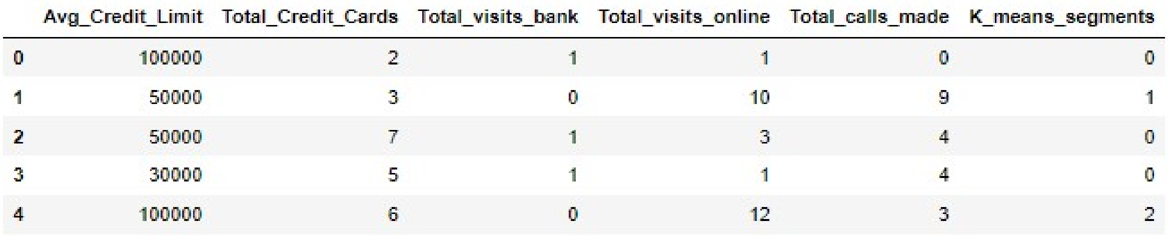
**Fig 16- Silhouette plot**

**>K-Means model**



```
array([0, 1, 0, 0, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

**>Adding K-Means segments into the original dataset**



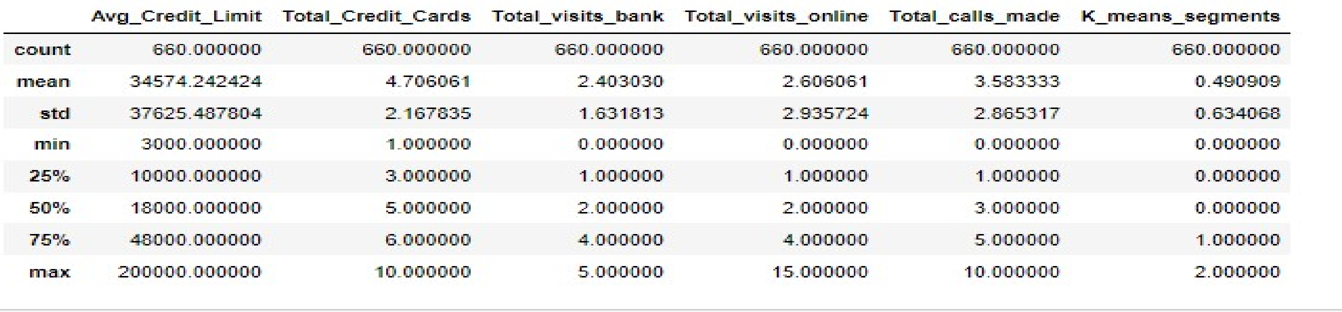
| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | K_means_segments |
|---|---|---|---|---|---|---|
| 0 | 100000 | 2 | 1 | 1 | 0 | 0 |
| 1 | 50000 | 3 | 0 | 10 | 9 | 1 |
| 2 | 50000 | 7 | 1 | 3 | 4 | 0 |
| 3 | 30000 | 5 | 1 | 1 | 4 | 0 |
| 4 | 100000 | 6 | 0 | 12 | 3 | 2 |

**Table 4- Adding k-means segments**

**>Cluster profiling**

```
K_means_segments
0    58.484848
1    33.939394
2     7.575758
Name: proportion, dtype: float64
```

The average of the k-means segments is 0.490.



| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | K_means_segments |
|---|---|---|---|---|---|---|
| count | 660.000000 | 660.000000 | 660.000000 | 660.000000 | 660.000000 | 660.000000 |
| mean | 34574.242424 | 4.706061 | 2.403030 | 2.606061 | 3.583333 | 0.490909 |
| std | 37625.487804 | 2.167835 | 1.631813 | 2.935724 | 2.865317 | 0.634068 |
| min | 3000.000000 | 1.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 25% | 10000.000000 | 3.000000 | 1.000000 | 1.000000 | 1.000000 | 0.000000 |
| 50% | 18000.000000 | 5.000000 | 2.000000 | 2.000000 | 3.000000 | 0.000000 |
| 75% | 48000.000000 | 6.000000 | 4.000000 | 4.000000 | 5.000000 | 1.000000 |
