# Data Mining Business Report

- **Name**: Hemant Patidar
- **Batch**: PGPDSBA Online Sep_A 2021
- **Date**: 23/01/2022

## Table of Contents

# List of Figures

# List of Tables

# Clustering

## Executive Summary

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

Purpose of our exercise would be to cluster profiles and give recommendation to strategic business management group.

## Data Description

- **spending**: Amount spent by the customer per month (in 1000s)
- **advance_payments**: Amount paid by the customer in advance by cash (in 100s)
- **probability_of_full_payment**: Probability of payment done in full by the customer to the bank
- **current_balance**: Balance amount left in the account to make purchases (in 1000s)
- **credit_limit**: Limit of the amount in credit card (10000s)
- **min_payment_amt** : minimum paid by the customer while making payments for purchases made monthly (in 100s)
- **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)

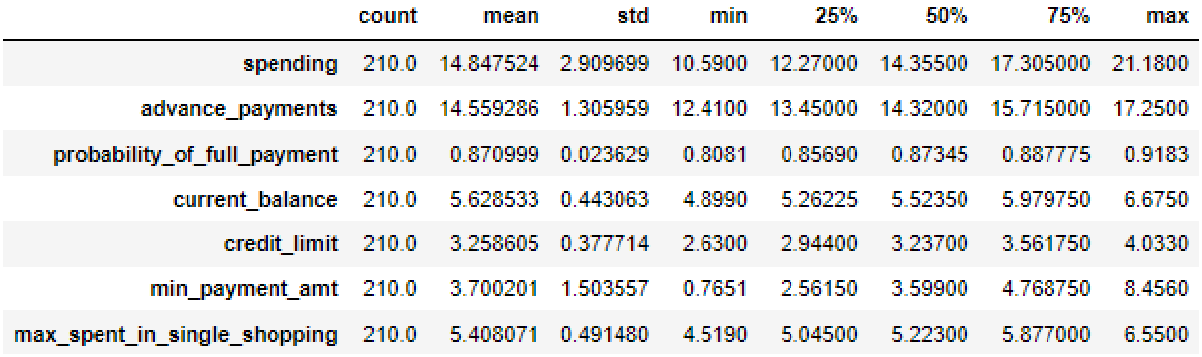
### Basic Data Information

The basic information about data can be derived based on Table 1 - Info - Market Segment Data

| Column | Non-Null Count | Dtype |
|---|---|---|
| spending | 210 non-null | float64 |
| advance_payments | 210 non-null | float64 |
| probability_of_full_payment | 210 non-null | float64 |
| current_balance | 210 non-null | float64 |
| credit_limit | 210 non-null | float64 |
| min_payment_amt | 210 non-null | float64 |
| max_spent_in_single_shopping | 210 non-null | float64 |

*Table 1 - Info - Market Segment Data*

- Data has 210 records and 7 different fields
- There is no null value in data
- All fields are float
- No duplicate record on the data



| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| spending | 210.0 | 14.847524 | 2.909699 | 10.5900 | 12.27000 | 14.35500 | 17.305000 | 21.1800 |
| advance_payments | 210.0 | 14.559286 | 1.305959 | 12.4100 | 13.45000 | 14.32000 | 15.715000 | 17.2500 |
| probability_of_full_payment | 210.0 | 0.870999 | 0.023629 | 0.8081 | 0.85690 | 0.87345 | 0.887775 | 0.9183 |
| current_balance | 210.0 | 5.628533 | 0.443063 | 4.8990 | 5.26225 | 5.52350 | 5.979750 | 6.6750 |
| credit_limit | 210.0 | 3.258605 | 0.377714 | 2.6300 | 2.94400 | 3.23700 | 3.561750 | 4.0330 |
| min_payment_amt | 210.0 | 3.700201 | 1.503557 | 0.7651 | 2.56150 | 3.59900 | 4.768750 | 8.4560 |
| max_spent_in_single_shopping | 210.0 | 5.408071 | 0.491480 | 4.5190 | 5.04500 | 5.22300 | 5.877000 | 6.5500 |

*Table 2 - General Stats About Data*

- Average spending of customers is around 14.8k, with a minimum of 10.59k and maximum of 21.18k.
- Minimum advance payment made by customers is 1.24k, having an average of 1.45k.
- Average probability of full payment by customer is 87%
- Customers are having around 5.62k left in their account on average, and their credit limit is 32.5k on average
- Maximum spent in single shopping has an average of 5.4k, and minimum payment amount for monthly purchases is 370 (on average)

## Data Visualization

- There is no categorical field in the data, all the information available is numerical in nature and most of them represents the amount.

### *Univariate Analysis:*

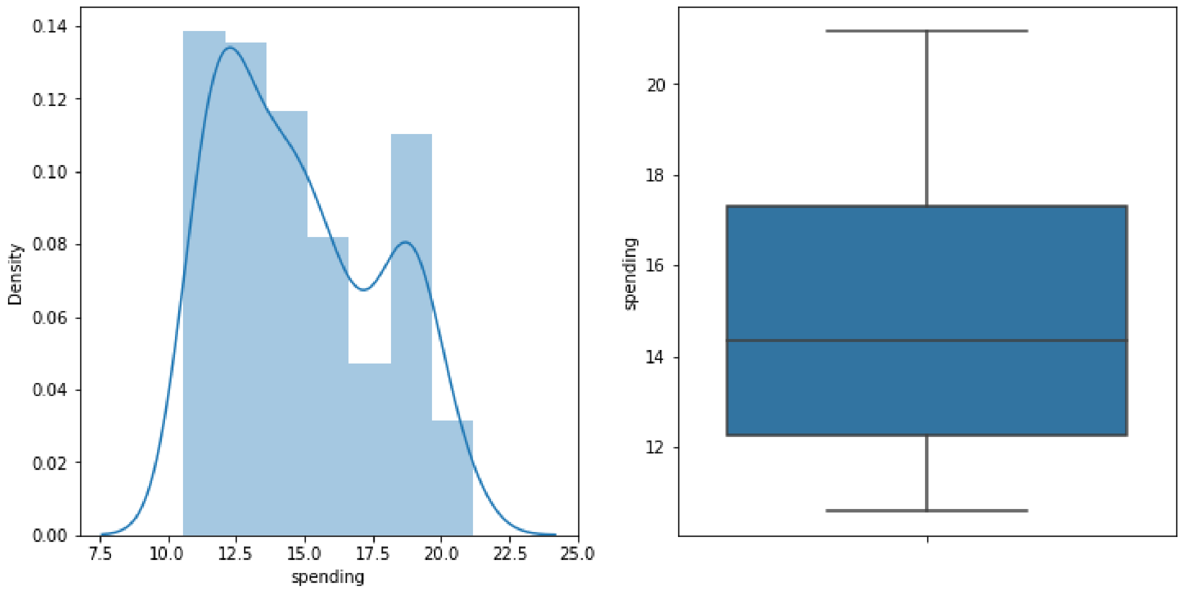


*Figure I - Spending Analysis*

**Spending**: There is no outlier present for spending data and lying between 10k & 22.5k, but the distribution looks positively skewed by value 0.397

