PROJECT : DATA MINING

CLUSTERING & CART-RF-ANN

Name: Varsha Srinivasan

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF EQUATIONS

PROBLEM 1: Clustering

Problem Statement

A leading bank wants to develop a customer segmentation to give promotional offers to its customers. They collected a sample that summarizes the activities of users during the past few months. You are given the task to identify the segments based on credit card usage.

Introduction

The purpose of this whole exercise is to perform exploratory data analysis and perform Clustering using KMeans and Hierarchical techniques like Fclusters, Agglomerative clustering. The dataset consists of 210 rows with their features like spending, advance payments, probability of full payment, current balance, credit limit, min payment amount, maximum spent in single shopping. The number of clusters are determined and customer segmentation is performed. Insights are derived and recommendations are made for each cluster group.

Data Description

1. spending: Amount spent by the customer per month (in 1000s)
2. advance_payments: Amount paid by the customer in advance by cash (in 100s)
3. probability_of_full_payment: Probability of payment done in full by the customer to the bank
4. current_balance: Balance amount left in the account to make purchases (in 1000s)
5. credit_limit: Limit of the amount in credit card (10000s)
6. min_payment_amt : minimum paid by the customer while making payments for purchases made monthly (in 100s)
7. max_spent_in_single_shopping: Maximum amount spent in one purchase (in 1000s)

1.1 Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis).

Sample of the dataset:

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
0	19.94	16.92	0.8752	6.675	3.763	3.252	6.550
1	15.99	14.89	0.9064	5.363	3.582	3.336	5.144
2	18.95	16.42	0.8829	6.248	3.755	3.368	6.148
3	10.83	12.96	0.8099	5.278	2.641	5.182	5.185
4	17.99	15.86	0.8992	5.890	3.694	2.068	5.837

Table 1: Sample of the Dataset1

The data is read from the csv file and the above tables shows the first 5 rows of the dataset.

EXPLORATORY DATA ANALYSIS

Data Type and Missing Values

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 210 entries, 0 to 209
Data columns (total 7 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   spending                      210 non-null    float64
 1   advance_payments              210 non-null    float64
 2   probability_of_full_payment   210 non-null    float64
 3   current_balance               210 non-null    float64
 4   credit_limit                  210 non-null    float64
 5   min_payment_amt               210 non-null    float64
 6   max_spent_in_single_shopping  210 non-null    float64
dtypes: float64(7)
memory usage: 11.6 KB
```

Table 2: Dataset Info1

```
spending                      0
advance_payments              0
```

```
probability_of_full_payment      0
current_balance                  0
credit_limit                     0
min_payment_amt                  0
max_spent_in_single_shopping     0
dtype: int64
```

There are no null values in the dataset

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
count	210.000000	210.000000	210.000000	210.000000	210.000000	210.000000	210.000000
mean	14.847524	14.559286	0.870999	5.628533	3.258605	3.700201	5.408071
std	2.909699	1.305959	0.023629	0.443063	0.377714	1.503557	0.491480
min	10.590000	12.410000	0.808100	4.899000	2.630000	0.765100	4.519000
25%	12.270000	13.450000	0.856900	5.262250	2.944000	2.561500	5.045000
50%	14.355000	14.320000	0.873450	5.523500	3.237000	3.599000	5.223000
75%	17.305000	15.715000	0.887775	5.979750	3.561750	4.768750	5.877000
max	21.180000	17.250000	0.918300	6.675000	4.033000	8.456000	6.550000

Table 3:Description of Dataset1

The total number of entries is 210 and there are 210 non-null values in the dataset as shown above. Hence there are no missing values. All features are of the float data type. Spending has a mean of around 14. Advance payments have a mean of 14. Probability of full payment has a mean of 0.87. Current balance has a mean of 5.63. Credit limit has a mean of 3.26. Min payment amount has a mean of 3.7. Maximum spent in single shopping has a mean of 5.40

```
0
```

There are no duplicates in the dataset

There are no duplicates in the dataset.

Outliers, Proportions and Treatment

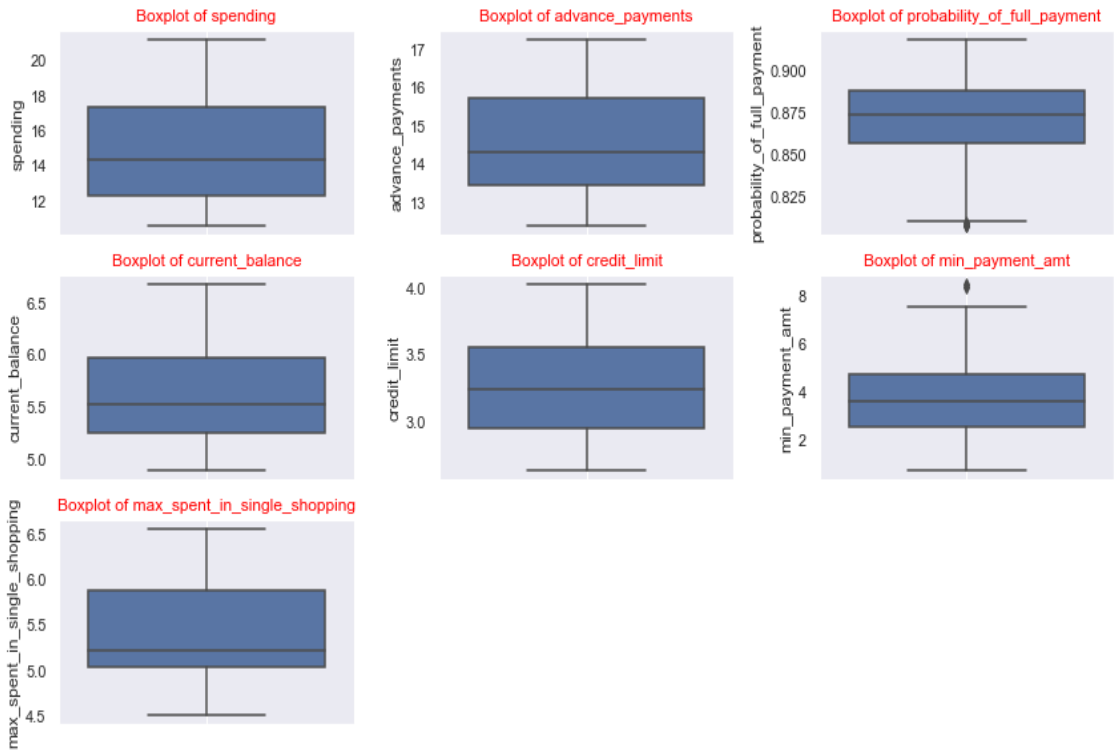

Figure 1: Boxplot with Outliers

There are few outliers in probability of full payment and minimum payment amount.

```
Lower outliers in spending is :  4.717499999999999
Upper outliers in spending is :  24.8575
Number of outliers in spending upper :  0
Number of outliers in spending lower :  0
% of Outlier in spending upper:  0 %
% of Outlier in spending lower:  0 %
-----------------------------------------------------
Lower outliers in advance_payments is :  10.052499999999998
Upper outliers in advance_payments is :  19.1125
Number of outliers in advance_payments upper :  0
Number of outliers in advance_payments lower :  0
% of Outlier in advance_payments upper:  0 %
% of Outlier in advance_payments lower:  0 %
-----------------------------------------------------
Lower outliers in probability_of_full_payment is :  0.8105875
Upper outliers in probability_of_full_payment is :  0.9340875
Number of outliers in probability_of_full_payment upper :  0
Number of outliers in probability_of_full_payment lower :  3
% of Outlier in probability_of_full_payment upper:  0 %
% of Outlier in probability_of_full_payment lower:  1 %
-----------------------------------------------------
Lower outliers in current_balance is :  4.186
Upper outliers in current_balance is :  7.056000000000001
Number of outliers in current_balance upper :  0
Number of outliers in current_balance lower :  0
% of Outlier in current_balance upper:  0 %
% of Outlier in current_balance lower:  0 %
-----------------------------------------------------
Lower outliers in credit_limit is :  2.017375
Upper outliers in credit_limit is :  4.488375
Number of outliers in credit_limit upper :  0
Number of outliers in credit_limit lower :  0
% of Outlier in credit_limit upper:  0 %
% of Outlier in credit_limit lower:  0 %
-----------------------------------------------------
Lower outliers in min_payment_amt is :  -0.7493749999999992
Upper outliers in min_payment_amt is :  8.079625
Number of outliers in min_payment_amt upper :  2
Number of outliers in min_payment_amt lower :  0
% of Outlier in min_payment_amt upper:  1 %
% of Outlier in min_payment_amt lower:  0 %
-----------------------------------------------------
Lower outliers in max_spent_in_single_shopping is :  3.797
Upper outliers in max_spent_in_single_shopping is :  7.125
Number of outliers in max_spent_in_single_shopping upper :  0
Number of outliers in max_spent_in_single_shopping lower :  0
% of Outlier in max_spent_in_single_shopping upper:  0 %
% of Outlier in max_spent_in_single_shopping lower:  0 %
-----------------------------------------------------
```

Min payment has 2 outliers above the upper whisker and Probability of full payment has 3 outliers below the lower whisker. Remaining features doesn't have any outliers.

Treatment of outliers is done by assigning the upper whisker value to the values of the outliers above the upper whisker and assigning the lower whisker value to the values of the outliers below the lower whisker.

Univariate Analysis

Box Plot

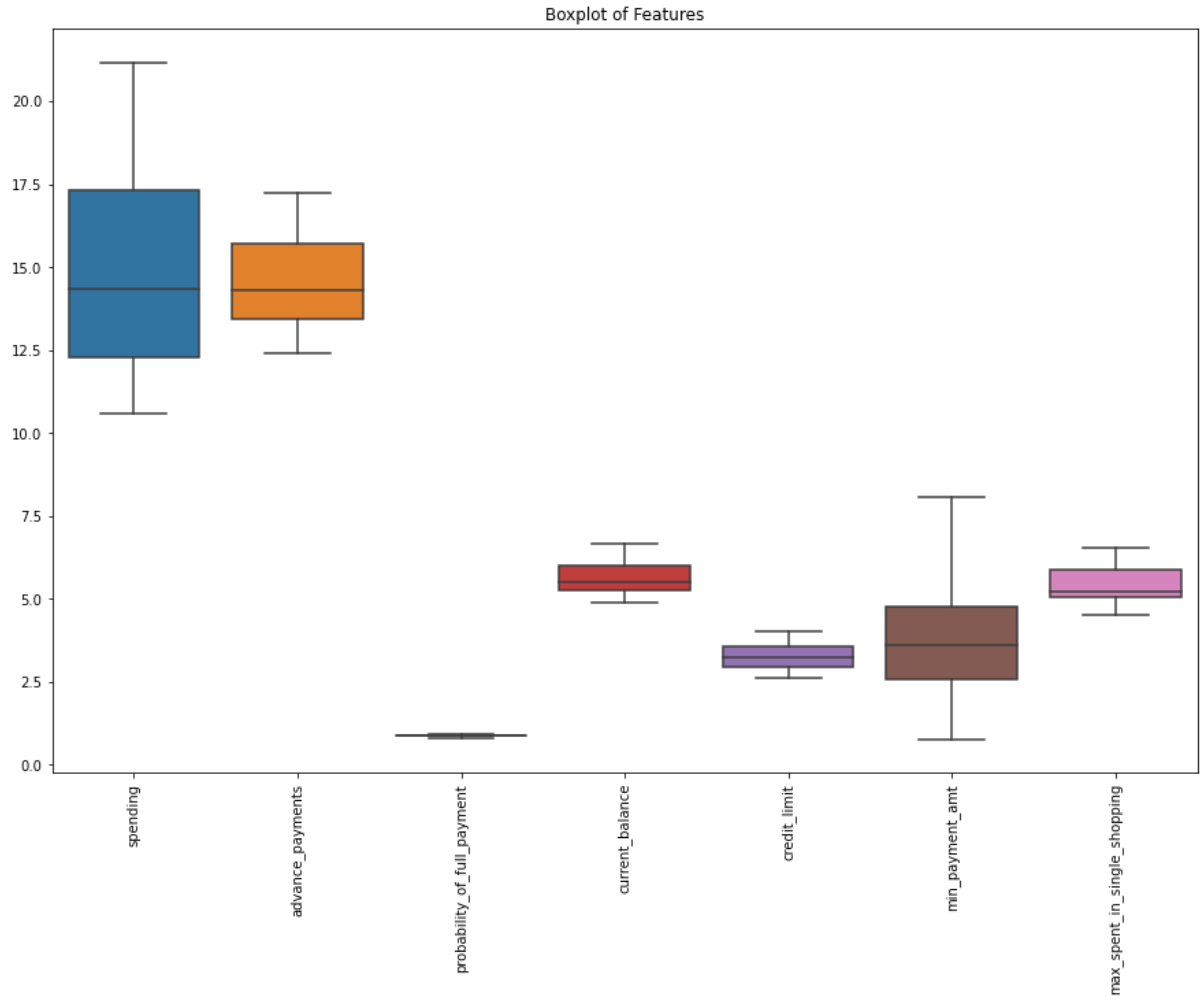

Figure 2:Boxplot without Outliers

Spending has a median of around 14. Advance payment has a median of around 14. Probability of full payment has a median of around 1. Current balance has a median of 5.5. Credit limit has a median of 3. Min payment amount has a median of 3.5. Max spent in single shopping has a median of 5.

Description of spending

```
-------------------------------------------------------------------------
-----
count    210.000000
mean      14.847524
std        2.909699
min       10.590000
25%       12.270000
50%       14.355000
75%       17.305000
max       21.180000
Name: spending, dtype: float64
```

```
Interquartile range (IQR) of spending is  5.035
Range of values:  10.59
```

Distribution of Spending

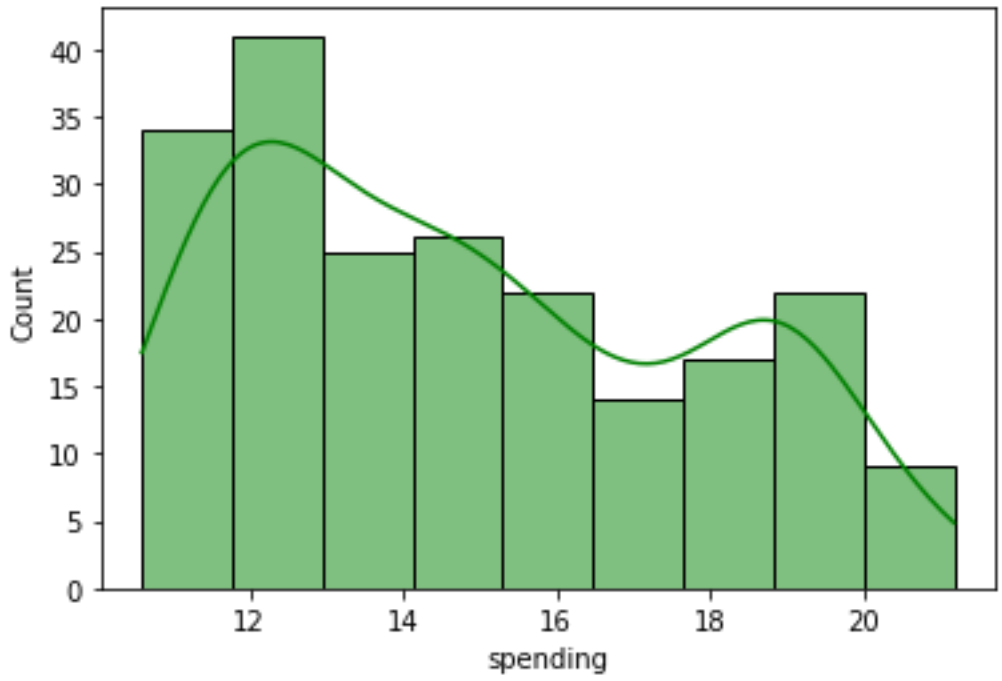

Figure 3: Distribution of Spending

Description of advance_payments

```
-------------------------------------------------------------------------
-----
count    210.000000
mean      14.559286
std        1.305959
min       12.410000
25%       13.450000
50%       14.320000
75%       15.715000
max       17.250000
Name: advance_payments, dtype: float64

Interquartile range (IQR) of spending is  2.265
Range of values:  4.84
```

Distribution of advance payments

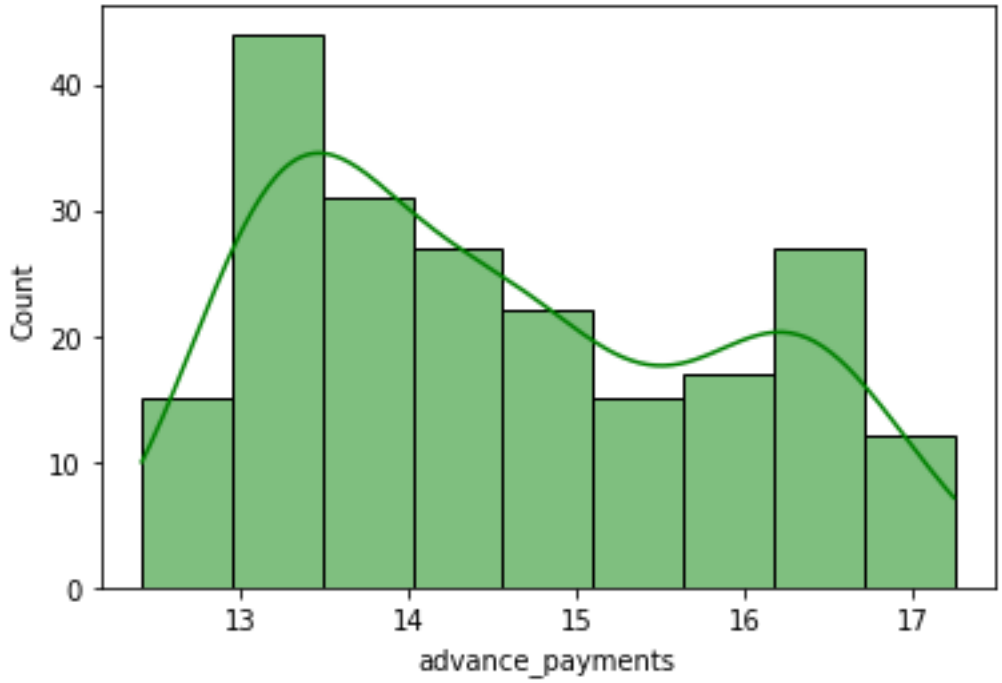

Figure 4: Distribution of Advance Payments

Description of probability_of_full_payment

```
-------------------------------------------------------------------------
-----
count    210.000000
mean       0.871025
std        0.023560
min        0.810588
25%        0.856900
50%        0.873450
75%        0.887775
max        0.918300
Name: probability_of_full_payment, dtype: float64

Interquartile range (IQR) of spending is  0.031
Range of values:  0.108
```

Distribution of probability_of_full_payment

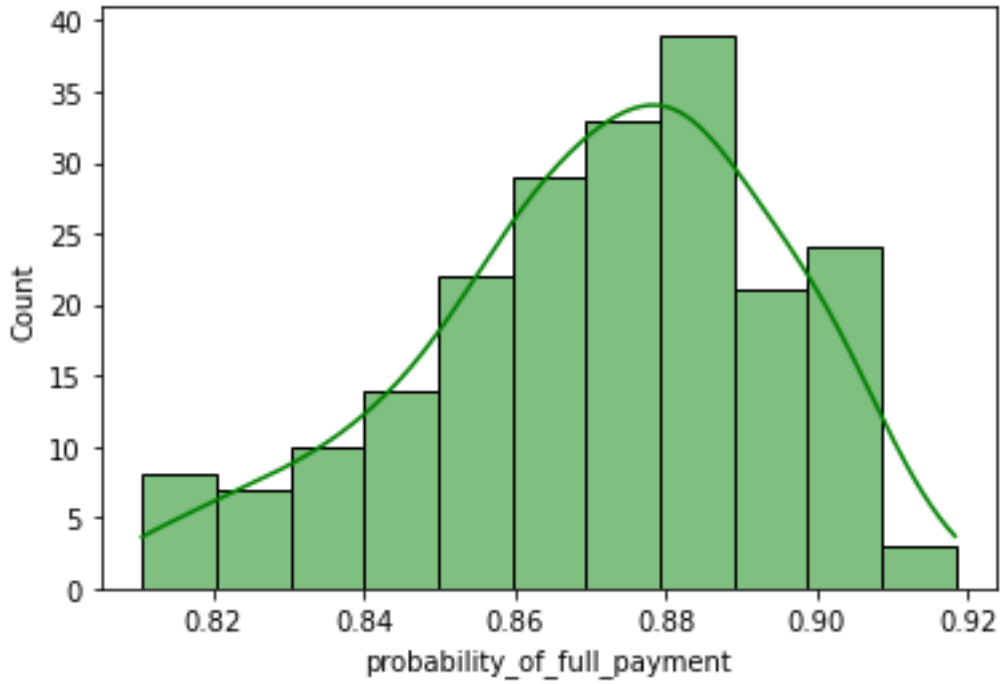

Figure 5: Distribution of Probability of Full Payment

Description of current_balance

```
-------------------------------------------------------------------------
-----
count    210.000000
mean       5.628533
std        0.443063
min        4.899000
25%        5.262250
50%        5.523500
75%        5.979750
max        6.675000
Name: current_balance, dtype: float64

Interquartile range (IQR) of spending is  0.718
Range of values:  1.776
```

Distribution of current_balance

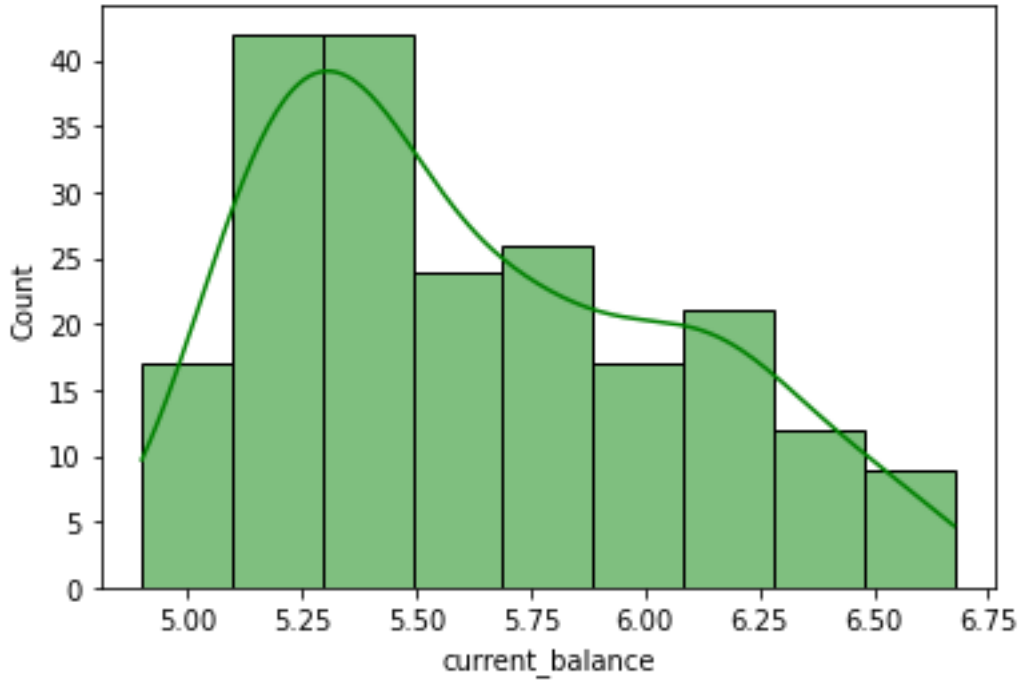

Figure 6: Distribution of Current Balance

Description of credit_limit

```
-------------------------------------------------------------------------
-----
count    210.000000
mean       3.258605
std        0.377714
min        2.630000
25%        2.944000
50%        3.237000
75%        3.561750
max        4.033000
Name: credit_limit, dtype: float64

Interquartile range (IQR) of spending is  0.618
Range of values:  1.403
```

Distribution of credit_limit

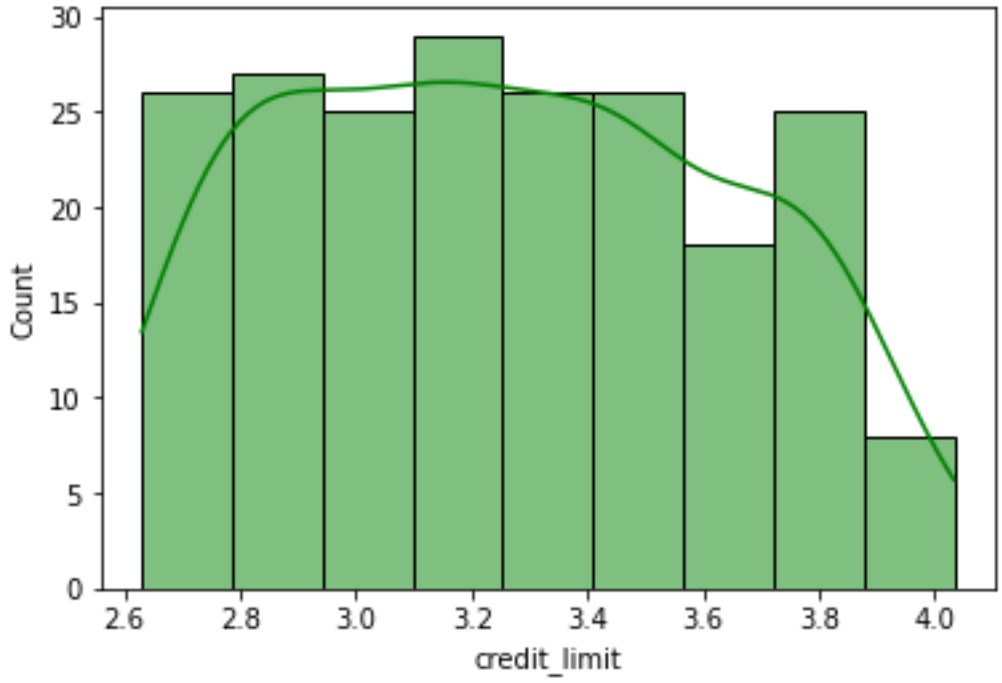

Figure 7: Distribution of Credit limit

Description of min_payment_amt