| max | 200000.000000 | 10.000000 | 5.000000 | 15.000000 | 10.000000 | 2.000000 |

**Table 5- Statistical summary of the dataset with k-means segments**

| K_means_segments | 0 | 1 | 2 |
|---|---|---|---|
| Avg_Credit_Limit | 33782.38 | 12174.11 | 141040.00 |
| Total_Credit_Cards | 5.52 | 2.41 | 8.74 |
| Total_visits_bank | 3.49 | 0.93 | 0.60 |
| Total_visits_online | 0.98 | 3.55 | 10.90 |
| Total_calls_made | 2.00 | 6.87 | 1.08 |
| freq | 386.00 | 224.00 | 50.00 |

## 4- Hierarchical Clustering

After applying hierarchical method we plot a dendogram plot

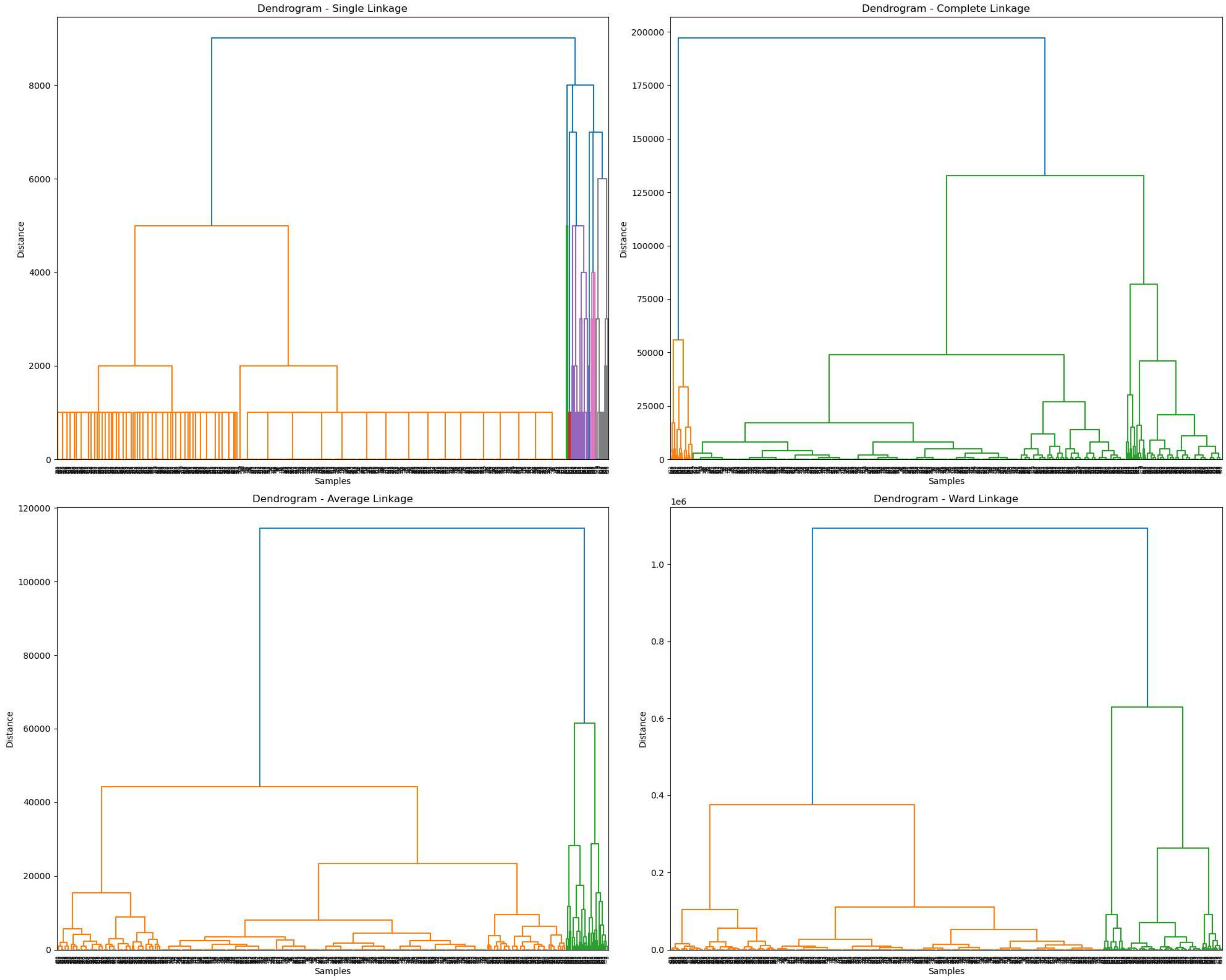
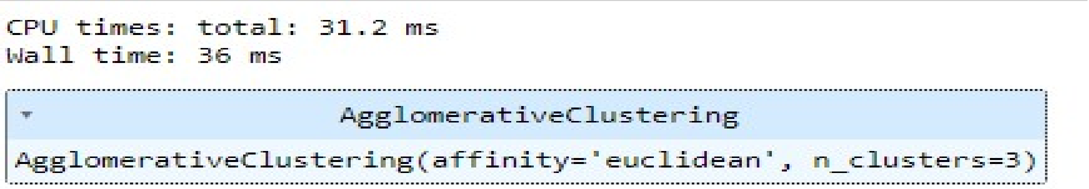


**Fig 17- Dendogram for different linkage method**

### >Lets check silhouette score

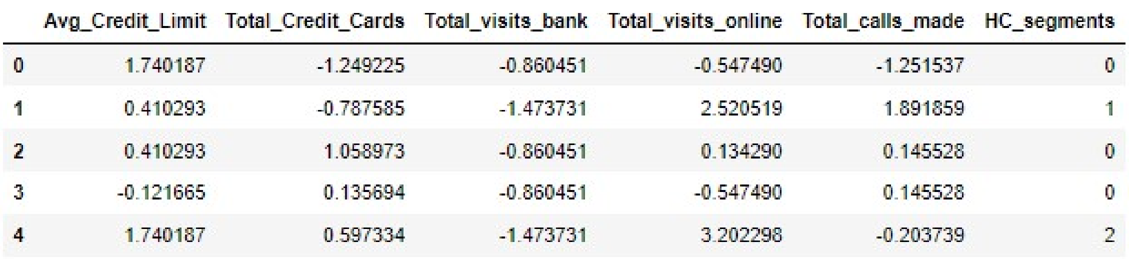
- For n_clusters = 2, silhouette score is 0.41770414762109936
- For n_clusters = 3, silhouette score is 0.5147639589979518
- For n_clusters = 4, silhouette score is 0.348082258126694
- For n_clusters = 5, silhouette score is 0.25691777326808435
- For n_clusters = 6, silhouette score is 0.2267784972566746
- For n_clusters = 7, silhouette score is 0.21629686854980873
- For n_clusters = 8, silhouette score is 0.21869490619485493
- For n_clusters = 9, silhouette score is 0.19527074669218447

### >Creating final model



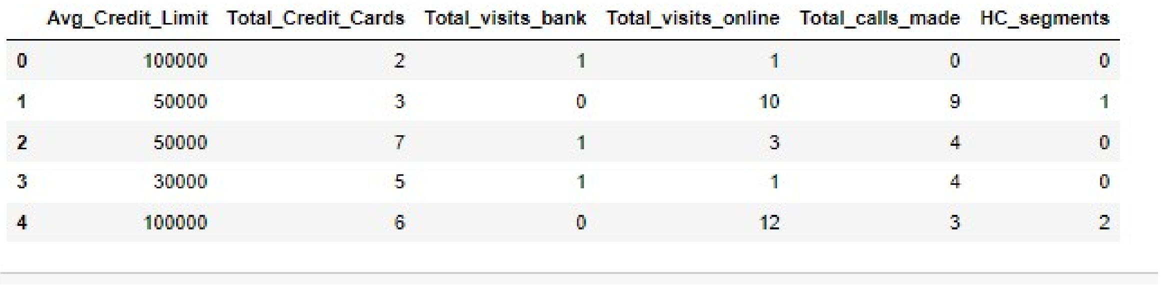
```
