Great Learning

DATA MINING

BUSINESS REPORT

Submitted by:

N. Aishwarya
PGP-DSBA

April 2022

Table of Contents

List of Figures

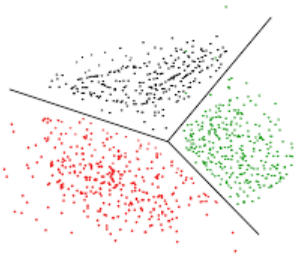
List of Tables

List of Formulas

Executive summary - Data Mining:

Data mining is a process to analyze massive volumes of data to discover business intelligence that helps to solve problems, mitigate risks, and seize new opportunities. It is based on mathematical algorithms and analytical skills to drive the desired results from the huge database collection.

Data Mining also helps professionals to develop competitive business strategies and manage operations effectively, including marketing, finance, customer support, HR, and other areas. It assists in risk management, fraud detection and cyber security planning, among other significant advantages.



Business 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. It is required to identify the segments based on credit card usage.

Solution Approach:

The purpose of the solutioning exercise is to explore the dataset using data mining techniques to arrive at the customer segmentation, thus enabling business strategies customized to them. Below is the data dictionary for clustering:

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)

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

Solution:

Firstly, import the necessary libraries required for the problem in the Jupiter Notebook file and run them. Read the " **bank_marketing_part1_Data-1.csv** " file for EDA.

Head of the data is obtained as below:

	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: Description of banking customers data

Output from shape command:

No. of rows: 210
No. of columns: 7

List of fields retrieval along with their data type:

```
<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: Information of banking customers data

Output for missing values:

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

Observations:

- Dataset consists 210 rows and 7 different attributes of credit card data.
- No missing records.
- All the variables are numeric type.

➢ Summary of the data, providing descriptive statistical variables:

Descriptive statistics help describe and understand the features of a specific data set by giving short summaries about the sample and measures of the data. The most recognized types of descriptive statistics are measures of center: the mean, median, and mode, which are used at almost all levels math and statistics.

	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: Summary Statistic of banking customers data

Summary statistics info:

- Based on summary descriptive, the data looks complete, without null values.
- We can also see there are no duplicates in the dataset.
- We can observe that the average spending of the customer per month is 14.8 with minimum spending of 10.6 and maximum spending of 21.2
- For most of the variables, mean/median are nearly equal and data is distributed evenly
- Standard Deviation is high for spending variable.
- There are considerable number of variables that are highly correlated.

Data Visualization:

Univariate analysis:

Helps us to understand the distribution of data in the dataset. With univariate analysis we can find patterns and we can summarize the data for.

1. Spending variable:

count	210
mean	14.85
std	2.90
min	10.59
25%	12.27

50%	14.35
75%	17.30
max	21.18

Table 4: Summary Statistic of Spending variable

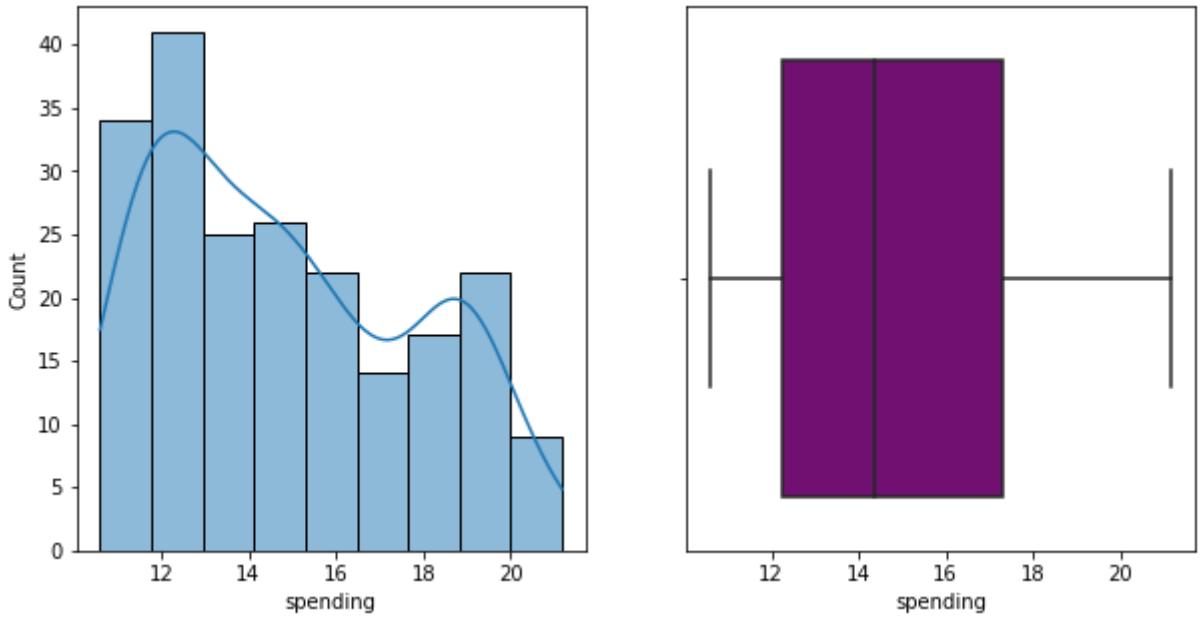


Figure 1: Univariate analysis of Spending variable

Observations:

- For Univariate Analysis of Spending, we are using histplot and boxplot to find information or patterns in the data.
- The Boxplot of Spending variable seems to have no outliers.
- The distribution of the data is right skewed.

2. Advance payments variable:

count	210
mean	14.56
std	1.30
min	12.41
25%	13.45
50%	14.32
75%	15.72
max	17.25

Table 5: Summary Statistic of Advance payments variable

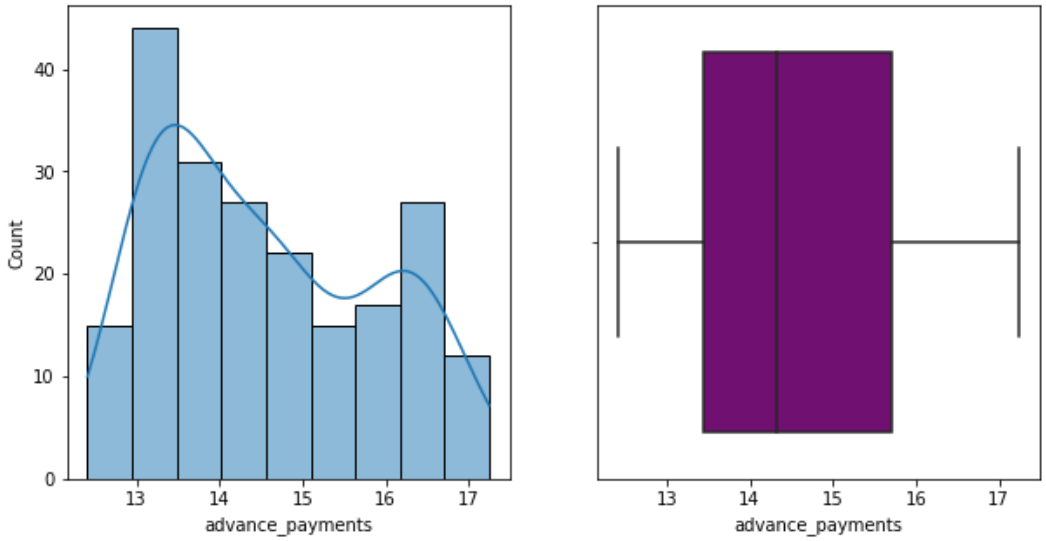


Figure 2: Univariate analysis of Advanced payments variable

Observations:

- The boxplot of advance payments variable has no outliers.
- The distribution of the data is positively skewed.

3. Probability_of_full_payment variable:

count	210
mean	0.870
std	0.023
min	0.808
25%	0.857
50%	0.873
75%	0.887
max	0.918

Table 6: Summary Statistic of Probability of full payment variable

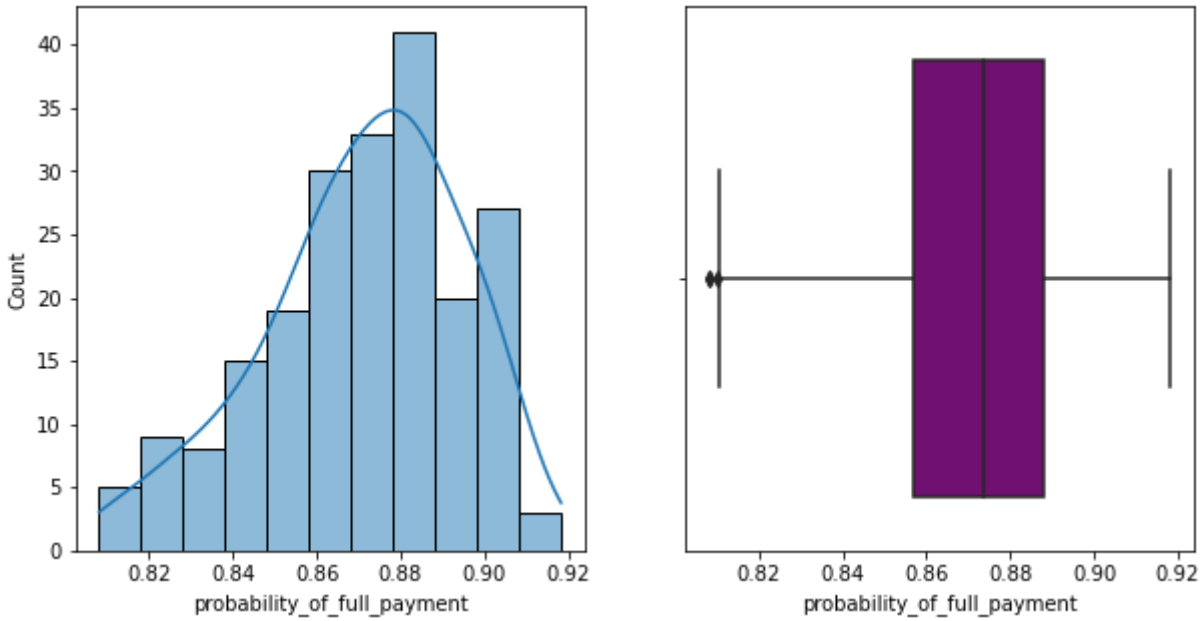


Figure 3: Univariate analysis of probability_of_full_payment variable

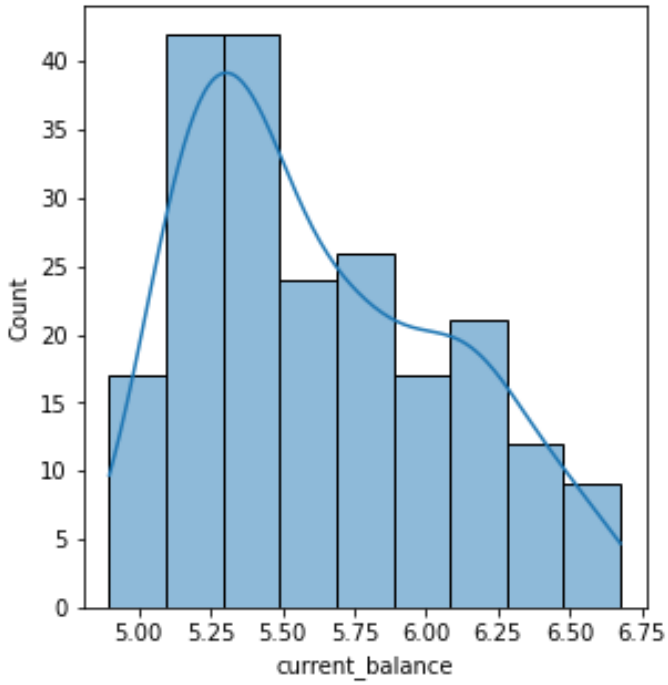
Observations:

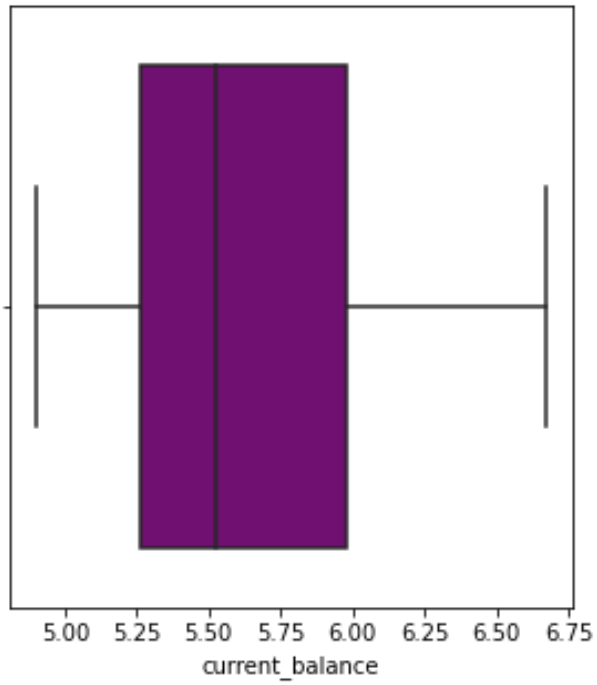
- The Boxplot of probability_of_full_payment variable has no outliers.
- The distribution of the data is left skewed.

4. Current_balance variable:

count	210
mean	5.628
std	0.443
min	4.899
25%	5.262
50%	5.523
75%	5.979
max	6.675

Table 7: Summary Statistic of Probability of Current balance variable





Observations:

- The boxplot of Current_balance variable has no outliers.
- The distribution of the data is right skewed.

5. Credit limit variable:

count	210
mean	3.259
std	0.378
min	2.630
25%	2.944
50%	3.237
75%	3.562
max	4.033

Table 8: Summary Statistic of Credit limit variable

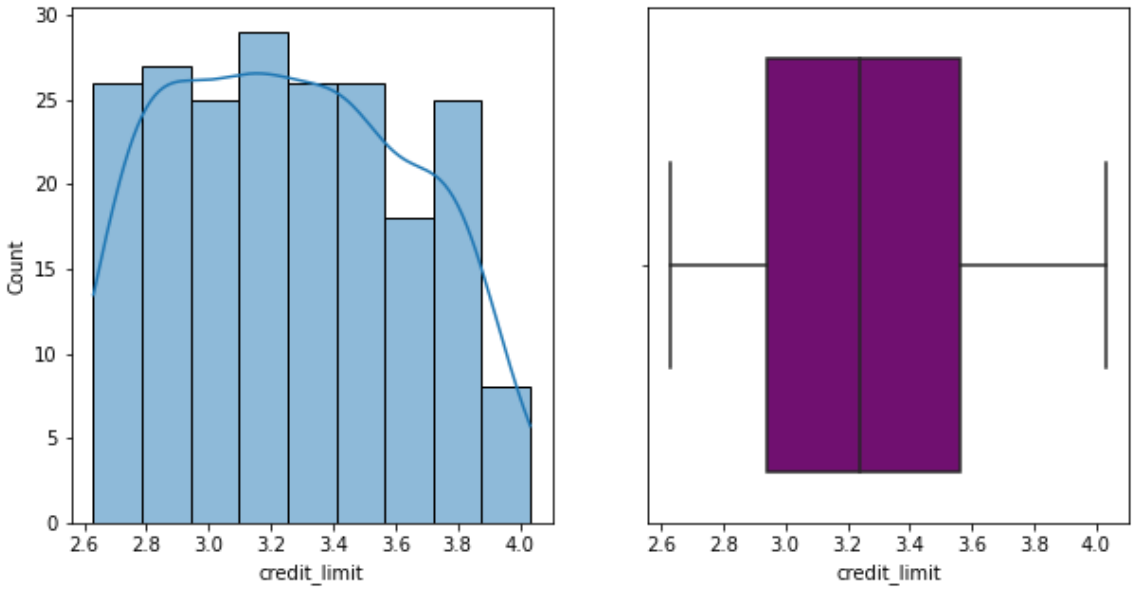


Figure 6: Univariate analysis of probability of Credit limit variable

Observations:

- The boxplot of credit limit variables has no outliers.
- The distribution of the data is spread normally.

6. min_payment_amt variable:

count	210.00
mean	3.70
std	1.50
min	0.77
25%	2.56
50%	3.60
75%	4.77
max	8.46

Table 9: Summary Statistic of min_payment_amt variable

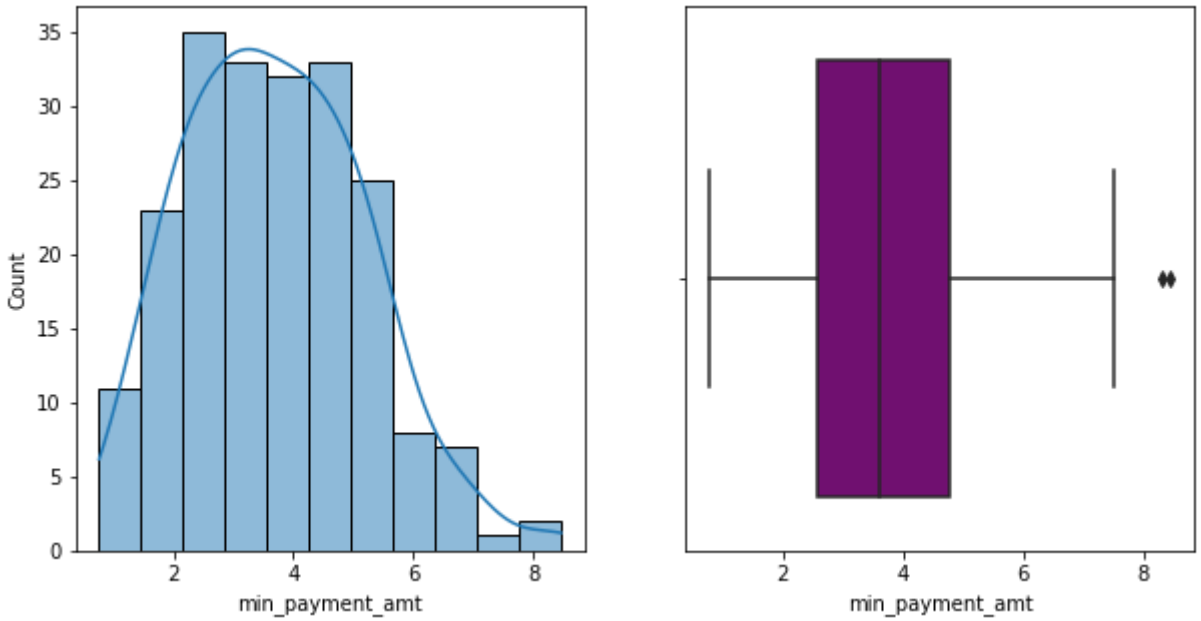


Figure 6: Univariate analysis of probability of min_payment_amt variable

Observations:

- From the above descriptive statistics and plots, we see that the min_payment_amt field is right skewed and has outliers.
- The distribution of the data is symmetric.

7. max_spent_in_single_shopping variable:

count	210.00
mean	5.41
std	0.49
min	4.52
25%	5.05
50%	5.22
75%	5.88
max	6.55

Table 10: Summary Statistic of max_spent_in_single_shopping variable

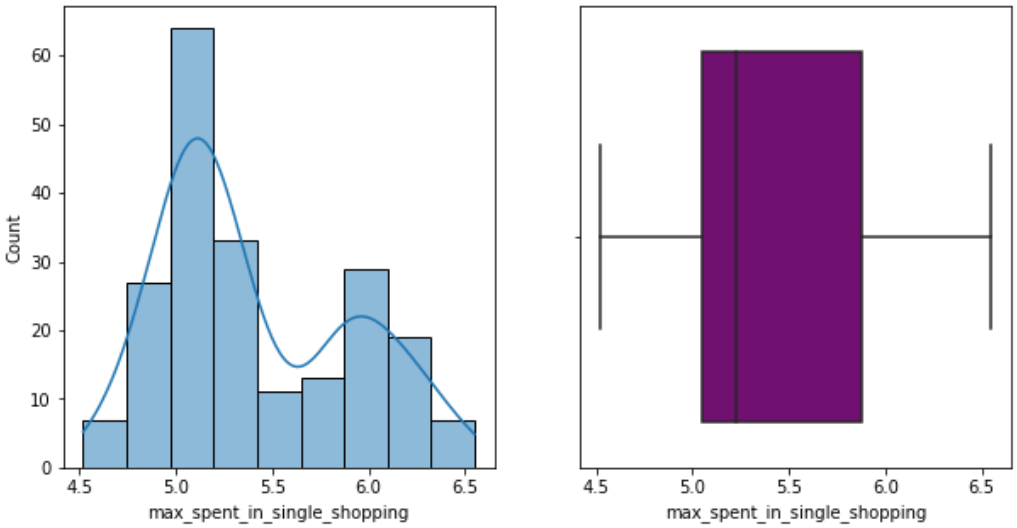


Figure 7: Univariate analysis of probability of max_spent_in_single_shopping variable

Observations:

- From the above descriptive statistics and plots, we see that the max_spent_in_single_shopping field is right skewed and has no outliers.

Histogram plot

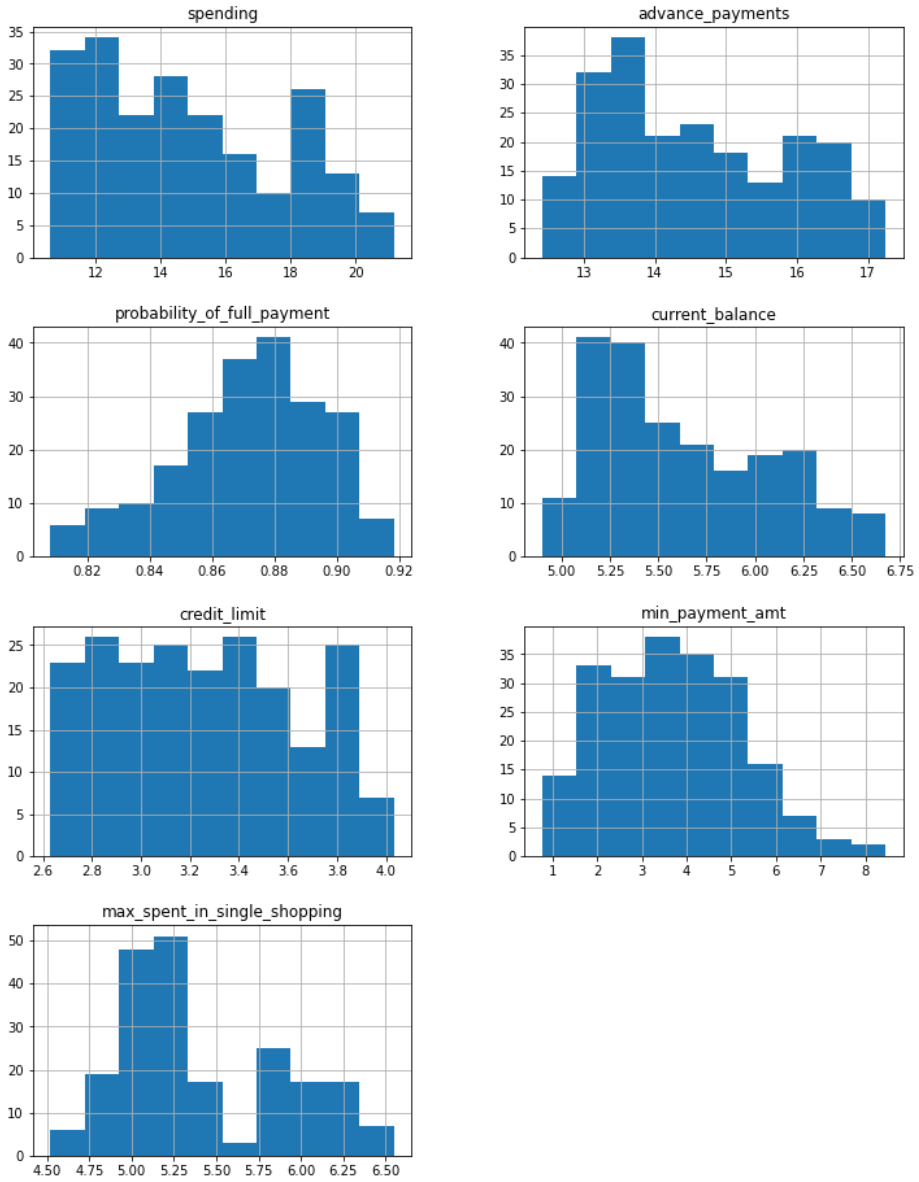


Figure 8: Histogram for independent attributes

Observations:

- Maximum number of customers use credit card for spending.
- Distribution is skewed to right tail for all the variable except probability_of_full_payment variable, which has left tail

Skewness values

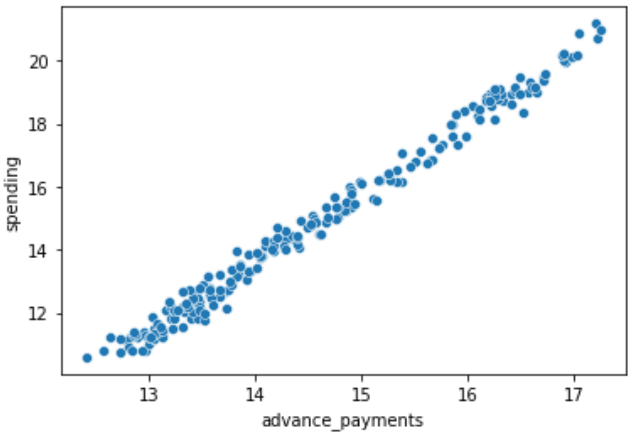
Max_spent_in_single_shopping	0.562
Current_balance	0.525
Min_payment_amt	0.402
Spending	0.400
Advance payments	0.387
Credit limit	0.134
Probability_of_full_payment	-0.538

Table 11: Skewness value between 7 independent variables

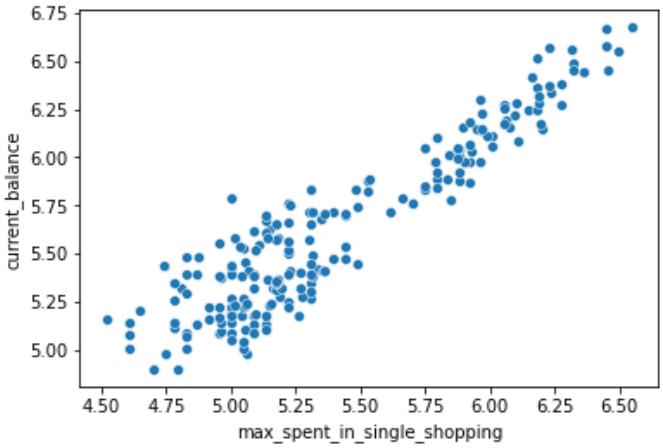
Bivariate Analysis:

Below charts provide bivariate analysis among key parameters:

A. Advanced payments vs. Spending



B. Max spent vs. current balance



C. Credit limit vs. Spending

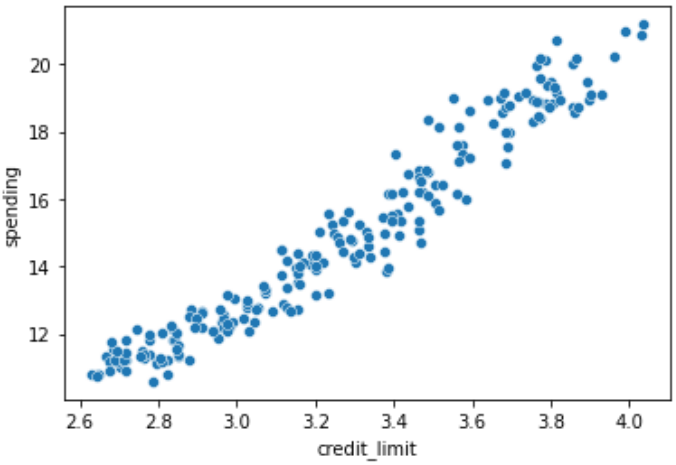


Figure 9: Bivariate analysis among various variables

Multivariate Analysis:

The Pair Plot helps us to understand the relationship between all the numerical values in the dataset. On comparing all the variables with each other we could understand the patterns. The pair plot function in seaborn makes it very easy to generate joint scatter plots for all the columns in the data.

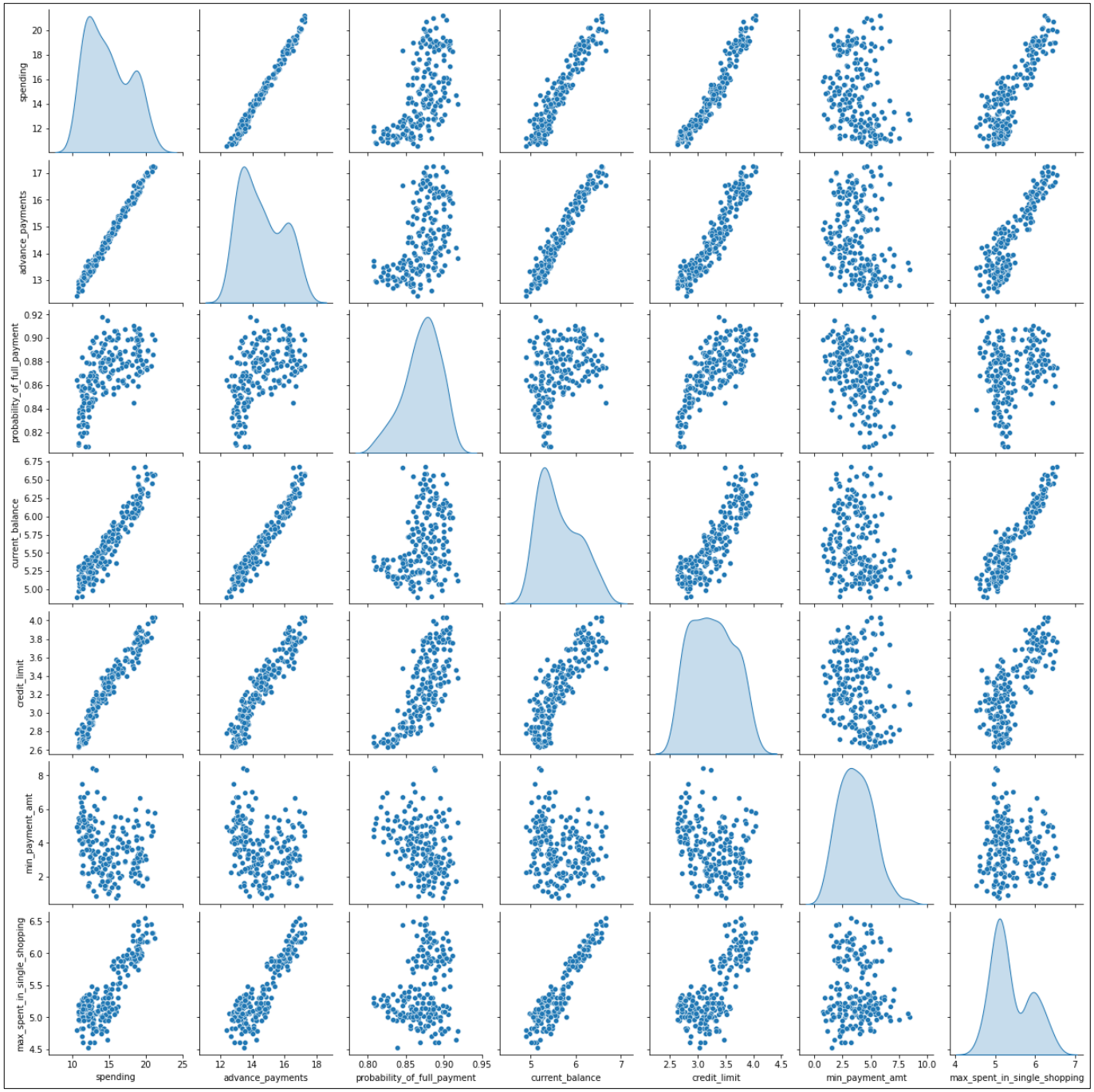


Figure 10: Pair Plot of independent variables

Clearly from the above pair plot, we can observe that all the attributes are not scaled and hence pre scaling will be required prior to performing clustering.

Correlation heatmap:

Correlation is a statistical measure that expresses the extent to which two variables are linearly related.

Below is the heatmap output from Python, for the 7 variables-

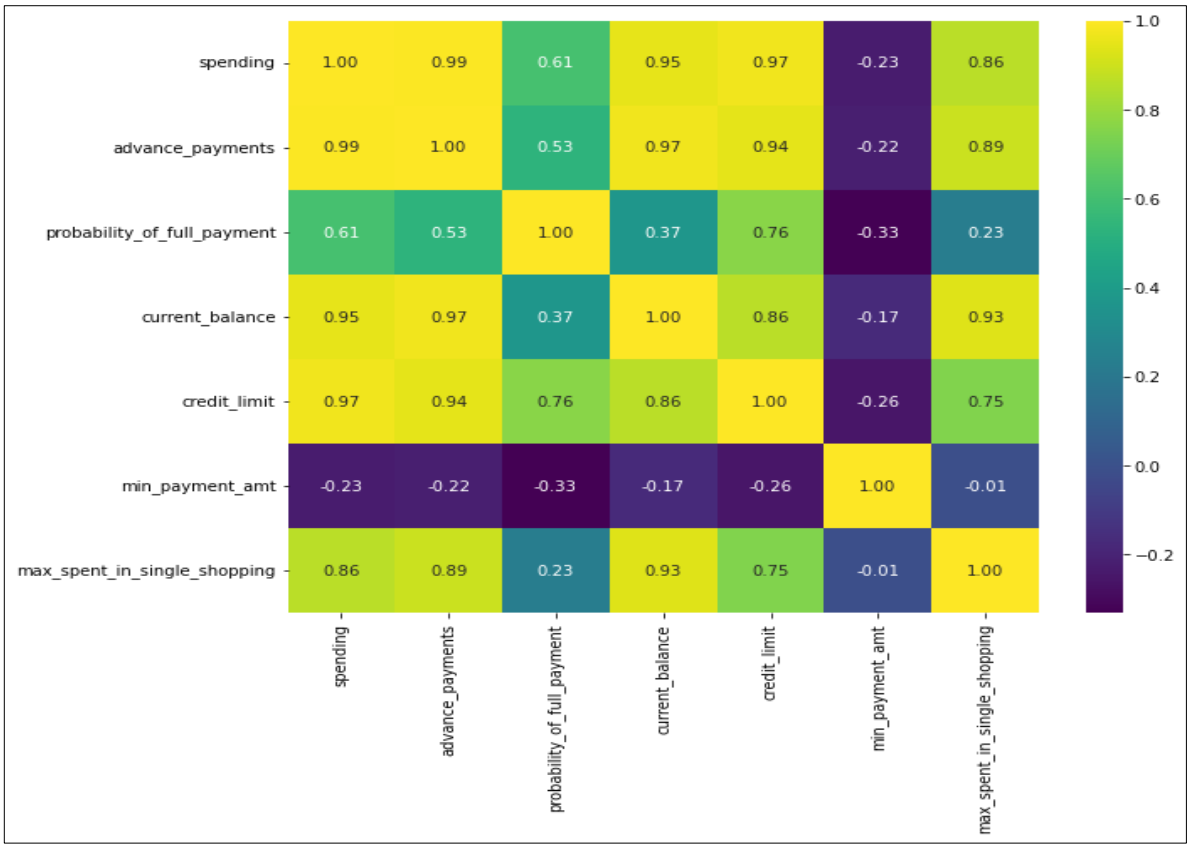


Figure 11: Correlation heatmap

Formula 3. Correlation

$$Correlation = \frac{Cov(x,y)}{\sigma_x * \sigma_y}$$

Where,

$Cov(x,y)$ = Covariance of x and y
σ_x = Standard deviation of x
σ_y = Standard deviation of y

Correlation matrix:

A correlation matrix displays the correlation coefficients for different variables. The matrix depicts the correlation between all the possible pairs of values in a table. The correlation matrix for the given set of variables is as follows:

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_sin
spending	1.00	0.99	0.61	0.95	0.97	-0.23	
advance_payments	0.99	1.00	0.53	0.97	0.94	-0.22	
probability_of_full_payment	0.61	0.53	1.00	0.37	0.76	-0.33	
current_balance	0.95	0.97	0.37	1.00	0.86	-0.17	
credit_limit	0.97	0.94	0.76	0.86	1.00	-0.26	
min_payment_amt	-0.23	-0.22	-0.33	-0.17	-0.26	1.00	
max_spent_in_single_shopping	0.86	0.89	0.23	0.93	0.75	-0.01	

Figure 12: Correlation matrix

Based on the correlation pair plot, strong positive correlation exists between:

- Spending & advance payments
- Advance payments & current balance
- Credit limit & spending
- Spending & current balance
- Credit limit & advance payments
- Max_spent_in_single_shopping & current balance

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

- ❖ Yes, scaling is required. Clustering algorithms such as K-means do need feature scaling. All the different variables need to be converted to one scale in order to perform meaningful analysis.
- ❖ Normalization is used to eliminate redundant data and ensures that good quality clusters are generated which can improve the efficiency of clustering algorithms. Hence it becomes an essential step before clustering as Euclidean distance is very sensitive to the changes in the differences.
- ❖ Feature scaling (also known as data normalization) is the method used to standardize the range of features of data. It helps to normalize the data within a particular range. Since, the range of values of data may vary widely, it becomes a necessary step in data preprocessing while using machine learning algorithms.

In the current dataset - spending, advance payments are in different values and this may get more weightage. Scaling will have all the values in the relative same range. Z score is used to standardize the data to relative same scale -3 to +3.

Below is the plot of the data prior and after scaling:

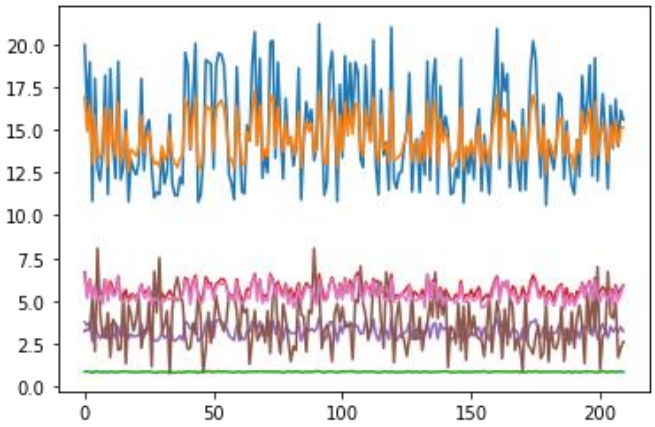


Figure 13: Data prior to scaling

Scaling the Variables:

Formula 1. Z Score

$$z = \frac{x-\mu}{\sigma}$$

Where, z = = standard score
x = Observed value
μ = mean of the sample
σ = standard deviation of the sample.

Snapshot of entire dataset after scaling is given below:

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
0	1.754355	1.811968	0.178230	2.367533	1.338579	-0.298806	2.328998
1	0.393582	0.253840	1.501773	-0.600744	0.858236	-0.242805	-0.538582
2	1.413300	1.428192	0.504874	1.401485	1.317348	-0.221471	1.509107
3	-1.384034	-1.227533	-2.591878	-0.793049	-1.639017	0.987884	-0.454961
4	1.082581	0.998364	1.196340	0.591544	1.155464	-1.088154	0.874813

Table 12: Scaling the Variables

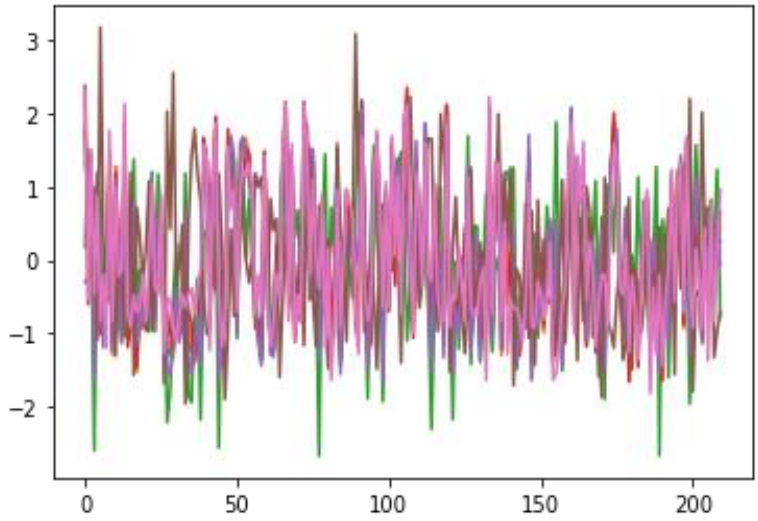


Figure 14: Data post scaling

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

- A dendrogram is a pictorial way to visualize hierarchical clustering. It is mainly used to show the outcome of hierarchical clustering tree like diagram that records the sequences of merges and splits. Dendrogram is created by importing the dendrogram and linkage module.

Below is the dendrogram created using average linkage method:

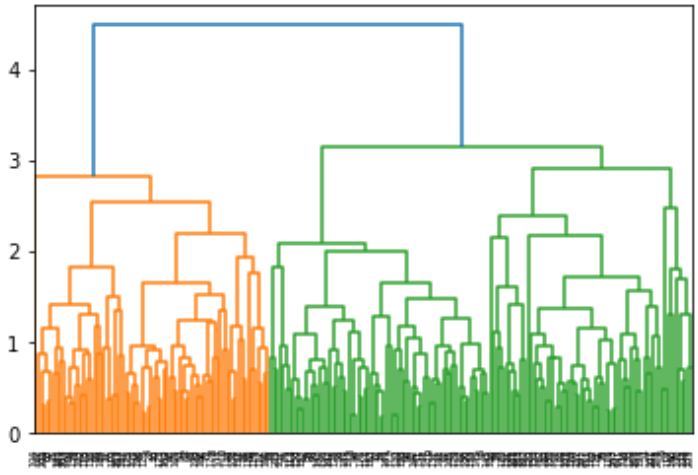
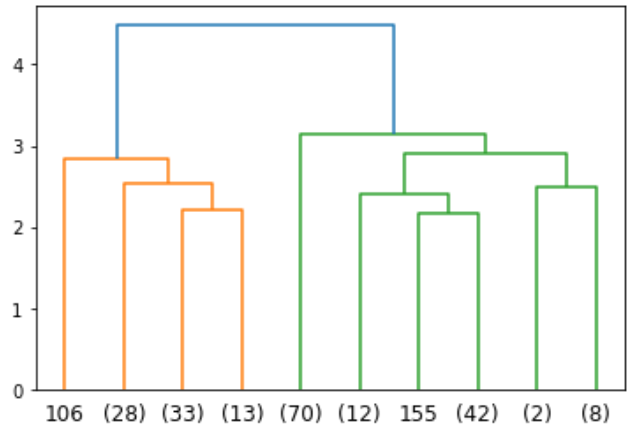


Figure 15: Dendrogram using hierarchal clustering

Cutting the Dendrogram with suitable clusters

For p =10



For p =17

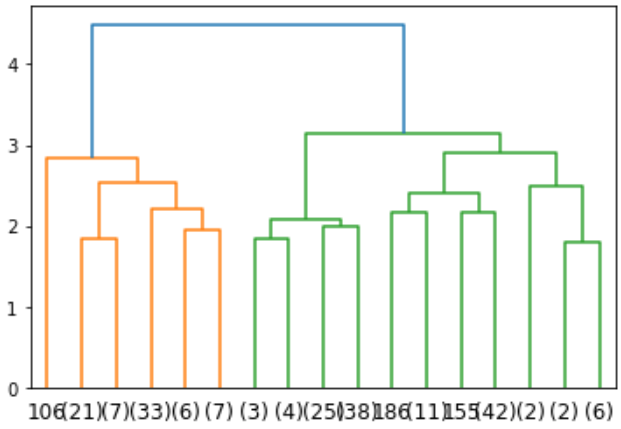


Figure 16: Cutting the Dendrogram with suitable clusters

Criterion is set as maxclust, then created 3 clusters, and stored the result in another object 'clusters'. The output is as below:

```
array([1, 3, 1, 2, 1, 3, 2, 2, 1, 2, 1, 1, 2, 1, 3, 3, 3, 2, 2, 2, 2, 2,
       1, 2, 3, 1, 3, 2, 2, 2, 2, 2, 2, 3, 2, 2, 2, 2, 2, 1, 1, 3, 1, 1,
       2, 2, 3, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 1,
       1, 3, 1, 2, 3, 2, 1, 1, 2, 1, 3, 2, 1, 3, 3, 3, 3, 1, 2, 1, 1, 1,
       1, 3, 3, 1, 3, 2, 2, 1, 1, 1, 2, 1, 3, 1, 3, 1, 3, 1, 1, 2, 3, 1,
       1, 3, 1, 2, 2, 1, 3, 3, 2, 1, 3, 2, 2, 2, 3, 3, 1, 2, 3, 3, 2, 3,
       3, 1, 2, 1, 1, 2, 1, 3, 3, 3, 2, 2, 2, 2, 1, 2, 3, 2, 3, 2, 3, 1,
       3, 3, 2, 2, 3, 1, 1, 2, 1, 1, 1, 2, 1, 3, 3, 2, 3, 2, 3, 1, 1, 1,
       3, 2, 3, 2, 3, 2, 3, 3, 1, 1, 3, 1, 3, 2, 3, 3, 2, 1, 3, 1, 1, 2,
       1, 2, 3, 3, 3, 2, 1, 3, 1, 3, 3, 1], dtype=int32)
```

Figure 17: Output "clusters"

Cluster frequency

1	75
2	70
3	65

Table 13: Cluster frequency

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

Table 14: Output of 3 clusters

Using Ward clustering

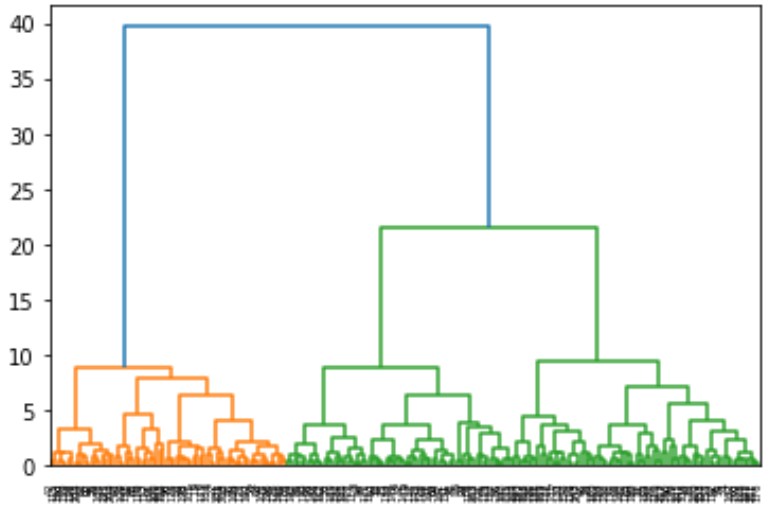


Figure 18: Dendrogram using ward clustering

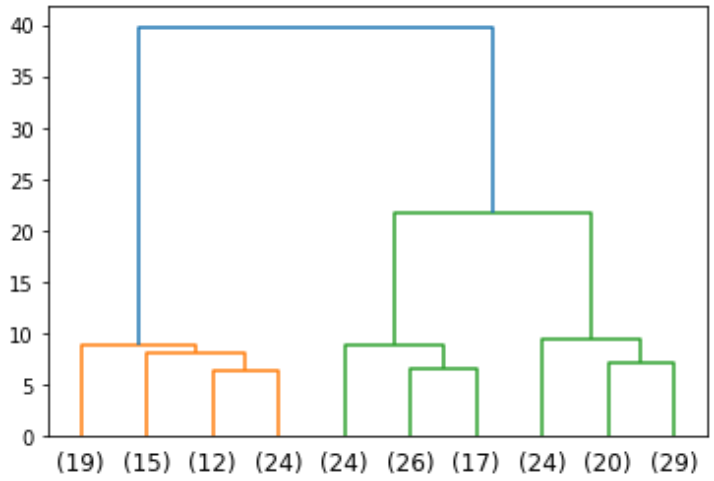


Figure 19: Dendrogram for p =10