```
-------------------------------------------------------------------------
-----
count    210.000000
mean       3.697288
std        1.494689
min        0.765100
25%        2.561500
50%        3.599000
75%        4.768750
max        8.079625
Name: min_payment_amt, dtype: float64

Interquartile range (IQR) of spending is  2.207
Range of values:  7.315
```

Distribution of min_payment_amt

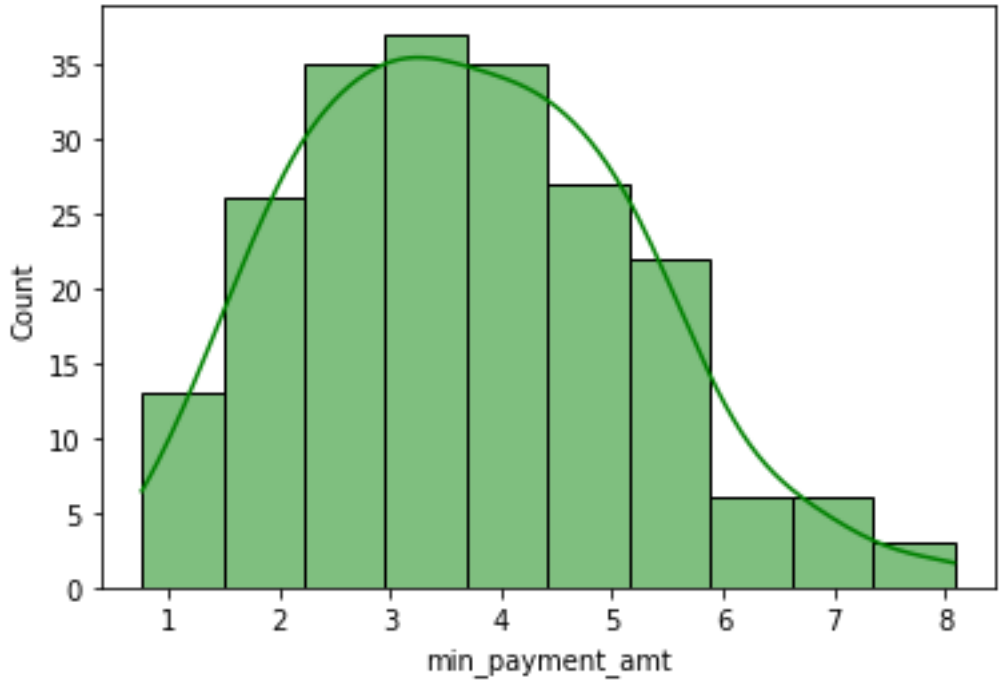

Figure 8: Distribution of min payment amount

Description of max_spent_in_single_shopping

```
-------------------------------------------------------------------------
-----
count    210.000000
mean       5.408071
std        0.491480
min        4.519000
25%        5.045000
50%        5.223000
75%        5.877000
max        6.550000
Name: max_spent_in_single_shopping, dtype: float64

Interquartile range (IQR) of spending is  0.832
Range of values:  2.031
```

Distribution of max_spent_in_single_shopping

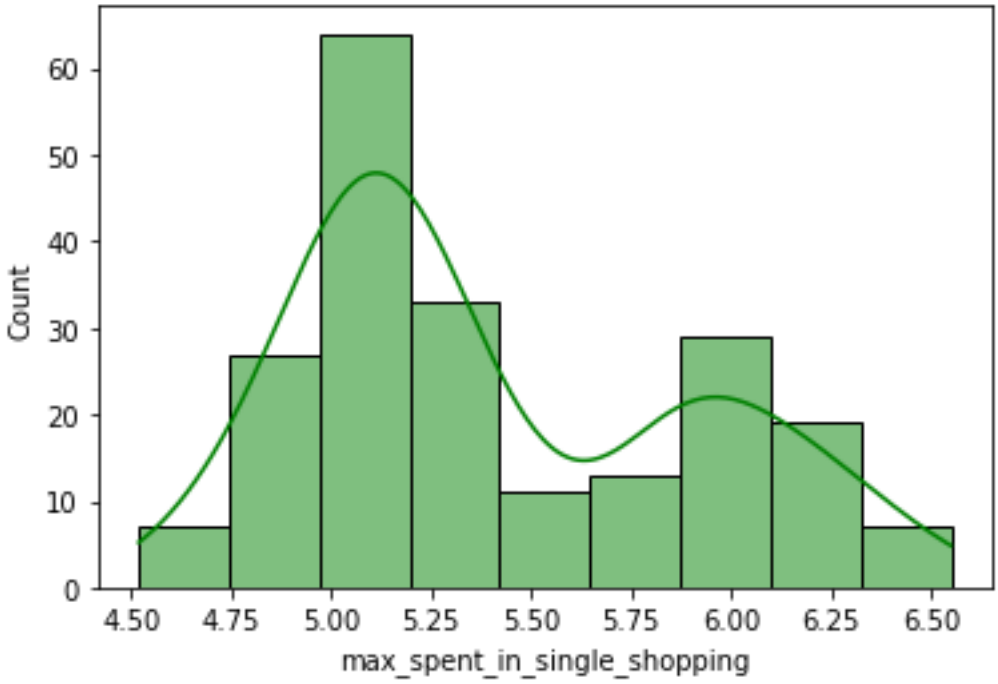

Figure 9: Distribution of max spent in single shopping

From the histogram we can conclude that maximum spent in single shopping, probability of full payment and current balance are nearly symmetrically distributed.

Skewness and Kurtosis

```
Skewness of spending is 0.4
Kurtosis of spending is -1.08
Skewness of advance_payments is 0.39
Kurtosis of advance_payments is -1.11
Skewness of probability_of_full_payment is -0.52
Kurtosis of probability_of_full_payment is -0.19
Skewness of current_balance is 0.53
Kurtosis of current_balance is -0.79
Skewness of credit_limit is 0.13
Kurtosis of credit_limit is -1.1
Skewness of min_payment_amt is 0.36
Kurtosis of min_payment_amt is -0.22
Skewness of max_spent_in_single_shopping is 0.56
Kurtosis of max_spent_in_single_shopping is -0.84
```

Skewness essentially measures the symmetry of the distribution. In positively skewed, the mean of the data is greater than the median as a large number of data-pushed on the right-hand side. In negatively skewed, the mean of the data is less than the median as a large number of data-pushed on the left-hand.

If the skewness is between -0.5 & 0.5, the data are nearly symmetrical. If the skewness is between -1 & -0.5 (negative/left skewed) or between 0.5 & 1(positive/right skewed), the data are slightly skewed. If the skewness is lower than -1 (negative/left skewed) or greater than 1 (positive/right skewed), the data are extremely skewed.

maximum spent in single shopping, probability of full payment and current balance are nearly symmetrically distributed.

Kurtosis refers to the degree of presence of outliers in the distribution. If kurtosis > 3, then it is called as Leptokurtic or heavy-tailed distribution as the kurtosis is more than normal distribution. If kurtosis = 3, then it is called as Mesokurtic as the kurtosis is same as the normal distribution. If kurtosis < 3, then it is called as Platykurtic or short-tailed distribution as the kurtosis is less than normal distribution.

Over here all of the features are short tailed distribution.

BIVARIATE ANALYSIS

PAIR PLOT

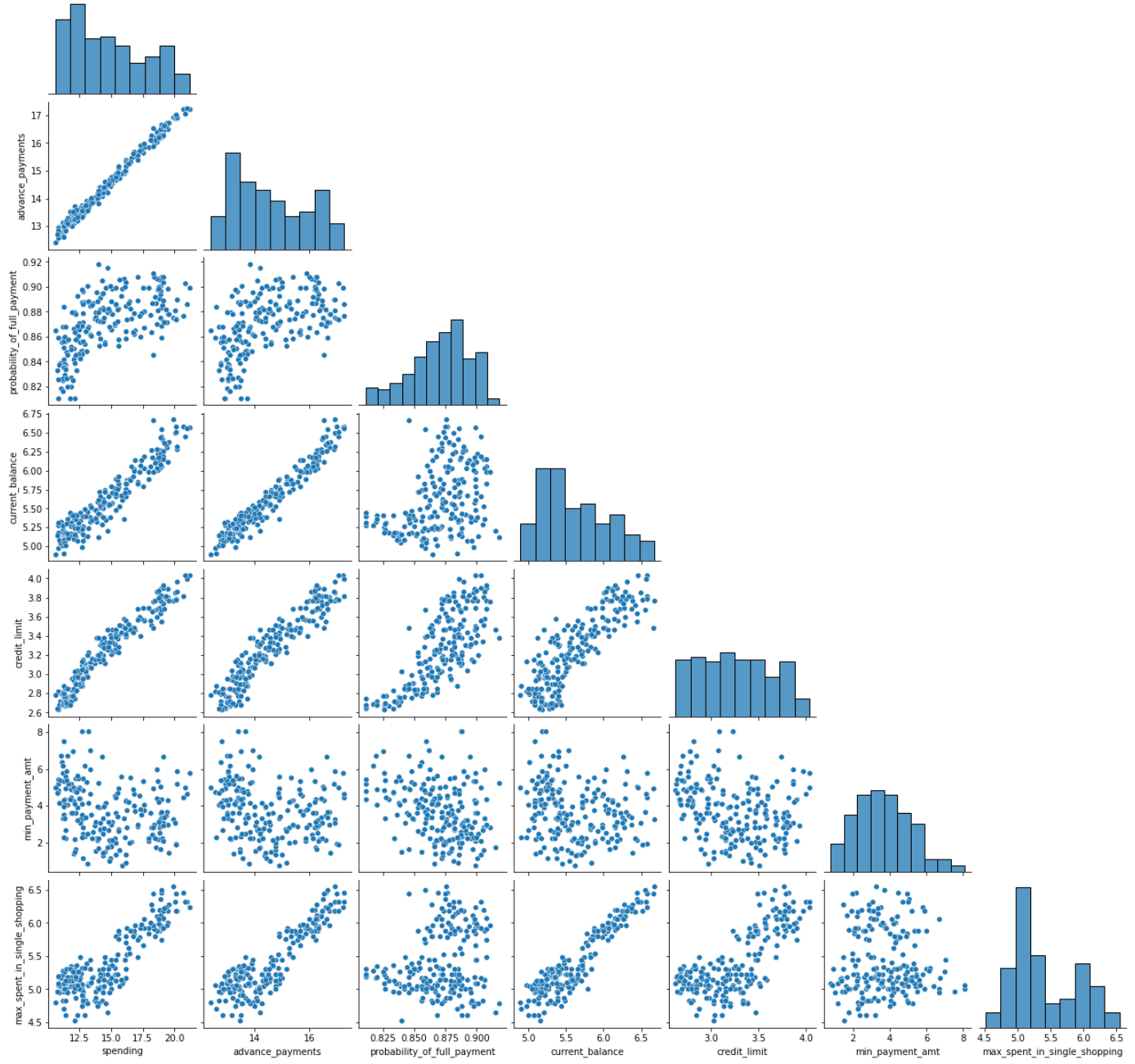

Figure 10: Pairplot

CORRELATION HEATMAP

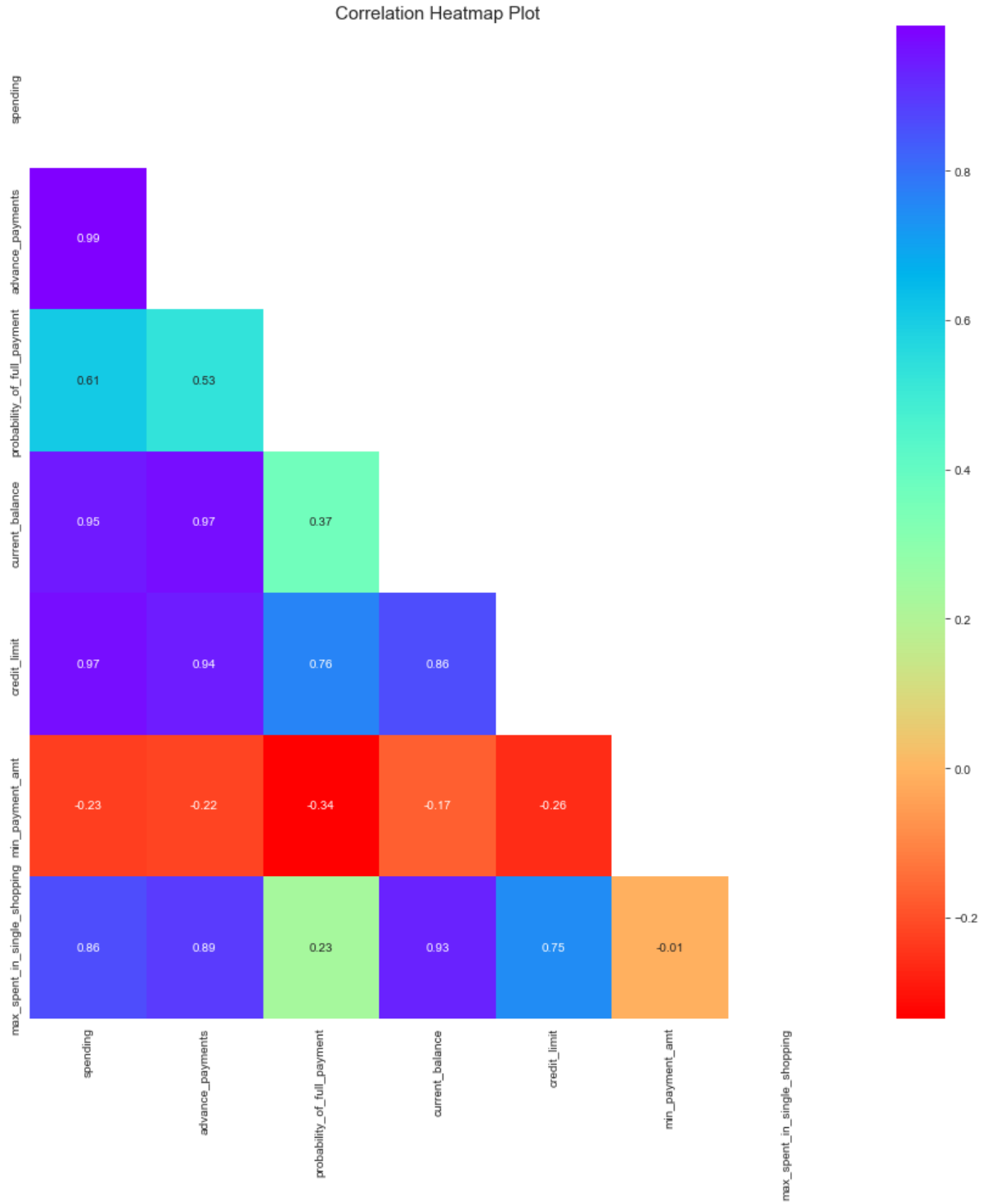

Figure 11: Correlation Heatmap

From pairplot and heatmap we can infer that there is a strong positive correlation between spending and credit_limit, advance_payments and credit_limit, current_balance and max_spent_in_single_shopping, spending and current_balance, advance_payments and current_balance, advance_payments and spending, spending and

max_spent_in_single_shopping, max_spent_in_single_shopping and advance_payments, current_balance and credit_limit.

1.2. Do you think scaling is necessary for clustering in this case? Justify.

Clustering algorithms such as K-means do need feature scaling before they are fed to the clustering algorithm as they use distance metrics like Euclidean Distance to form the clusters. The Standard Scaler method is used for scaling the data. This method will calculate the z-score for the data points and then scale the data such that mean is equal to 0 and standard deviation is equal to 1.

Before Scaling

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
count	210.000000	210.000000	210.000000	210.000000	210.000000	210.000000	210.000000
mean	14.847524	14.559286	0.871025	5.628533	3.258605	3.697288	5.408071
std	2.909699	1.305959	0.023560	0.443063	0.377714	1.494689	0.491480
min	10.590000	12.410000	0.810588	4.899000	2.630000	0.765100	4.519000
25%	12.270000	13.450000	0.856900	5.262250	2.944000	2.561500	5.045000
50%	14.355000	14.320000	0.873450	5.523500	3.237000	3.599000	5.223000
75%	17.305000	15.715000	0.887775	5.979750	3.561750	4.768750	5.877000
max	21.180000	17.250000	0.918300	6.675000	4.033000	8.079625	6.550000

Table 4:Data Description before Scaling

From the above table we can see that different features have different scales- advance payments in 100s , current balance in 1000s, credit limit 10000s . If we don't do the scaling credit limit will have more weightage by the model while fitting. They aren’t suitable for clustering algorithm as distance metric needs scaled data.

```
spending                    8.466351
advance_payments            1.705528
probability_of_full_payment 0.000555
current_balance             0.196305
credit_limit                0.142668
min_payment_amt             2.234095
max_spent_in_single_shopping 0.241553
dtype: float64
```

Above are the variances of unscaled data. Spending has a variance of 8.46 whereas other variables variance lie between 0 and 2.3.

After Scaling

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
count	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02
mean	9.148766e-16	1.097006e-16	1.642601e-15	-1.089076e-16	-2.994298e-16	1.512018e-16	-1.935489e-15
std	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00
min	-1.466714e+00	-1.649686e+00	-2.571391e+00	-1.650501e+00	-1.668209e+00	-1.966425e+00	-1.813288e+00
25%	-8.879552e-01	-8.514330e-01	-6.009681e-01	-8.286816e-01	-8.349072e-01	-7.616981e-01	-7.404953e-01
50%	-1.696741e-01	-1.836639e-01	1.031721e-01	-2.376280e-01	-5.733534e-02	-6.591519e-02	-3.774588e-01
75%	8.465989e-01	8.870693e-01	7.126469e-01	7.945947e-01	8.044956e-01	7.185591e-01	9.563941e-01
max	2.181534e+00	2.065260e+00	2.011371e+00	2.367533e+00	2.055112e+00	2.938945e+00	2.328998e+00

Table 5:Data Description after Scaling

After scaling the data the features have a mean of around 0 and standard deviation of around 1.

```
spending                      1.004785
advance_payments              1.004785
probability_of_full_payment   1.004785
current_balance               1.004785
credit_limit                  1.004785
min_payment_amt               1.004785
max_spent_in_single_shopping  1.004785
dtype: float64
```

Variance values of all the features are the same in the scaled data

1.3. Apply hierarchical clustering to scaled data. Identify the number of optimum clusters using Dendrogram and briefly describe them.

Dendrogram with Average Linkage

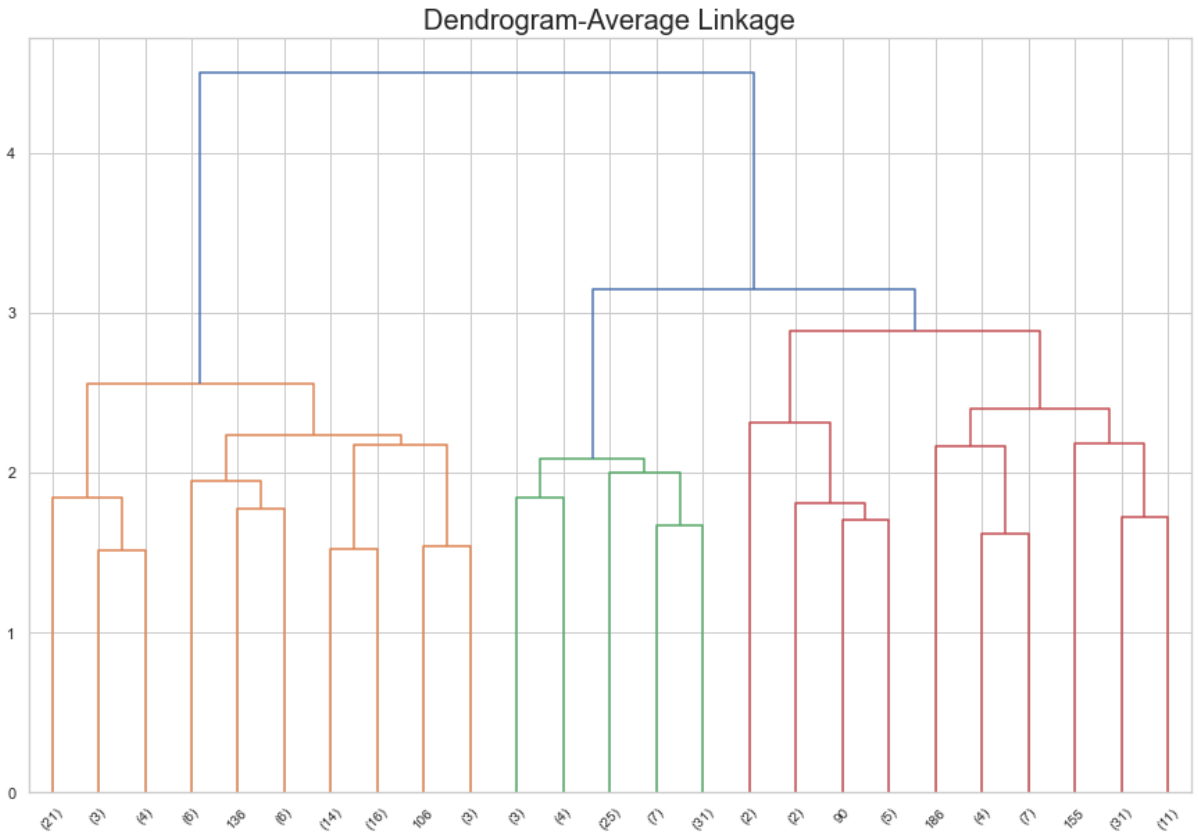

Figure 12: Dendrogram-Average Linkage

Dendrogram-Ward Linkage

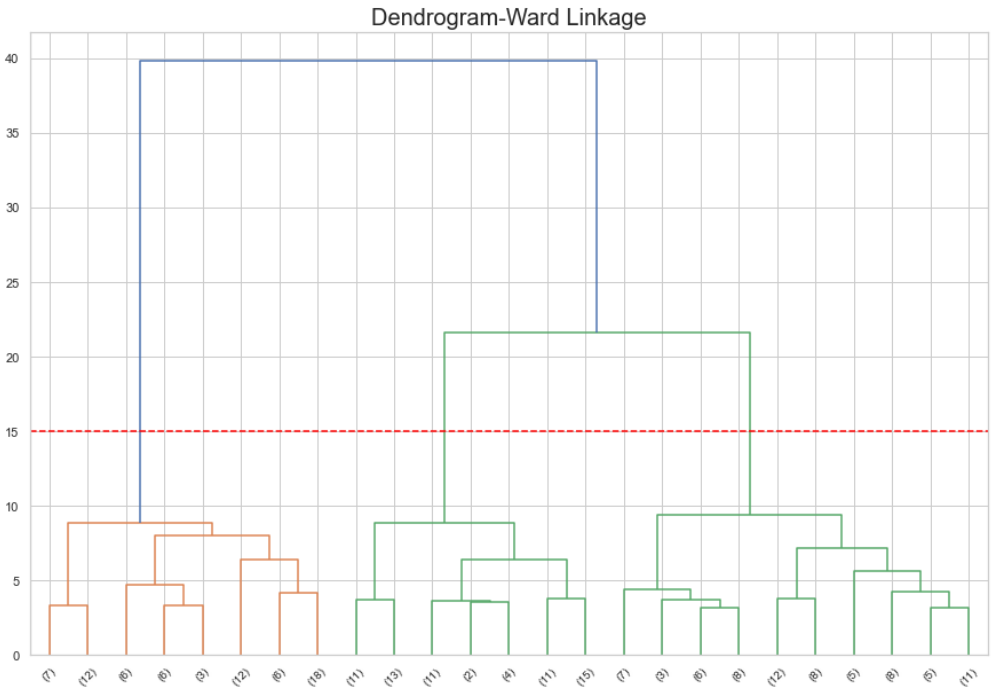

Figure 13: Dendrogram- Ward Linkage

Dendrogram from both the linkage methods (average and ward) shows that there are 3 clusters of data points based on the business context of having 3 types of customers such as low spenders of money, medium spenders of money and high spenders of money. Red line is drawn at 15 in the Dendrogram with Ward Linkage because there is no horizontal line intersecting the green vertical lines. The red line intersects the 3 lines so the number of clusters is taken as 3. Also in Dendrogram with average linkage we can see that there are 3 coloured clusters: red, green and orange.

Clusters derived from Fcluster method

Form flat clusters from the hierarchical clustering using the criterion maxclust which sets the maximum number of clusters and with 3 clusters.

clusters	1	2	3
spending	18.371429	11.872388	14.199041
advance_payments	16.145429	13.257015	14.233562
probability_of_full_payment	0.884400	0.848155	0.879190
current_balance	6.158171	5.238940	5.478233
credit_limit	3.684629	2.848537	3.226452
min_payment_amt	3.639157	4.940302	2.612181
max_spent_in_single_shopping	6.017371	5.122209	5.086178

Table 6: FCluster mean1