CPU times: total: 31.2 ms
Wall time: 36 ms
```

```
AgglomerativeClustering(affinity='euclidean', n_clusters=3)
```



| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | HC_segments |
|---|---|---|---|---|---|---|
| 0 | 1.740187 | -1.249225 | -0.860451 | -0.547490 | -1.251537 | 0 |
| 1 | 0.410293 | -0.787585 | -1.473731 | 2.520519 | 1.891859 | 1 |
| 2 | 0.410293 | 1.058973 | -0.860451 | 0.134290 | 0.145528 | 0 |
| 3 | -0.121665 | 0.135694 | -0.860451 | -0.547490 | 0.145528 | 0 |
| 4 | 1.740187 | 0.597334 | -1.473731 | 3.202298 | -0.203739 | 2 |

**Table 6- Adding HC to the original dataset and top 5 rows of the dataset**

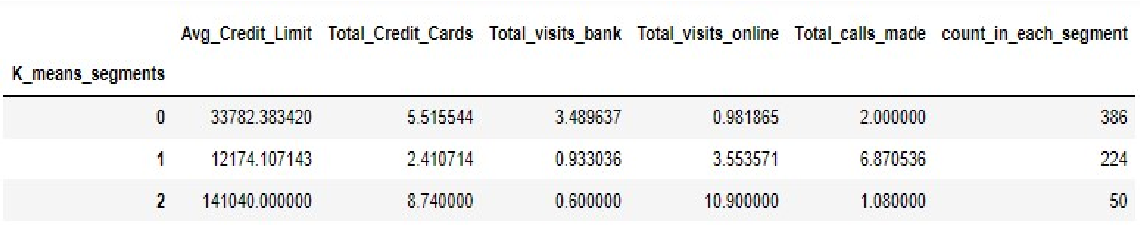


| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | HC_segments |
|---|---|---|---|---|---|---|
| 0 | 100000 | 2 | 1 | 1 | 0 | 0 |
| 1 | 50000 | 3 | 0 | 10 | 9 | 1 |
| 2 | 50000 | 7 | 1 | 3 | 4 | 0 |
| 3 | 30000 | 5 | 1 | 1 | 4 | 0 |
| 4 | 100000 | 6 | 0 | 12 | 3 | 2 |

**Table 7- Copy of the dataset with HC and top 5 rows of the dataset**

# 5- K-means vs Hierarchical Clustering

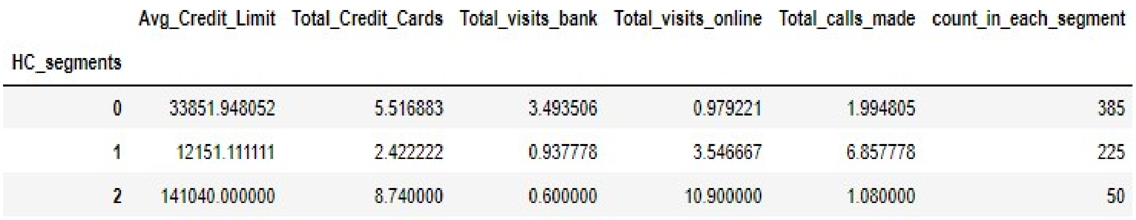
### >Cluster Profiling: K-means Clustering



| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | count_in_each_segment |
|---|---|---|---|---|---|---|
| K_means_segments | | | | | | |
| 0 | 33782.383420 | 5.515544 | 3.489637 | 0.981865 | 2.000000 | 386 |