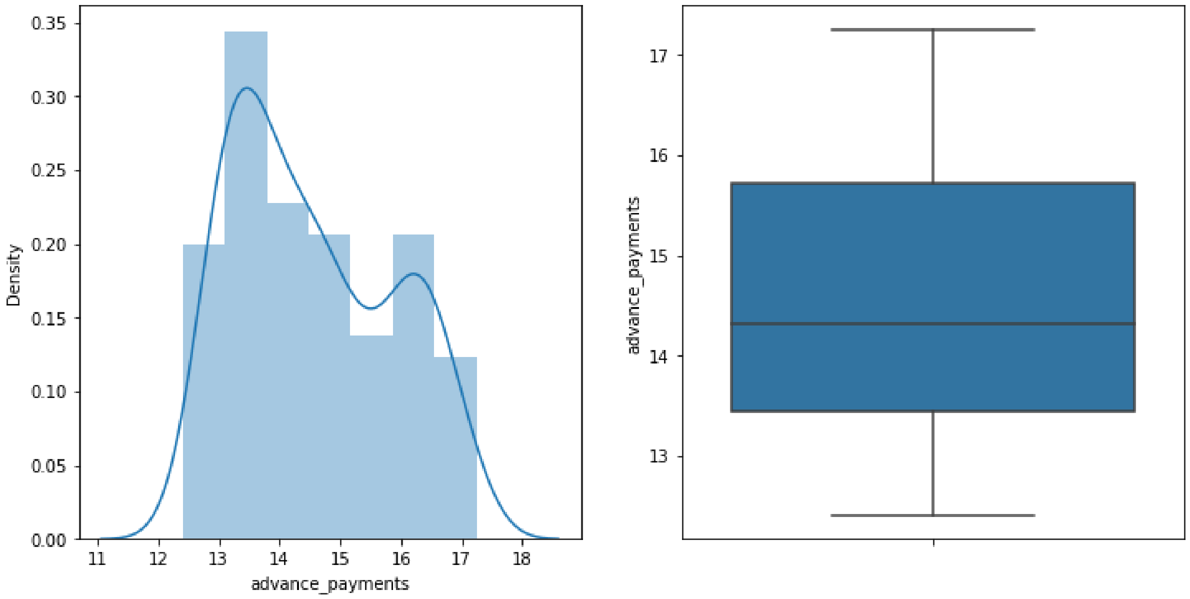


*Figure II - Advance Payments Analysis*

**Advance Payments:** we do not have any outlier on Advance payments data, but the distribution seem positively skewed (0.383) and range lies between 1.25k to 1.75k.

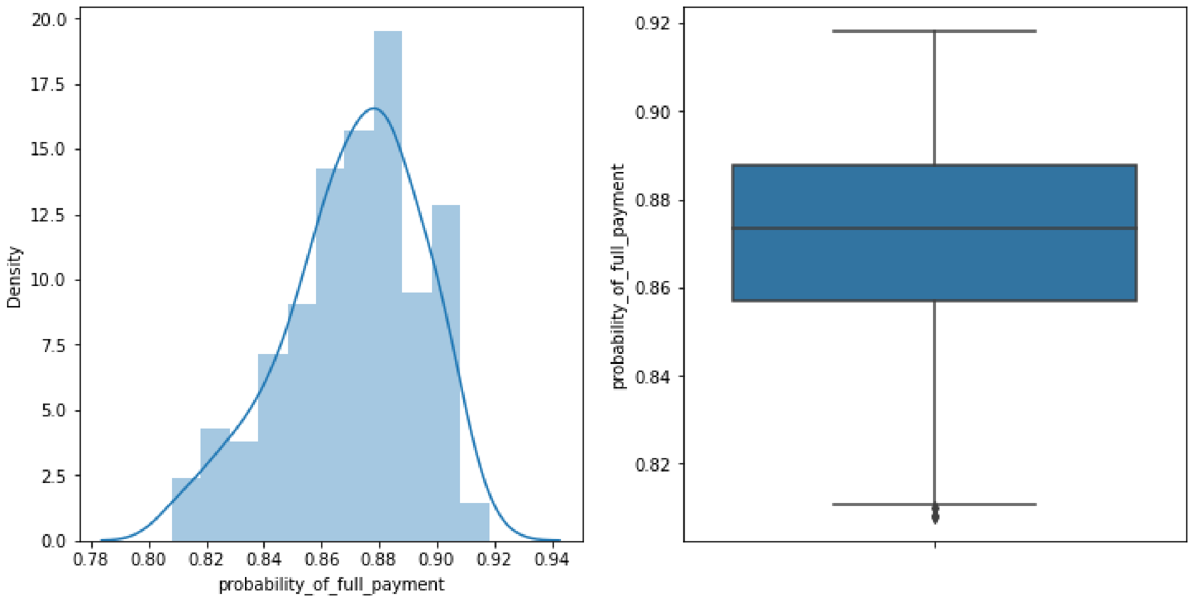


*Figure III - Probability of full payment Analysis*

**Probability of Full Payment**: We have 3 outliers (1.42% data) in probability of full payments, and data seems negatively skewed (-0.534) which means probability of full payment for most of the customer is high.