```
3    73
1    70
2    67
Name: clusters, dtype: int64
```

There are 70 customers belonging to Group 1, 67 customers belonging to Group 2, 73 customers belonging to Group 3.

Clusters derived from Agglomerative Clustering method

Agglomerative clustering is done using the euclidean distance as the value in the affinity parameter ,linkage method as average and with 3 clusters.

clusters	0	1	2
spending	14.217077	18.129200	11.916857
advance_payments	14.195846	16.058000	13.291000
probability_of_full_payment	0.884869	0.881595	0.846845
current_balance	5.442000	6.135747	5.258300
credit_limit	3.253508	3.648120	2.846000
min_payment_amt	2.759007	3.650200	4.619000
max_spent_in_single_shopping	5.055569	5.987040	5.115071

Table 7:Agglomerative Cluster

```
1    75
2    70
0    65
Name: clusters, dtype: int64
```

There are 65 customers belonging to Group 0, 75 customers belonging to Group 1, 70 customers belonging to Group 2.

Scatterplot using Fcluster

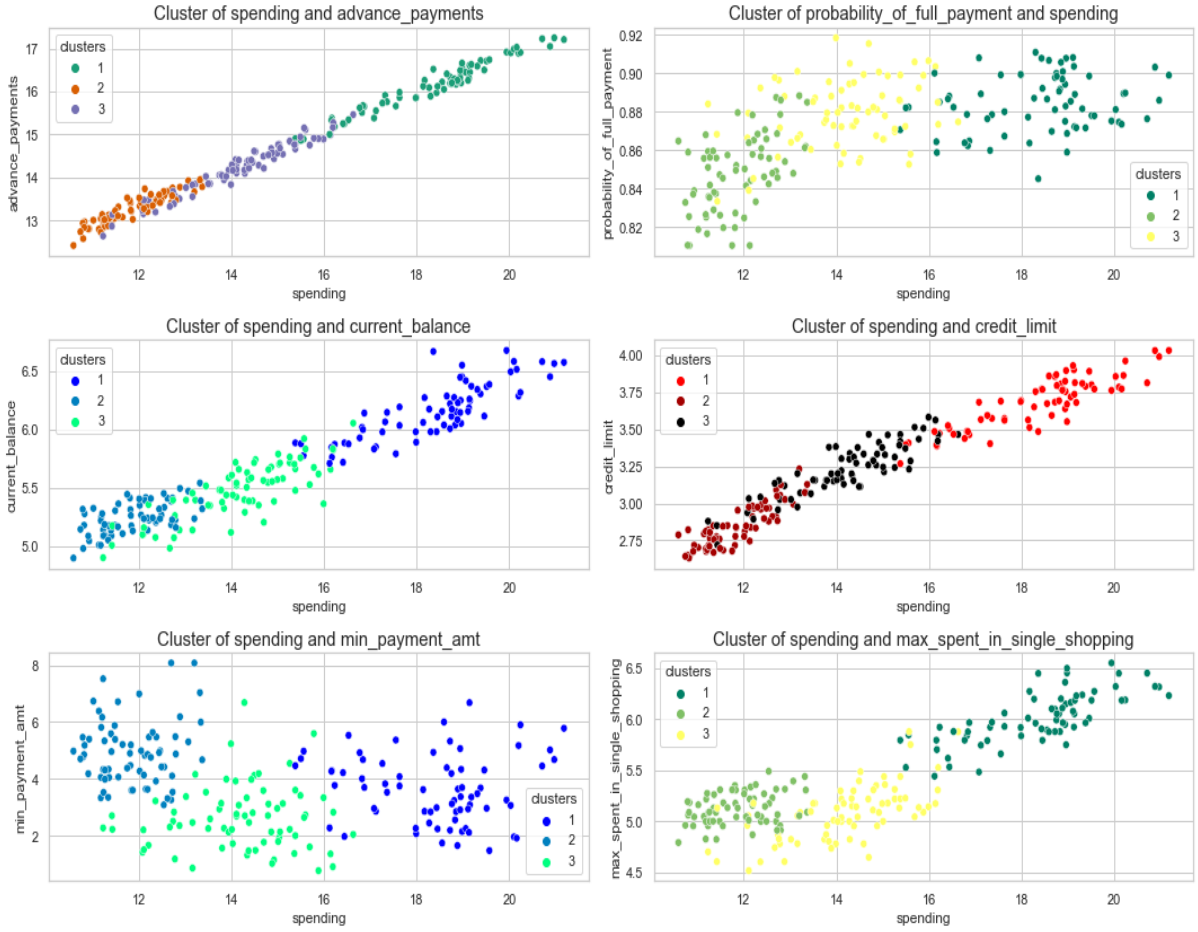

Figure 14:Cluster plots for Spending variable

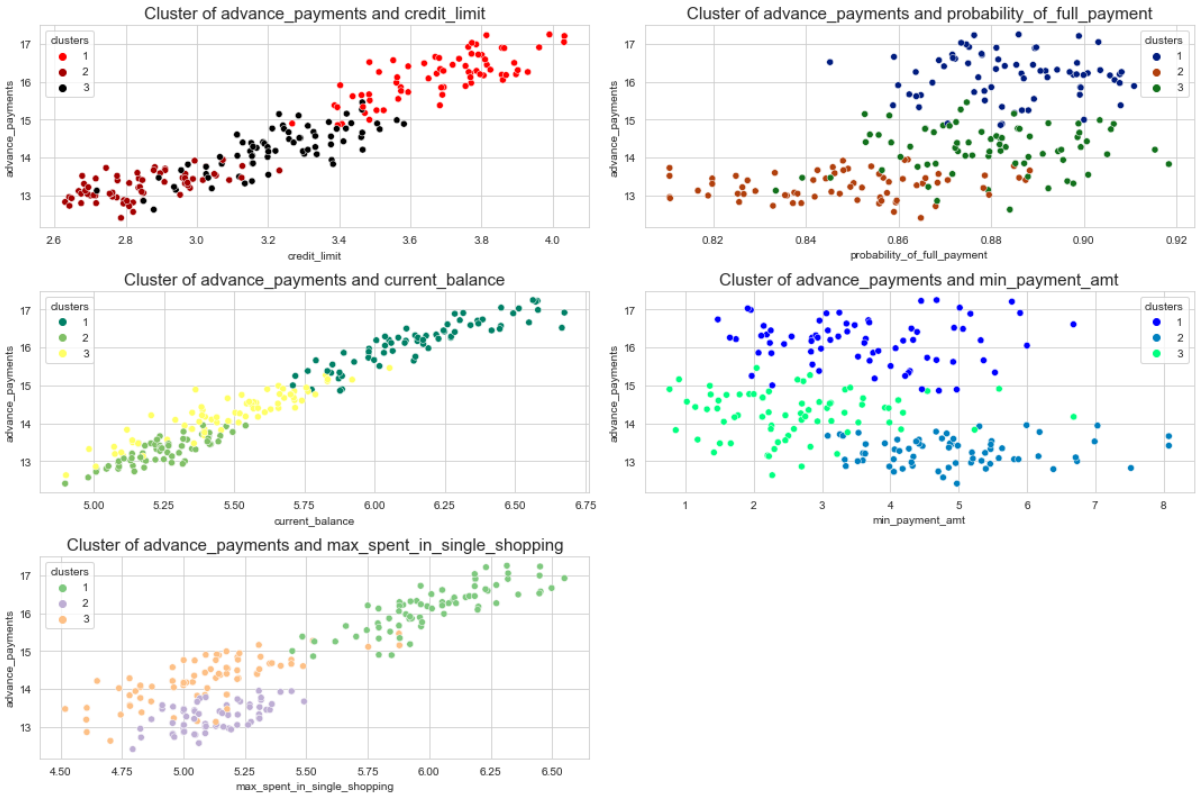

Figure 15:Cluster plots for advance payment variable

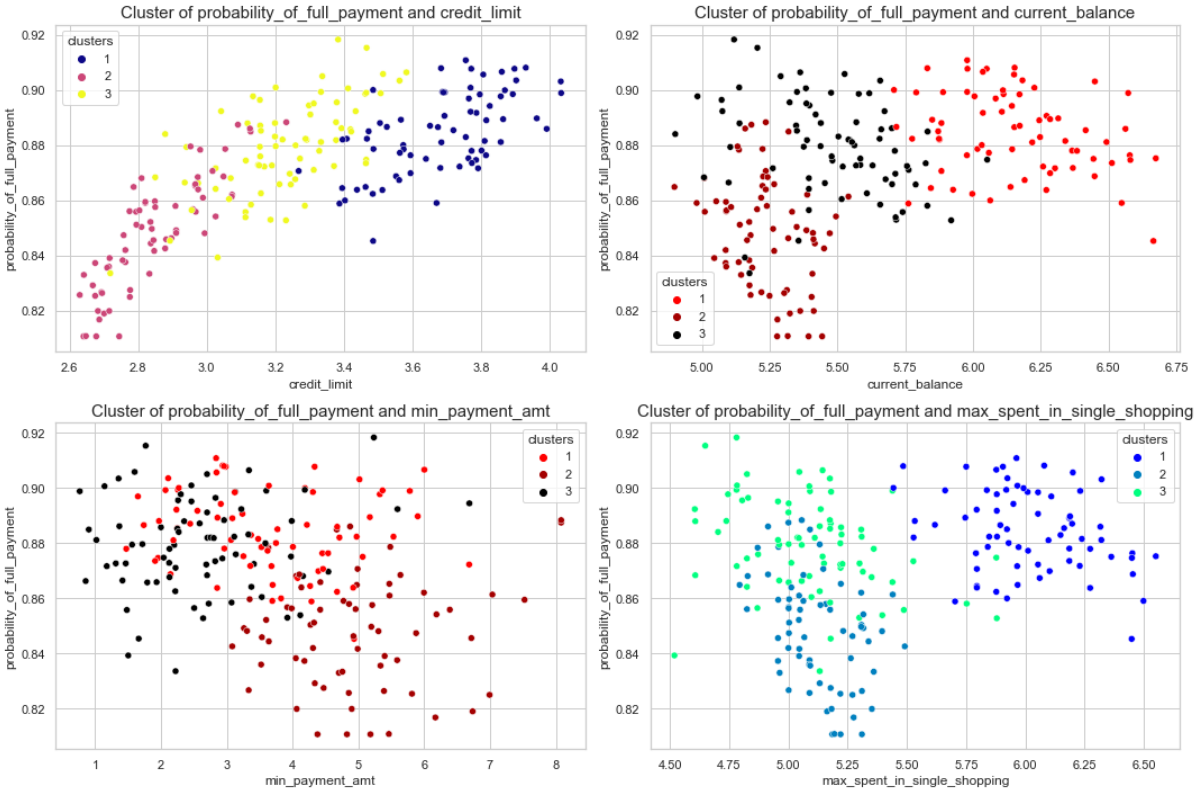

Figure 16:Cluster plots for probability of full payment variable

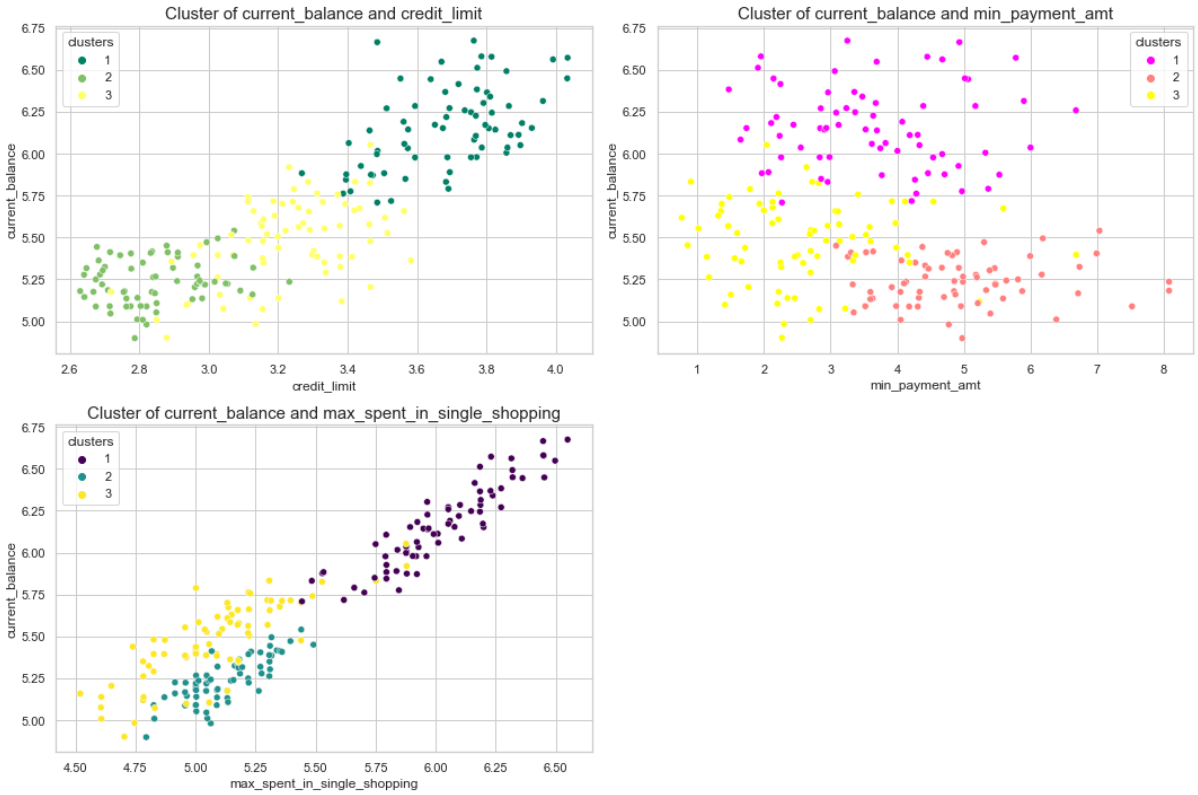

Figure 17:Cluster plots for current balance variable

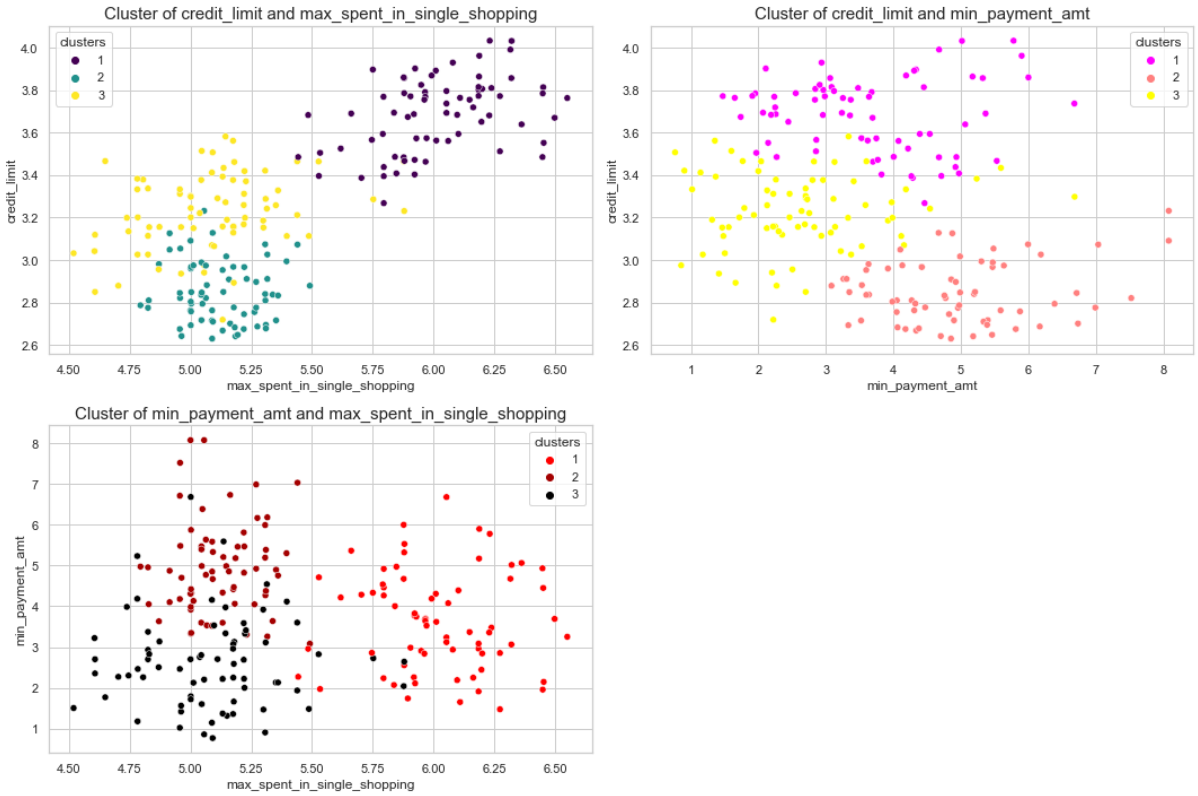

Figure 18:Cluster plots for credit limit and max spent in single shopping variable

In all the above plots we can see that the 3 clusters have distinguishable boundaries. Hence 3 is the optimum number of clusters as it is also suits the business context as categorising the people as low spenders, medium spenders and high spenders.

1.4 Apply K-Means clustering on scaled data and determine optimum clusters. Apply elbow curve and silhouette score. Explain the results properly. Interpret and write inferences on the finalized clusters.

Kmeans clustering is applied with different number of clusters ranging from 1 to 10 , Their corresponding total wss and silhouette score is calculated and plotted to determine the optimum number of clusters. The total wss values for 1 cluster to 10 clusters are:

```
[1469.999999999999,
 659.14740095485,
 430.298481751223,
 371.0356644664012,
 325.9741284729876,
 289.45524862464833,
 263.859944426353,
 239.94446635017925,
 220.59353946108112,
 205.7633419678701]
```

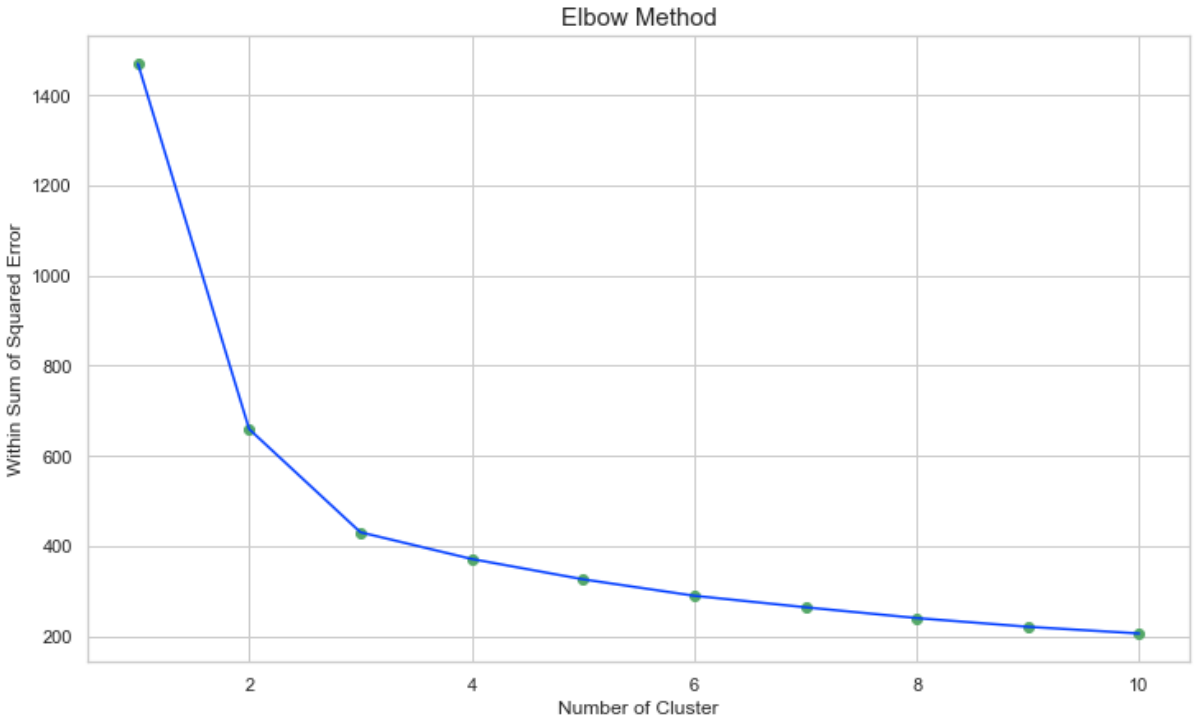

Figure 19: Elbow Method

3 clusters is chosen to be optimum, where addition of one more cluster does not lower the value of total WCSS appreciably. From 1->2 there was a drop of around 800, 2->3 there is a decrease of around 200. Since there is no significant drop of total wss from 3->4 and 4->5 which is around 60 and 45 respectively I pick 3 and 4 as the for optimum number of clusters.Elbow occurs at these points.

The Silhouette Score for 2 clusters to 10 clusters are:

```
Number of Clusters : 2  Silhouette Score : 0.46560100442748986
Number of Clusters : 3  Silhouette Score : 0.4008059221522216
Number of Clusters : 4  Silhouette Score : 0.3275850791949873
Number of Clusters : 5  Silhouette Score : 0.28665397420054717
Number of Clusters : 6  Silhouette Score : 0.28202953848971773
Number of Clusters : 7  Silhouette Score : 0.26638963568371493
Number of Clusters : 8  Silhouette Score : 0.26022036479418587
Number of Clusters : 9  Silhouette Score : 0.25509640529778443
Number of Clusters : 10  Silhouette Score : 0.24595921136139773
```

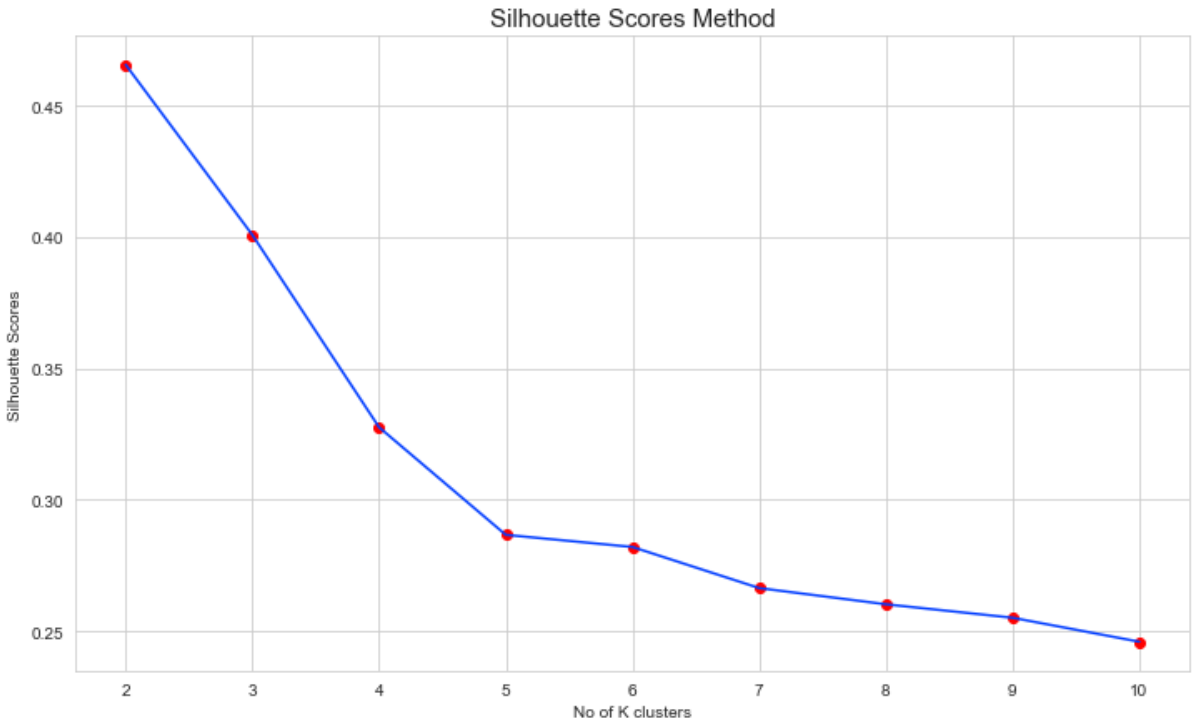

Figure 20:Silhouette Scores Plot

The maximum value of the statistic indicates the optimum value of clusters. Cluster 3 has more silhouette score than Cluster 4. Hence 3 is the optimum value of cluster.

First 5 rows of the dataset after appending kmeans clusters to it.

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	clusters
0	19.94	16.92	0.875200	6.675	3.763	3.252	6.550	1
1	15.99	14.89	0.906400	5.363	3.582	3.336	5.144	2
2	18.95	16.42	0.882900	6.248	3.755	3.368	6.148	1
3	10.83	12.96	0.810588	5.278	2.641	5.182	5.185	0
4	17.99	15.86	0.899200	5.890	3.694	2.068	5.837	1

Table 8: Dataset with KMeans Cluster

Pairplot for KMeans Cluster

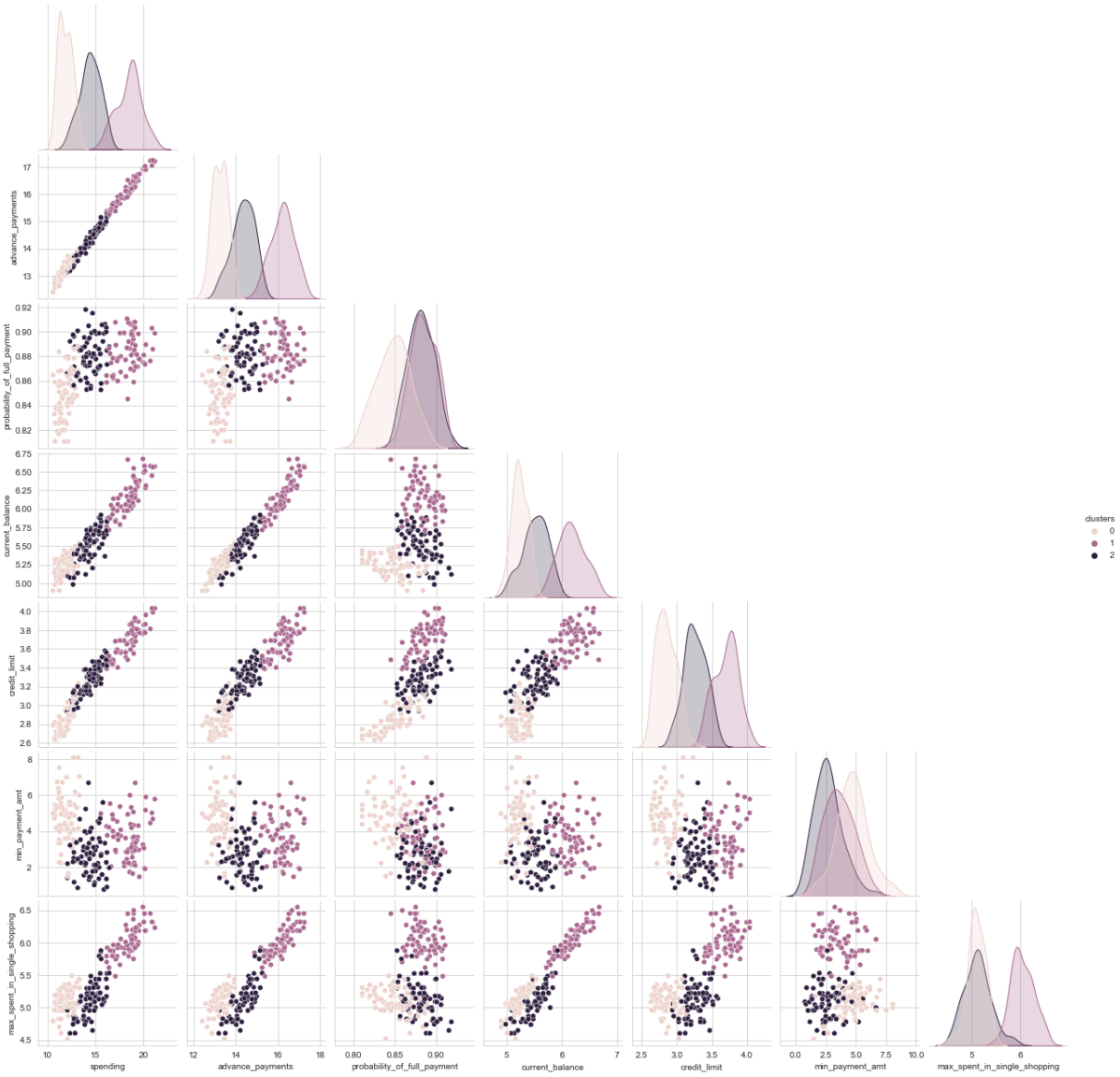

Figure 21: Pairplot for KMeans Cluster

We can observe that the 3 clusters are plotted with distinguishable boundaries.

1.5 Describe cluster profiles for the clusters defined. Recommend different promotional strategies for different clusters.

KMeans Group Profile

clusters	0	1	2
spending	11.856944	18.495373	14.437887
advance_payments	13.247778	16.203433	14.337746
probability_of_full_payment	0.848330	0.884210	0.881597
current_balance	5.231750	6.175687	5.514577
credit_limit	2.849542	3.697537	3.259225
min_payment_amt	4.733892	3.632373	2.707341
max_spent_in_single_shopping	5.101722	6.041701	5.120803

Table 9: KMeans Cluster Mean

Group 0 - Has lowest values in spending,advance payments,current balance,probability of full payment ,current balance,credit limit and max xspent in single shopping. High in minimum payment amount.

Group 1 - Has highest values in spending,advance payments,current balance,probability of full payment ,current balance,credit limit and max spent in single shopping. Second highest value in minimum payment amount.

Group 2 - Has second highest values in spending,advance payments,current balance,probability of full payment ,current balance,credit limit and max spent in single shopping. Lowest value in minimum payment amount.

Fcluster Group Profile

clusters	1	2	3
spending	18.371429	11.872388	14.199041
advance_payments	16.145429	13.257015	14.233562
probability_of_full_payment	0.884400	0.848155	0.879190
current_balance	6.158171	5.238940	5.478233
credit_limit	3.684629	2.848537	3.226452
min_payment_amt	3.639157	4.940302	2.612181
max_spent_in_single_shopping	6.017371	5.122209	5.086178

Table 10:FCluster Cluster Mean2

Group 1 - Has highest values in spending, advance payments,current balance, probability of full payment ,current balance, credit limit and max spent in single shopping. Second highest value in minimum payment amount.

Group 2 - Has lowest values in spending,advance payments,current balance,probability of full payment ,current balance,credit limit . High in minimum payment amount and second highest value in max spent in single shopping.

Group 3 - Has second highest values in spending,advance payments,current balance, probability of full payment ,current balance,credit limit. Lowest value in minimum payment amount and max spent in single shopping.

Promotional Strategies based on Hierarchical Clustering – Fcluster

Group 1 - High Spenders

- It has the highest current balance so new credit cards can be offered to this group
- It has high probability of full payment and advance payment so loan can be given against credit cards to earn interests.
- We can also increase their credit limits as they are having highest probability of full payment .

Group 2 - Low Spenders

- We can give reward points for their loan repayments since this group of customers has lowest spending, advance payment ,lowest probability of full payment & least Credit limit among all.
- We can offer insurance so that they can save their income tax to increase current balance.
- We can send them reminders for their loan repayments since the probability of full payment is low.

- Since these Customers are having lowest spending habits we can provide tie up link their municipality tax, electricity bills,gas bills,groceries .

Group 3 - Medium Spenders

- We can also offer vouchers and gifts to increase their spending and maximum spending in one purchase. This can be done to move them to Group 1.

- We can tie them up with popular brands to increase their spending.

- The minimum payment amount is lowest among all 3 groups so we can offer a significant discount for making full payment.

PROBLEM 2 : CART-RF-ANN

Problem Statement

An Insurance firm providing tour insurance is facing higher claim frequency. The management decides to collect data from the past few years. You are assigned the task to make a model which predicts the claim status and provide recommendations to management. Use CART, RF & ANN and compare the models' performances in train and test sets.

Introduction

The purpose of this whole exercise is to perform exploratory data analysis and predict the Claim Status using algorithms like CART, Random Forest and Artificial Neural Network. Using the predictions recommendations are to be provided to the management. The dataset consists of 3000 rows with their features like Claimed, Agency Code, Type, Channel, Product Name, Duration, Destination, Sales, Age and Commission.

Data Description

1. Target: Claim Status (Claimed)
2. Code of tour firm (Agency Code)
3. Type of tour insurance firms (Type)
4. Distribution channel of tour insurance agencies (Channel)
5. Name of the tour insurance products (Product)
6. Duration of the tour (Duration in days)
7. Destination of the tour (Destination)
8. Amount worth of sales per customer in procuring tour insurance policies in rupees (in 100’s)
9. The commission received for tour insurance firm (Commission is in percentage of sales)
10. Age of insured (Age)

2.1. Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis)

Sample of the dataset

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
0	48	C2B	Airlines	No	0.70	Online	7	2.51	Customised Plan	ASIA
1	36	EPX	Travel Agency	No	0.00	Online	34	20.00	Customised Plan	ASIA
2	39	CWT	Travel Agency	No	5.94	Online	3	9.90	Customised Plan	Americas
3	36	EPX	Travel Agency	No	0.00	Online	4	26.00	Cancellation Plan	ASIA
4	33	JZI	Airlines	No	6.30	Online	53	18.00	Bronze Plan	ASIA

Table 11: Sample of the Dataset2

Here Claimed is the Target variable which is categorical. Hence classification using CART, Random Forest and Artificial Neural Network.

EXPLORATORY DATA ANALYSIS

Data Type and Missing Values

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Age           3000 non-null   int64
 1   Agency_Code   3000 non-null   object
 2   Type          3000 non-null   object
 3   Claimed       3000 non-null   object
 4   Commision     3000 non-null   float64
 5   Channel       3000 non-null   object
 6   Duration      3000 non-null   int64
 7   Sales         3000 non-null   float64
 8   Product Name  3000 non-null   object
 9   Destination   3000 non-null   object
dtypes: float64(2), int64(2), object(6)
memory usage: 234.5+ KB
```

Table 12:Dataset Info2

Age, Commision, Duration and Sales are of numeric data type. Remaining features are of Object datatype.

```
Age              0
Agency_Code      0
Type             0
Claimed          0
Commision        0
Channel          0
Duration         0
Sales            0
Product Name     0
Destination      0
dtype: int64
```

There are no missing values.

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
count	3000.000000	3000	3000	3000	3000.000000	3000	3000.000000	3000.000000	3000	3000
unique	NaN	4	2	2	NaN	2	NaN	NaN	5	3
top	NaN	EPX	Travel Agency	No	NaN	Online	NaN	NaN	Customised Plan	ASIA
freq	NaN	1365	1837	2076	NaN	2954	NaN	NaN	1136	2465
mean	38.091000	NaN	NaN	NaN	14.529203	NaN	70.001333	60.249913	NaN	NaN
std	10.463518	NaN	NaN	NaN	25.481455	NaN	134.053313	70.733954	NaN	NaN
min	8.000000	NaN	NaN	NaN	0.000000	NaN	-1.000000	0.000000	NaN	NaN
25%	32.000000	NaN	NaN	NaN	0.000000	NaN	11.000000	20.000000	NaN	NaN
50%	36.000000	NaN	NaN	NaN	4.630000	NaN	26.500000	33.000000	NaN	NaN
75%	42.000000	NaN	NaN	NaN	17.235000	NaN	63.000000	69.000000	NaN	NaN
max	84.000000	NaN	NaN	NaN	210.210000	NaN	4580.000000	539.000000	NaN	NaN

Table 13: Description of Dataset2

Treating duplicates and bad data

There are 139 duplicates in the dataset. We can eliminate if there is an unique identifier but when there is no unique identifier in the dataset we can drop the duplicates if their number is less than 5% of total record. Since there 139 is less than 150(5% of 3000) we can drop it.

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
1508	25	JZI	Airlines	No	6.3	Online	-1	18.0	Bronze Plan	ASIA

Table 14:Bad Data1

Duration cannot have a negative value. Hence it can imputed with the median value of Duration. Median is used since there are outliers in the data.

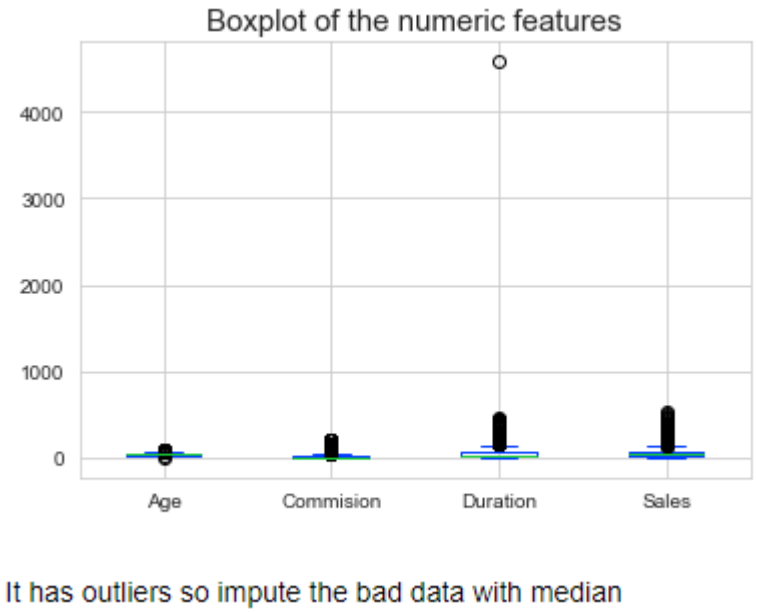

Figure 22: Boxplot-Outliers

After imputing with the median the record has the following values

```
Age                      25
Agency_Code             JZI
Type                Airlines
Claimed                   No
Commision                6.3
Channel               Online
Duration                  28
Sales                   18.0
Product Name     Bronze Plan
Destination             ASIA
Name: 1508, dtype: object
```

The Column sales 52 records with 0 as values .

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
131	53	JZI	Airlines	No	12.95	Online	93	0.0	Bronze Plan	ASIA
162	36	EPX	Travel Agency	No	0.00	Online	2	0.0	Customised Plan	ASIA
323	54	CWT	Travel Agency	No	100.98	Online	18	0.0	Customised Plan	Americas
483	44	CWT	Travel Agency	No	11.88	Online	10	0.0	Customised Plan	ASIA
513	31	CWT	Travel Agency	No	83.16	Online	99	0.0	Customised Plan	EUROPE

Table 15: Bad Data2

Hence it is imputed with the median.

After treating the data there are 2861 records.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2861 entries, 0 to 2999
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Age           2861 non-null   int64
 1   Agency_Code   2861 non-null   object
 2   Type          2861 non-null   object
 3   Claimed       2861 non-null   object
 4   Commision     2861 non-null   float64
 5   Channel       2861 non-null   object
 6   Duration      2861 non-null   int64
 7   Sales         2861 non-null   float64
 8   Product Name  2861 non-null   object
 9   Destination   2861 non-null   object
dtypes: float64(2), int64(2), object(6)
memory usage: 310.4+ KB
```

Table 16:Changed Dataset Info

After treating the bad data there is a slight change in the value of measures of central tendency.

	Age	Commision	Duration	Sales
count	2861.000000	2861.000000	2861.000000	2861.000000
mean	38.204124	15.080996	72.130374	62.366756
std	10.678106	25.826834	135.972828	71.012142
min	8.000000	0.000000	0.000000	0.190000
25%	31.000000	0.000000	12.000000	21.000000
50%	36.000000	5.630000	28.000000	33.500000
75%	43.000000	17.820000	66.000000	69.300000
max	84.000000	210.210000	4580.000000	539.000000

Table 17:Changed Dataset Description

Univariate Analysis

Boxplot

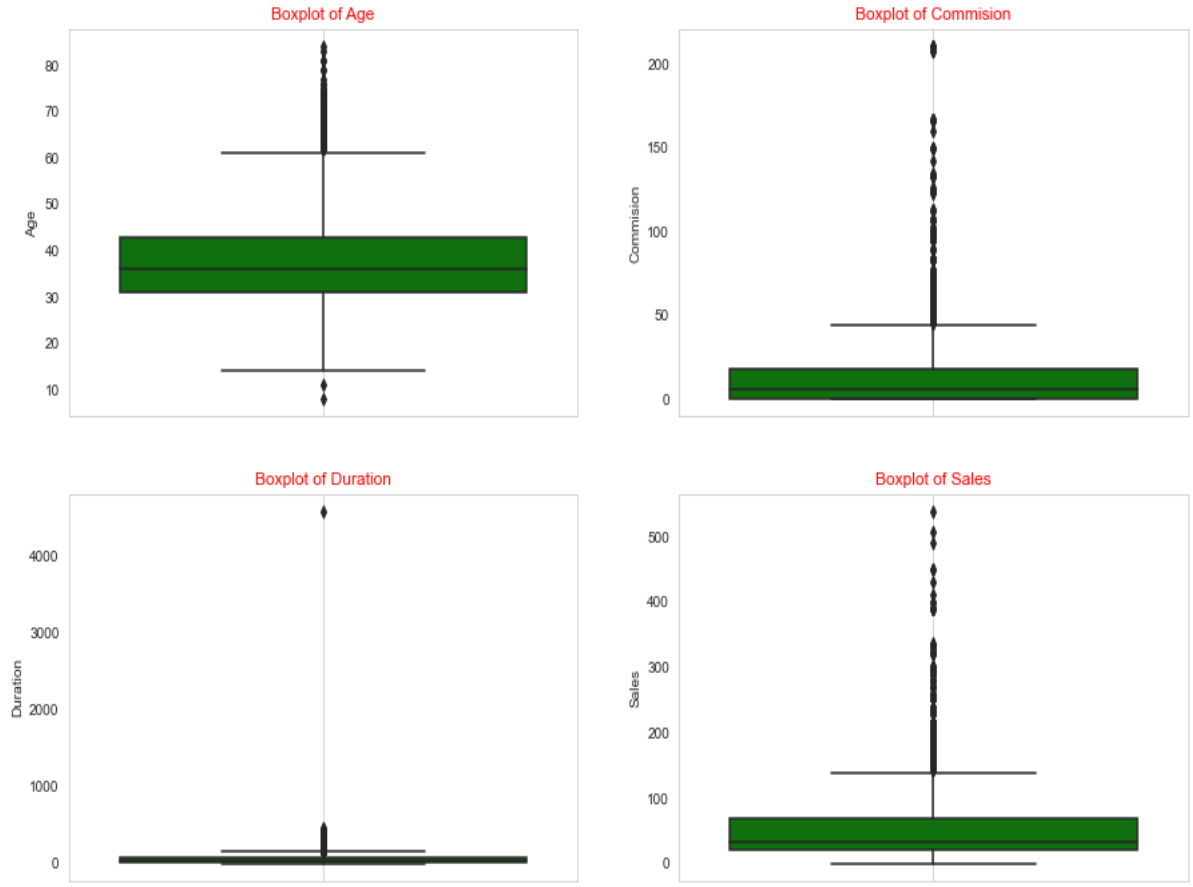

Figure 23:Univariate Analysis - Boxplot

There are outliers in all the features. Median of age is around 35. Commission has a median around 5. Duration and Sales around 30.

Proportion of Outliers

```
Lower outliers in Age is :  13.0
Upper outliers in Age is :  61.0
Number of outliers in Age upper :  128
Number of outliers in Age lower :  2
% of Outlier in Age upper:  4 %
% of Outlier in Age lower:  0 %
-----------------------------------------------------
Lower outliers in Commision is :  -26.73
Upper outliers in Commision is :  44.55
Number of outliers in Commision upper :  354
Number of outliers in Commision lower :  0
% of Outlier in Commision upper:  12 %
% of Outlier in Commision lower:  0 %
-----------------------------------------------------
Lower outliers in Duration is :  -69.0
Upper outliers in Duration is :  147.0
Number of outliers in Duration upper :  362
Number of outliers in Duration lower :  0
% of Outlier in Duration upper:  13 %
% of Outlier in Duration lower:  0 %
-----------------------------------------------------
Lower outliers in Sales is :  -51.44999999999999
Upper outliers in Sales is :  141.75
Number of outliers in Sales upper :  347
Number of outliers in Sales lower :  0
% of Outlier in Sales upper:  12 %
% of Outlier in Sales lower:  0 %
-----------------------------------------------------
```

The above image shows the proportion of outliers of each feature in the dataset.

Analysis of Agency_Code

```
Value Count of Agency_Code
-------------------------------------------------------------------------------
-
EPX    1238
C2B     913
CWT     471
JZI     239
Name: Agency_Code, dtype: int64

Description of Agency_Code
-------------------------------------------------------------------------------
-
count     2861
unique       4
top        EPX
```

```
freq      1238
Name: Agency_Code, dtype: object

Countplot of Agency_Code
--------------------------------------------------------------------------------
-
```

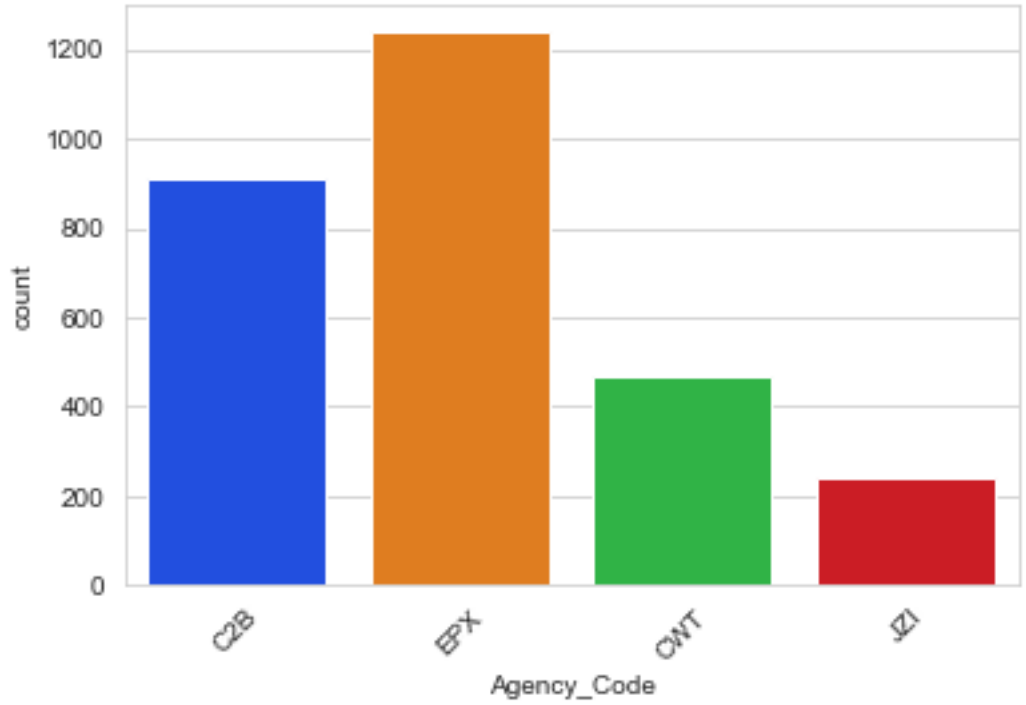

Figure 24:Countplot of Agency_Code

Most of the insurances are Agency Code EPX Agency Code.

Analysis of Type

```
Value Count of Type
--------------------------------------------------------------------------------
-
Travel Agency    1709
Airlines         1152
Name: Type, dtype: int64

Description of Type
--------------------------------------------------------------------------------
-
count              2861
unique                2
top       Travel Agency
freq               1709
Name: Type, dtype: object
```

```
Countplot of Type
--------------------------------------------------------------------------------
-
```

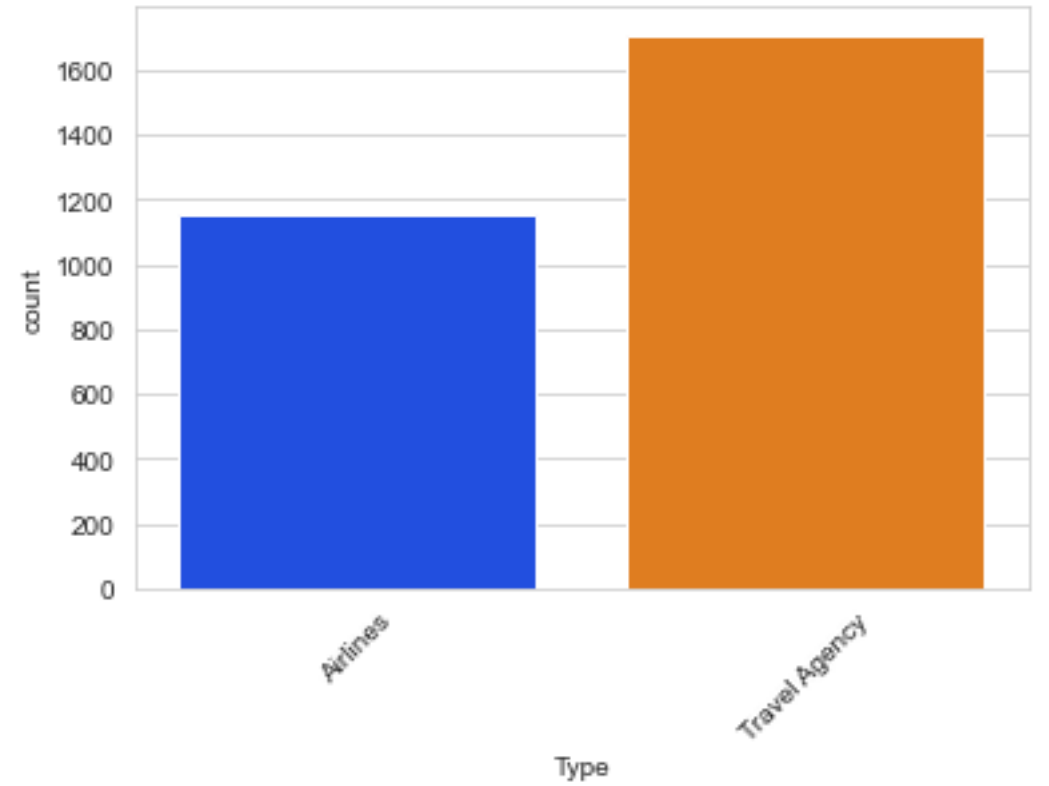

Figure 25:Countplot of Type

Most of the insurance has type Travel Agency.

Analysis of Claimed

```
Value Count of Claimed
--------------------------------------------------------------------------------
-
No     1947
Yes     914
Name: Claimed, dtype: int64

Description of Claimed
--------------------------------------------------------------------------------
-
count     2861
unique       2
top         No
freq      1947
```