| 1 | 12174.107143 | 2.410714 | 0.933036 | 3.553571 | 6.870536 | 224 |
| 2 | 141040.000000 | 8.740000 | 0.600000 | 10.900000 | 1.080000 | 50 |

**Table 8- K-means clustering**

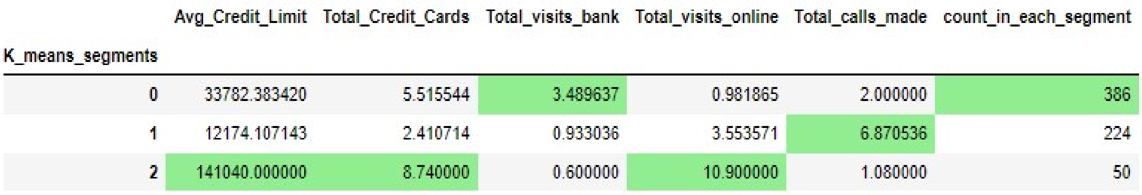
### >Cluster Profiling: Hierarchical Clustering



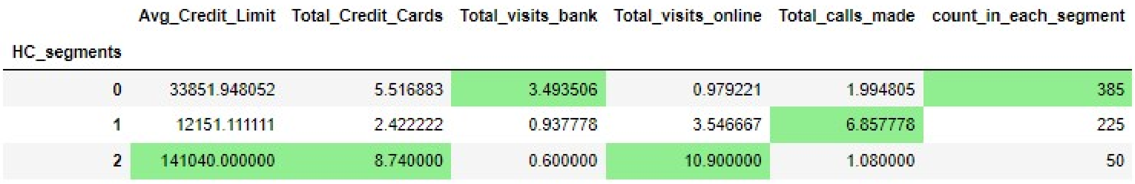
| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | count_in_each_segment |
|---|---|---|---|---|---|---|
| HC_segments | | | | | | |
| 0 | 33851.948052 | 5.516883 | 3.493506 | 0.979221 | 1.994805 | 385 |
| 1 | 12151.111111 | 2.422222 | 0.937778 | 3.546667 | 6.857778 | 225 |
| 2 | 141040.000000 | 8.740000 | 0.600000 | 10.900000 | 1.080000 | 50 |

**Table 9- Hierarchical Clustering**

### >K-means vs Hierarchical Clustering



| K_means_segments | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | count_in_each_segment |
|---|---|---|---|---|---|---|
| 0 | 33782.383420 | 5.515544 | 3.489637 | 0.981865 | 2.000000 | 386 |
| 1 | 12174.107143 | 2.410714 | 0.933036 | 3.553571 | 6.870536 | 224 |
| 2 | 141040.000000 | 8.740000 | 0.600000 | 10.900000 | 1.080000 | 50 |



| HC_segments | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | count_in_each_segment |
|---|---|---|---|---|---|---|
| 0 | 33851.948052 | 5.516883 | 3.493506 | 0.979221 | 1.994805 | 385 |
| 1 | 12151.111111 | 2.422222 | 0.937778 | 3.546667 | 6.857778 | 225 |
| 2 | 141040.000000 | 8.740000 | 0.600000 | 10.900000 | 1.080000 | 50 |