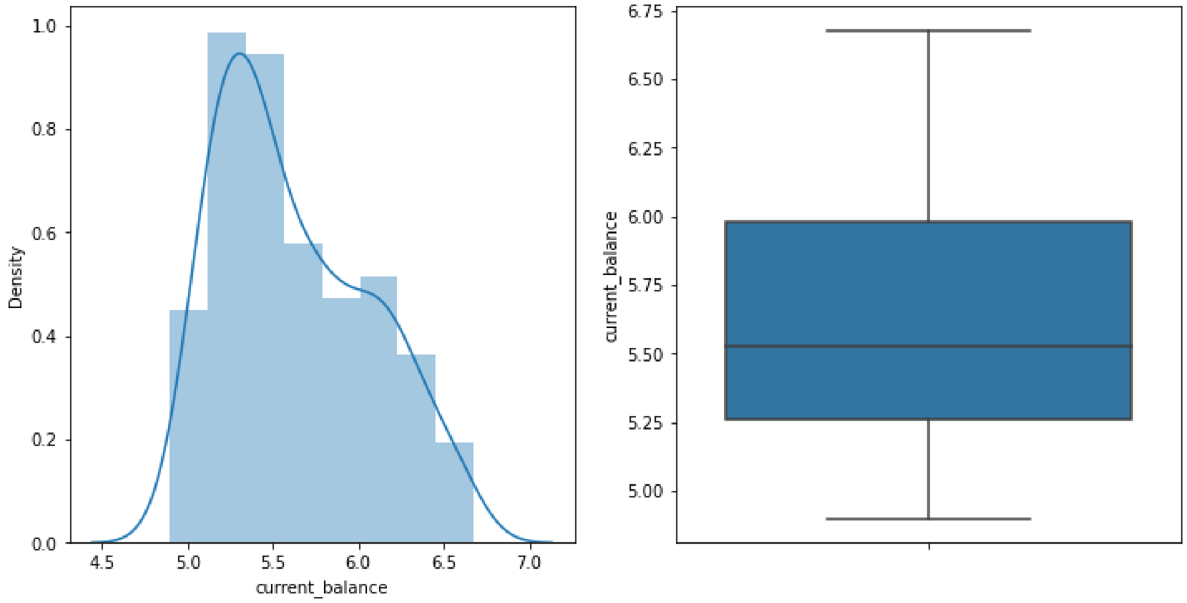


*Figure IV - Current Balance Analysis*

**Current Balance**: Balance left in account for purchase doesn't seem extreme or lowest for any customer (no outlier), but the distribution looks positively skewed (0.521). Current balance lies between 4.8k to 6.8k.

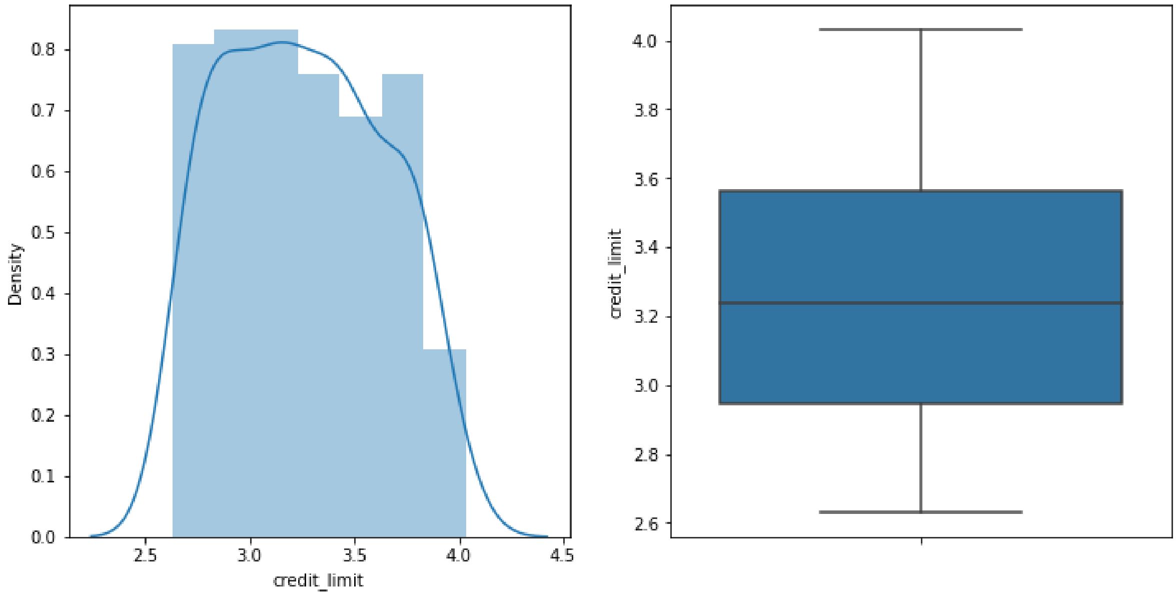


*Figure V - Credit Limit Analysis*

**Credit Limit:** The data looks following normal distribution (skewness 0.133) and have no outlier, and the limit seems lying between 25k to 41k

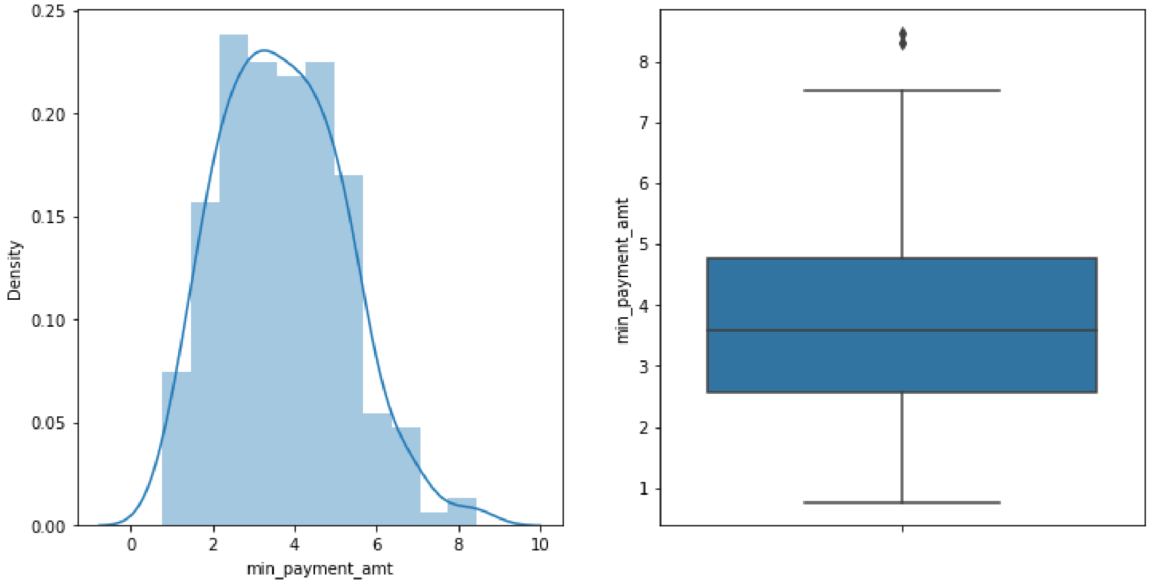


*Figure VI - Minimum Payment Amount Analysis*

**Minimum Payment Amount:** We have 2 extreme values present on data (0.95% of data as outlier). The payment amount (minimum) lies in a range 0-900. And distribution looks slightly positively skewed (0.398)