```
array([1, 3, 1, 2, 1, 2, 2, 3, 1, 2, 1, 3, 2, 1, 3, 2, 3, 2, 3, 2, 2, 2,
       1, 2, 3, 1, 3, 2, 2, 2, 3, 2, 2, 3, 2, 2, 2, 2, 2, 1, 1, 3, 1, 1,
       2, 2, 3, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 2, 2, 1, 3, 2, 2, 3, 3, 1,
       1, 3, 1, 2, 3, 2, 1, 1, 2, 1, 3, 2, 1, 3, 3, 3, 3, 1, 2, 3, 3, 1,
       1, 2, 3, 1, 3, 2, 2, 1, 1, 1, 2, 1, 2, 1, 3, 1, 3, 1, 1, 2, 2, 1,
       3, 3, 1, 2, 2, 1, 3, 3, 2, 1, 3, 2, 2, 2, 3, 3, 1, 2, 3, 3, 2, 3,
       3, 1, 2, 1, 1, 2, 1, 3, 3, 3, 2, 2, 3, 2, 1, 2, 3, 2, 3, 2, 3, 3,
       3, 3, 3, 2, 3, 1, 1, 2, 1, 1, 1, 2, 1, 3, 3, 3, 3, 2, 3, 1, 1, 1,
       3, 3, 1, 2, 3, 3, 3, 3, 1, 1, 3, 3, 3, 2, 3, 3, 2, 1, 3, 1, 1, 2,
       1, 2, 3, 1, 3, 2, 1, 3, 1, 3, 1, 3], dtype=int32)
```

Figure 20: Output "clusters"

Cluster frequency

1	70
2	67
3	73

Table 15: Cluster frequency of Ward clustering

Inference:

- Both the methods report minor variation in the mean
- We form cluster grouping based on the dendrogram, 3 or 4 looks good. Based on further analysis, and based on the dataset, 3 group cluster solution is ideal.
- And three group cluster solution gives a pattern based on high/medium/low spending with max_spent_in_single_shopping (high value item) and probability_of_full_payment (payment made).

Analysis 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.

The K-means algorithm's goal is to keep the size of each cluster as small as possible, the small WSS indicates that every data point is close to its nearest centroids, or say the model has returned good results.

Cluster Output for all the observations:

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

Figure 21: Output "clusters"

Calculation of WSS for other values of K - Elbow Method:

Values of K	Wss (The sum distance within the centroids)
K=1	1470.0000
K=2	659.1718
K=3	430.6590
K=4	371.3509
K=5	326.7053
K=6	288.6618
K=7	262.0122
K=8	239.3096
K=9	223.4699
K=10	207.5570

Table 16: WSS for other values of K - Elbow Method

Inference:

- WSS reduces as K keeps increasing

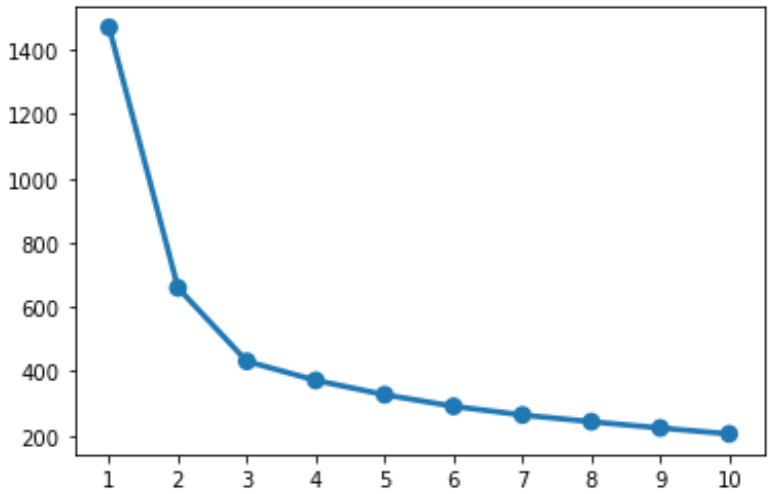


Figure 22: Point plot

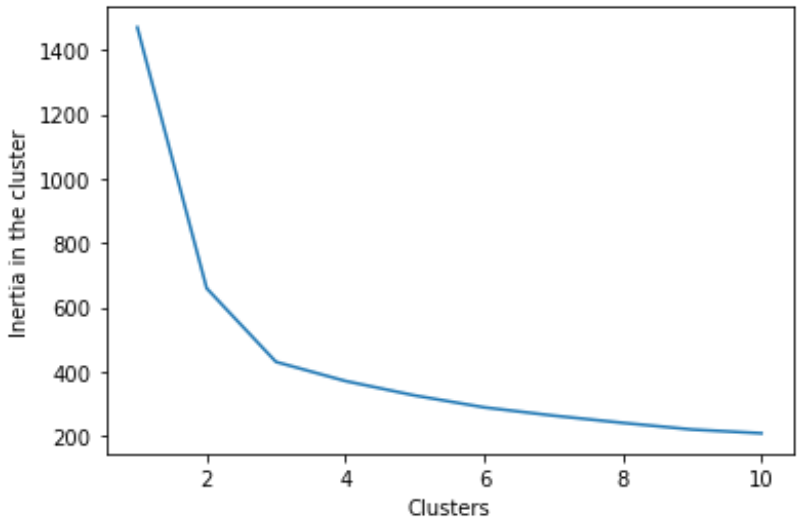


Figure 23: Plot depicting number of clusters vs. Inertia in the cluster

Formula 2:

$$\textbf{Silhouette Width} = \frac{b-a}{max(a,b)}$$

Where,

b = distance between observation and the neighbor cluster centroid (c2)
a = distance between observation and its own cluster centroid(c1)

Cluster evaluation for 3 clusters: the silhouette score:

- Silhouette score for 3 cluster is 0.40

Cluster evaluation for 4 clusters: the silhouette score:

- Silhouette score for 3 cluster is 0.328

Inference:

- Silhouette score is better for 3 clusters than for 4 clusters. hence, final clusters will be 3.

Appending Clusters to the original dataset

After appending the clusters to the original dataset, we get the following output:

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_kmeans4
0	19.94	16.92	0.8752	6.675	3.763	3.252	6.550	1
1	15.99	14.89	0.9064	5.363	3.582	3.336	5.144	0
2	18.95	16.42	0.8829	6.248	3.755	3.368	6.148	1
3	10.83	12.96	0.8099	5.278	2.641	5.182	5.185	3
4	17.99	15.86	0.8992	5.890	3.694	2.068	5.837	1

Cluster Profiling

Output of Cluster Profiling for 3 cluster is given below:

Clus_kmeans3	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
0	13.954179	14.095970	0.881048	5.425612	3.207896	2.593242	5.024149
1	19.120000	16.459184	0.886686	6.267265	3.768612	3.472980	6.125878
2	16.317333	15.288000	0.876877	5.864800	3.444433	3.868567	5.686533
3	11.822656	13.248125	0.845712	5.241187	2.834109	4.954094	5.129891

Plotting the Silhouette Coefficient scores

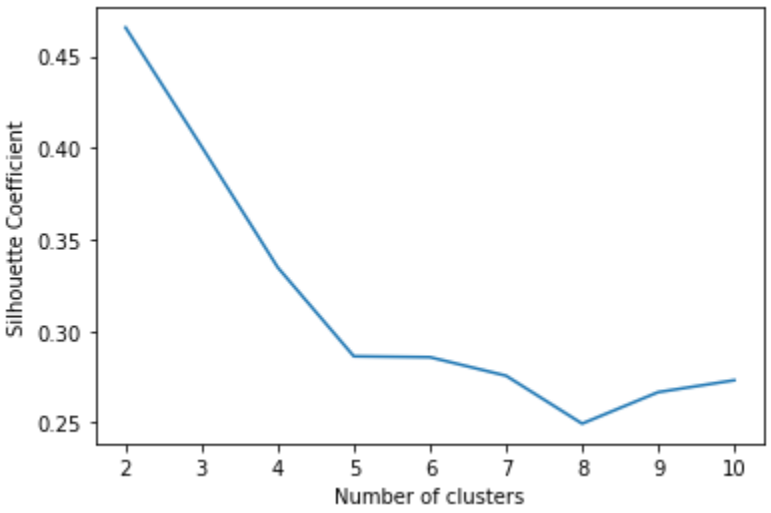


Figure 24: Silhouette Coefficient Plot

3 Cluster Solution:

Fitting the K means:

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

Figure 25: K means output for 3 Cluster

Proportion of labels classified:

0	71
1	67
2	72

Fitting K-Means to the dataset:

cluster	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_km
1	14.4	14.3	0.9	5.5	3.3	2.7	5.1	
2	11.9	13.2	0.8	5.2	2.8	4.7	5.1	
3	18.5	16.2	0.9	6.2	3.7	3.6	6.0	

Cluster Percentage:

Cluster numbering	Cluster_Size	Cluster_Percentage
1	71	33.81
2	72	34.29
3	67	31.90

Table 17: K Means Cluster Percentage for 3 cluster

Transposing the cluster:

Cluster	1	2	3
Spending	14.4	11.9	18.5
Advance_payments	14.3	13.2	16.2
probability_of_full_payment	0.9	0.8	0.9
current_balance	5.5	5.2	6.2
credit_limit	3.3	2.8	3.7
min_payment_amt	2.7	4.7	3.6
max_spent_in_single_shopping	5.1	5.1	6.0
Clus_kmeans3	0.3	2.7	1.3

Table 18: K means transposing the cluster for 3 cluster

We are going with 3 clusters via k means, but based on the analysis of 4 and 5 k means cluster, we observe that based on current dataset given, 3 cluster solution makes sense based on the spending pattern (High, Medium, Low)

4-Cluster Solution:

Fitting the K-means:

```
array([0, 3, 0, 1, 0, 1, 1, 3, 0, 1, 0, 2, 1, 0, 3, 1, 3, 1, 3, 1, 1, 1,
       0, 1, 3, 2, 3, 1, 1, 1, 3, 1, 1, 3, 1, 1, 1, 1, 1, 0, 0, 3, 2, 0,
       1, 1, 3, 0, 0, 0, 1, 0, 0, 0, 0, 2, 1, 1, 1, 0, 3, 1, 1, 2, 3, 0,
       0, 3, 0, 3, 3, 1, 0, 0, 1, 0, 3, 1, 2, 3, 3, 3, 3, 0, 1, 2, 2, 2,
       2, 1, 3, 0, 3, 1, 3, 0, 0, 2, 1, 0, 3, 0, 2, 0, 3, 0, 0, 1, 3, 0,
       2, 3, 0, 1, 1, 2, 3, 2, 1, 0, 3, 1, 1, 1, 3, 3, 0, 1, 3, 3, 1, 3,
       3, 0, 1, 0, 0, 1, 2, 3, 2, 3, 1, 1, 3, 1, 0, 1, 3, 1, 3, 1, 3, 2,
       3, 3, 3, 1, 3, 0, 0, 1, 0, 2, 0, 1, 2, 3, 3, 1, 3, 1, 3, 0, 0, 0,
       3, 3, 2, 1, 3, 3, 3, 3, 2, 2, 3, 2, 3, 1, 3, 3, 1, 0, 3, 2, 0, 1,
       0, 1, 3, 2, 3, 1, 2, 3, 2, 3, 2, 2])
```

Proportion of labels classified

0	51
1	64
2	30
3	65

K-Means Clustering & Cluster Information (for 4-cluster):

cluster	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_km
1	19.1	16.4	0.9	6.3	3.8	3.5	6.1	
2	11.9	13.3	0.8	5.2	2.8	4.9	5.1	
3	16.1	15.2	0.9	5.8	3.4	4.0	5.6	
4	14.1	14.1	0.9	5.4	3.2	2.4	5.0	

Cluster Percentage:

Cluster numbering	Cluster_Size	Cluster_Percentage
1	51	24.29
2	68	32.38
3	30	14.29
4	61	29.05

Table 19: K Means Cluster Percentage for 4 cluster

Transposing the cluster:

Cluster	1	2	3	4
Spending	19.1	11.9	16.1	14.1
Advance_payments	16.4	13.3	15.2	14.1
probability_of_full_payment	0.9	0.8	0.9	0.9
current_balance	6.3	5.2	5.8	5.4
credit_limit	3.8	2.8	3.4	3.2
min_payment_amt	3.5	4.9	4.0	2.4
max_spent_in_single_shopping	6.1	5.1	5.6	5.0
Clus_kmeans3	1.0	2.8	1.9	0.0

Table 20: K means transposing the cluster for 4 cluster

5-cluster solution:

Fitting the K-means:

```
array([1, 0, 1, 4, 1, 2, 2, 0, 1, 2, 1, 2, 2, 1, 2, 2, 0, 2, 2, 2, 2, 4,
       1, 2, 0, 3, 0, 4, 4, 4, 0, 2, 2, 0, 4, 4, 4, 2, 4, 1, 1, 0, 3, 1,
       4, 2, 0, 1, 1, 1, 2, 1, 1, 1, 1, 3, 2, 4, 4, 1, 0, 2, 4, 3, 0, 1,
       1, 0, 1, 2, 0, 2, 1, 1, 2, 1, 0, 4, 3, 0, 0, 0, 0, 1, 4, 3, 3, 3,
       3, 2, 0, 1, 0, 4, 2, 1, 1, 3, 4, 3, 2, 1, 0, 1, 0, 1, 1, 4, 2, 1,
       3, 0, 1, 4, 4, 3, 0, 2, 4, 1, 2, 4, 2, 2, 0, 0, 1, 4, 0, 0, 4, 0,
       2, 1, 4, 3, 1, 2, 3, 0, 3, 0, 4, 2, 2, 2, 1, 4, 0, 4, 0, 2, 0, 3,
       2, 0, 2, 4, 0, 3, 1, 2, 1, 3, 1, 2, 3, 0, 0, 2, 0, 4, 0, 1, 1, 1,
       0, 2, 3, 2, 0, 2, 0, 0, 3, 3, 2, 3, 0, 4, 2, 0, 4, 1, 0, 3, 1, 2,
       1, 4, 0, 3, 0, 4, 3, 0, 3, 0, 0, 3])
```

K-Means Clustering & Cluster Information (for 5-cluster):

cluster	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_km
1	11.6	13.2	0.8	5.3	2.8	4.7	5.2	
2	19.1	16.4	0.9	6.3	3.8	3.5	6.1	
3	14.2	14.2	0.9	5.5	3.2	2.3	5.0	
4	12.3	13.3	0.9	5.2	3.0	5.0	5.0	
5	16.2	15.2	0.9	5.9	3.4	3.9	5.7	

Cluster Percentage:

Cluster numbering	Cluster_Size	Cluster_Percentage
1	39	18.57
2	50	23.81
3	55	26.19
4	36	17.14
5	30	14.29

Table 21: K Means Cluster Percentage for 5 cluster

Transposing the cluster:

Cluster	1	2	3	4	5
Spending	11.6	19.1	14.2	12.3	16.2
Advance_payments	13.2	16.4	14.2	13.3	15.2
probability_of_full_payment	0.8	0.9	0.9	0.9	0.9
current_balance	5.3	6.3	5.5	5.2	5.9
credit_limit	2.8	3.8	3.2	3.0	3.4
min_payment_amt	4.7	3.5	2.3	5.0	3.9

max_spent_in_single_shopping	5.2	6.1	5.0	5.0	5.7
Clus_kmeans3	3.0	1.0	0.0	2.1	1.9

Table 22: K means transposing the cluster for 5 cluster

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

3 group cluster via K-means:

Cluster	1	2	3
Spending	14.4	11.9	18.5
Advance payments	14.3	13.2	16.2
probability_of_full_payment	0.9	0.8	0.9
Current_balance	5.5	5.2	6.2
Credit limit	3.3	2.8	3.7
min_payment_amt	2.7	4.7	3.6
max_spent_in_single_shopping	5.1	5.1	6.0
Clus_kmeans3	0.3	2.7	1.3

Table 23: 3 group cluster via K-means

3 group cluster via hierarchical clustering

Cluster-3	1	2	3
Spending	18.4	11.9	14.2
Advance payments	16.1	13.3	14.2
probability_of_full_payment	0.9	0.8	0.9
Current_balance	6.2	5.2	5.5
credit limit	3.7	2.8	3.2
min_payment_amt	3.6	4.9	2.6
max_spent_in_single_shopping	6.0	5.1	5.1
Frequency	70.0	67.0	73.0

Table 24: 3 group cluster via hierarchical clustering

Cluster Group Profiles

Basis the analysis, below are the cluster groups identified:

Cluster 1: High Spending customers:

Cluster 2: Low Spending

Cluster 3: Medium Spending

Hence it is recommended to tailor promotional strategies customized for each cluster, as below:

Cluster 1: High Spending Group

- Creating customer delight by awarding “reward points” to increase their purchases.
- Maximum “max_spent_in_single_shopping” is high for this group, so can be offered discount/offer on next transactions upon full payment
- Reward this cluster with higher credit limit and thus increase additional spending
- Give loan against the credit card, as they are customers with good repayment record.
- Tie up with luxury brands, which will drive more one_time_maximum spending

Cluster 2: Low Spending Group

- Customers should be given reminders for payments. Offers can be provided on early payments to improve their payment rate.
- Increase there is spending habits by tying up with grocery stores, utilities (electricity, phone, gas, others)

Cluster 3: Medium Spending Group

- They are potential target customers who are paying bills and doing purchases and maintaining comparatively good credit score. Hence, we can increase credit limit or can lower down interest rate.
- Promote premium cards/loyalty cars to increase transactions.
- Increase spending habits by trying with premium ecommerce sites, travel portal, travel airlines/hotel, as this will encourage them to spend more.

Business 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.

Solution Approach:

The purpose of the solutioning exercise is to explore the dataset using data mining techniques to predict the claim status. Below is the data dictionary for the problem:

Attribute Information:

- Target: Claim Status (Claimed)
- Code of tour firm (Agency Code)
- Type of tour insurance firms (Type)
- Distribution channel of tour insurance agencies (Channel)
- Name of the tour insurance products (Product)
- Duration of the tour (Duration)
- Destination of the tour (Destination)
- Amount of sales of tour insurance policies (Sales)
- The commission received for tour insurance firm (Commission)
- Age of insured (Age)

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

Firstly, import the necessary libraries required for the problem in the Jupiter Notebook file and run them. Read the “insurance_part2_data.csv ” file for EDA.

Head of the data is obtained as below:

	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 25: Description of banking customers data

Output from shape command:

Number of rows:	3000
Number of Columns:	10

List of fields retrieval along with their data type:

```
<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 26: Information of banking customers data

Output for missing values:

Age	0
Agency Code	0
Type	0
Claimed	0
Commission	0
Channel	0
Duration	0
Sales	0
Product Name	0
Destination	0

➢ Summary of the data, providing descriptive statistical variables:

Descriptive statistics help describe and understand the features of a specific data set by giving short summaries about the sample and measures of the data. The most recognized types of descriptive statistics are measures of center: the mean, median, and mode, which are used at almost all levels math and statistics.