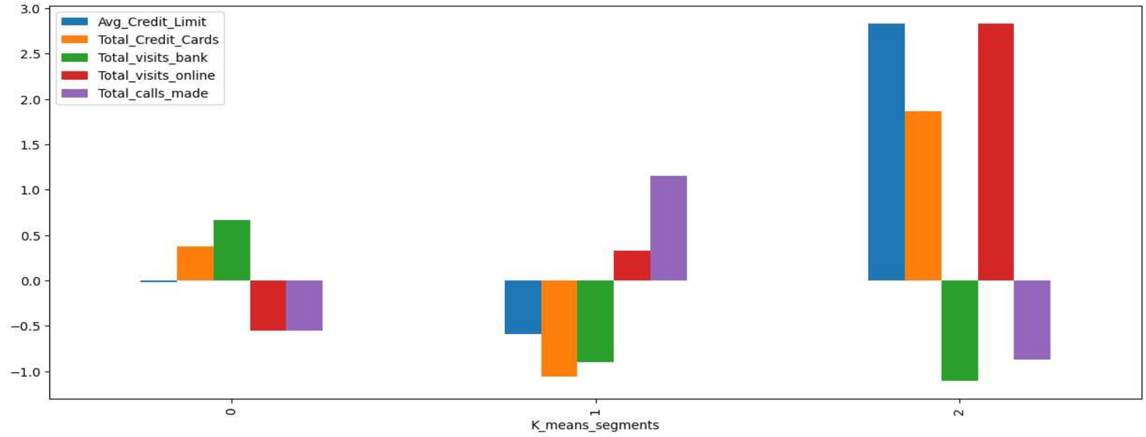


**Fig 18- K-means segments**

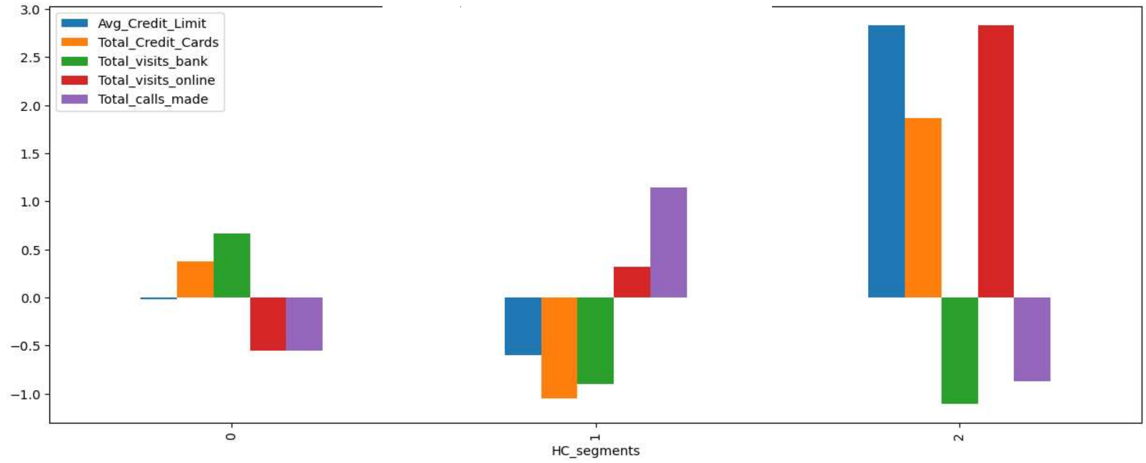


**Fig 19- HC Segments**

Observations:

- K-Means Clustering faster than the Hierarchical Clustering.
- From the above plot we say that both methods recommend 3 clusters as ideal for classification based on Communication method.
- From the above plots customers with higher Credit Limit and More Credit cards are identified by both clustering mechanism.
- We see that k-means and hierarchical clustering methods return similar Silhouette scores.

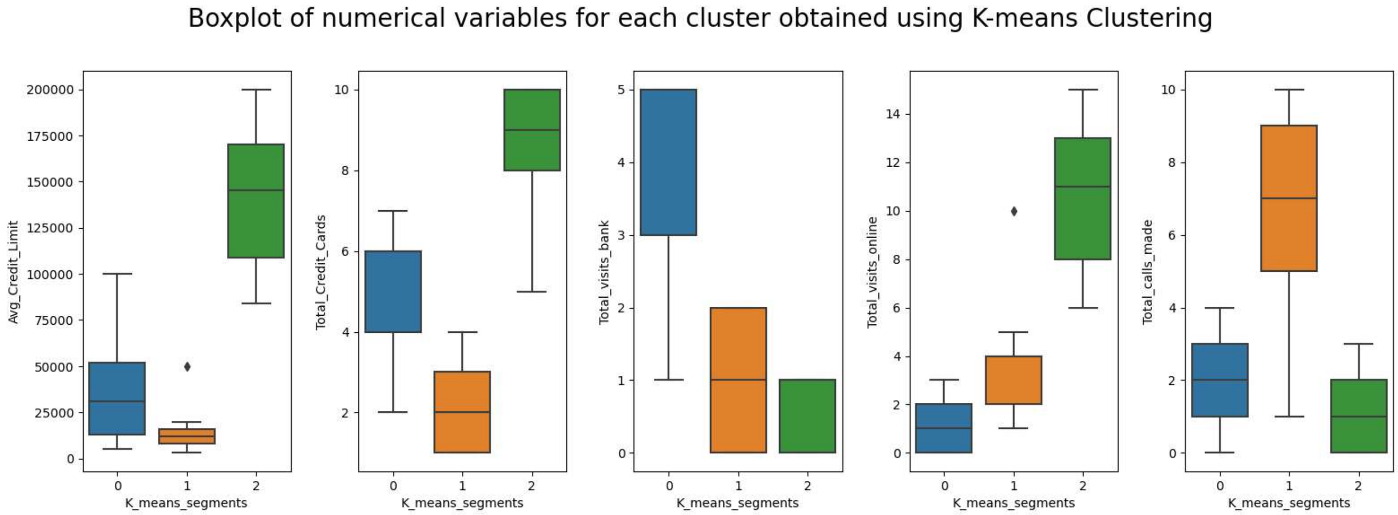


**Fig 20- Boxplot of numerical variables for each cluster obtained using k-means clustering**

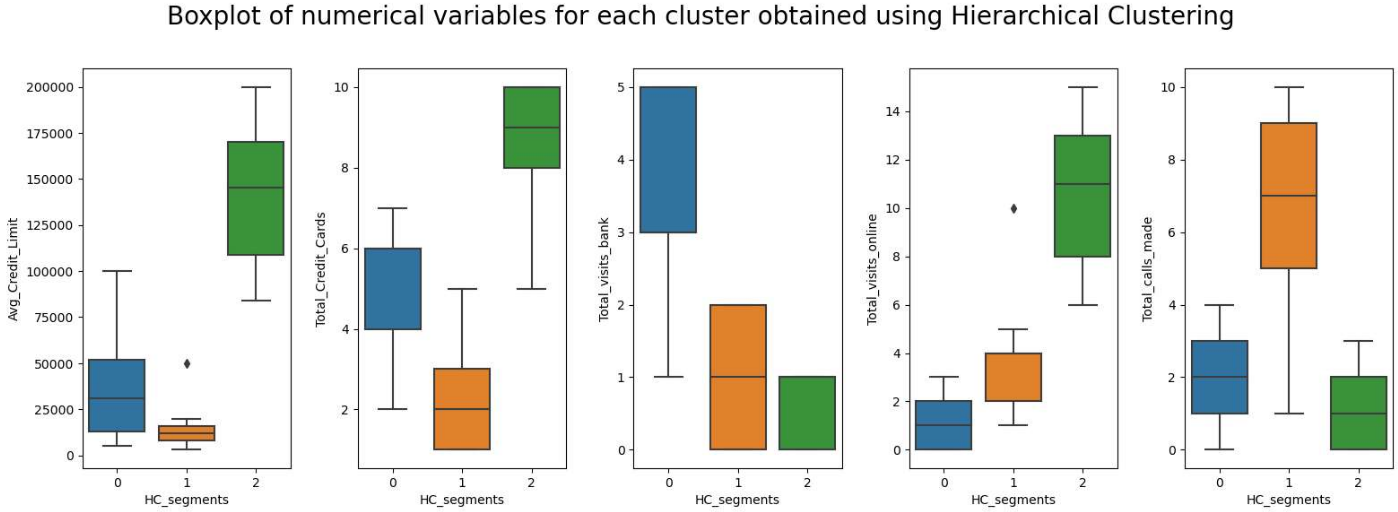


**Fig 21- Boxplot of numerical variables for each cluster obtained using hierarchical clustering**

## 6- Actionable Insights & Recommendations

### >Cluster Comparison

- K-Means Clustering faster than the Hierarchical Clustering.
- From the above plot we say that both methods recommend 3 clusters as ideal for classification based on Communication method.
- From the above plots customers with higher Credit Limit and More Credit cards are identified by both clustering mechanism.
- We see that k-means and hierarchical clustering methods return similar Silhouette scores.