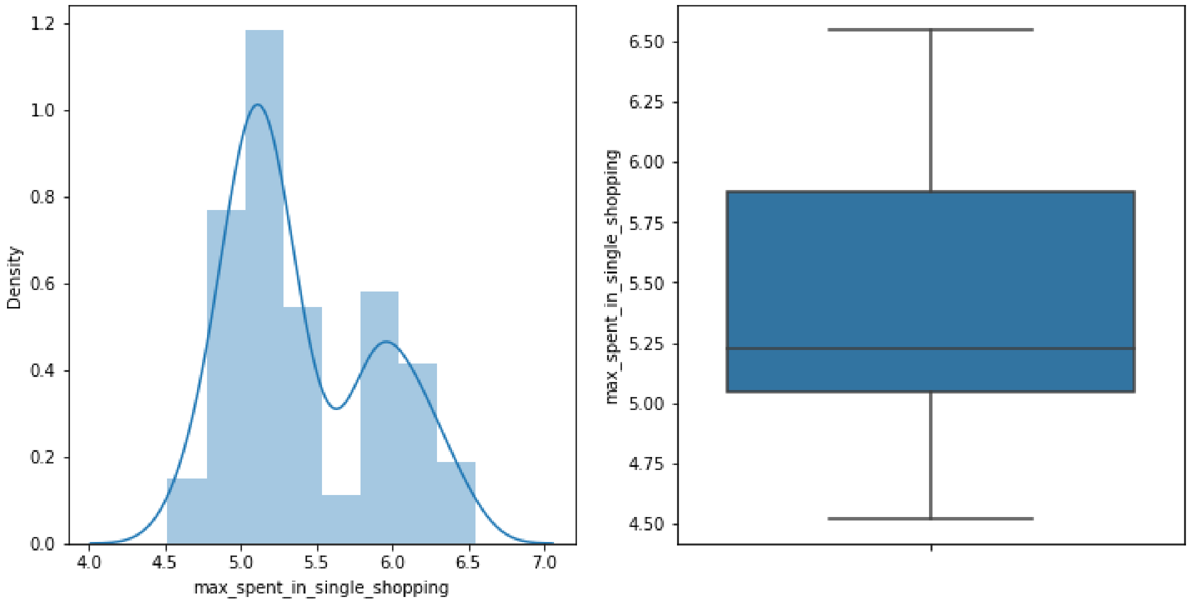


*Figure VII - Maximum Spent In Single Shopping Analysis*

**Maximum Spent In Single Shopping:** Data does not have any outlier but distribution seems positively skewed with 0.557 skewness. Range of spent looks 4.5k – 6.5k

## *Bi-Variate Analysis*

**Pair Plot:**

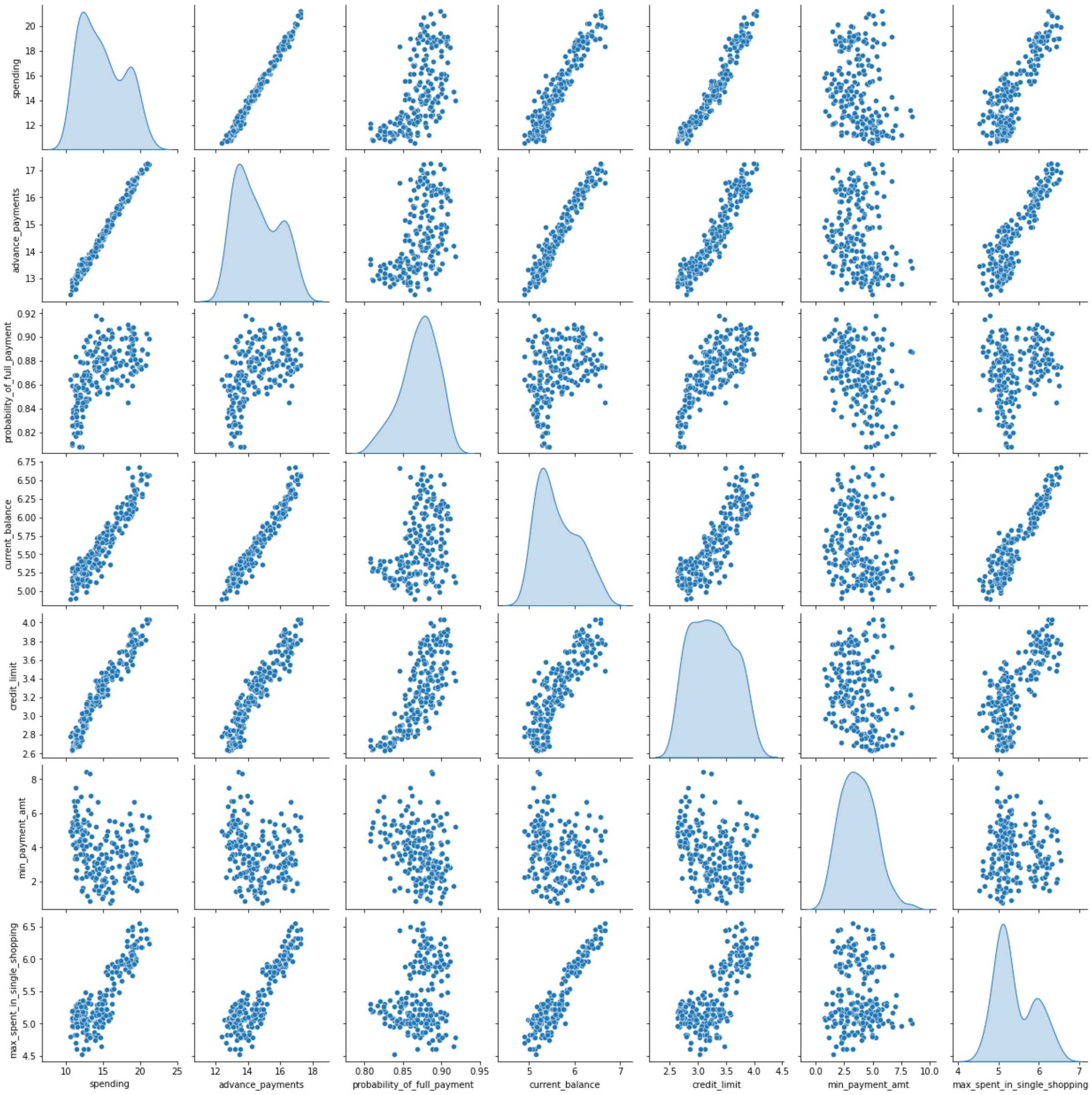


*Figure VIII - Pair Plot*

From this pair plot, we can observe that –

- Spending, advance payments, current balance, credit limit and maximum spent on one shopping look proportional to each other and increases/decreases together.
- Minimum payment amount looks scattered in its' range and does not follow any other variable.