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Age	3000.0	NaN	NaN	NaN	38.091	10.463518	8.0	32.0	36.0	42.0	84.0
Agency_Code	3000	4	EPX	1365	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Type	3000	2	Travel Agency	1837	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Claimed	3000	2	No	2076	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Commision	3000.0	NaN	NaN	NaN	14.529203	25.481455	0.0	0.0	4.63	17.235	210.21
Channel	3000	2	Online	2954	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Duration	3000.0	NaN	NaN	NaN	70.001333	134.053313	-1.0	11.0	26.5	63.0	4580.0
Sales	3000.0	NaN	NaN	NaN	60.249913	70.733954	0.0	20.0	33.0	69.0	539.0
Product Name	3000	5	Customised Plan	1136	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Destination	3000	3	ASIA	2465	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Table 27: Summary Statistic of banking customers data

Inferences:

- The dataset consists of 3000 records with 9 independent variables and one target variable “claimed”.
- Age, Commission, Duration, Sales are numeric variables.
- All other variables are of object datatype.
- No missing values in the entire dataset.
- Duration has negative value, which is not possible.
- Commission & Sales- mean and median varies significantly.
- Categorial code variable maximum unique count is 5.
- We can also see there are no duplicates in the dataset.
- We can observe that the average spending of the customer per month is 14.8 with minimum spending of 10.6 and maximum spending of 21.2
- For most of the variables, mean/median are nearly equal and data is distributed evenly
- Number of duplicate rows is 139

Though it shows there are 139 records, but it can be of different customers, there is no customer ID or any unique identifier, hence not dropping them off.

Getting unique counts of all Nominal Variables:

AGENCY_CODE: 4
JZI 239
CWT 472
C2B 924
EPX 1365
Name: Agency_Code, dtype: int64

TYPE: 2
Airlines 1163
Travel Agency 1837
Name: Type, dtype: int64

CLAIMED: 2
Yes 924
No 2076
Name: Claimed, dtype: int64

CHANNEL: 2
Offline 46
Online 2954
Name: Channel, dtype: int64

PRODUCT NAME: 5
Gold Plan 109
Silver Plan 427
Bronze Plan 650
Cancellation Plan 678
Customised Plan 1136
Name: Product Name, dtype: int64

DESTINATION: 3
EUROPE 215
Americas 320
ASIA 2465
Name: Destination, dtype: int64

Univariate Analysis:

Age variable:

count	3000.00
mean	38.09
std	10.46
min	8.00
25%	32.00
50%	36.00
75%	42.00
max	84.00

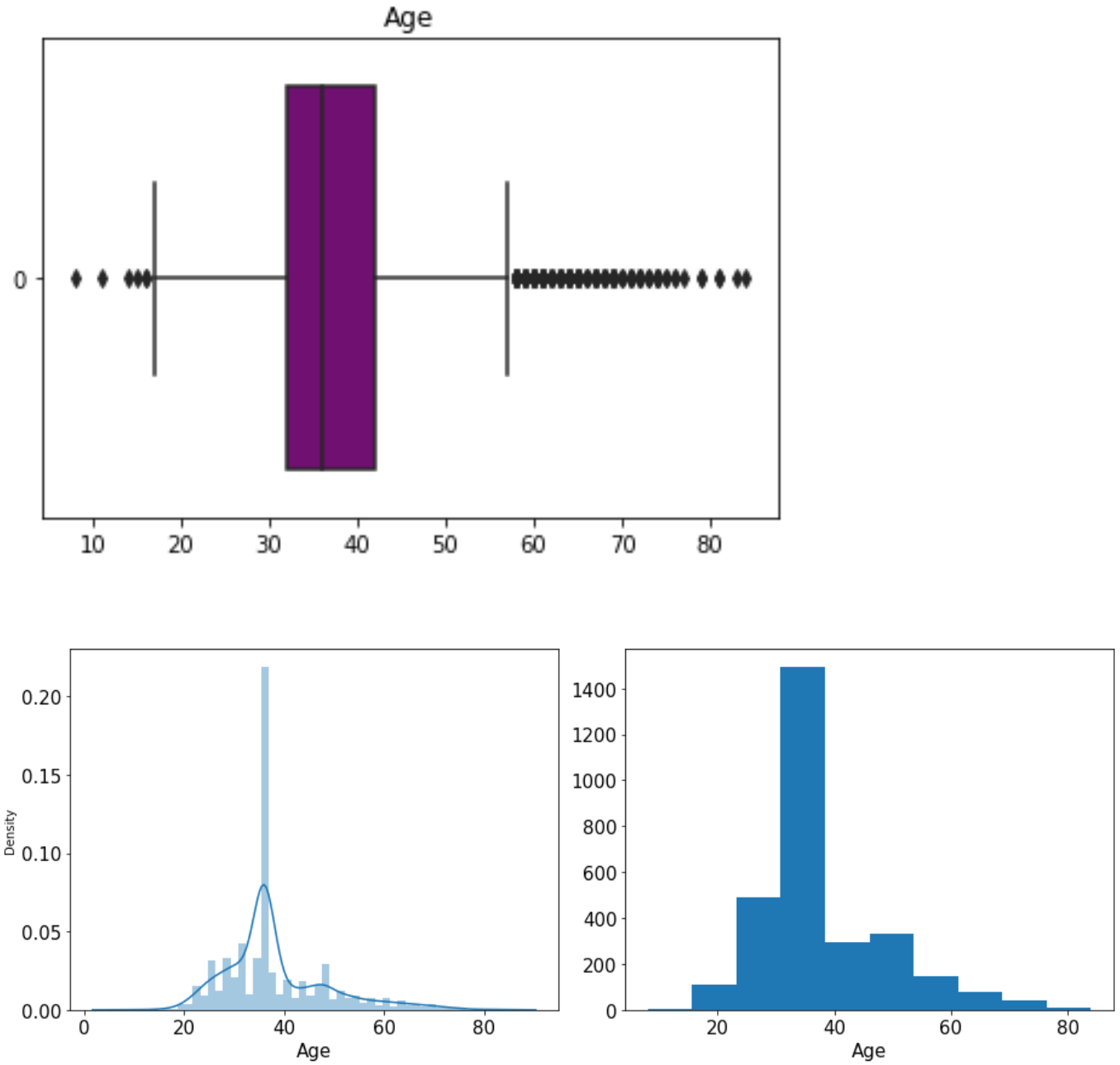


Figure 26: Univariate analysis of Age variable

Commission variable:

count	3000.00
mean	14.53
std	25.48
min	0.00
25%	0.00
50%	4.63
75%	17.24
max	210.21

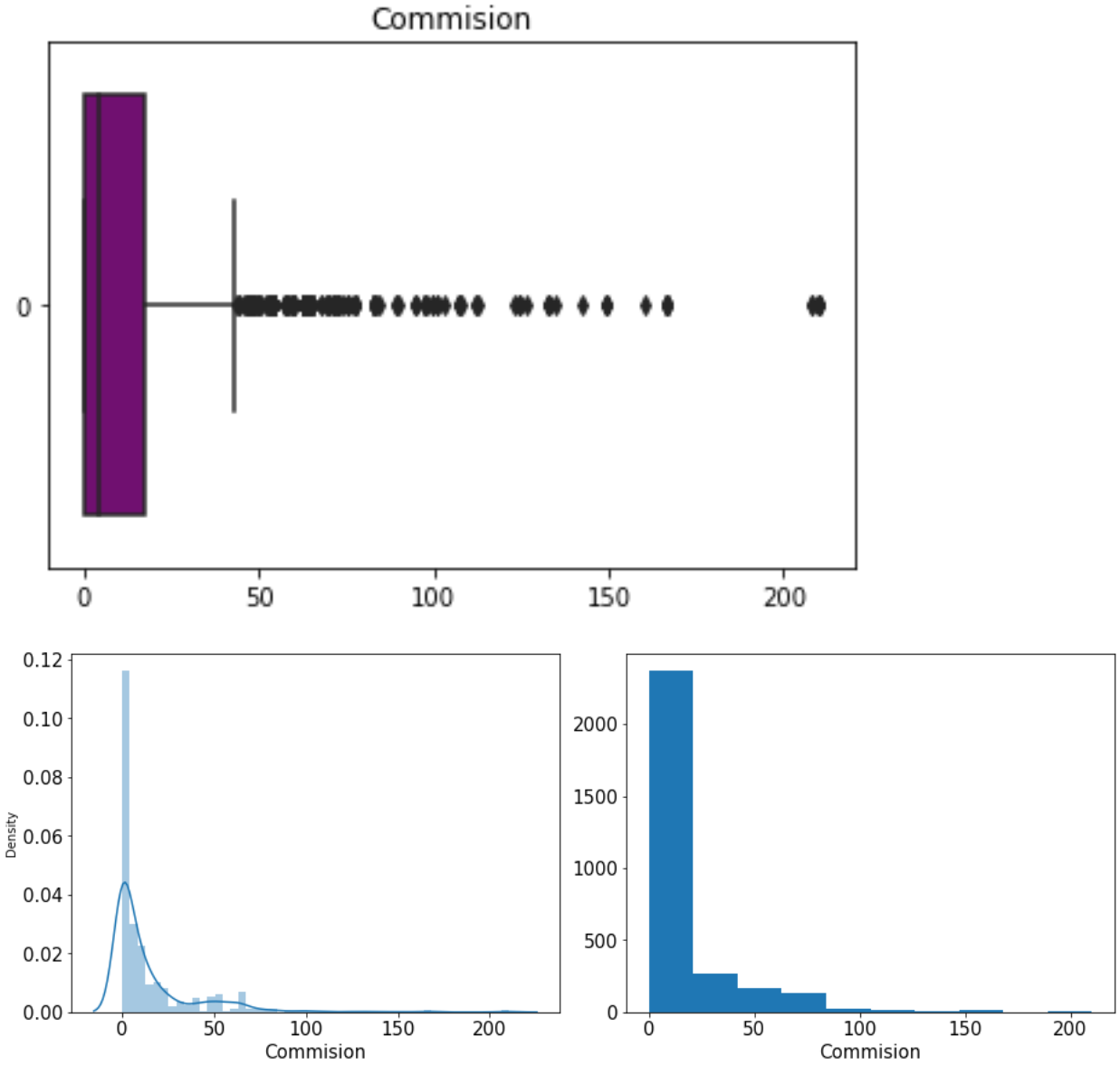


Figure 27: Univariate analysis of Commission variable

Duration variable:

count	3000.00
mean	14.53
std	25.48
min	0.00
25%	0.00
50%	4.63
75%	17.24
max	210.21

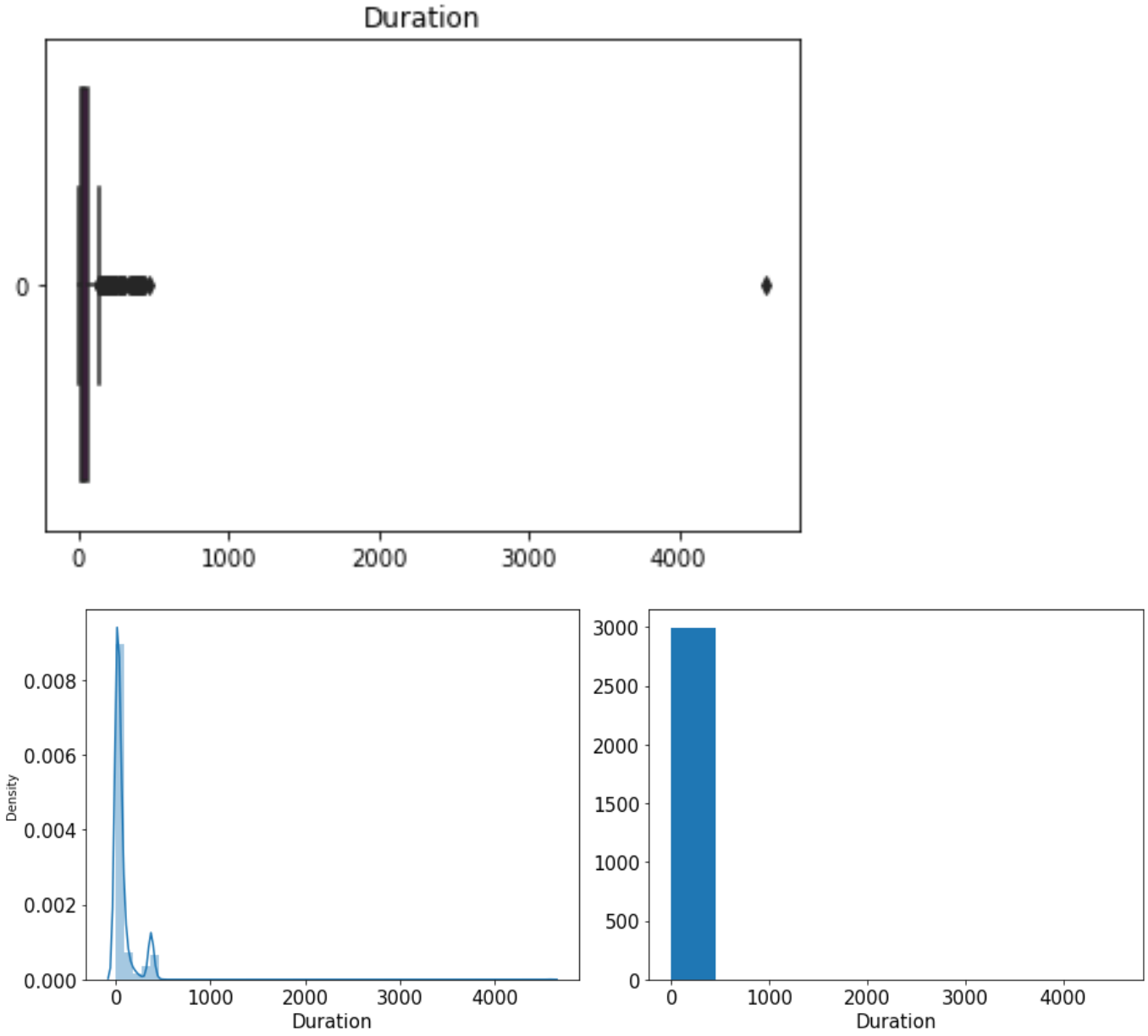


Figure 28: Univariate analysis of Duration variable

Sales variable:

count	3000.00
mean	60.25
std	70.73
min	0.00
25%	20.00
50%	33.00
75%	69.00

max	539.00

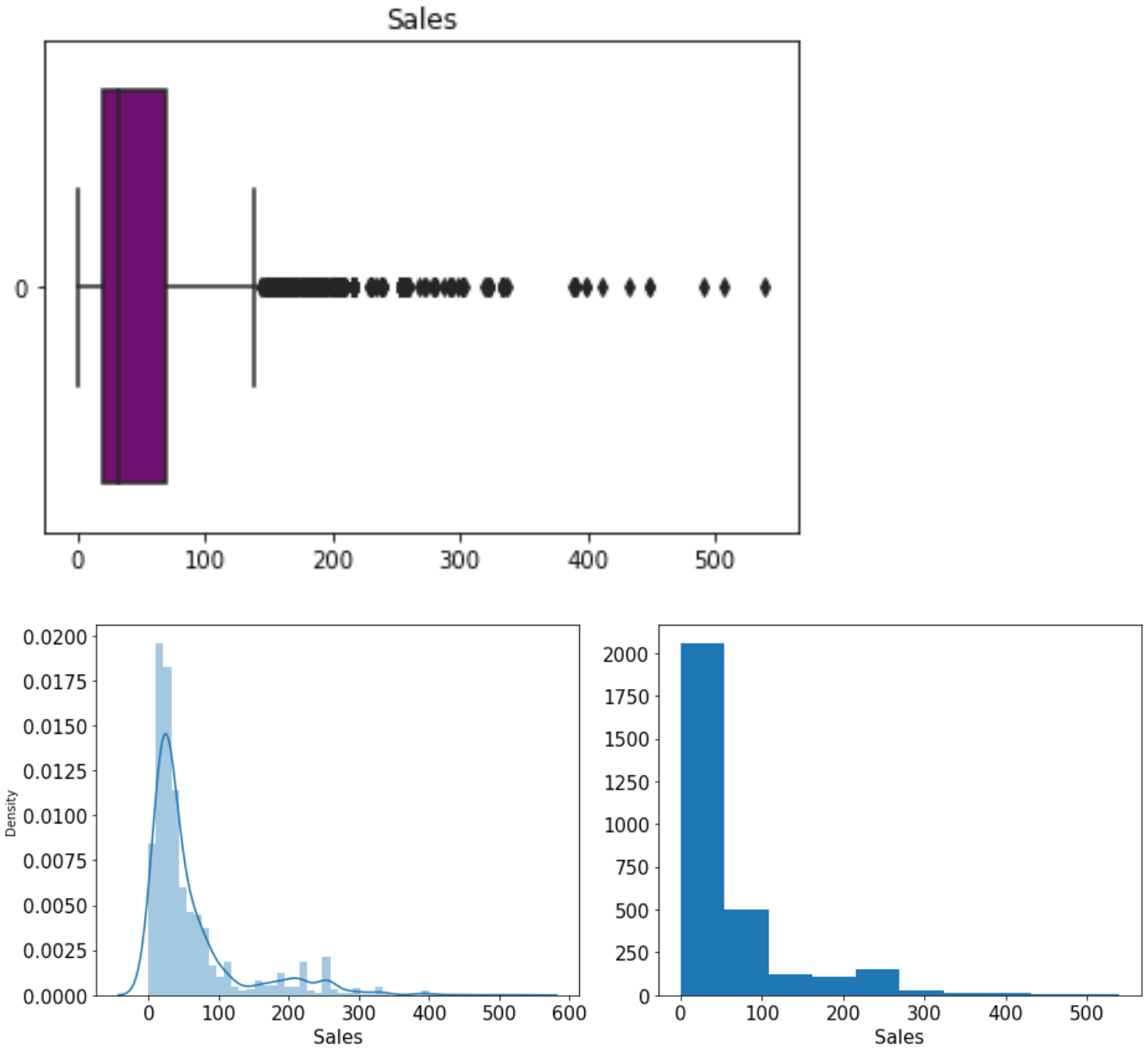


Figure 29: Univariate analysis of Sales variable

There are outliers in all the variables, but the sales and commission can be a genuine business value. Random Forest and CART can handle the outliers. Hence, Outliers are not treated for now, we will keep the data as it is.

The outliers will be treated for the ANN model to compare the same after the all the steps just for comparison.

Categorical Variables:

Agency_Code:

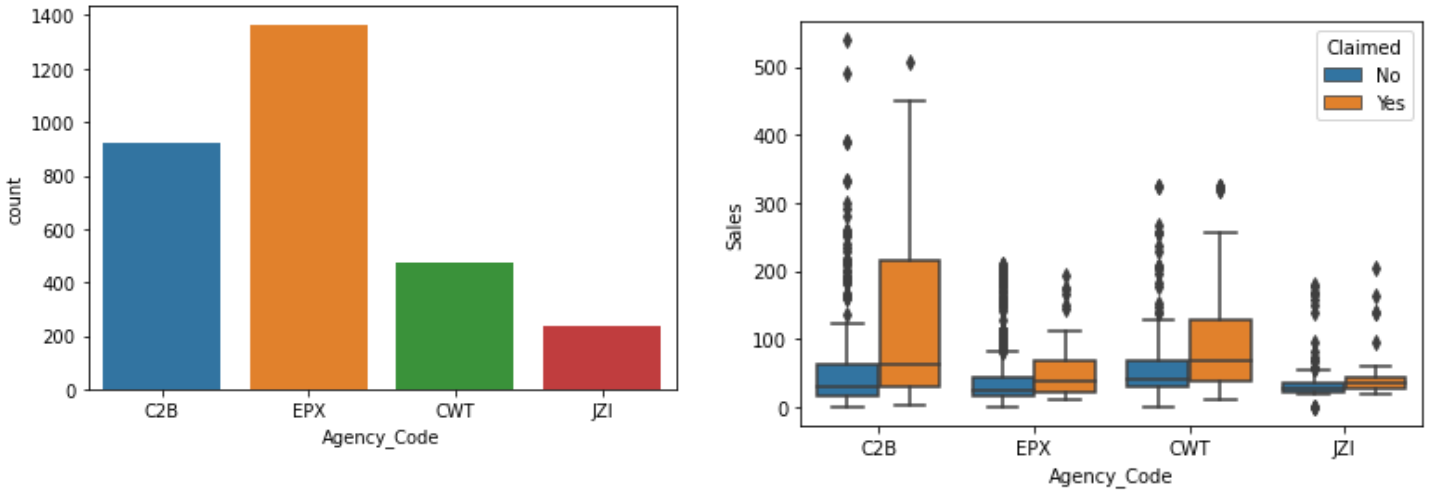


Figure 30: Count plot and boxplot of Agency code variable

Type:

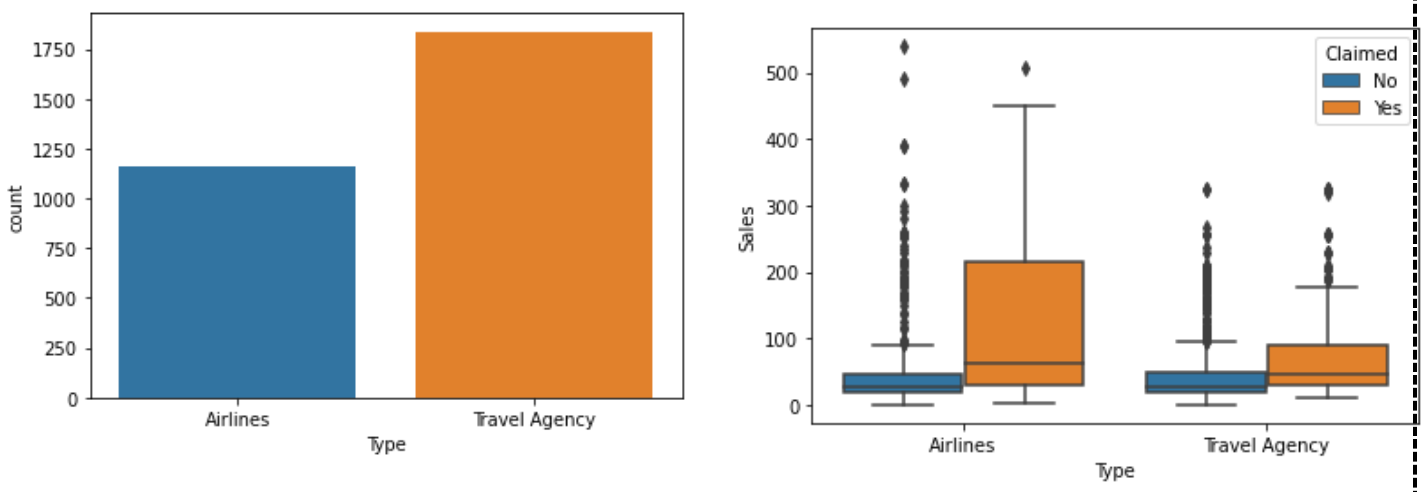


Figure 31: Count plot and boxplot of Type variable

Channel:

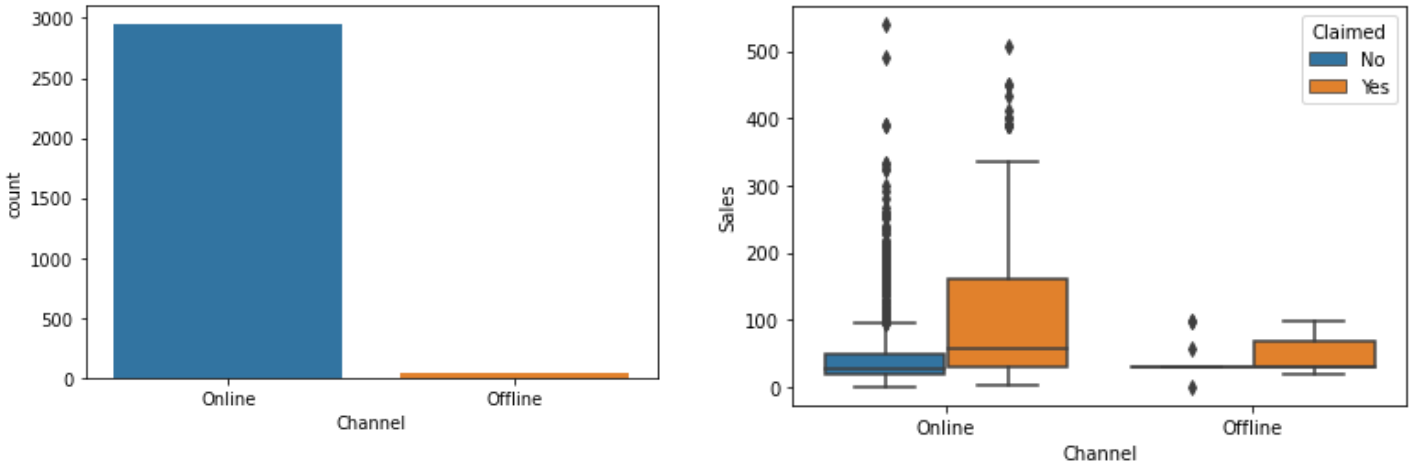


Figure 32: Count plot and boxplot of Channel variable

Product Name:

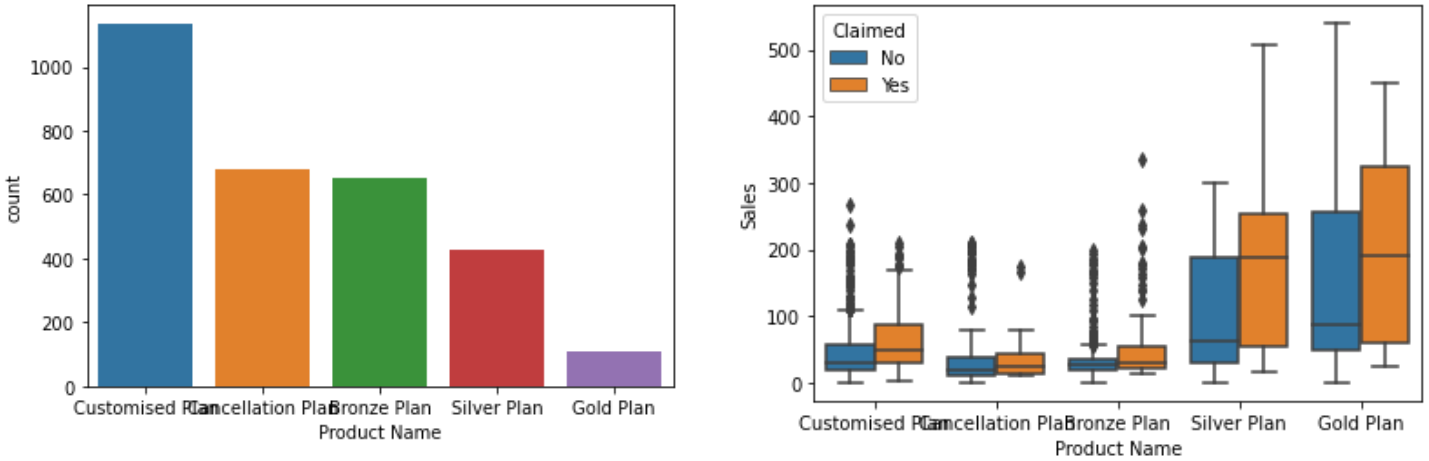


Figure 33: Count plot and boxplot of Product Name variable

Destination:

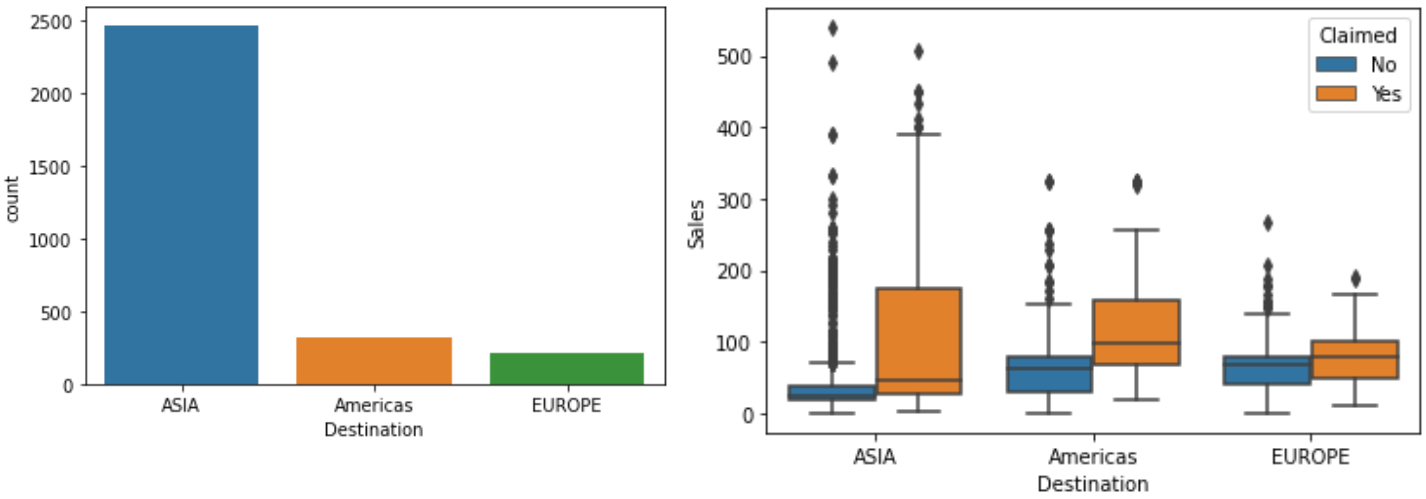


Figure 34: Count plot and boxplot of Destination variable

Checking pairwise distribution of the continuous variables

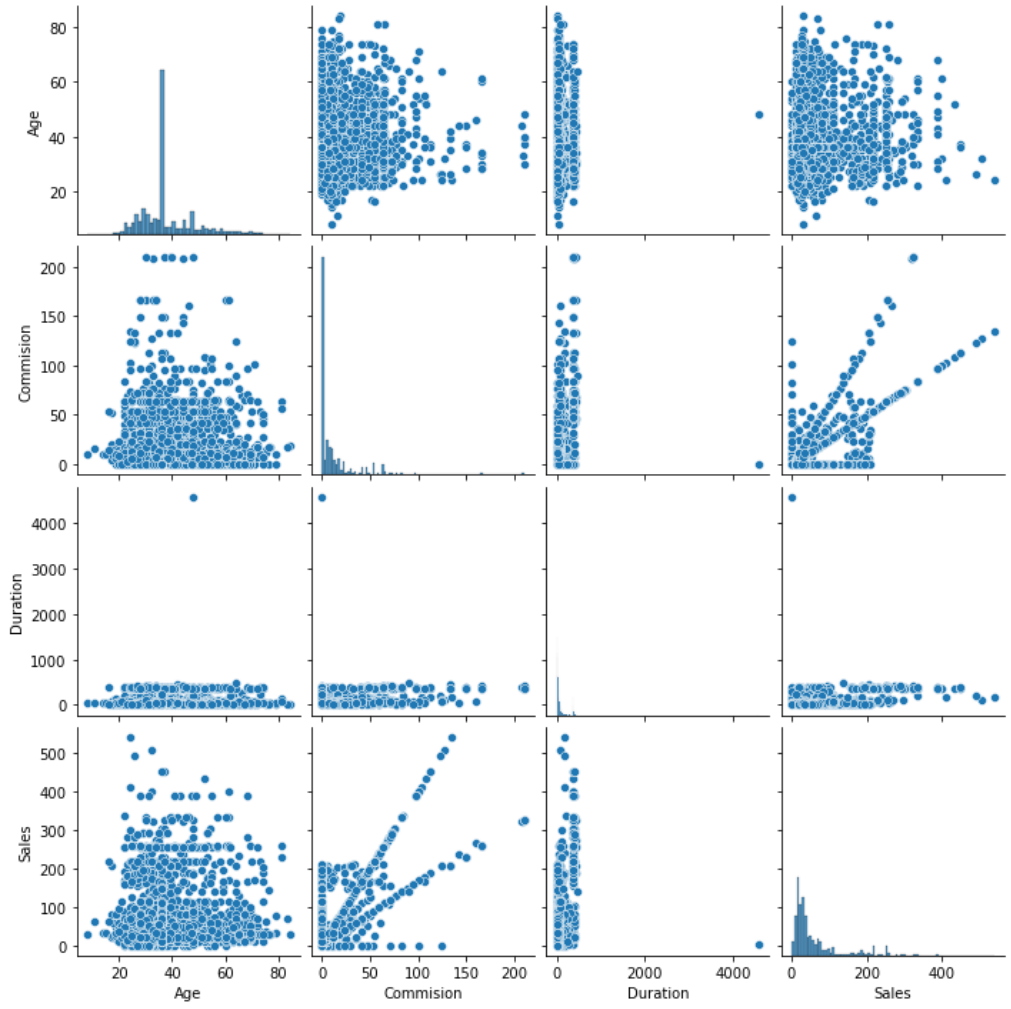


Figure 35: Multivariate analysis of independent attributes

Checking for Correlations

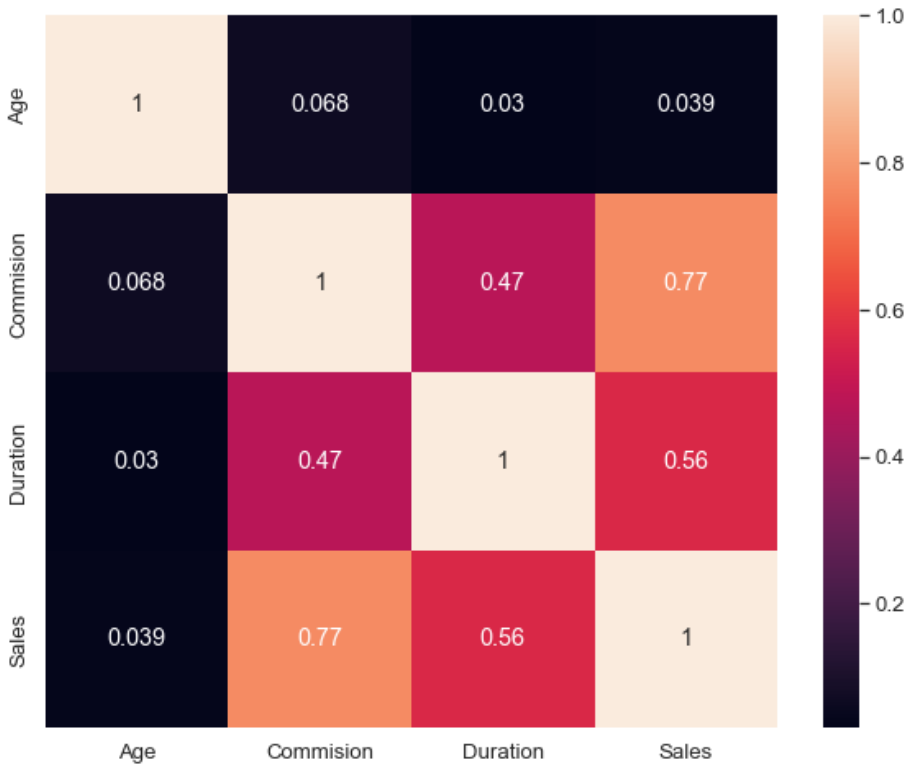


Figure 36: Correlation heatmap between numerical values

Conclusion:

- The variable Commission has high correlation with Sales variables.
- The Age Variable is least correlated with Duration variable.
- The Age variable is least correlated with all other numeric variables.

Converting all objects to categorical codes

Below is the output:

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 Pl
an']
[2 1 0 4 3]

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

Rechecking the information of the given data:

```
<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   int8
 2   Type          3000 non-null   int8
 3   Claimed       3000 non-null   int8
 4   Commision     3000 non-null   float64
 5   Channel       3000 non-null   int8
 6   Duration      3000 non-null   int64
 7   Sales         3000 non-null   float64
 8   Product Name  3000 non-null   int8
 9   Destination   3000 non-null   int8
dtypes: float64(2), int64(2), int8(6)
memory usage: 111.5 KB
```

Dataset:

	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

Proportion of 1s and 0s:

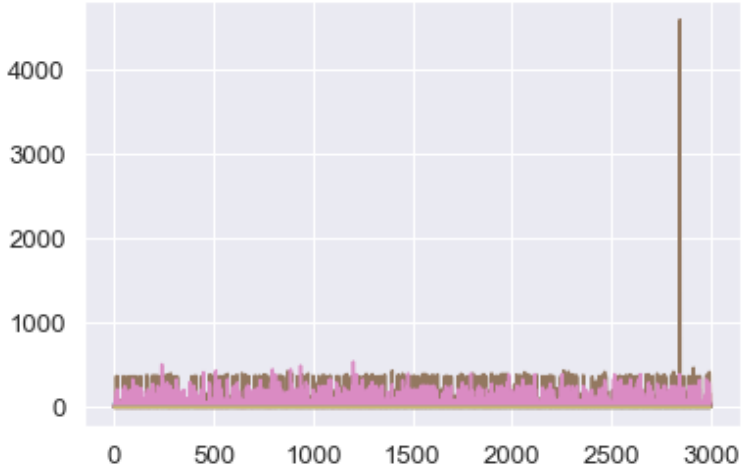
Values	Proportions
0	0.692
1	0.308

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

Extracting the target column into separate vectors for training set and test set

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

Prior to scaling:



Scaling the attributes:

	Age	Agency_Code	Type	Commision	Channel	Duration	Sales	Product Name	Destination
0	0.947162	-1.314358	-1.256796	-0.542807	0.124788	-0.470051	-0.816433	0.268835	-0.434646
1	-0.199870	0.697928	0.795674	-0.570282	0.124788	-0.268605	-0.569127	0.268835	-0.434646
2	0.086888	-0.308215	0.795674	-0.337133	0.124788	-0.499894	-0.711940	0.268835	1.303937
3	-0.199870	0.697928	0.795674	-0.570282	0.124788	-0.492433	-0.484288	-0.525751	-0.434646
4	-0.486629	1.704071	-1.256796	-0.323003	0.124788	-0.126846	-0.597407	-1.320338	-0.434646

Post scaling:

First Splitted the data into training and test set and checked the dimensions of the train and test data.

```
X_train (2100, 9)
X_test (900, 9)
train_labels (2100,)
test_labels (900,)
```

Building a Decision Tree Classifier

Formula 3 : Gini Index Calculation

$$\text{Gini(D)}=1-\sum_{i=1}^{m}(p_i)^2$$

Where , m : Number of Classes

p : Probability that a record in D belongs to class Ci

Snapshot of the Generated tree is given below:

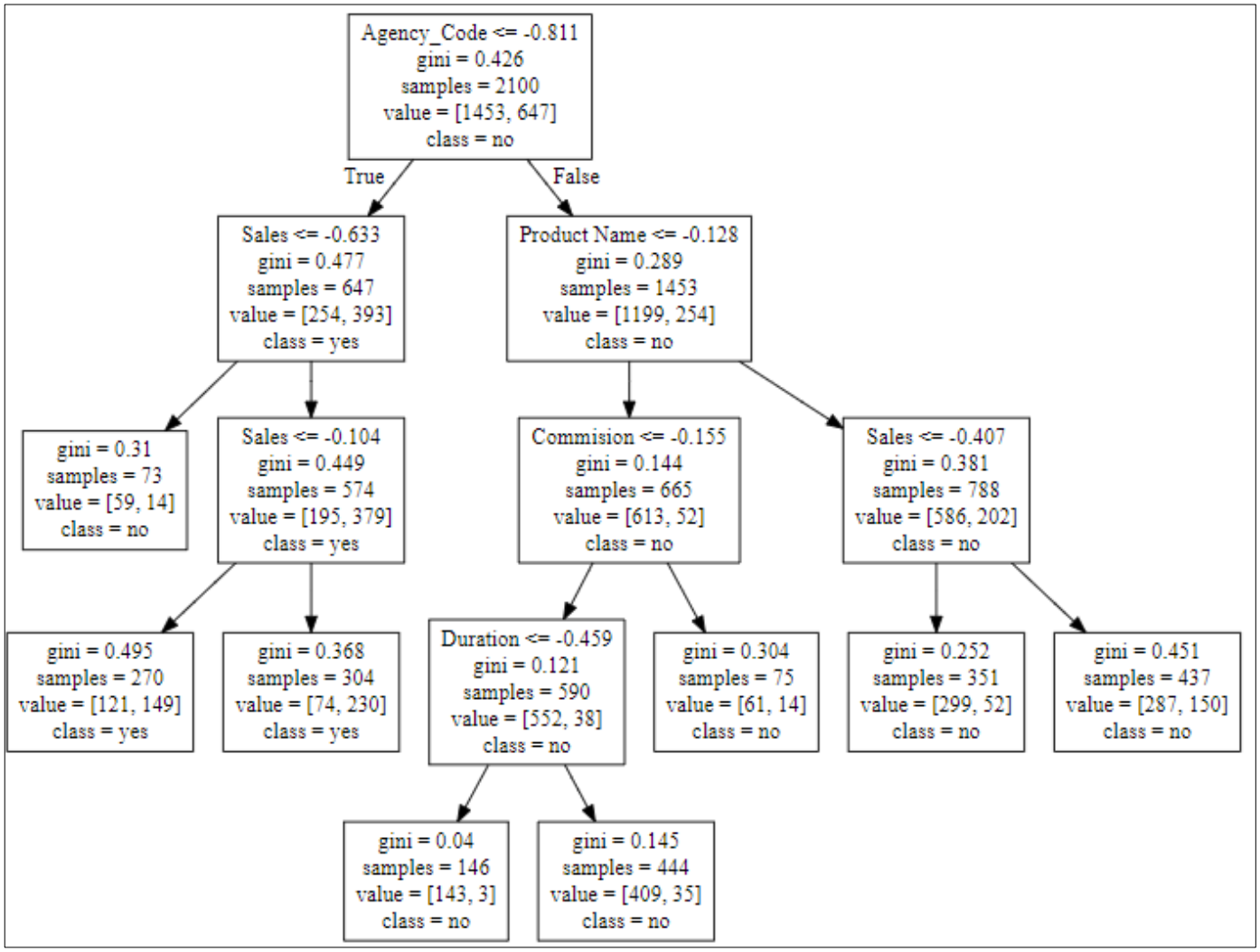


Figure 37: Snapshot of the Generated tree

Inferences:

- The First condition required for splitting the decision tree is Agency_Code <= -0.811.
- The Gini of the parent is 0.426.
- The number of samples in the root node is 2100.
- The proportion of index is [1453,647]
- The majority of the class is “No” for the parent node with the highest value 1453 and minority class is “Yes” with least value 647.
- We can observe that the left child node of the parent node has Gini value 0.289 with majority of “No” class.
- We can also observe that the right child node of the parent node has Gini value 0.477 with majority of “Yes” class.
- The two terminal nodes have Gini index of 0.04 and 0.145 with sample sizes of 146 and 444 majority of class falls under the value “No”.
- All the terminal nodes have sufficient number of data.

Variable Importance

	Imp
Agency_Code	0.634112
Sales	0.220899
Product Name	0.086632
Commision	0.021881
Age	0.019940
Duration	0.016536
Type	0.000000
Channel	0.000000
Destination	0.000000

Getting the Predicted Classes and Probability

	0	1
0	0.656751	0.343249
1	0.979452	0.020548
2	0.921171	0.078829
3	0.656751	0.343249
4	0.921171	0.078829

Analysis 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.

1. CART - AUC and ROC for the training and testing data:

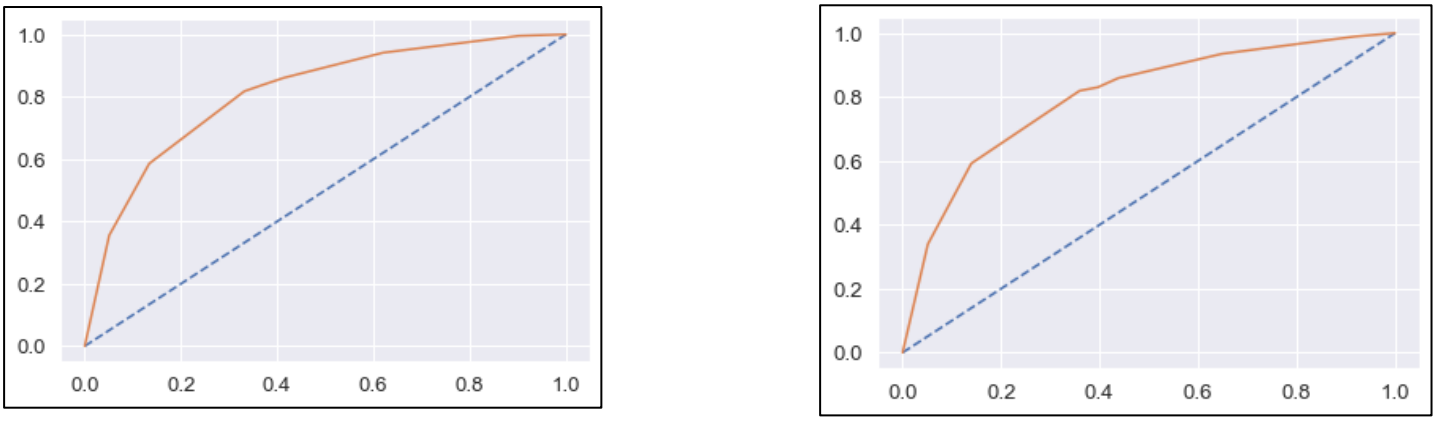


Figure 38: AUC and ROC for CART for training and testing

Confusion Matrix for the training data

```
array([[1258,  195],
       [ 268,  379]], dtype=int64)
```

Train Data Accuracy - 0.78

Classification Report:

	precision	recall	f1-score	support
0	0.82	0.87	0.84	1453
1	0.66	0.59	0.62	647
accuracy			0.78	2100
macro avg	0.74	0.73	0.73	2100
weighted avg	0.77	0.78	0.78	2100

Inferences:

- cart_train_precision value is 0.66
- cart_train_recall is 0.59
- cart_train_f1 is 0.62

CART Confusion Matrix and Classification Report for the testing data

Confusion Matrix for test data

```
array([[536,  87],
       [113, 164]], dtype=int64)
```

Test Data Accuracy is 0.778

Classification Report:

	precision	recall	f1-score	support
0	0.83	0.86	0.84	623
1	0.65	0.59	0.62	277
accuracy			0.78	900
macro avg	0.74	0.73	0.73	900
weighted avg	0.77	0.78	0.77	900

Inferences:

- cart_test_precision value is 0.65
- cart_test_recall value is 0.59
- cart_test_f1 value is 0.62

CART conclusion

Train Data:

- AUC: 81%
- Accuracy: 78%
- Precision: 66%
- f1-Score: 62%

Test Data:

- AUC: 80%
- Accuracy: 77%
- Precision: 65%
- f1-Score: 62%

Training and Test set results are almost similar, and with the overall measures high, the hence the model prediction is good.