### >Insights

- From the whole analysis we prefer the bank to handle the customer who visits banks, and provide them credits cards.
- Which of the customers called the bank try to handle phone calling transaction as well.
- Which of the customers visited online try to prefer them digital transaction because they have also the most credits cards.

**>Recommendations**

- Bank need to target Cluster 1 to publish credit cards services.
- Bank need to start searching for communication change so that the contact methods are changed according to customer preferences.
- From the above analysis by clustering we say that both clustering methods are classifying customers based on the Communication Method that customer uses to interact with bank, Business can choose either of the mechanism and implement recommendations based on the chosen clustering.

## 7- PCA Transformation

**>Bartletts Test of Sphericity**

Bartlett's test of sphericity tests the hypothesis that the variables are uncorrelated in the population.
$HO$: All variables in the data are uncorrelated

$HA$: At least one pair of variables in the data are correlated

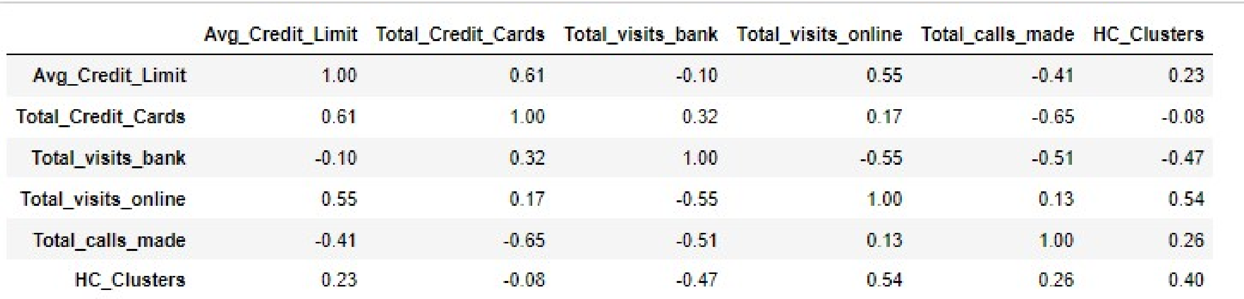
If the null hypothesis cannot be rejected, then PCA is not advisable.

The p_value of Bartlett sphericity is 0.0

**>KMO test**

Calculated KMO value is 0.718 so it’s expected to provide a considerable reduction is the dimension and extraction of meaningful components.