**Correlation Matrix (Heat map):**

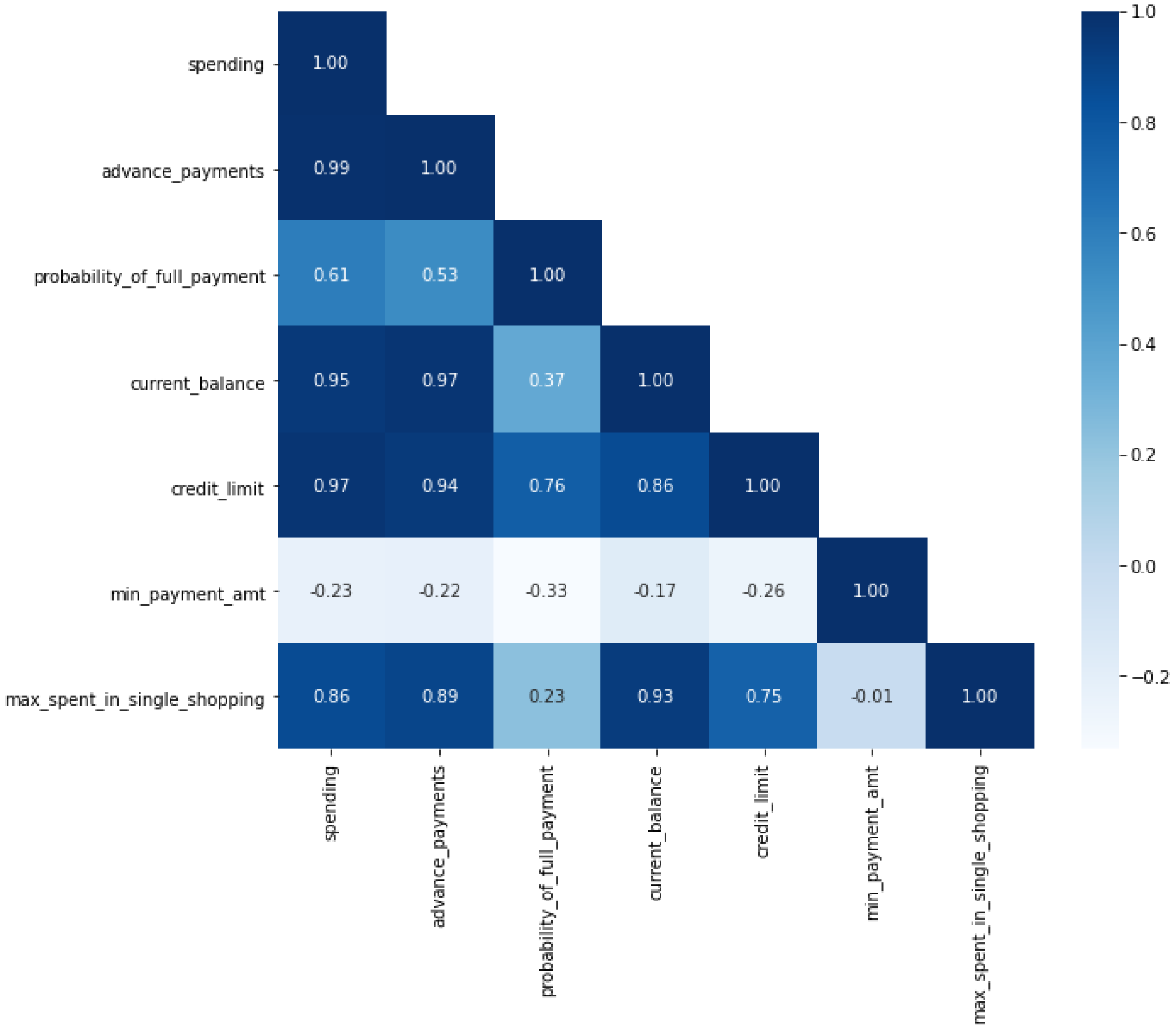


*Figure IX - Correlation Matrix (Heat Map)*

From above matrix, we can infer that –

- Spending, advance payments, current balance, credit limit and maximum spent on one shopping are highly correlated with each other as seen in pair plot.
- Minimum payment amount doesn't seem correlated with any other variable.
- Advance payment and Spending have highest correlation as 0.99 followed by Credit limit and spending as 0.95

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

**Yes,** features' scaling is necessary for clustering, the clustering techniques uses different types of distances between points and un-even weights of feature will give feature-bias results.

As specified in data dictionary, the features' unit are not consistent (i.e. some in 100s and 1000s and probability would be in 0-1)

Feature values in 10000s – Credit Limit

Feature values in 1000s – Spending, Current Balance, Maximum spent in one purchase

Feature values in 100s – Advance payments, Minimum payment amount

Feature values in 0-1 – Probability of Full Payment

We need to bring all these in one scale, and we will be using Standard scaling (Z-score)

**Z-Score Scaling** – This technique would represent data values in terms of how many standard deviations it is far from the mean.

Formula -

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

Where,

Z – Z-score value

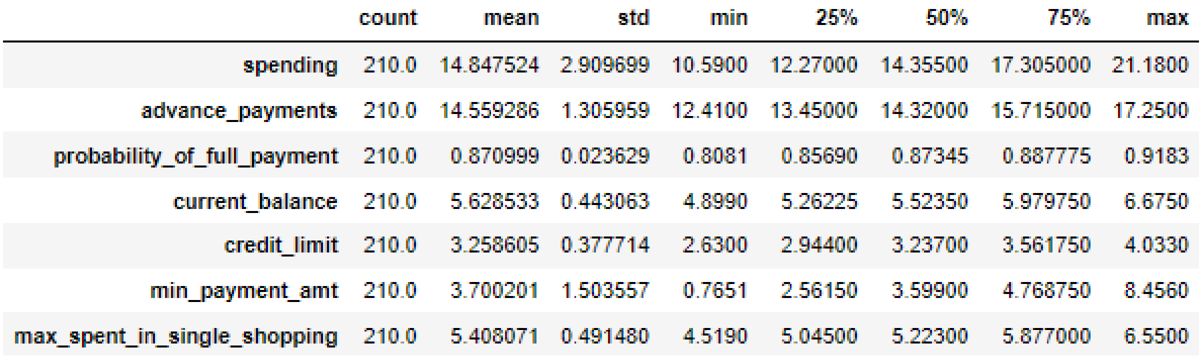
x – Data point value

µ – Mean of feature

σ – Standard deviation

We will treat outlier before scaling as our goal would be to cluster similar profiles and outlier may affect clustering.