```
Name: Claimed, dtype: object

Countplot of Claimed
--------------------------------------------------------------------------------
-
```

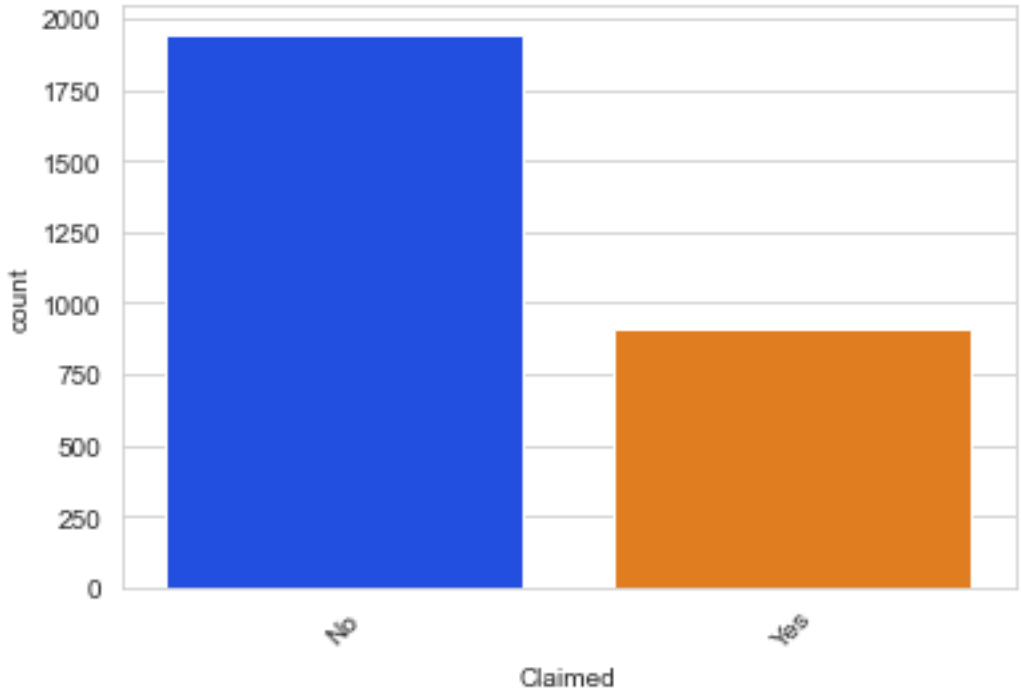

Figure 26:Countplot of Claimed

There are more unclaimed insurances than claimed.

Analysis of Channel

```
Value Count of Channel
--------------------------------------------------------------------------------
-
Online     2815
Offline      46
Name: Channel, dtype: int64

Description of Channel
--------------------------------------------------------------------------------
-
count       2861
unique         2
top       Online
freq        2815
Name: Channel, dtype: object

Countplot of Channel
```

```
--------------------------------------------------------------------------------
-
```

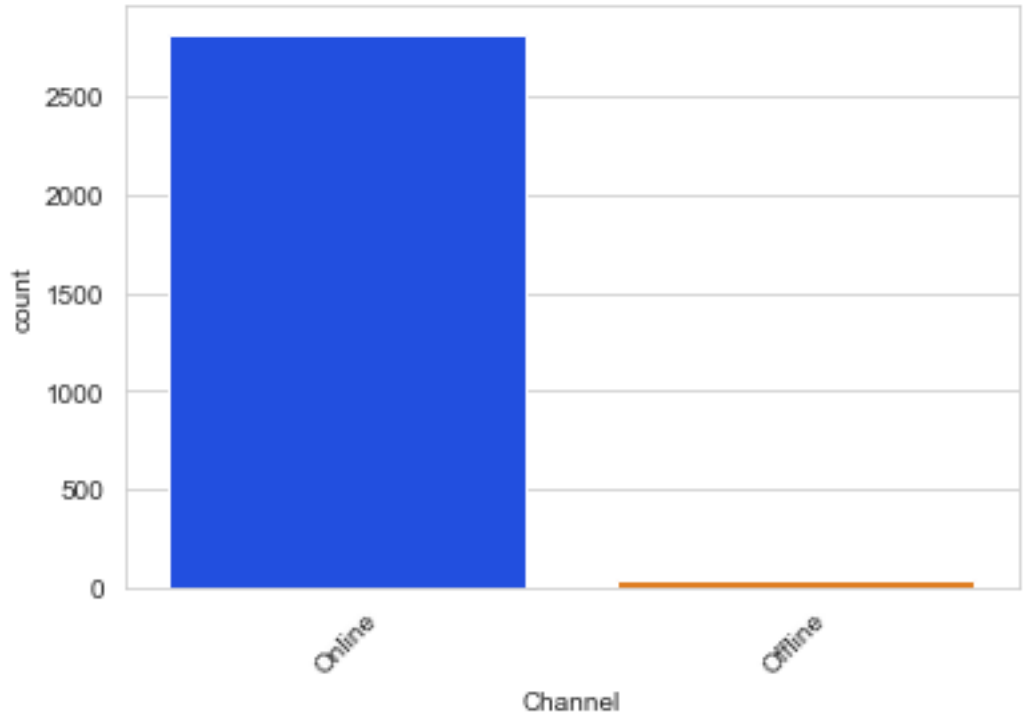

Figure 27: Countplot of Channel

Mostly people use Online Channel than Offline.

Analysis of Product Name

```
Value Count of Product Name
--------------------------------------------------------------------------------
-
Customised Plan      1071
Bronze Plan           645
Cancellation Plan     615
Silver Plan           421
Gold Plan             109
Name: Product Name, dtype: int64

Description of Product Name
--------------------------------------------------------------------------------
-
count                2861
unique                  5
top       Customised Plan
freq                 1071
Name: Product Name, dtype: object
```

```
Countplot of Product Name
--------------------------------------------------------------------------------
-
```

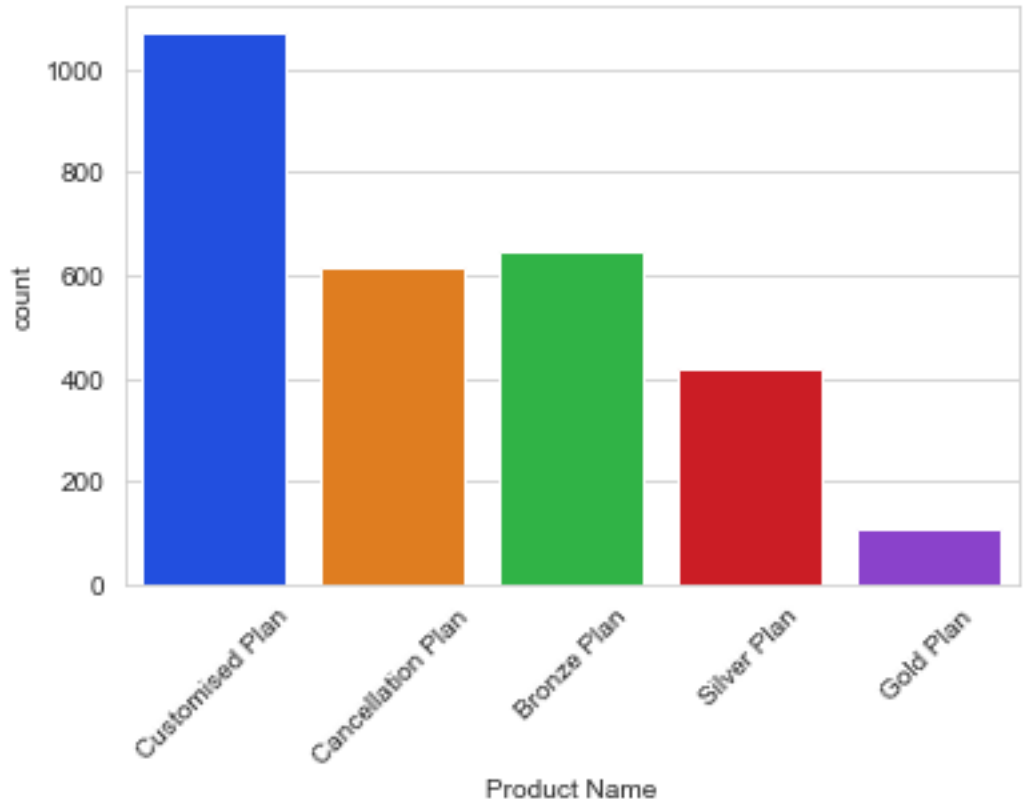

Figure 28:Countplot of Product Name

Most people go for Customised Plan Insurance than other plans.

Analysis of Destination

```
Value Count of Destination
--------------------------------------------------------------------------------
-
ASIA        2327
Americas     319
EUROPE       215
Name: Destination, dtype: int64

Description of Destination
--------------------------------------------------------------------------------
-
count     2861
unique       3
top       ASIA
freq      2327
```

```
Name: Destination, dtype: object

Countplot of Destination
--------------------------------------------------------------------------------
-
```

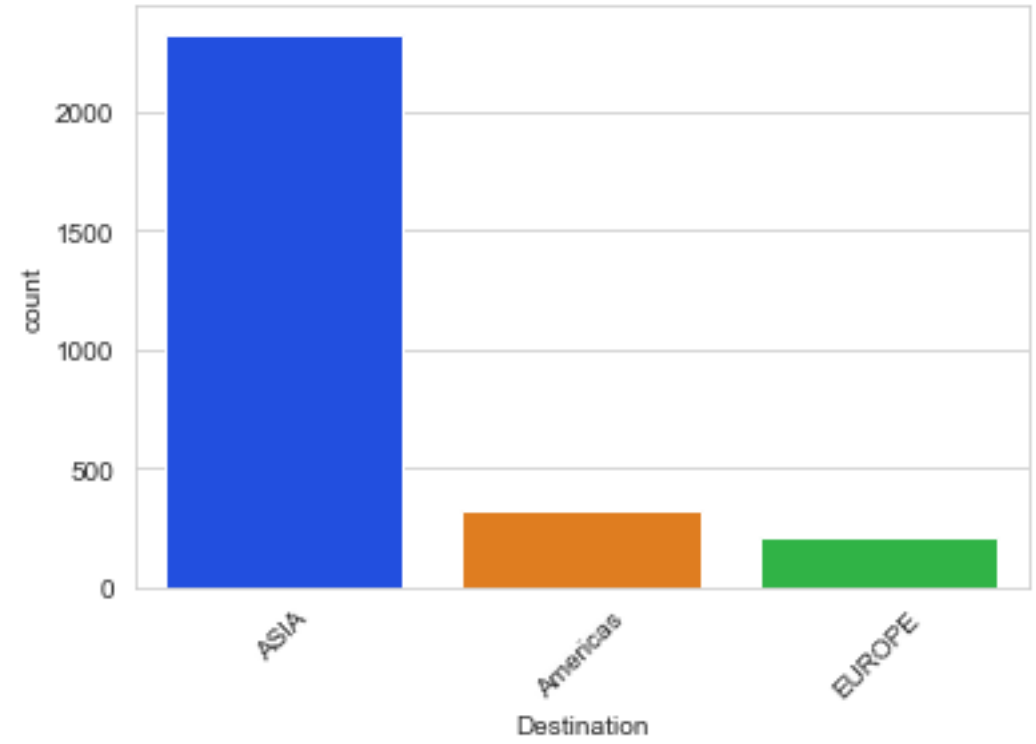

Figure 29:Countplot of Destination

Asia as Destination is chosen by most.

Skewsness and Kurtosis

```
Skewness of Age is 1.1
Kurtosis of Age is 1.44
Skewness of Commision is 3.1
Kurtosis of Commision is 13.59
Skewness of Duration is 13.79
Kurtosis of Duration is 422.68
Skewness of Sales is 2.37
Kurtosis of Sales is 6.07
```

Skewness essentially measures the symmetry of the distribution. In positively skewed, the mean of the data is greater than the median as a large number of data-pushed on the right-hand side. In negatively skewed, the mean of the data is less than the median as a large number of data-pushed on the left-hand.

If the skewness is between -0.5 & 0.5, the data are nearly symmetrical. If the skewness is between -1 & -0.5 (negative/left skewed) or between 0.5 &

1(positive/right skewed), the data are slightly skewed. If the skewness is lower than -1 (negative/left skewed) or greater than 1 (positive/right skewed), the data are extremely skewed.

Kurtosis refers to the degree of presence of outliers in the distribution. If kurtosis > 3, then it is called as Leptokurtic or heavy-tailed distribution as the kurtosis is more than normal distribution. If kurtosis = 3, then it is called as Mesokurtic as the kurtosis is same as the normal distribution. If kurtosis < 3, then it is called as Platykurtic or short-tailed distribution as the kurtosis is less than normal distribution.

Age has a short tailed distribution while Commision, Duration and Sales has a heavy tailed distribution.

Analysis of Age

```
Description of Age
-------------------------------------------------------------------------
-
count    2861.000000
mean       38.204124
std        10.678106
min         8.000000
25%        31.000000
50%        36.000000
75%        43.000000
max        84.000000
Name: Age, dtype: float64

Interquartile range (IQR) of is  12.0
Range of values:  76

Distribution of Age
-------------------------------------------------------------------------
-
```

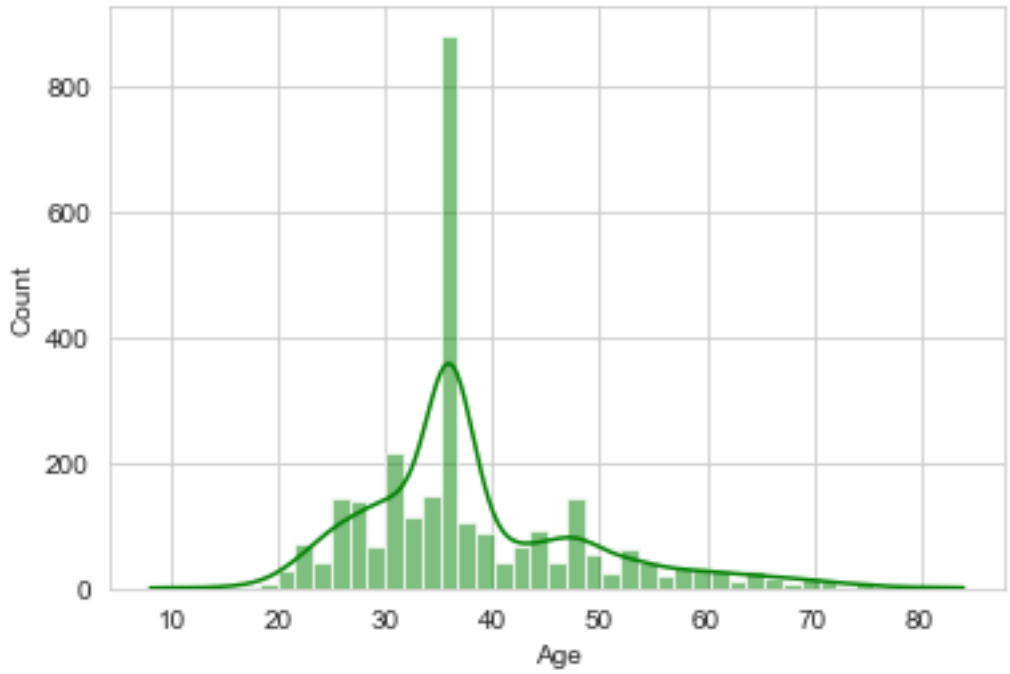

Figure 30: Distribution of Age

Age distribution is extremely right skewed.

Analysis of Commission

```
Description of Commision
-------------------------------------------------------------------------
-
count    2861.000000
mean       15.080996
std        25.826834
min         0.000000
25%         0.000000
50%         5.630000
75%        17.820000
max       210.210000
Name: Commision, dtype: float64

Interquartile range (IQR) of is  17.82
Range of values:  210.21

Distribution of Commision
-------------------------------------------------------------------------
-
```

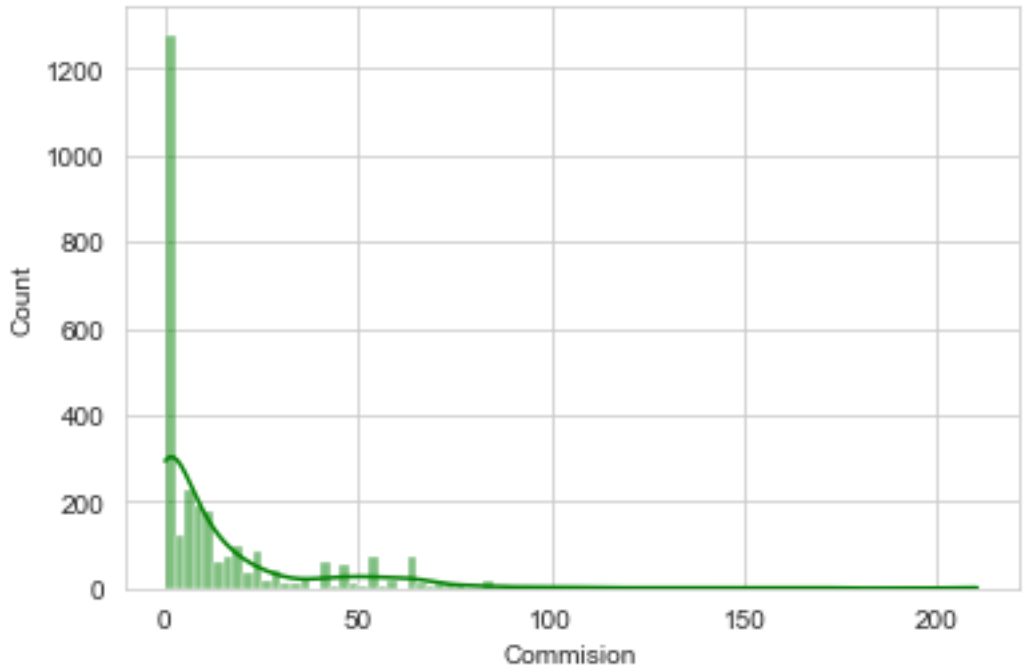

Figure 31:Distribution of Commision

Commission distribution is extremely right skewed.

Analysis of Duration

```
Description of Duration
--------------------------------------------------------------------------------
-
count    2861.000000
mean       72.130374
std       135.972828
min         0.000000
25%        12.000000
50%        28.000000
75%        66.000000
max      4580.000000
Name: Duration, dtype: float64

Interquartile range (IQR) of is  54.0
Range of values:  4580

Distribution of Duration
--------------------------------------------------------------------------------
-
```

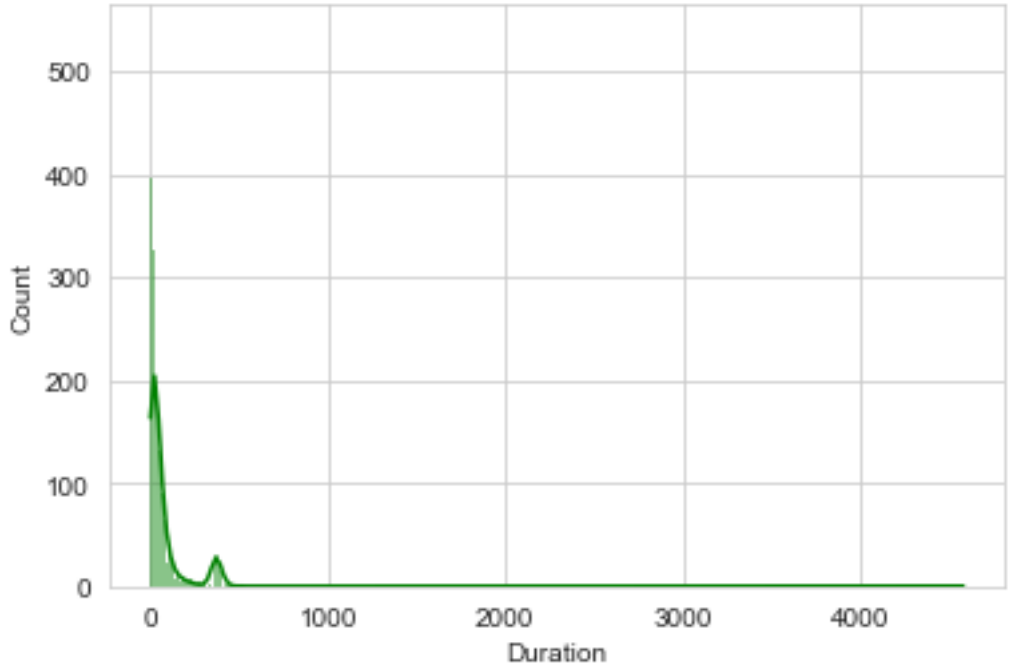

Figure 32:Distribution of Duration

Duration distribution is extremely right skewed.

Analysis of Sales

```
Description of Sales
-------------------------------------------------------------------------------
-
count    2861.000000
mean       62.366756
std        71.012142
min         0.190000
25%        21.000000
50%        33.500000
75%        69.300000
max       539.000000
Name: Sales, dtype: float64