**>Step 1- Create the covariance Matrix**



| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | HC_Clusters |
|---|---|---|---|---|---|---|
| Avg_Credit_Limit | 1.00 | 0.61 | -0.10 | 0.55 | -0.41 | 0.23 |
| Total_Credit_Cards | 0.61 | 1.00 | 0.32 | 0.17 | -0.65 | -0.08 |
| Total_visits_bank | -0.10 | 0.32 | 1.00 | -0.55 | -0.51 | -0.47 |
| Total_visits_online | 0.55 | 0.17 | -0.55 | 1.00 | 0.13 | 0.54 |
| Total_calls_made | -0.41 | -0.65 | -0.51 | 0.13 | 1.00 | 0.26 |
| HC_Clusters | 0.23 | -0.08 | -0.47 | 0.54 | 0.26 | 0.40 |

**Table 10- covariance matrix**

**>Step 2- Get eigen values and eigen vector**

- Eigenvectors:
  ```
  [[-0.27 -0.52 -0.48  0.17  0.58  0.23]
   [ 0.55  0.3  -0.35  0.61 -0.07  0.33]
   [ 0.02 -0.25 -0.66 -0.31 -0.63 -0.08]
   [-0.23  0.74 -0.44 -0.32  0.32 -0.1 ]
   [-0.75  0.18  0.02  0.5  -0.38  0.12]
   [ 0.05 -0.06 -0.14  0.39  0.09 -0.9 ]]
  ```
- Eigenvalues: [2.33 2.18 0.32 0.28 0.25 0.05]
- Explained variance ratio: array([233., 218., 32., 28., 25., 5.])

**>Step 3 View Scree Plot to identify the number of components to be built**

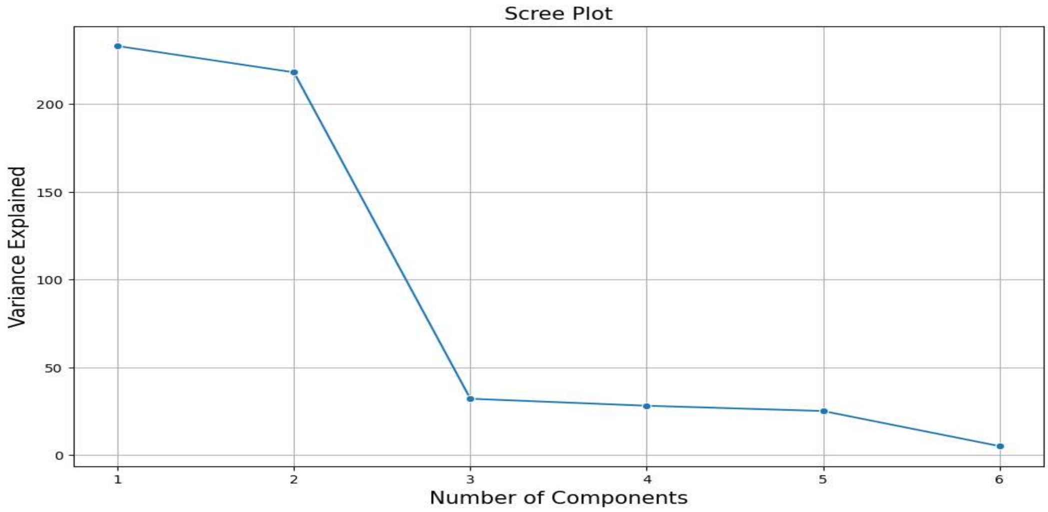
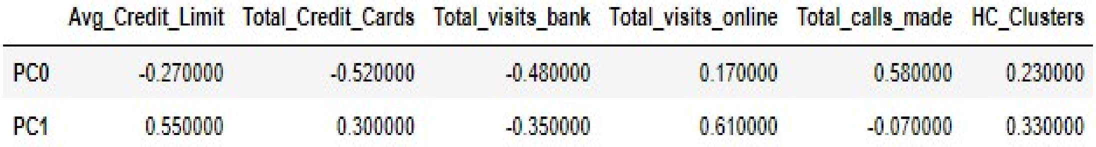


**Fig 22- Scree plot for number of components**

**>Step 4 Apply PCA for the number of decided components (n=2)**

```
array([[-0.35576078,  2.65667863, -0.25611274, ..., -0.98892453,
        -1.52152243, -0.7166506 ],
       [ 0.49447317,  2.07482131,  0.75630897, ...,  4.26280292,
         6.2028015 ,  5.53705454]])
```

The shape of the PCA components dataFrame is 2 rows and 6 columns.



| | Avg_Credit_Limit | Total_Credit_Cards | Total_visits_bank | Total_visits_online | Total_calls_made | HC_Clusters |
|---|---|---|---|---|---|---|
| PC0 | -0.270000 | -0.520000 | -0.480000 | 0.170000 | 0.580000 | 0.230000 |
| PC1 | 0.550000 | 0.300000 | -0.350000 | 0.610000 | -0.070000 | 0.330000 |

**Table 11- DataFrame of PCA**

**>Linear equation of first PC**

```
( -0.27 ) * Avg_Credit_Limit + ( -0.52 ) * Total_Credit_Cards + ( -0.48 ) *
Total_visits_bank + ( 0.17 ) * Total_visits_online + ( 0.58 ) * Total_calls_made + ( 0.23
) * HC_Clusters +
```

**>Linear equation of second PC**

```
( 0.55 ) * Avg_Credit_Limit + ( 0.3 ) * Total_Credit_Cards + ( -0.35 ) *
Total_visits_bank + ( 0.61 ) * Total_visits_online + ( -0.07 ) * Total_calls_made + (
0.33 ) * HC_Clusters +
```

## 8- Interpretation of Principal Components

- Avg_credit_limit has a negative suggesting that higher values of Avg_credit_limit are associated lower values of first PC.