**Before Scaling** –

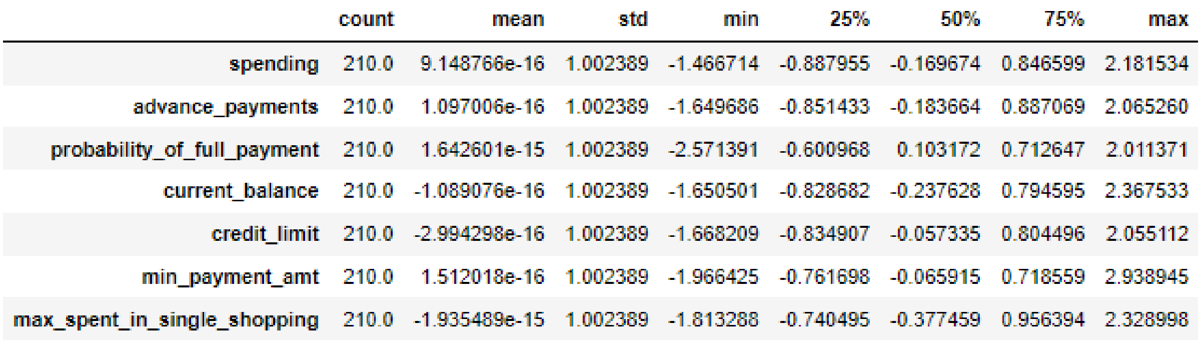


| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| spending | 210.0 | 14.847524 | 2.909699 | 10.5900 | 12.27000 | 14.35500 | 17.305000 | 21.1800 |
| advance_payments | 210.0 | 14.559286 | 1.305959 | 12.4100 | 13.45000 | 14.32000 | 15.715000 | 17.2500 |
| probability_of_full_payment | 210.0 | 0.870999 | 0.023629 | 0.8081 | 0.85690 | 0.87345 | 0.887775 | 0.9183 |
| current_balance | 210.0 | 5.628533 | 0.443063 | 4.8990 | 5.26225 | 5.52350 | 5.979750 | 6.6750 |
| credit_limit | 210.0 | 3.258605 | 0.377714 | 2.6300 | 2.94400 | 3.23700 | 3.561750 | 4.0330 |
| min_payment_amt | 210.0 | 3.700201 | 1.503557 | 0.7651 | 2.56150 | 3.59900 | 4.768750 | 8.4560 |
| max_spent_in_single_shopping | 210.0 | 5.408071 | 0.491480 | 4.5190 | 5.04500 | 5.22300 | 5.877000 | 6.5500 |

*Table 3 - Data before Scaling*

The means and standard deviations for feature are not consistent for units (as stated earlier, features are in different units)

**After Scaling** –



| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| spending | 210.0 | 9.148766e-16 | 1.002389 | -1.466714 | -0.887955 | -0.169674 | 0.846599 | 2.181534 |
| advance_payments | 210.0 | 1.097006e-16 | 1.002389 | -1.649686 | -0.851433 | -0.183664 | 0.887069 | 2.065260 |
| probability_of_full_payment | 210.0 | 1.642601e-15 | 1.002389 | -2.571391 | -0.600968 | 0.103172 | 0.712647 | 2.011371 |
| current_balance | 210.0 | -1.089076e-16 | 1.002389 | -1.650501 | -0.828682 | -0.237628 | 0.794595 | 2.367533 |
| credit_limit | 210.0 | -2.994298e-16 | 1.002389 | -1.668209 | -0.834907 | -0.057335 | 0.804496 | 2.055112 |
| min_payment_amt | 210.0 | 1.512018e-16 | 1.002389 | -1.966425 | -0.761698 | -0.065915 | 0.718559 | 2.938945 |
| max_spent_in_single_shopping | 210.0 | -1.935489e-15 | 1.002389 | -1.813288 | -0.740495 | -0.377459 | 0.956394 | 2.328998 |

*Table 4 - Data after Scaling*

Z-Score ensure the feature distribution mean as 0 and standard deviation as 1, and above table displays the same information.

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

**Hierarchical Clustering Techniques**

Agglomerative –

Agglomerative clustering uses bottom-up technique in grouping. Based on increasing distance between pairs it would identify similar clusters.

FCluster –

Form flat clusters from the hierarchical clustering defined by the given linkage matrix.

**Method Used**

Dendrogram –

The graphical structure helps in identifying number of optimal clusters based on distance between similar group/pairs.

Linkage –

Criterion defines which distance/algorithm to be used to merge the pairs. We are using ward linkage that minimizes the variance of the clusters being merged.

Affinity –

Metric used to compute the linkage, in fcluster cutting we don't need to specify it but for 'ward' linkage the affinity would be always 'Euclidian'

### Dendrogram visualization –

The dendrogram can be hard to read when the original observation matrix from which the linkage is derived is large. Truncation is used to condense the dendrogram.

We are truncating the visual from 20 clusters to top.