Interquartile range (IQR) of is  48.3
Range of values:  538.81

Distribution of Sales
-------------------------------------------------------------------------------
-
```

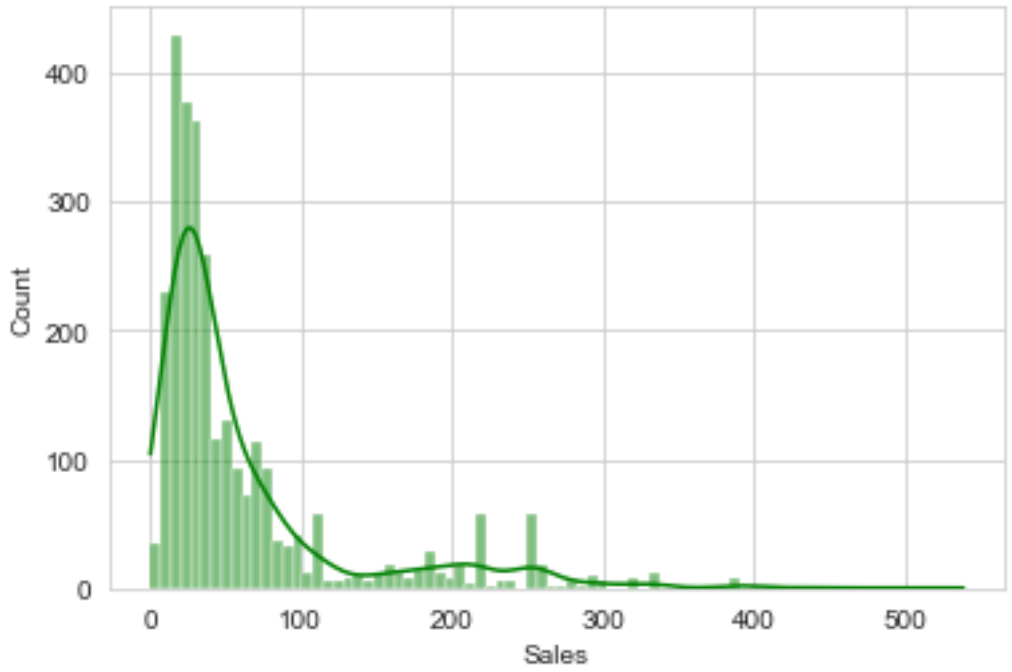

Figure 33:Distribution of Sales

Sales distribution is extremely right skewed.

Bivariate Analysis

Countplot

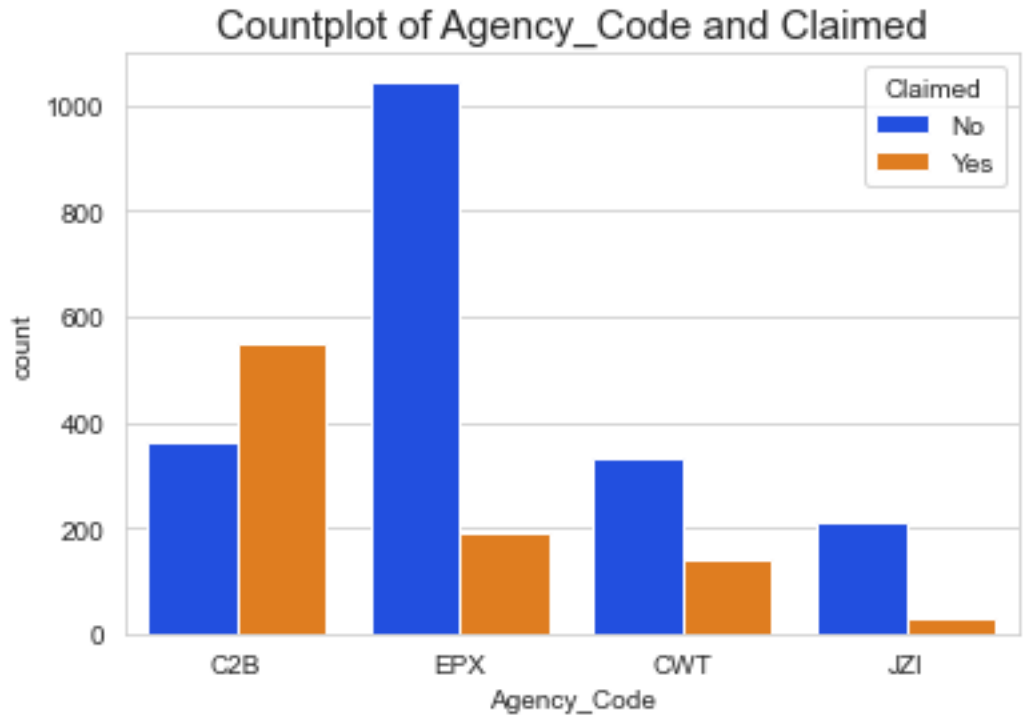

Figure 34:Countplot of Agency Code and Claimed

In C2B there are more claimed and Unclaimed.

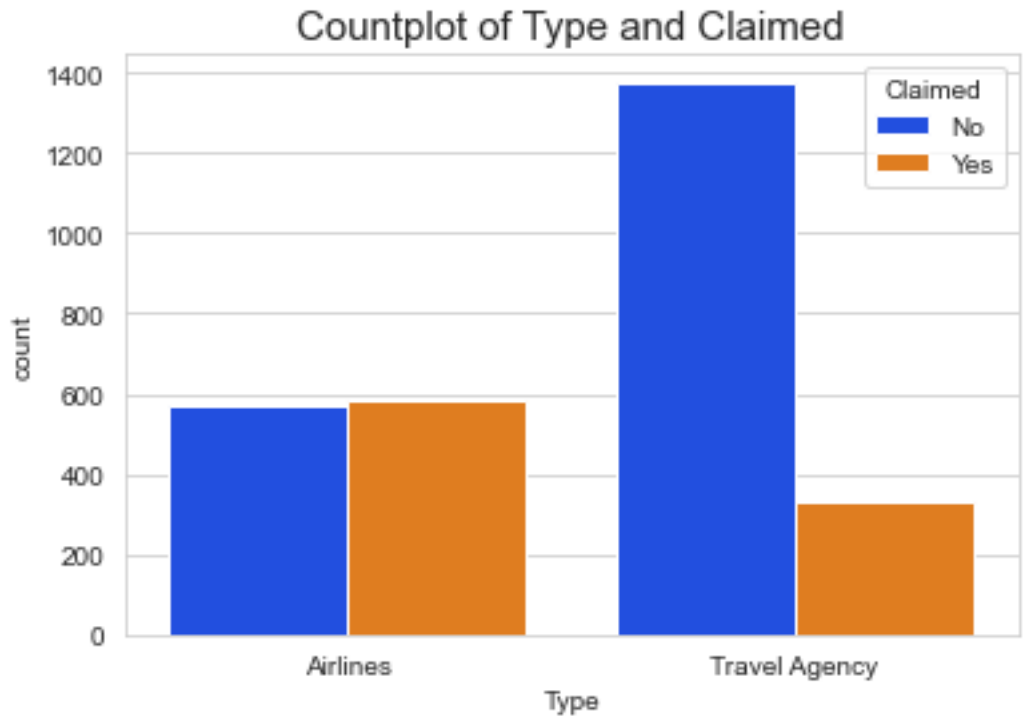

Figure 35:Countplot of Type and Claimed

Airlines type of Insurance firms has nearly the same number of Claimed and Unclaimed.

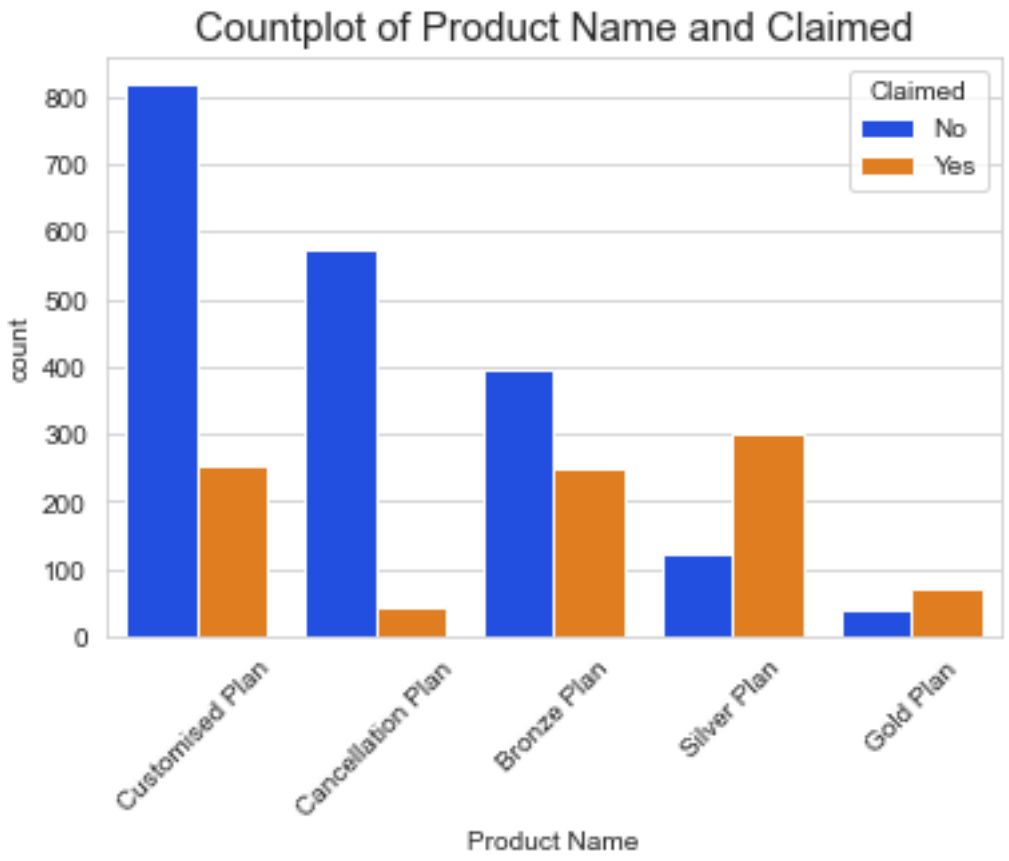

Figure 36:Countplot of Product Name and Claimed

Among all the plans, Silver Plan and Gold Plan has more Claimed than Unclaimed.

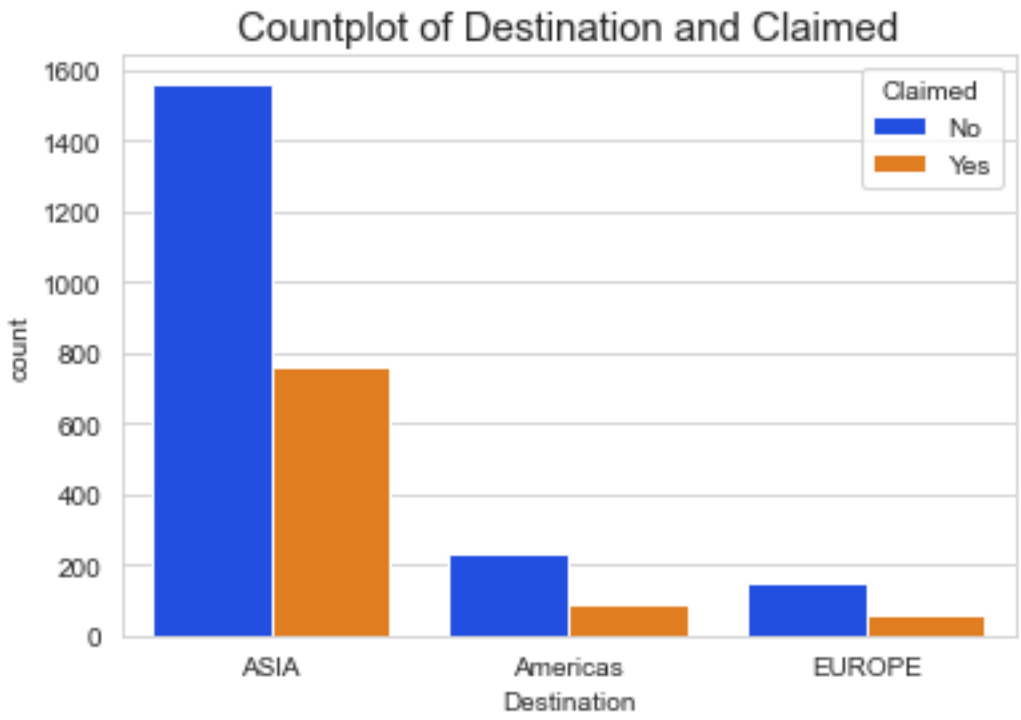

Figure 37:Countplot of Destination and Claimed

In all the destination there are more Unclaimed than Claimed.

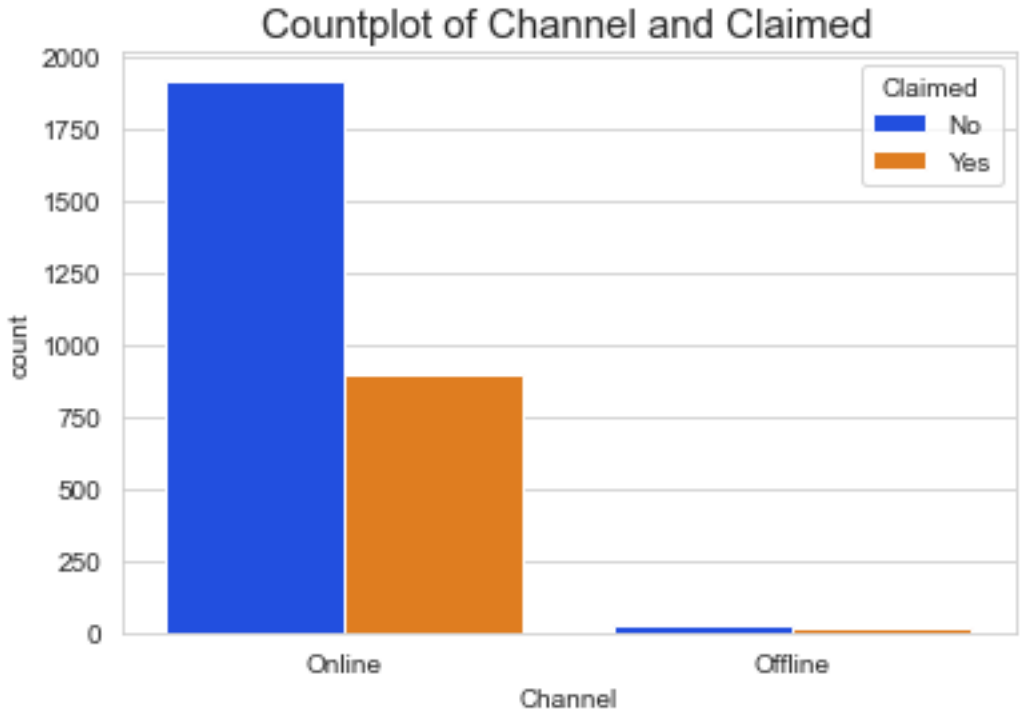

Figure 38:Countplot of Channel and Climed

Online has more unclaimed than claimed. Online has more Claimed than Offline.

Boxplot

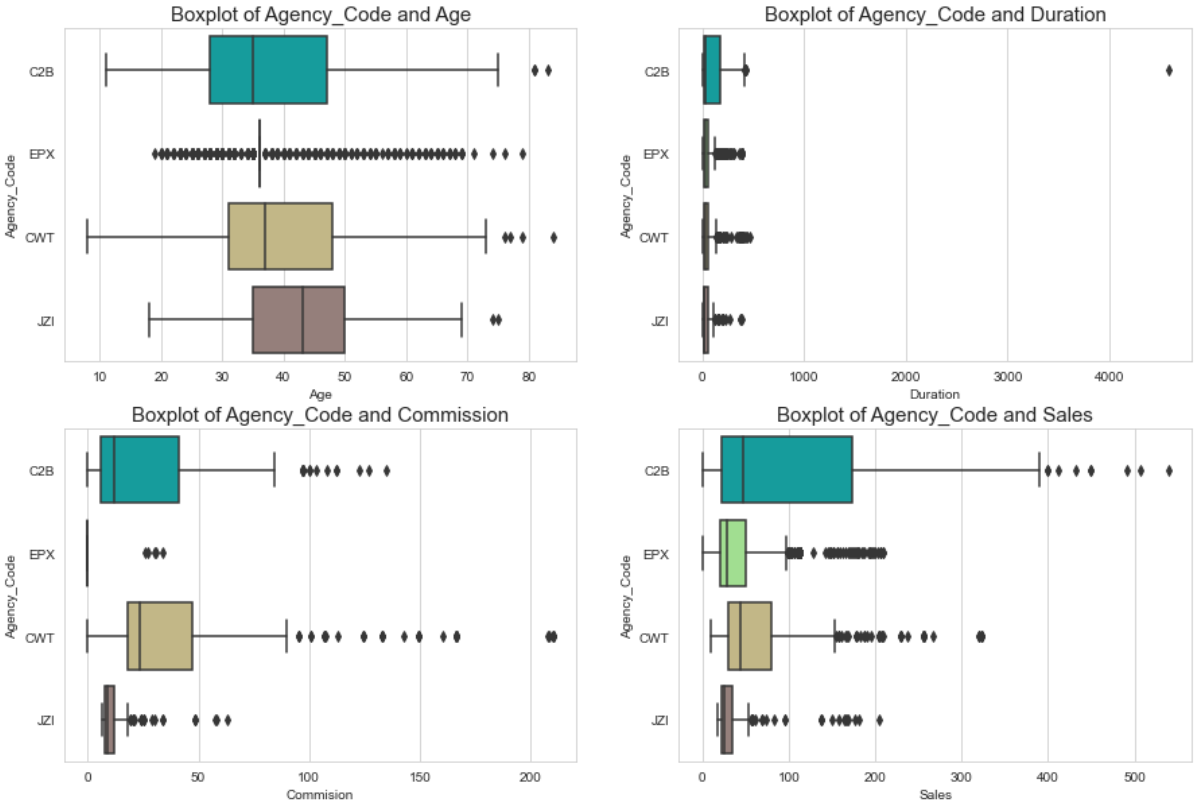

Figure 39:Boxplot of Agency Code and numeric variables

Agency Code C2B has more sales above Rs 5000.

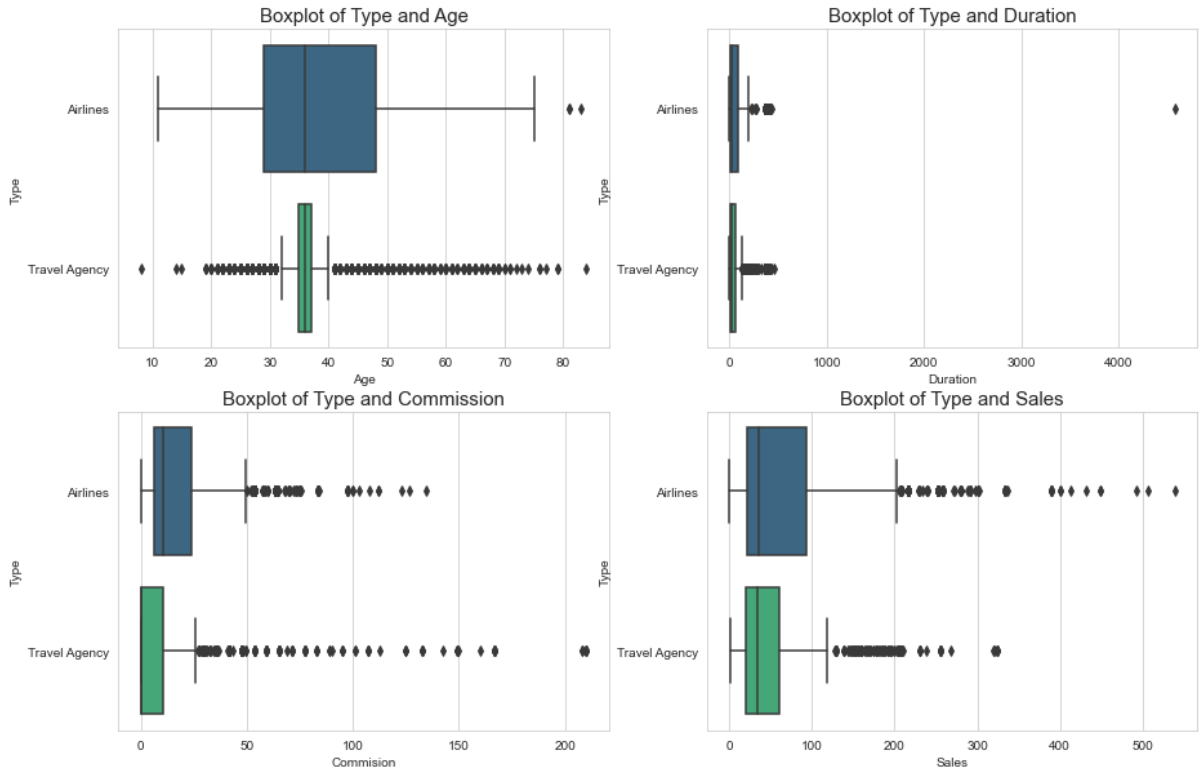

Figure 40: Boxplot of Type and numeric variables

Airline type of tour insurance firms has most people aged above 35 .

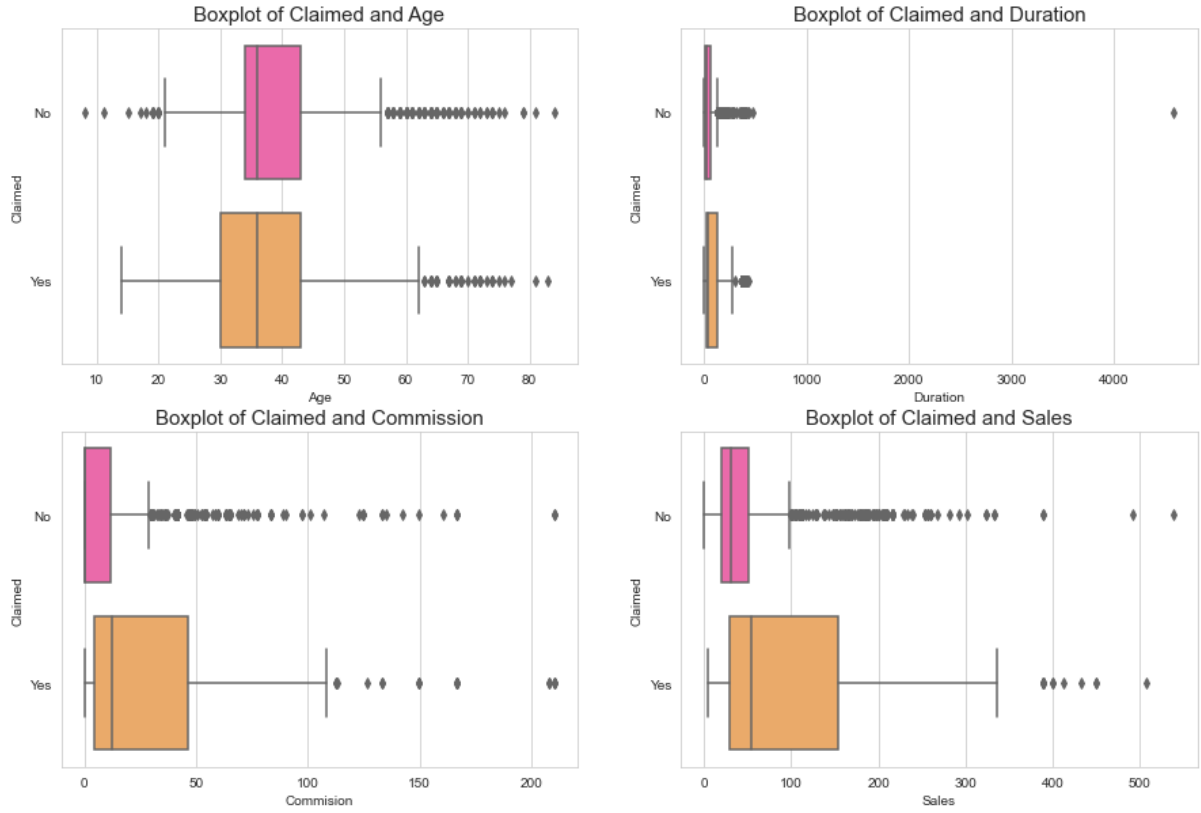

Figure 41:Boxplot of Claimed and numeric variables

When there is more commission then there is high claims frequency.

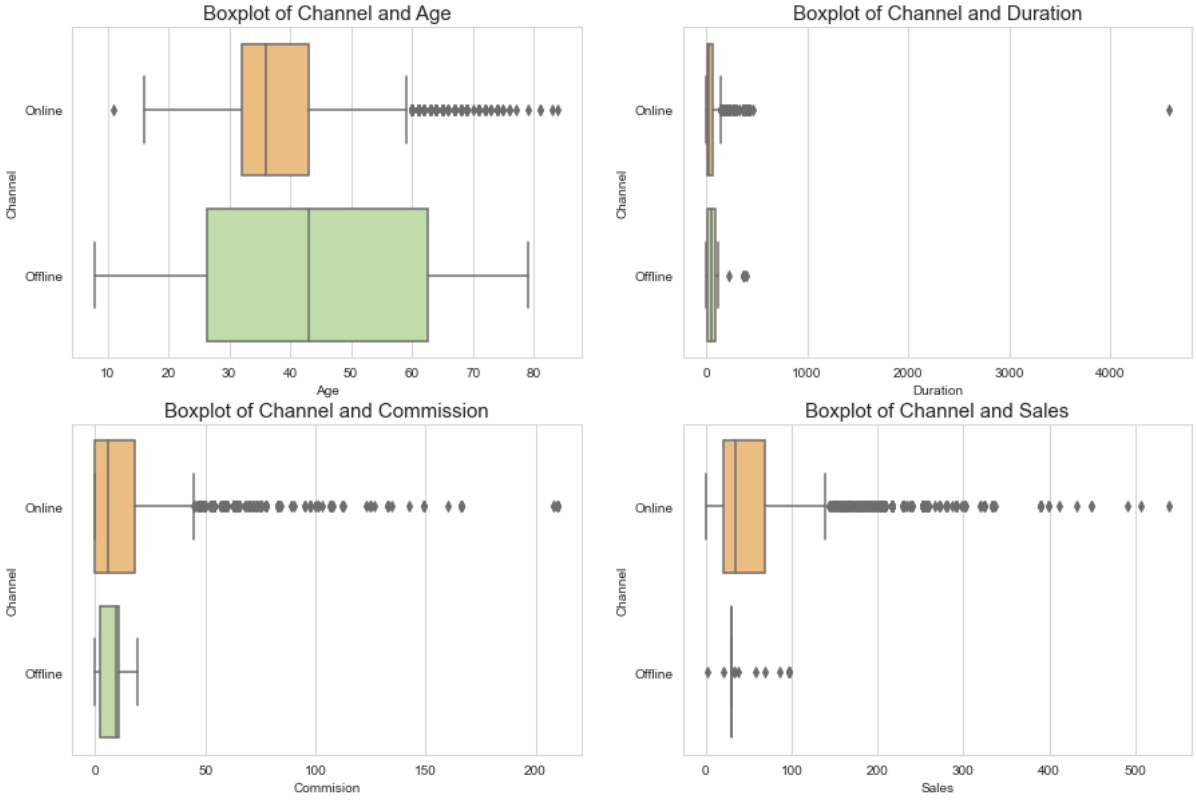

Figure 42: Boxplot of Channel and numeric variables

People with age more than 40 mostly uses Offline.

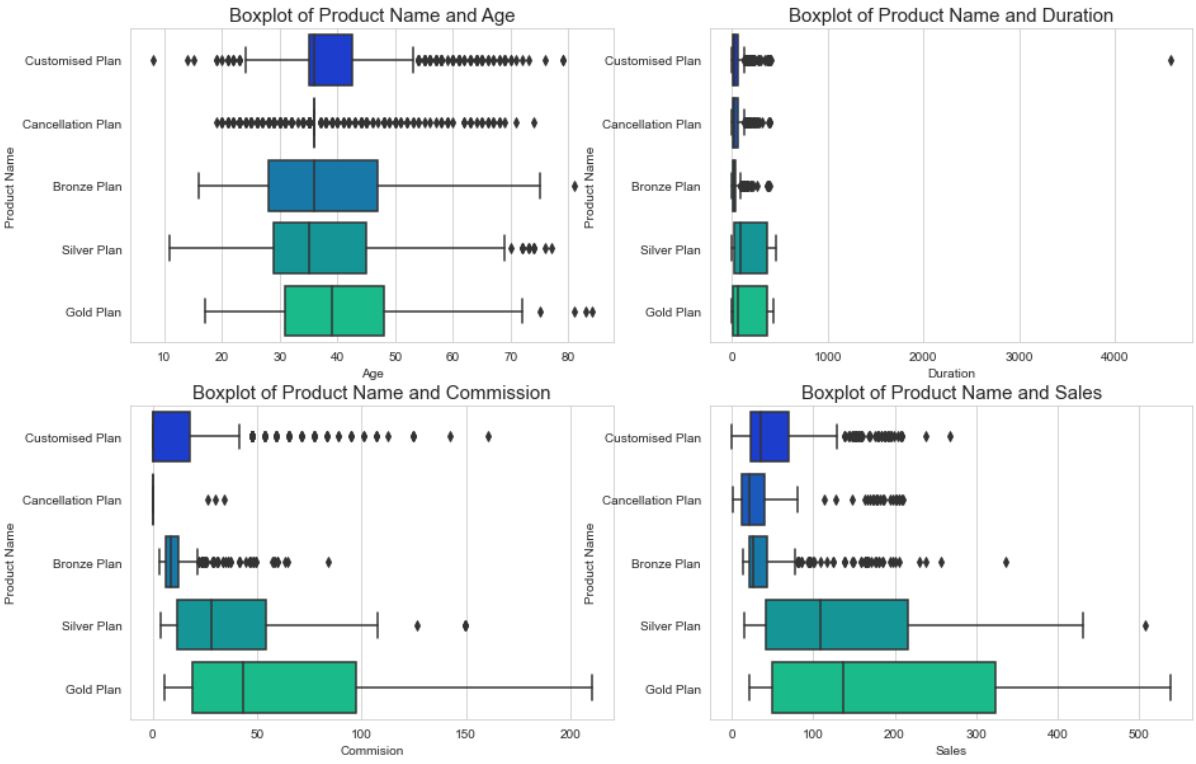

Figure 43:Boxplot of Product Name and numeric variables

Gold Plan and Silver Plan has more Sales and Commission. The median age is 40 for people having Gold Plan.

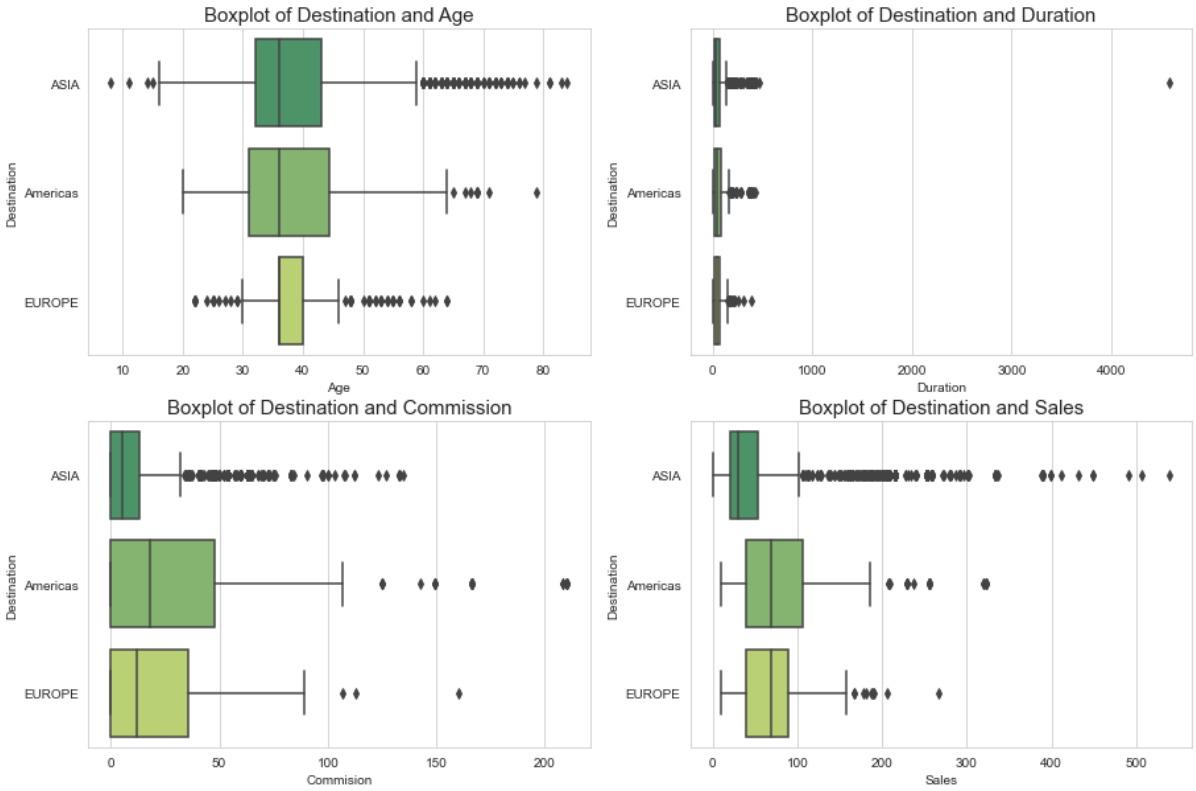

Figure 44:Boxplot of Destination and numeric variables

Destination as Asia has lower sales and commission than America and Europe.

PAIRPLOT

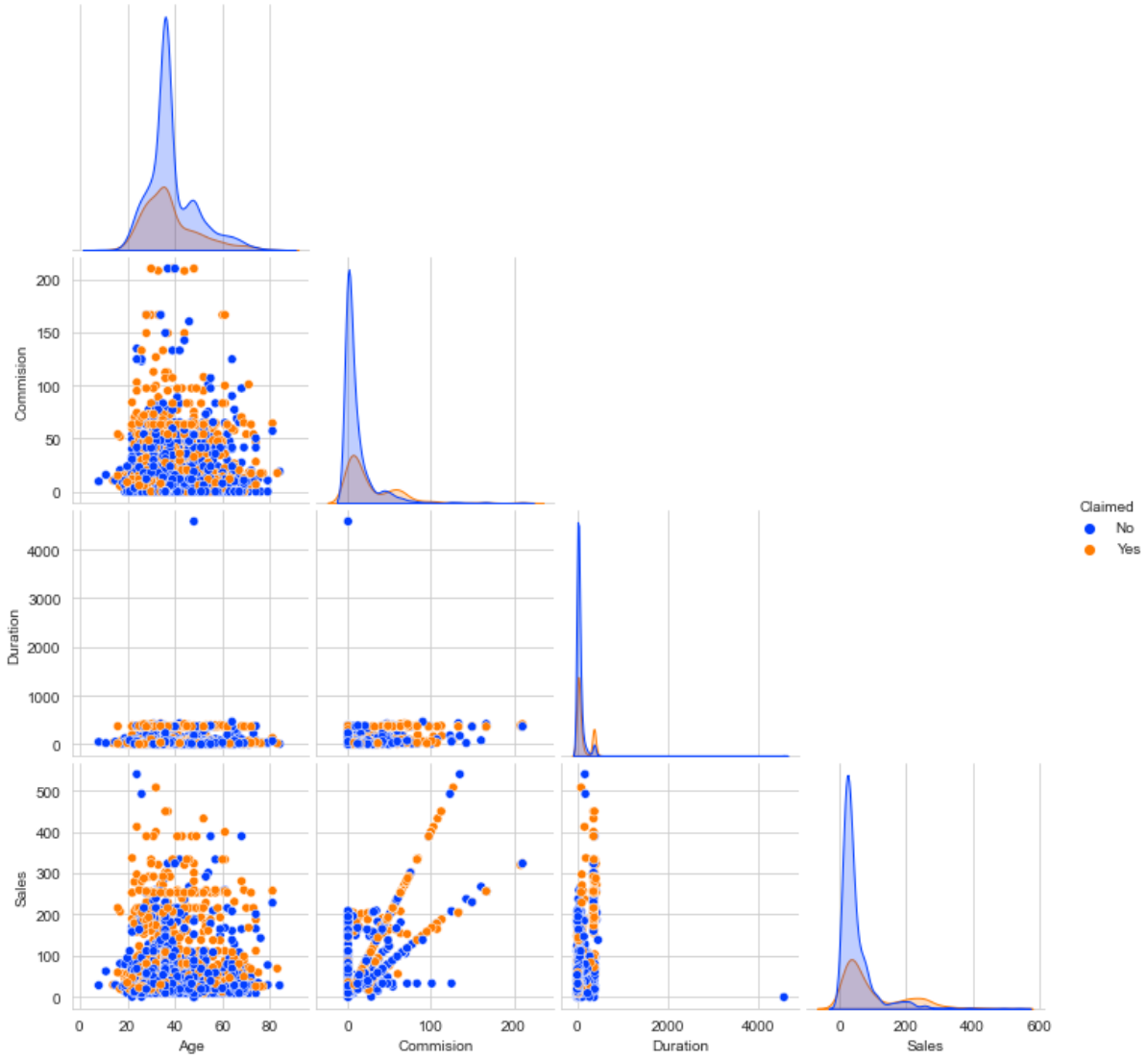

Figure 45:Pairplot2

When commission increases Sales also increases. Sales above Rs 15000 has more claims.

CORRELATION HEATMAP

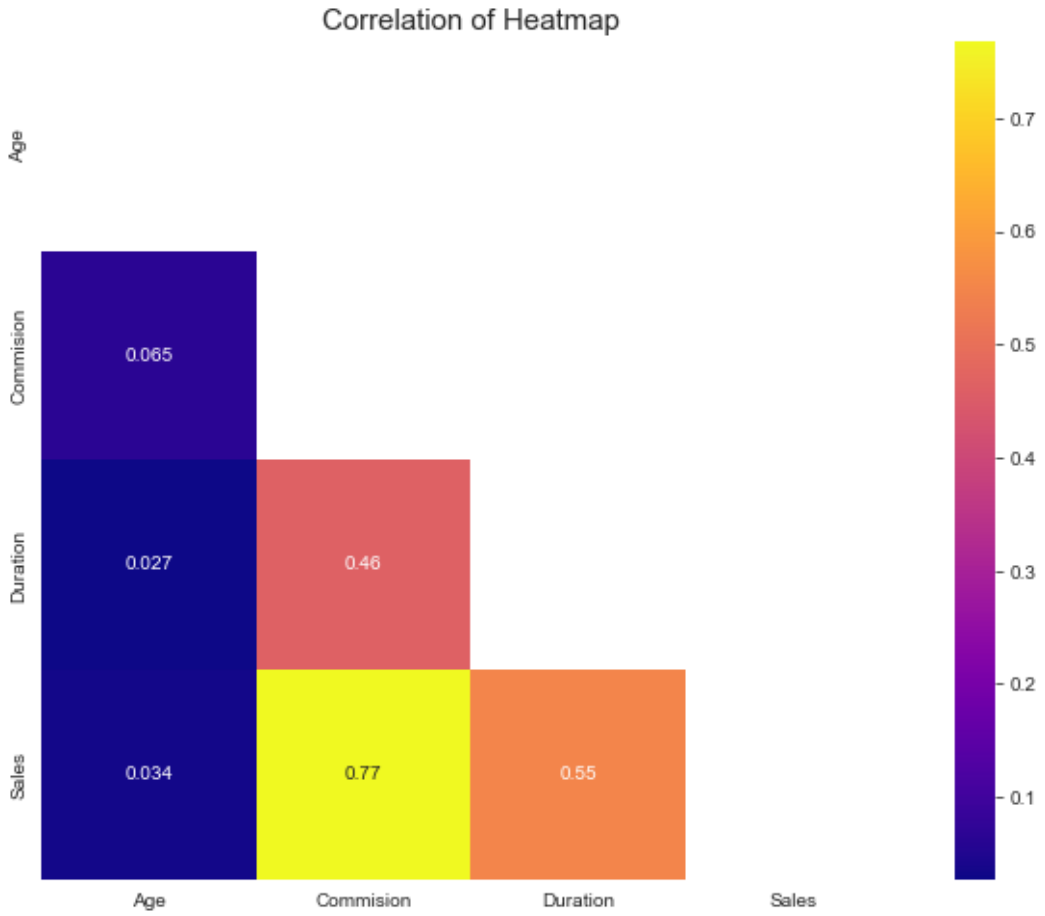

Figure 46:Correlation Heatmap

From the heatmap and pairplot we can say that Commission has a positive correlation with Sales.

Multivariate Analysis

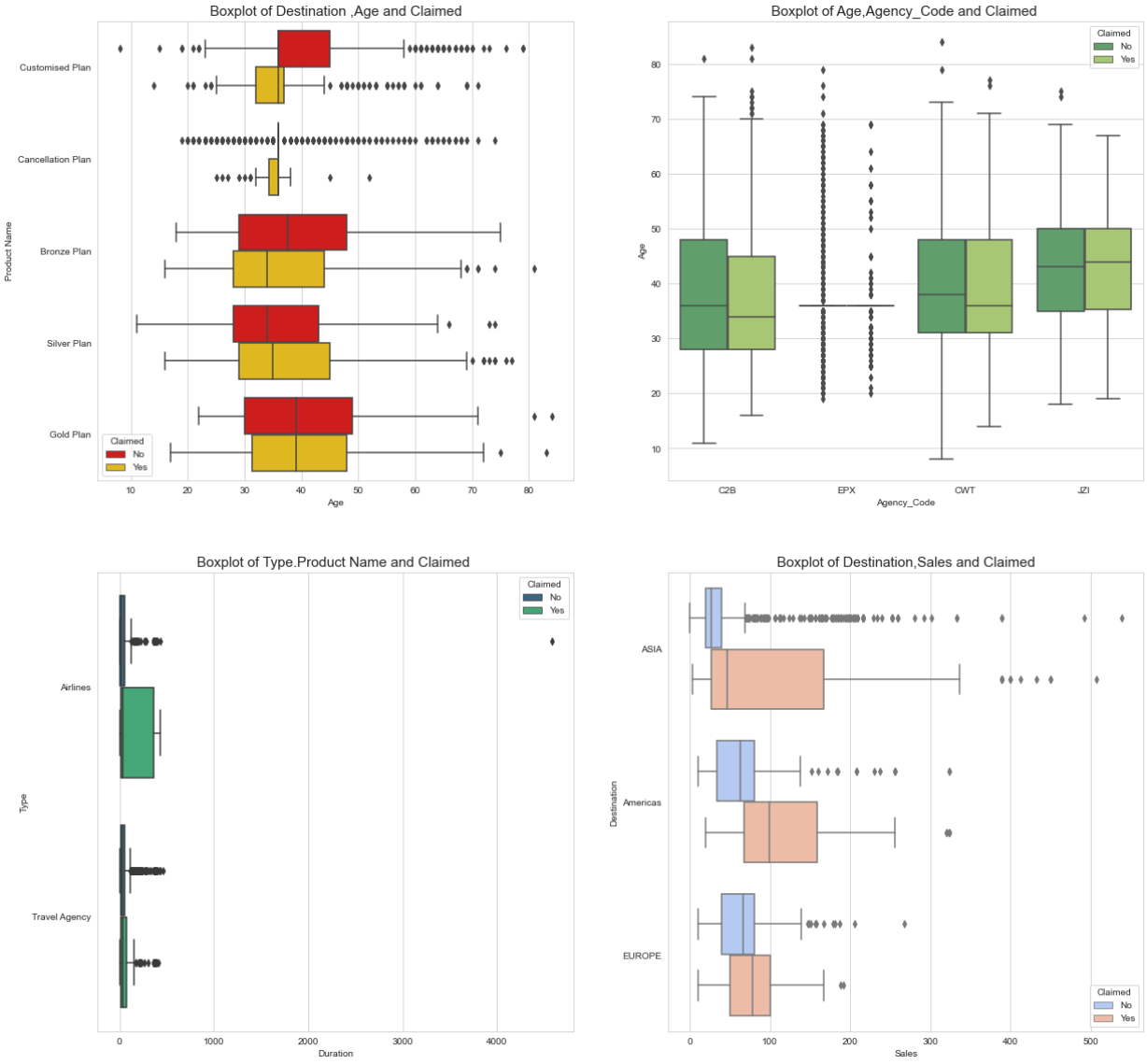

Figure 47:Multivariate Analysis

Most of the Insurance of Airlines have been claimed. For Asia, the sales going above 4000 Rs has more claim frequency.

2.2. Data Split: Split the data into test and train, build classification model CART, Random Forest, Artificial Neural Network.

Decision tree in Python can take only numerical / categorical columns. It cannot take string / object types. Hence, object types are converted into categorical.