- Total_credit_cards also shows a negative value so that means it associated with lower values of first PC.
- Total_visits_bank also shows a negative value so that means it associated with lower values of first PC.
- Total_visits_online shows a positive value that means it associated with higher values of first PC.
- Total_calls_made shows a positive value that means it associated with higher values of first PC.
- HC_cluster shows positive value that means it associated with higher value of first PC.

## >Variance Captured by PCs

The first two principal components explain 83.41% of the variance in the data.

## >Visualization of clusters formed

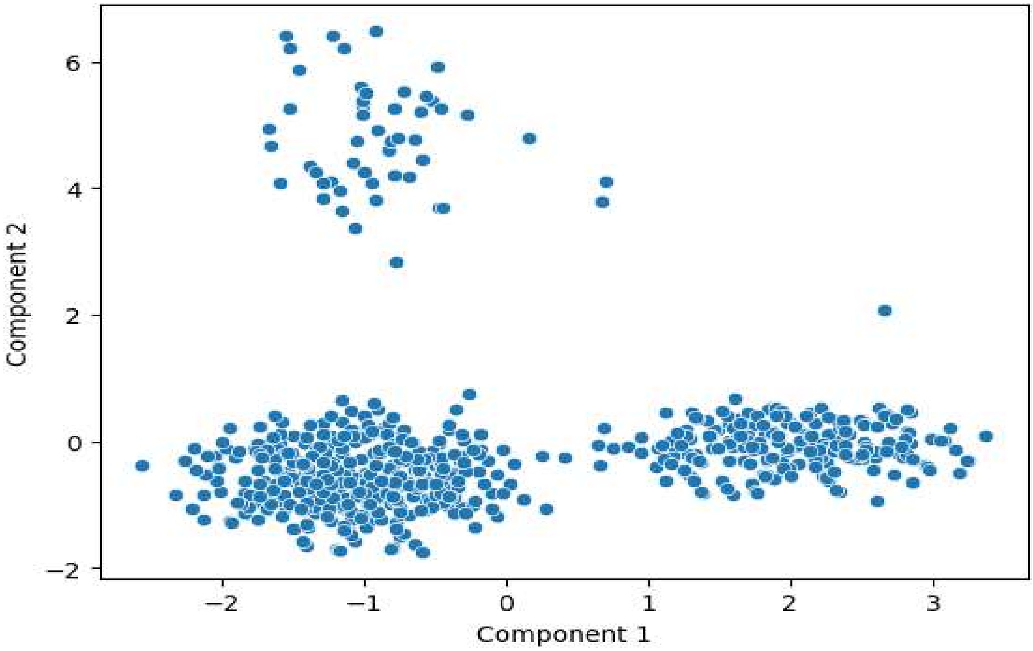


**Fig 23- Scatter plot of Clusters formed without hue**

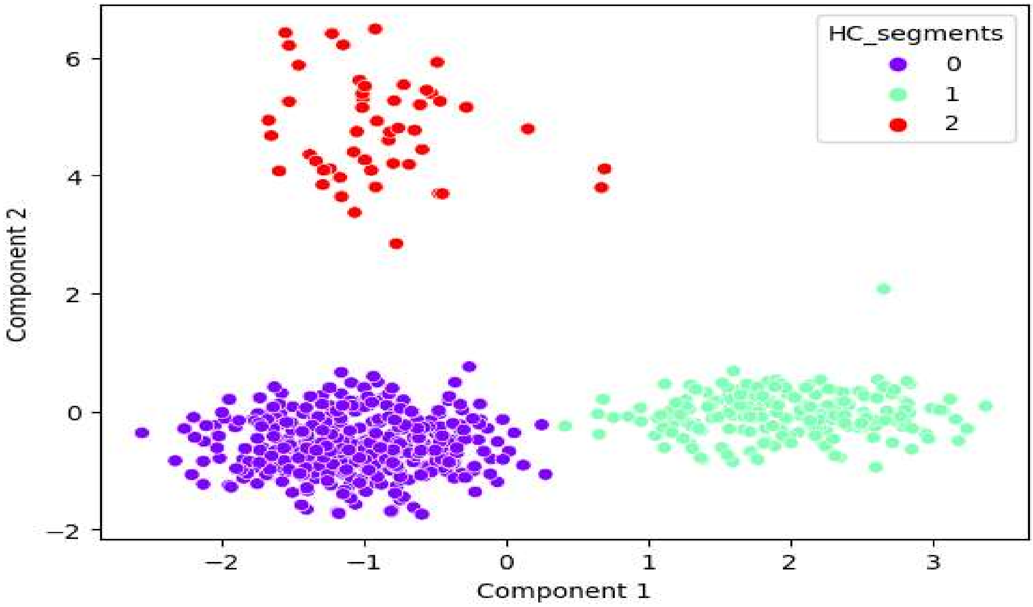


**Fig 24- Scatter plot of Clusters formed with hue**