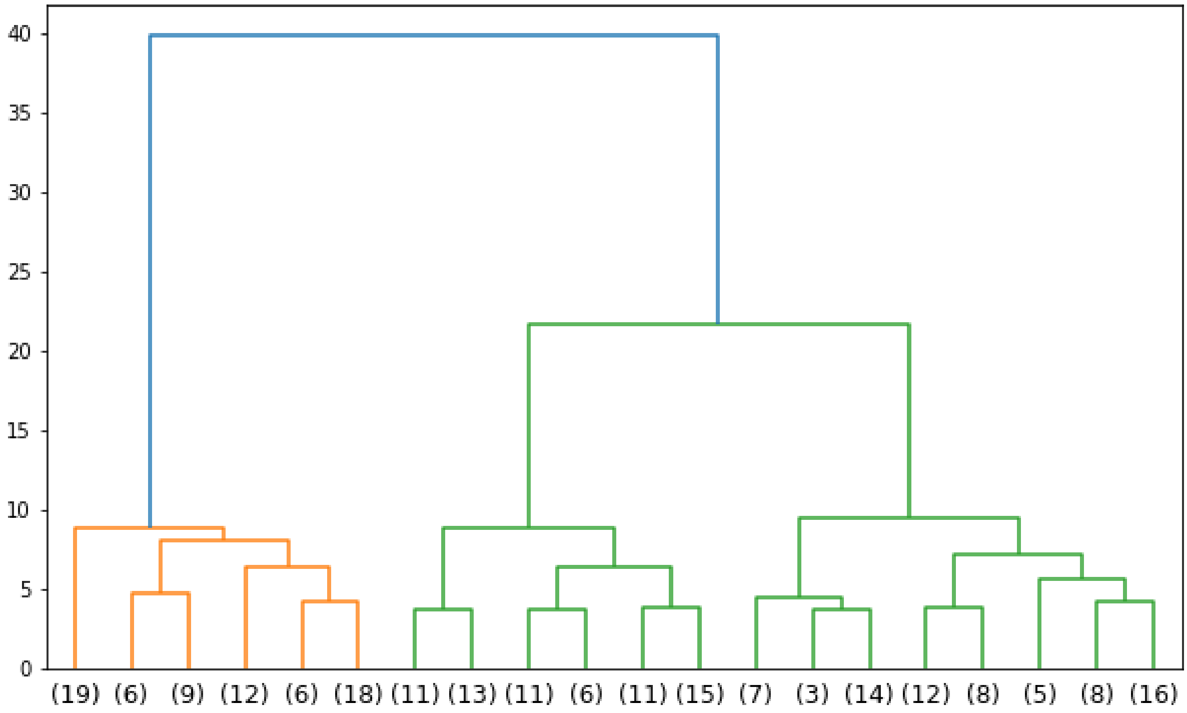


*Figure X - Dendrogram for Data*

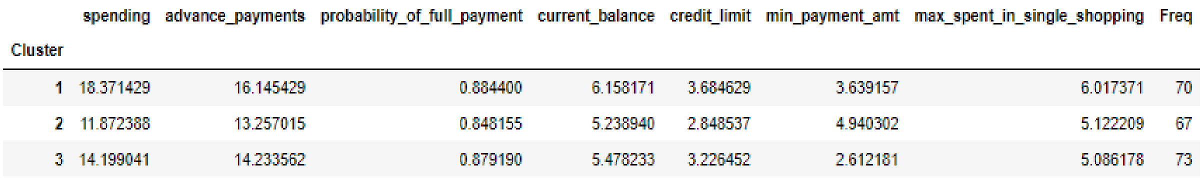
With above visual, we can observe that distance above 10 would not give much similarity between pairs, we can cut the dendrogram above 10 (distance) and consider optimal number of clusters as 3 (can identify from the figure)

## Identify clusters –

Since we know the optimal number of clusters (i.e. 3), we can cut our data into 3 clusters using fclusters.

Parameters – We are using ward linkage with maxclust criteria (which would limit the clusters to 3 only)

## Customer Profiling (Hierarchical) –



| | spending | advance_payments | probability_of_full_payment | current_balance | credit_limit | min_payment_amt | max_spent_in_single_shopping | Freq |
|---|---|---|---|---|---|---|---|---|
| Cluster | | | | | | | | |
| 1 | 18.371429 | 16.145429 | 0.884400 | 6.158171 | 3.684629 | 3.639157 | 6.017371 | 70 |
| 2 | 11.872388 | 13.257015 | 0.848155 | 5.238940 | 2.848537 | 4.940302 | 5.122209 | 67 |
| 3 | 14.199041 | 14.233562 | 0.879190 | 5.478233 | 3.226452 | 2.612181 | 5.086178 | 73 |

*Table 5 - Customer Profiling (Hierarchical Clustering)*

- The clusters formed have 70, 67 and 73 items respectively.
- Cluster#1 is privileged customers category having high average spending and high credit limit. The spent in one time purchase is also high.
- Cluster#2 have minimum average spending and probability of full payment is also lesser compared to other 2 profiles.
- Cluster#3 are the mid-range customers, having average spending lesser than rich category but higher than the other group and their current balance left in account is also not very high or less.

## 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.

K-Means clustering uses predefined number of clusters in which data frame should be cut into, this technique is widely used because of guaranteed convergence.

WSS (Within Sum of Squares) is a solution under the K-Means algorithm which helps to decide the value of K (number of clusters)

For a range of clusters (say 1-15), we will calculate the inertia (WSS) after clusters were formed, and the difference between inertia for 2 consecutive number of clusters would not be significant, we will take the lesser one as our optimal number of clusters (K).

In WSS plot, this can be identified by an elbow in most cases.

### WSS Plot -

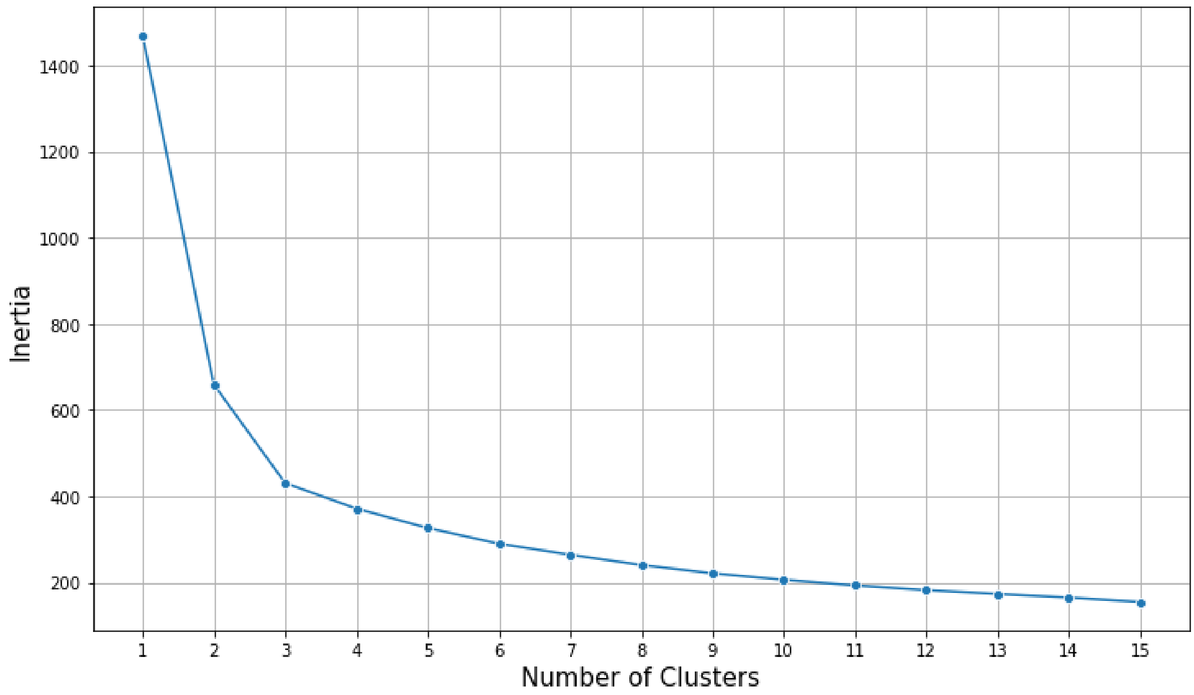


*Figure XI - WSS Plot (K-Means)*

With above figure, we can infer that difference between inertia of K=3 and K=4 is not that significant and seem to form an elbow from K=3.

Hence, we will take 3 as our optimal number of clusters.

## Silhouette Score –

Silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1. 1: Means clusters are well apart from each other and clearly distinguished.