```
feature: Agency_Code
['C2B', 'EPX', 'CWT', 'JZI']
Categories (4, object): ['C2B', 'CWT', 'EPX', 'JZI']
[0 2 1 3]


feature: Type
['Airlines', 'Travel Agency']
Categories (2, object): ['Airlines', 'Travel Agency']
[0 1]


feature: Claimed
['No', 'Yes']
Categories (2, object): ['No', 'Yes']
[0 1]


feature: Channel
['Online', 'Offline']
Categories (2, object): ['Offline', 'Online']
[1 0]


feature: Product Name
['Customised Plan', 'Cancellation Plan', 'Bronze Plan', 'Silver Plan', 'Gold Plan']
Categories (5, object): ['Bronze Plan', 'Cancellation Plan', 'Customised Plan', 'Gold Plan', 'Silver Plan']
[2 1 0 4 3]


feature: Destination
['ASIA', 'Americas', 'EUROPE']
Categories (3, object): ['ASIA', 'Americas', 'EUROPE']
[0 1 2]
```

:

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
0	48	0	0	0	0.70	1	7	2.51	2	0
1	36	2	1	0	0.00	1	34	20.00	2	0
2	39	1	1	0	5.94	1	3	9.90	2	1
3	36	2	1	0	0.00	1	4	26.00	1	0
4	33	3	0	0	6.30	1	53	18.00	0	0

Table 18: Dataset after conversion

The testing data size is 30% of total records. There are 2002 records in the train set.

The proportion of target class in training data is.

```
0    0.678821
1    0.321179
Name: Claimed, dtype: float64
```

The proportion of target class in testing data is.

```
0    0.684517
1    0.315483
Name: Claimed, dtype: float64
```

There is an imbalance in the dataset as there is a huge difference in the proportion.

CART

The parameters given in Grid Search for CART Model is

```
GridSearchCV(cv=5, estimator=DecisionTreeClassifier(),
             param_grid={'max_depth': [8, 10, 12],
                         'min_samples_leaf': [30, 50, 60],
                         'min_samples_split': [60, 100, 150]})
```

min sample split is 2%-3% of training set. Values of max-depth is suggested to be taken from 8-15 to avoid overfitting and underfitting. Min sample leaf is taken as min sample split divided by 3.

The model with best parameter is:

```
DecisionTreeClassifier(max_depth=8, min_samples_leaf=30,
min_samples_split=150)
```

The value of cross validation is taken as 5.

The feature importance values are:

	Feature_Imp
Agency_Code	0.553262
Sales	0.271909
Duration	0.059774
Commision	0.046018
Product Name	0.043635
Age	0.025403
Type	0.000000
Channel	0.000000
Destination	0.000000

Agency code has more feature importance.

Random Forest

The parameters given in Grid Search for Random Forest Model is

```
GridSearchCV(cv=5, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [8, 10, 12], 'max_features': [3, 4, 5],
                         'min_samples_leaf': [20, 30, 50, 60],
                         'min_samples_split': [60, 100, 150],
                         'n_estimators': [100, 200]})
```

min sample split is 2%-3% of training set. Values of max-depth is suggested to be taken from 8-15 to avoid overfitting and underfitting. Value of Max feature is taken as square root of number of independent variable to half of the number of independent variables(10).Hence 3,4 and 5 is taken.

The model with best parameter is:

```
RandomForestClassifier(max_depth=10, max_features=5, min_samples_leaf=20,
                       min_samples_split=100)
```

In [107]:

The value of cross validation is taken as 5.

The feature importance values are:

	Feature_Imp
Agency_Code	0.371063
Product Name	0.225863
Sales	0.191553
Commision	0.080464
Duration	0.060117
Age	0.042875
Type	0.019806
Destination	0.008260
Channel	0.000000

Agency code has more feature importance

Artificial Neural Network(ANN)

ANN algorithm requires the model to be scaled hence Standard Scaler is used. Below are the values of scaled test data.

```
array([[-0.68701032, -0.27289013,  0.83463176, ...,  0.50264304,
         0.24339146, -0.44775345],
       [ 2.79357258,  0.71683095,  0.83463176, ..., -0.46611137,
        -0.53499465, -0.44775345],
       [ 0.34775757, -1.2626112 , -1.19813318, ...,  0.31743999,
         1.80016368, -0.44775345],
       ...,
       [ 1.19438584, -1.2626112 , -1.19813318, ..., -0.65131442,
        -1.31338076, -0.44775345],
       [ 1.38252546,  0.71683095,  0.83463176, ..., -0.58008248,
         0.24339146, -0.44775345],
       [-0.21666128,  0.71683095,  0.83463176, ..., -0.58008248,
         0.24339146, -0.44775345]])
```

The parameters given in Grid Search for ANN Model is

```
GridSearchCV(cv=5, estimator=MLPClassifier(),
             param_grid={'activation': ['logistic', 'relu'],
                         'hidden_layer_sizes': [6], 'max_iter': [10000],
                         'solver': ['sgd', 'adam'],
                         'tol': [0.0001, 0.01, 0.001]})
```

SGD and Adam are the important solvers for MLP Classifier. Adam is built on top of sgd. Max iteration must be above 2500.tol(tolerance) or Learning rate is kept as low as possible. It is a tradeoff between processing time and accuracy. If Learning rate increases accuracy will drop and processing time will increase. 0.0001 is the industry recommended value.

Hidden layer size is taken as the sum of number of independent variable and 1(which is the number of output neuron) divided by 12. Hence here 10+1 is 11 and divided by 2 is around 6.

The best parameters are:

```
{'activation': 'logistic',
 'hidden_layer_sizes': 6,
 'max_iter': 10000,
 'solver': 'adam',
 'tol': 0.0001}
```

2.3 Performance Metrics: Comment and Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score, classification reports for each model.

Confusion Matrix *:*
TN / True Negative: when a case was negative and predicted negative
TP / True Positive: when a case was positive and predicted positive
FN / False Negative: when a case was positive but predicted negative(Type 2 error)
FP / False Positive: when a case was negative but predicted positive(Type 1 error)

In this problem **False Negative** is an important metric as it denotes claimed cases as unclaimed ones which generates loss for the Insurance firm.

Classification Report

Precision is the ability of a classifier not to label an instance positive that is actually negative.

$$Precision = TP/(TP + FP)$$

Equation 1:Precision

Recall is the fraction of positives that were correctly identified.

$$Recall = TP/(TP+FN)$$

Equation 2:Recall

F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0.

$$F1\ Score = 2*(Recall * Precision) / (Recall + Precision)$$

Equation 3:F1 Score

Recall is the important metric as it is inversely proportional to Type 2 error

CART

Classification Report – Train Data

	precision	recall	f1-score	support
0	0.80	0.90	0.85	1359
1	0.71	0.53	0.61	643
accuracy			0.78	2002
macro avg	0.75	0.71	0.73	2002
weighted avg	0.77	0.78	0.77	2002

Table 19:Classification Report-CART-Train Data

Classification Report – Test Data

	precision	recall	f1-score	support
0	0.80	0.89	0.84	588
1	0.69	0.51	0.58	271
accuracy			0.77	859
macro avg	0.74	0.70	0.71	859
weighted avg	0.76	0.77	0.76	859

Table 20:Classification Report-CART-Test Data

Confusion Matrix – Train Data

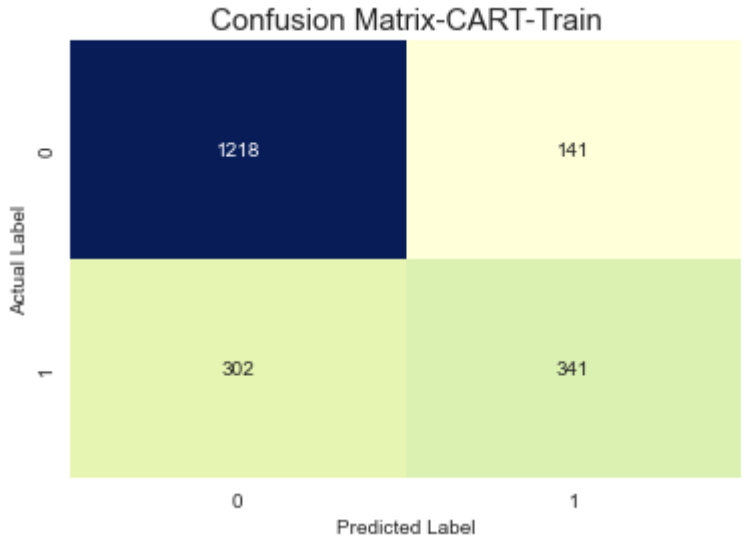

Figure 48:Confusion Matrix-CART-Train Data

Confusion Matrix – Test Data

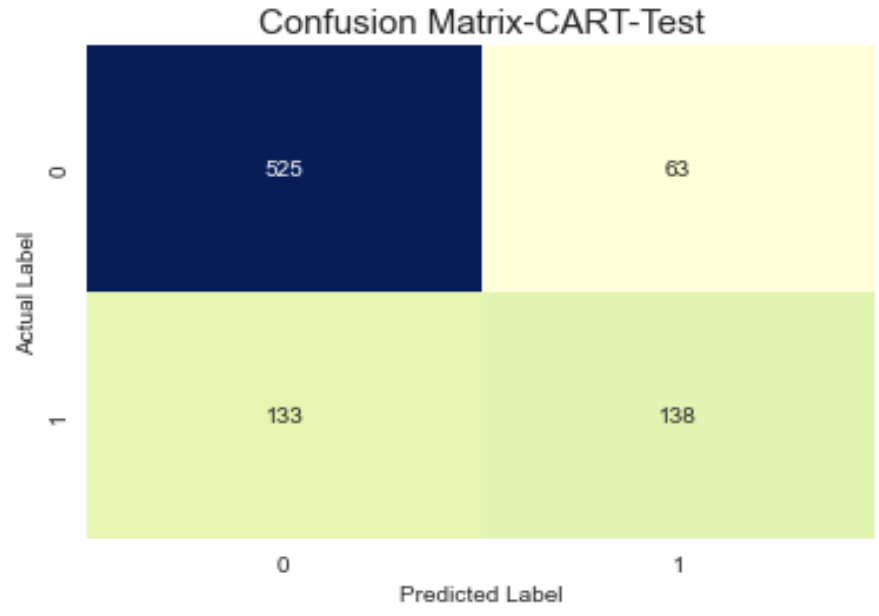

Figure 49:Confusion Matrix-CART-Test Data

Accuracy for Train Data – 0.7787212787212787

Accuracy for Test Data – 0.7718277066356228

The accuracies are nearly the same. Hence there is no underfitting and Overfitting. It has a good score hence a good model.

ROC Curve and AUC Score – Train Data

AUC Score for CART train data: 0.823

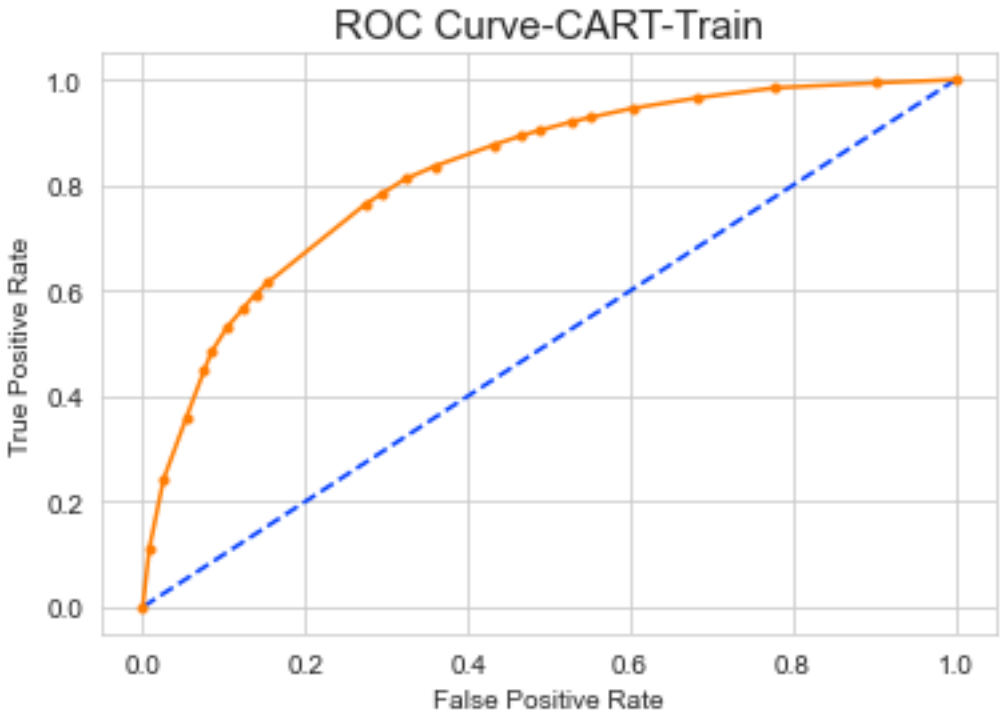

Figure 50:ROC Curve-CART-Train Data

ROC Curve and AUC Score – Test Data

AUC Score for CART test data: 0.778

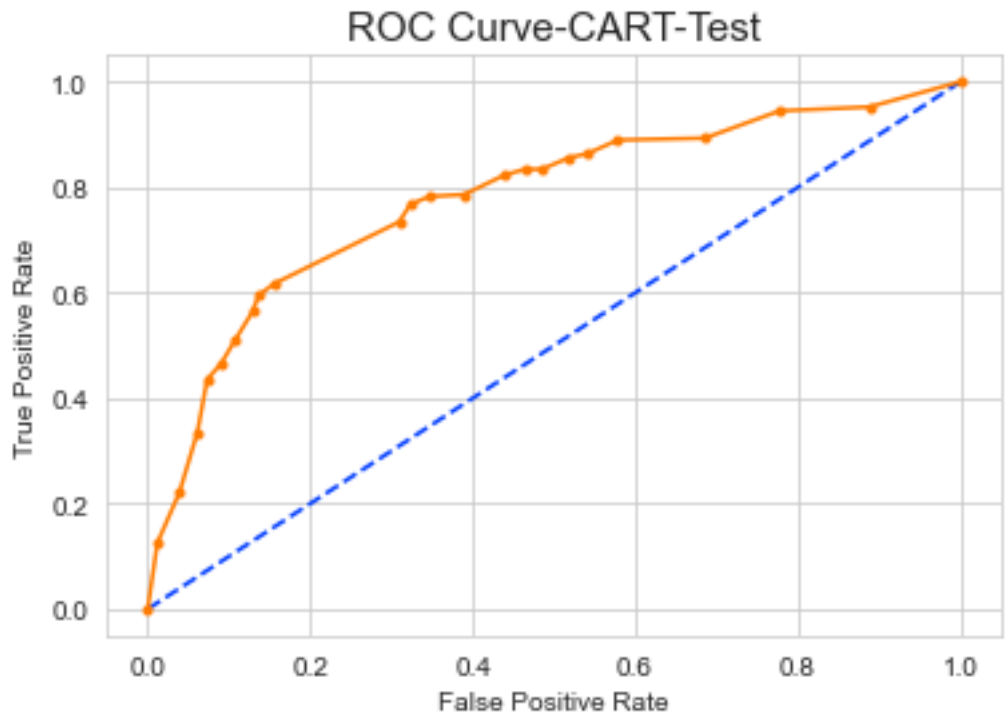

Figure 51:ROC Curve-CART-Test Data

Random Forest

Classification Report – Train Data

	precision	recall	f1-score	support
0	0.81	0.89	0.85	1359
1	0.71	0.56	0.63	643
accuracy			0.79	2002
macro avg	0.76	0.73	0.74	2002
weighted avg	0.78	0.79	0.78	2002

Table 21:Classification Report-RF-Train Data

Classification Report – Test Data

	precision	recall	f1-score	support
0	0.81	0.89	0.85	588
1	0.70	0.54	0.61	271
accuracy			0.78	859
macro avg	0.75	0.72	0.73	859
weighted avg	0.77	0.78	0.77	859

Table 22:Classification Report-RF-Test Data

Confusion Matrix – Train Data

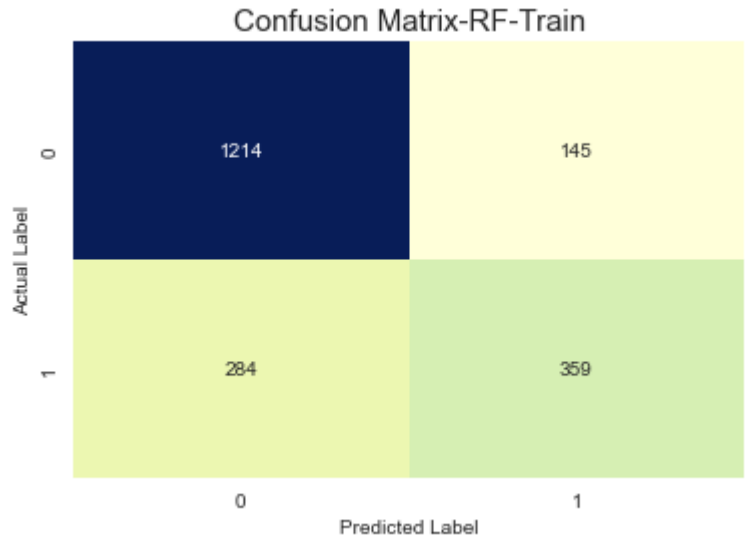

Figure 52: Confusion Matrix-RF-Train Data

Confusion Matrix – Test Data

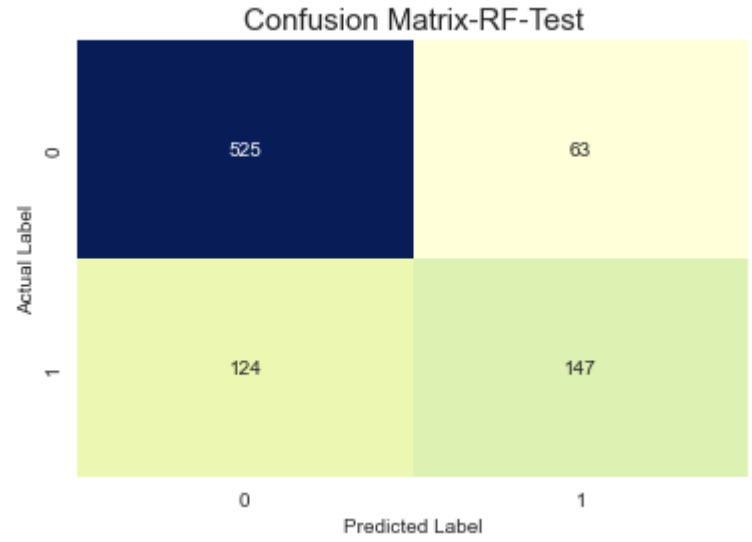

Figure 53:Confusion Matrix-RF-Test Data

Accuracy for Train Data – 0.7857142857142857

Accuracy for Test Data – 0.7823050058207218

The accuracies are nearly the same. Hence there is no underfitting and Overfitting. It has a good score hence a good model.

ROC Curve and AUC Score – Train Data

AUC Score for Random Forest train data: 0.835

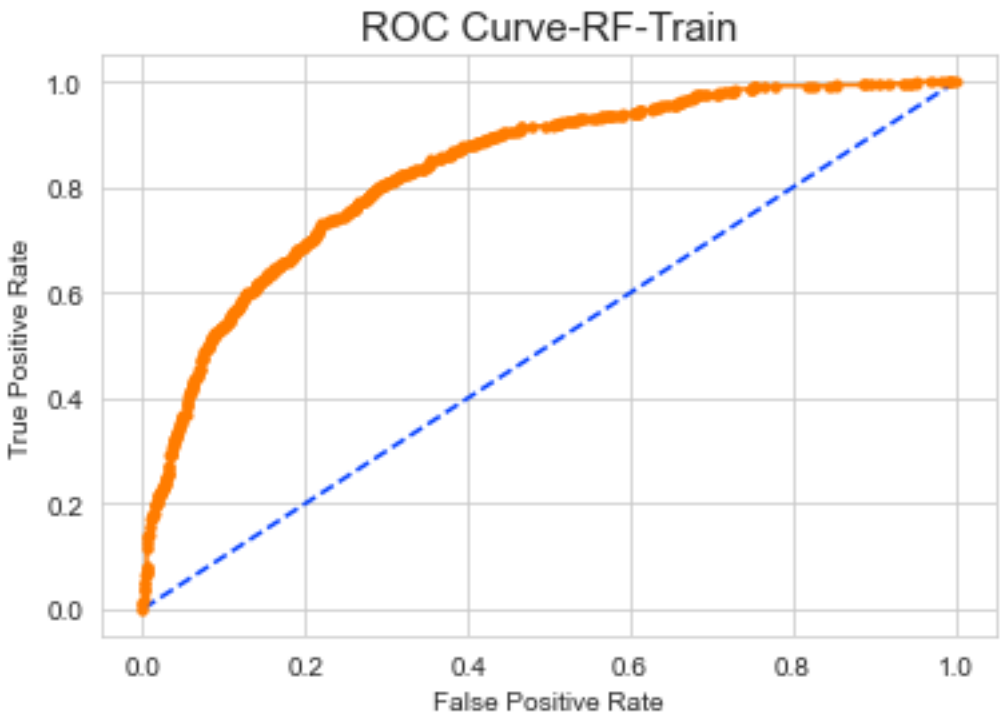

Figure 54:ROC Curve-RF-Train Data

ROC Curve and AUC Score – Test Data

AUC Score for Random Forest test data: 0.815

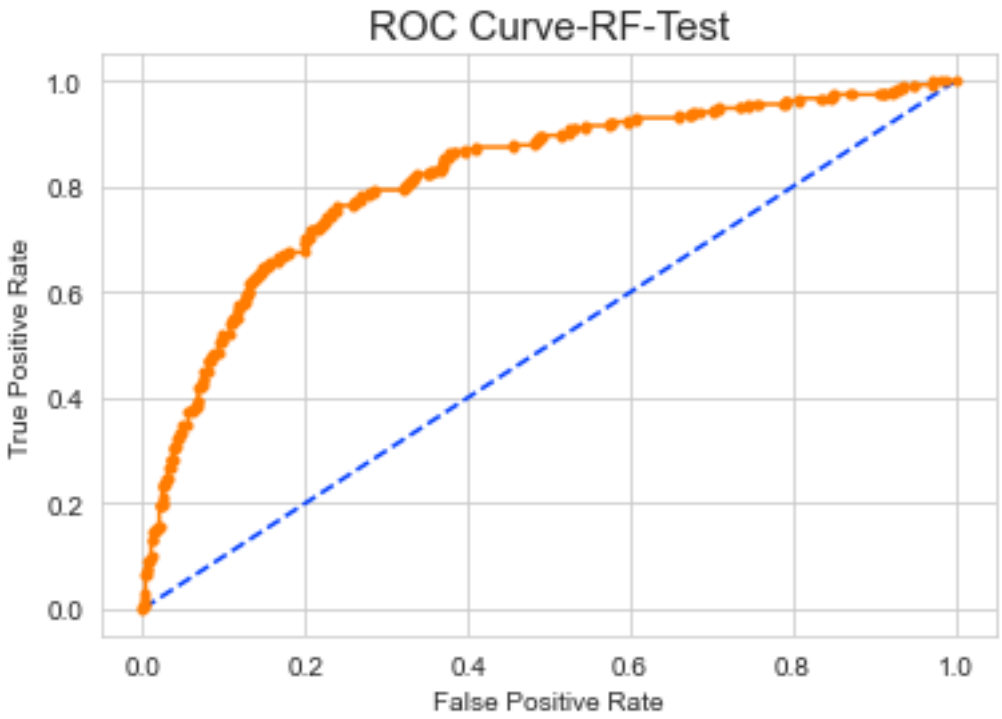

Figure 55:ROC Curve-RF-Test Data

Artificial Neural Network

Classification Report – Train Data

	precision	recall	f1-score	support
0	0.78	0.89	0.84	1359
1	0.68	0.48	0.56	643
accuracy			0.76	2002
macro avg	0.73	0.69	0.70	2002
weighted avg	0.75	0.76	0.75	2002

Table 23:Classification Report-ANN-Train Data

Classification Report – Test Data

	precision	recall	f1-score	support
0	0.78	0.90	0.84	588
1	0.69	0.46	0.55	271
accuracy			0.76	859
macro avg	0.74	0.68	0.69	859
weighted avg	0.75	0.76	0.75	859

Table 24:Classification Report-ANN-Test Data

Confusion Matrix – Train Data

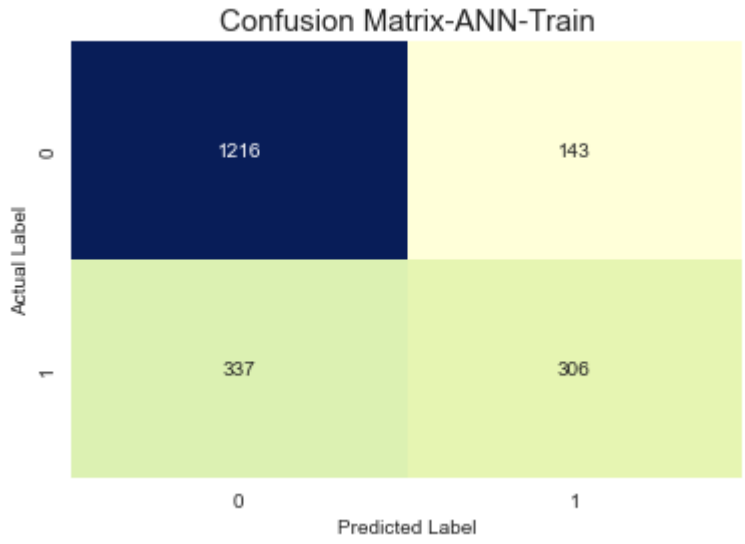

Figure 56: Confusion Matrix-ANN-Train Data

Confusion Matrix – Test Data

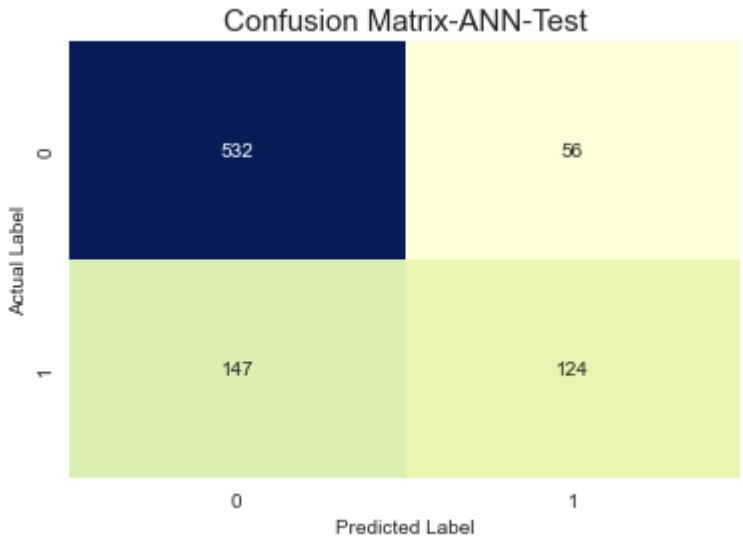

Figure 57:Confusion Matrix-ANN-Test Data

Accuracy for Train Data – 0.7602397602397603

Accuracy for Test Data – 0.7636786961583236
The accuracies are nearly the same. Hence there is no underfitting and Overfitting. It has a good score hence a good model.

ROC Curve and AUC Score – Train Data
AUC Score for ANN train data: 0.785

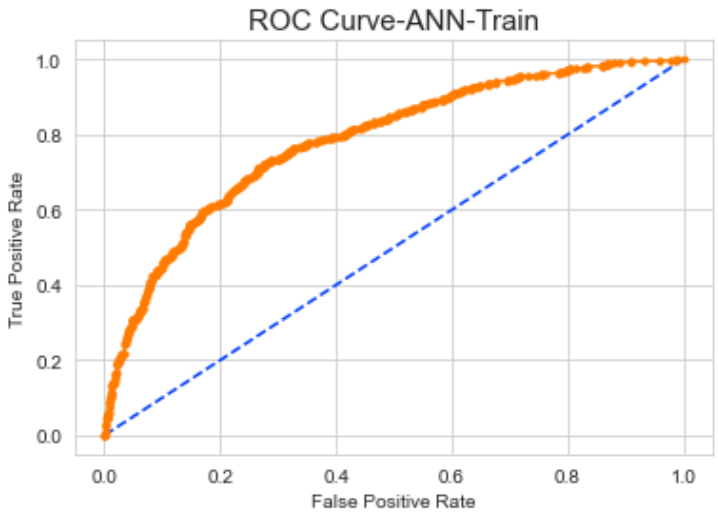

Figure 58:ROC Curve-ANN-Train Data

ROC Curve and AUC Score – Test Data

AUC Score for ANN test data: 0.795

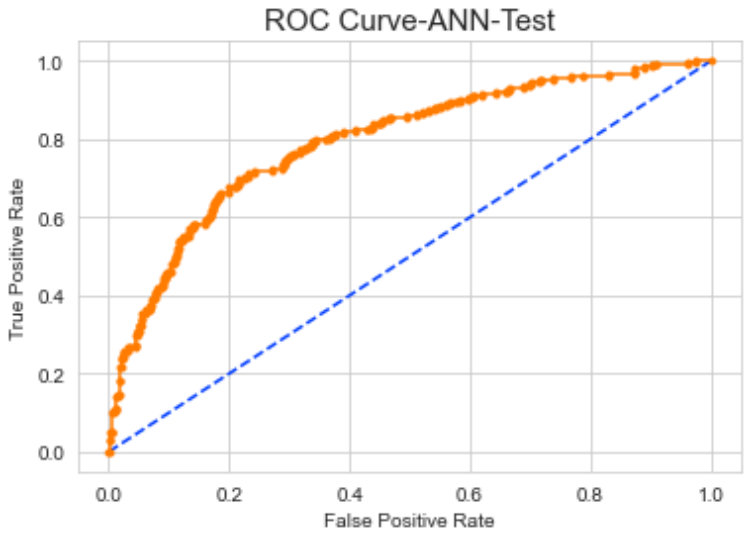

Figure 59: ROC Curve-ANN-Test Data

For all the above 3 models we have a good precision value for both train data and test data.

2.4 Final Model: Compare all the models and write an inference which model is best/optimized.

	CART Train	CART Test	Random Forest Train	Random Forest Test	Neural Network Train	Neural Network Test
Accuracy	0.78	0.77	0.79	0.78	0.76	0.76
AUC	0.82	0.78	0.83	0.81	0.79	0.80
Recall	0.53	0.51	0.56	0.54	0.48	0.46
Precision	0.71	0.69	0.71	0.70	0.68	0.69
F1 Score	0.61	0.58	0.63	0.61	0.56	0.55

Table 25: Performance Metrics

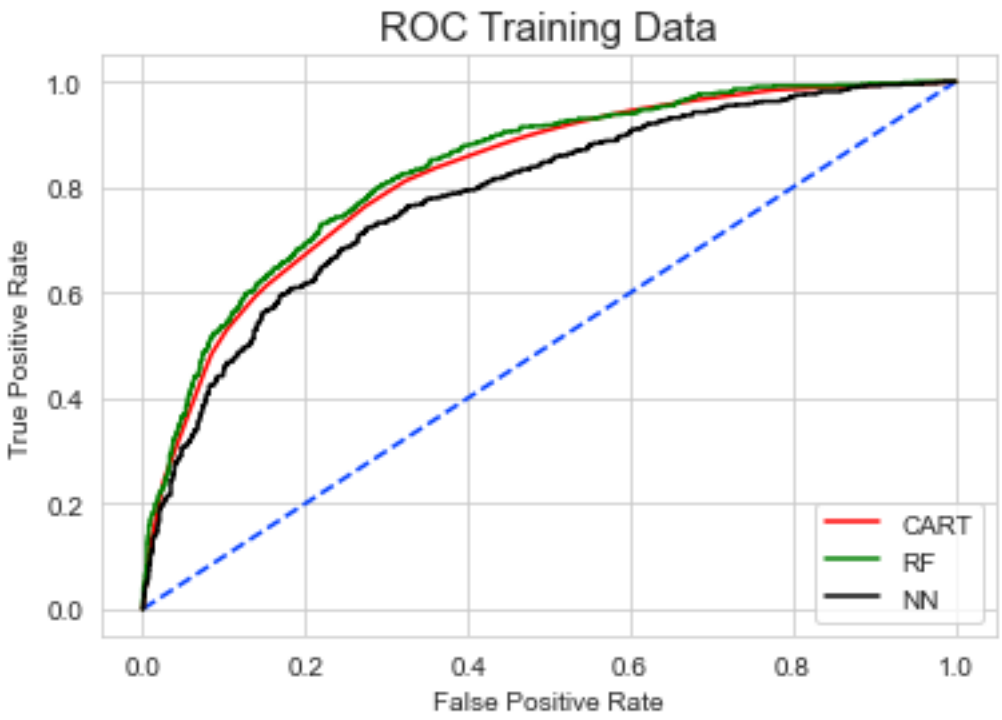

Figure 60:ROC Curve-All 3 Models-Train Data

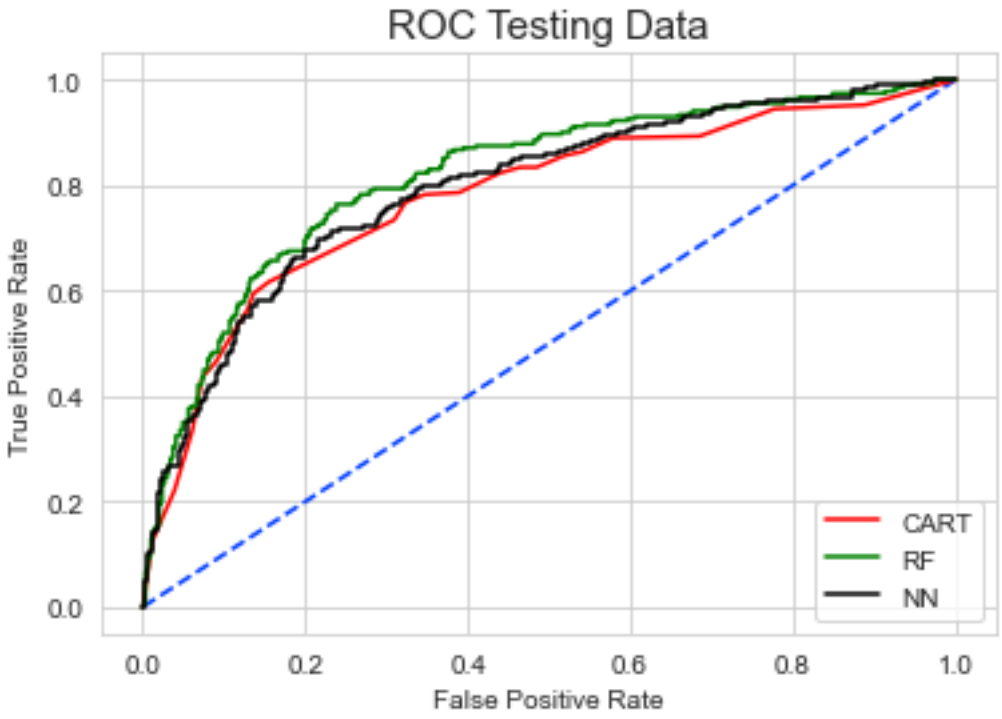

Figure 61:ROC Curve-All 3 Models-Test Data

Out of all the 3 models, Random Forest has slightly better performance than the CART and ANN model which can be inferred from the above performance metrics. It also has a better recall value than others. From the ROC Curve we

can see that the area under the green line(RF) is the largest of all 3. So Random Forest model is the best/optimized model.

2.5 Inference: Based on the whole Analysis, what are the business insights and recommendations

From running algorithms like CART, Random Forest and Artificial Neural Networks I can infer that Random Forest performed better than the other algorithms. Adding more features like insurance amount and income status can lead to better analysis. Insurance with higher sales has more claim frequency than lower sales.

Recommendations

- Gold Plan Insurance has less number of purchases. Hence it is to be marketed more.
- Since data suggest that more than 90% of insurance is done by Online mode so we can infer that offline experiences must be improved
- More sales happen through Travel Agency than Airlines and the claims are processed more at Airlines. To improve sales through Travel Agency proper advertising can be done and gifts can be given.
- JZI & CWT agency should increase sales as they have lower sales. To improve sales through JZI & CWT proper advertising can be done and gifts can be given.
- Asia as Destination has less number of sales but less claim frequency than its unclaimed frequency. This destination package can be promoted more.

----------X-------X-------X----------