Change is the most important variable for predicting diabetes.

Building a Random Forest Classifier

Grid Search used for finding out the optimal values for the hyper parameters.

RF Model Performance Evaluation on Training data

Confusion matrix:

```
array([[1311,  142],
       [ 243,  404]], dtype=int64)
```

Random Forest train accuracy value is 0.82

Classification Report:

	precision	recall	f1-score	support
0	0.84	0.90	0.87	1453
1	0.74	0.62	0.68	647
accuracy			0.82	2100
macro avg	0.79	0.76	0.77	2100
weighted avg	0.81	0.82	0.81	2100

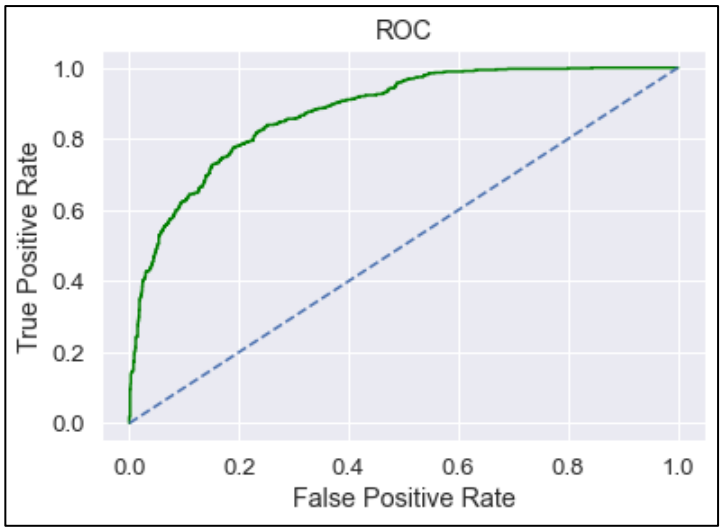


Figure 39: Classification report and ROC for Random Forest algorithm

RF Model Performance Evaluation on Test data

Confusion matrix

```
array([[547,  76],
       [122, 155]], dtype=int64)
```

Random Forest Accuracy value is 0.78

Report Classification

	precision	recall	f1-score	support
0	0.82	0.88	0.85	623
1	0.67	0.56	0.61	277
accuracy			0.78	900
macro avg	0.74	0.72	0.73	900
weighted avg	0.77	0.78	0.77	900

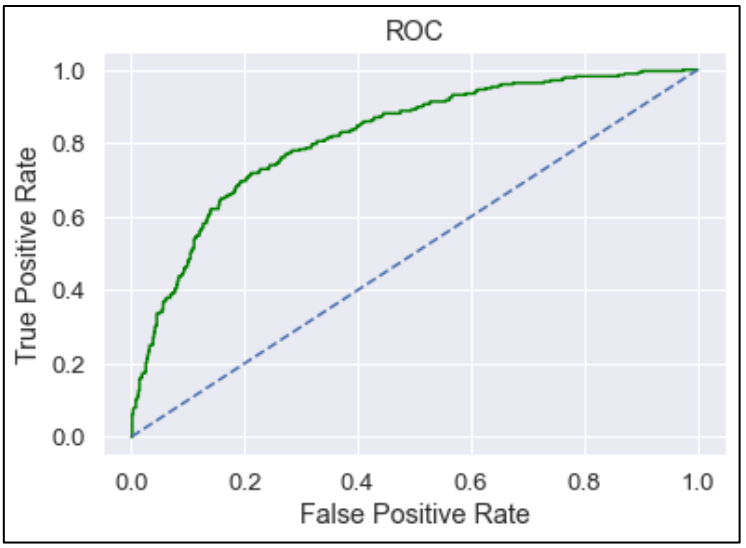


Figure 39: Classification report and ROC for Random Forest algorithm

Random Forest Conclusion

Train Data:

- AUC: 88%
- Accuracy: 82%
- Precision: 74%
- f1-Score: 68%

Test Data:

- AUC: 82%
- Accuracy: 78%
- Precision: 67%
- f1-Score: 61

Training and Test set results are almost similar, and with the overall measures high, the model is a good model. Change is again the most important variable for predicting diabetes

III. Neural Network

Neural Network Model Performance Evaluation on Training data

Confusion matrix:

```
array([[1311,  142],
       [ 243,  404]], dtype=int64)
```

Neural Network training accuracy is 0.82

Classification report:

	precision	recall	f1-score	support
0	0.84	0.90	0.87	1453
1	0.74	0.62	0.68	647
accuracy			0.82	2100
macro avg	0.79	0.76	0.77	2100
weighted avg	0.81	0.82	0.81	2100

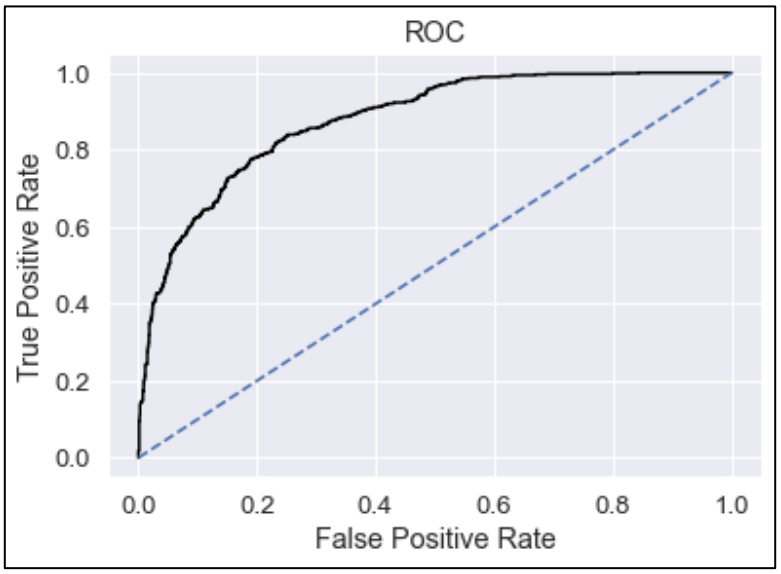


Figure 40: Classification report and ROC for Neural Network algorithm – Training data

Neural Network Model Performance Evaluation on Test data

Confusion matrix

```
array([[547,  76],
       [122, 155]], dtype=int64)
```

Neural Network accuracy for test data is 0.78

Classification report:

	precision	recall	f1-score	support
0	0.82	0.88	0.85	623
1	0.67	0.56	0.61	277
accuracy			0.78	900
macro avg	0.74	0.72	0.73	900
weighted avg	0.77	0.78	0.77	900

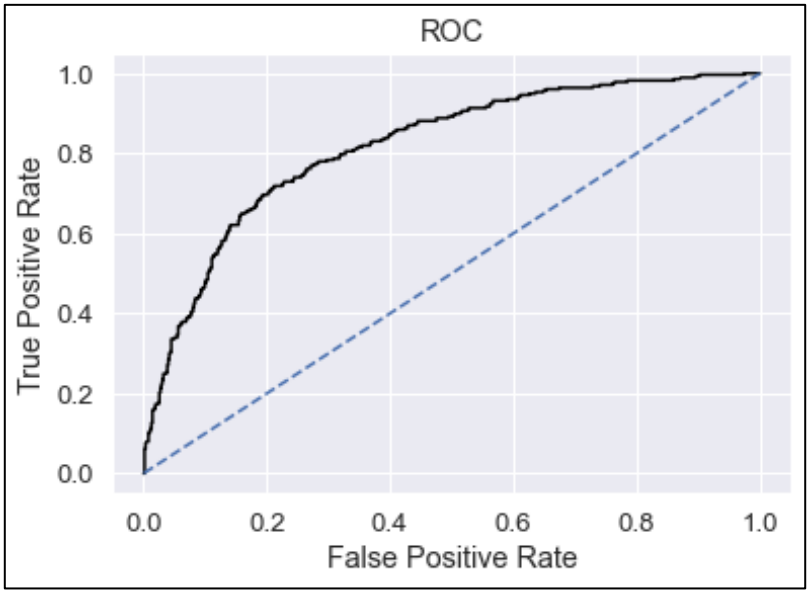


Figure 41: Classification report and ROC for Neural Network algorithm – Test data

Neural Network Conclusion

Train Data:

- AUC: 88%
- Accuracy: 82%
- Precision: 74%
- f1-Score: 68

Test Data:

- AUC: 82%
- Accuracy: 78%
- Precision: 67%
- f1-Score: 61%

Training and Test set results are almost similar, and with the overall measures high, the model is a good model.

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

Comparison of the performance metrics from the 3 models

	CART Train	CART Test	Random Forest Train	Random Forest Test	Neural Network Train	Neural Network Test
Accuracy	0.78	0.78	0.82	0.78	0.82	0.78
AUC	0.81	0.80	0.88	0.82	0.88	0.82
Recall	0.59	0.59	0.62	0.56	0.62	0.56
Precision	0.66	0.65	0.74	0.67	0.74	0.67
F1 Score	0.62	0.62	0.68	0.61	0.68	0.61

ROC Curve for the 3 models on the Training data

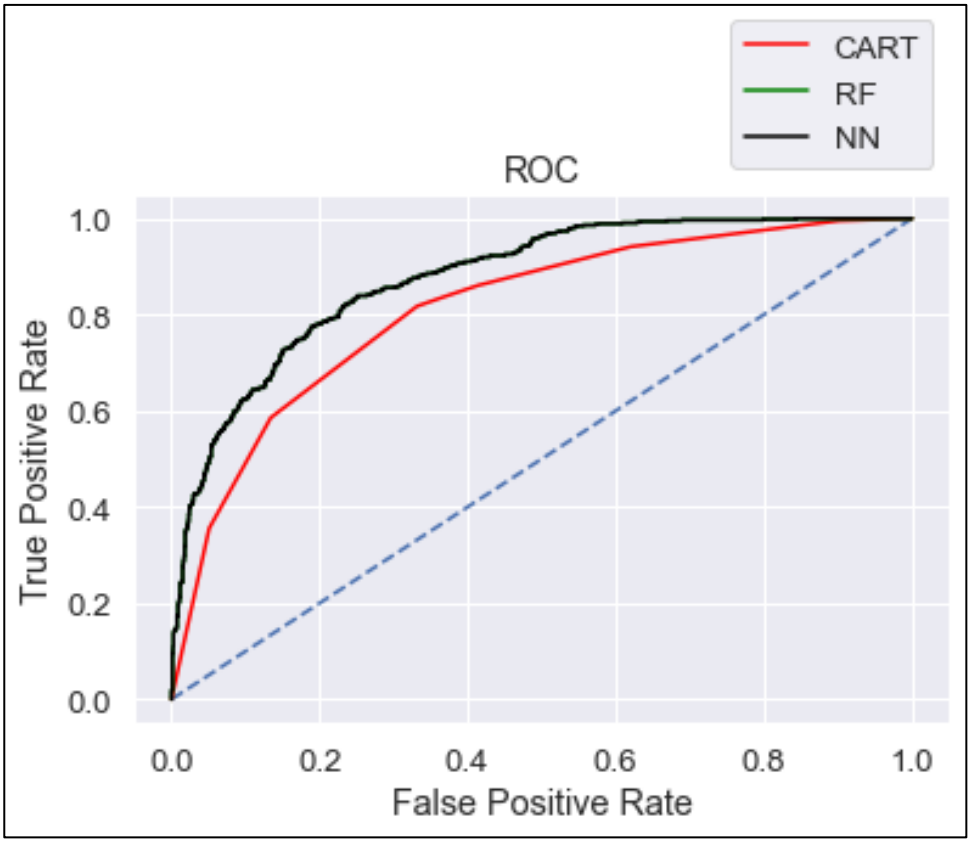


Figure 42: ROC Curve for the 3 models on the Training data

ROC Curve for the 3 models on the Test data

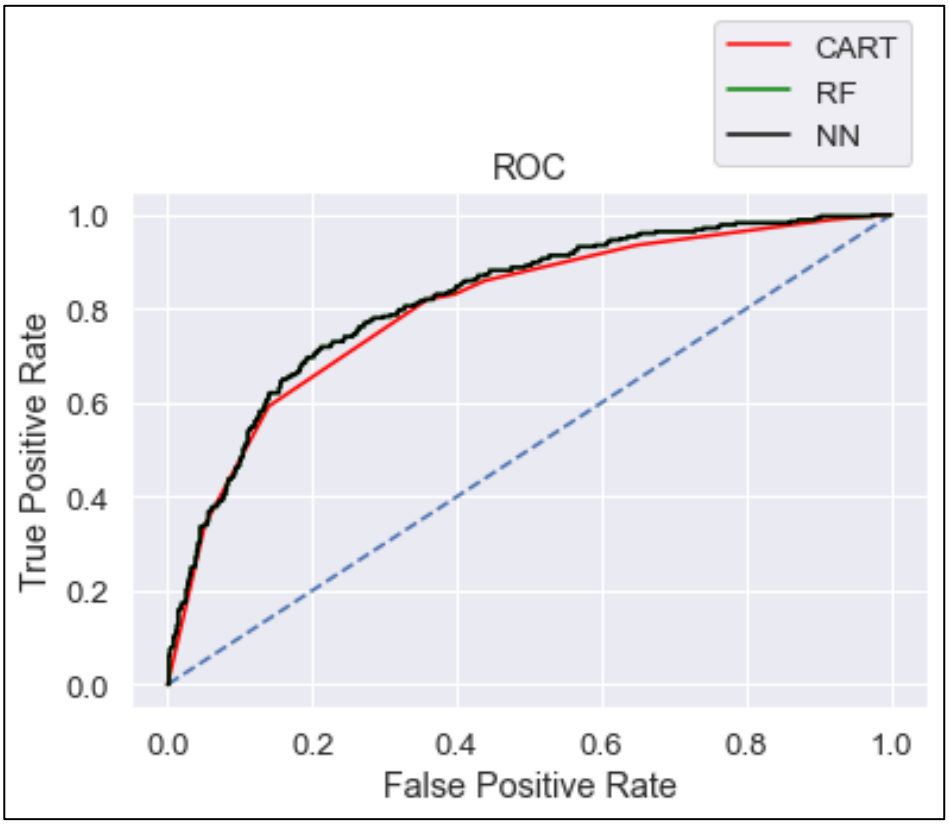


Figure 43: ROC Curve for the 3 models on the Test data

Conclusion:

Random Forest model should be selected, as it provides better accuracy, precision, recall, f1 score and scores better than CART & Neural Network on the same parameters.

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

- This is understood by looking at the insurance data by drawing relations between different variables such as day of the incident, time, age group, and associating it with other external information such as location, behavior patterns, weather information, airline/vehicle types, etc.
- Streamlining online experiences benefitted customers, leading to an increase in conversions, which subsequently raised profits.
- As per the data 90% of insurance is done by online channel.

- Other interesting fact, is almost all the offline business has a claimed associated.

- Need to train the JZI agency resources to pick up sales as they are in bottom, need to run promotional marketing campaign or evaluate if we need to tie up with alternate agency.

- Also based on the model we are getting 80% accuracy, so we need customer books airline tickets or plans, cross sell the insurance based on the claim data pattern. Other interesting fact is more sales happen via Agency than Airlines and the trend shows the claim are processed more at Airline.

- **The KPI's of insurance claims are**:

  - Reduce claims cycle time
  - Increase customer satisfaction
  - Combat fraud
  - Optimize claims recovery
  - Reduce claim handling costs Insights gained from data and AI-powered analytics could expand the boundaries of insurability, extend existing products, and give rise to new risk transfer solutions in areas like a non-damage business interruption and reputational damage.