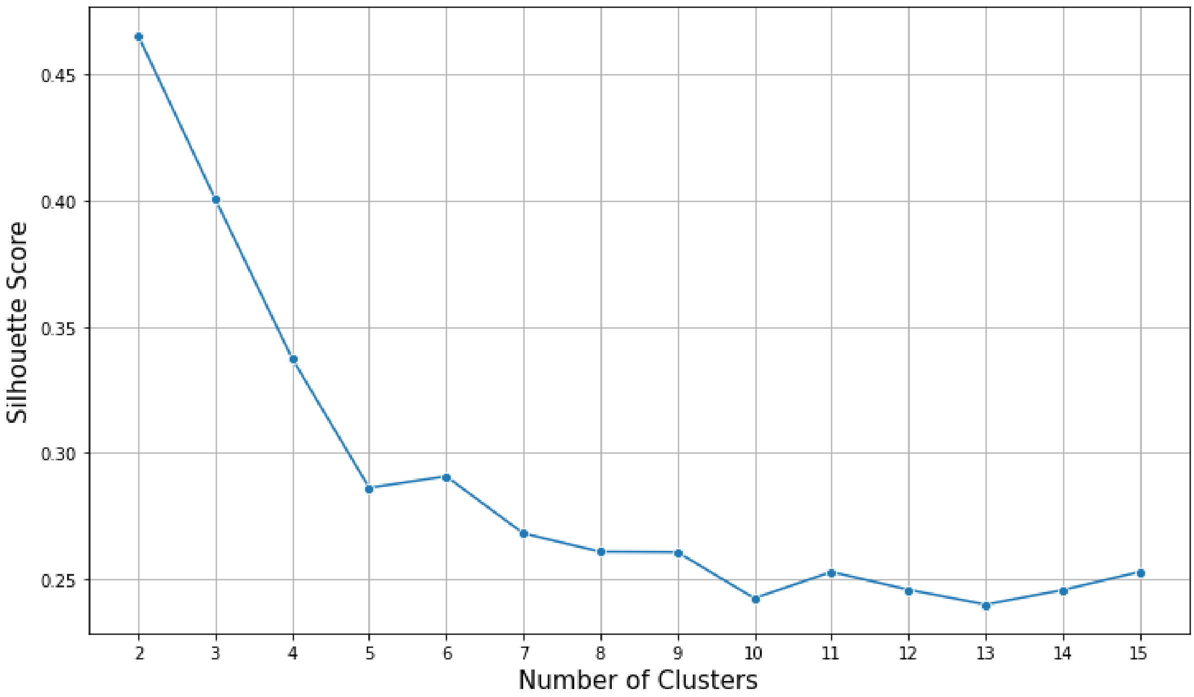


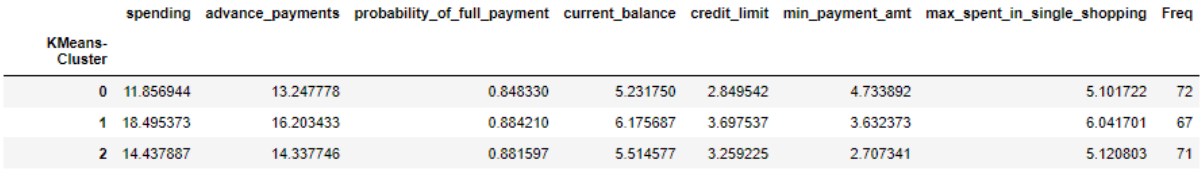
*Figure XII - Silhouette Score (K-Means)*

It is evident by above graph that silhouette score is better for 3 clusters instead of 4, so we will profile the customers in 3 different clusters.

**We want more than 2 clusters to make sense of different profiles, hence ignoring K=2

- Silhouette Score for K=3 is: 0.40

## Customer Profiling (K-Means) –



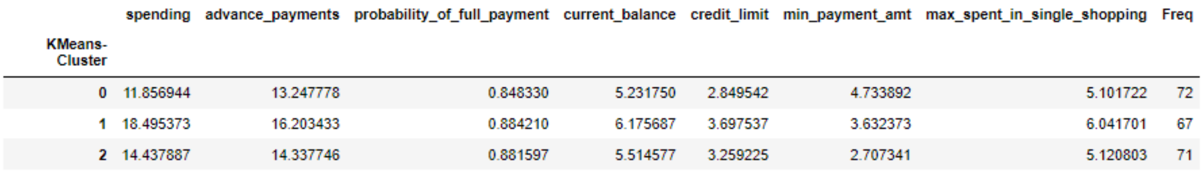
| KMeans-Cluster | spending | advance_payments | probability_of_full_payment | current_balance | credit_limit | min_payment_amt | max_spent_in_single_shopping | Freq |
|---|---|---|---|---|---|---|---|---|
| 0 | 11.856944 | 13.247778 | 0.848330 | 5.231750 | 2.849542 | 4.733892 | 5.101722 | 72 |
| 1 | 18.495373 | 16.203433 | 0.884210 | 6.175687 | 3.697537 | 3.632373 | 6.041701 | 67 |
| 2 | 14.437887 | 14.337746 | 0.881597 | 5.514577 | 3.259225 | 2.707341 | 5.120803 | 71 |

*Table 6 - Customer Profiling (K-Means)*

- The clusters formed have 72, 67 and 71 items respectively.
- Cluster#1 is privileged customers category having high average spending and high credit limit. The spent in one time purchase is also high.
- Cluster#0 have minimum average spending and probability of full payment is also lesser compared to other 2 profiles.
- Cluster#2 are the mid-range customers, having average spending lesser than rich category but higher than the other group and their current balance left in account is also not very high or less.

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

### Profiling –



| | spending | advance_payments | probability_of_full_payment | current_balance | credit_limit | min_payment_amt | max_spent_in_single_shopping | Freq |
|---|---|---|---|---|---|---|---|---|
| KMeans-Cluster | | | | | | | | |
| 0 | 11.856944 | 13.247778 | 0.848330 | 5.231750 | 2.849542 | 4.733892 | 5.101722 | 72 |
| 1 | 18.495373 | 16.203433 | 0.884210 | 6.175687 | 3.697537 | 3.632373 | 6.041701 | 67 |
| 2 | 14.437887 | 14.337746 | 0.881597 | 5.514577 | 3.259225 | 2.707341 | 5.120803 | 71 |

*Table 7 - Custer Profiling*

The profiling looked same for hierarchical and K-Means clustering with a difference of only few records grouped into different clusters.

We have taken K-Means cluster for profiling, grouped the data and displayed average values of features among the group.

By above table, we can infer that –

- Cluster#0 (Group 1): Low-range category of customers
    - The customers in this category have less average spending and their balance left in account for purchase is also lesser compared to other groups
    - They have less credit limit, and their probability of full payment is also slightly less.
    - We have 72 customers' records in this group
- Cluster#1(Group 2): High-range category of customers
    - This category has rich/privileged customers, they have high average spending, and their average spent is more on one purchase
    - Balance left in their account for purchase is high compared to others
    - We have 67 customers in this profile
- Cluster#2 (Group 3): Mid-range category of customers
    - We have 71 customers in mid-range category
    - Their spending is average compared to other groups, so it their advance payments.
    - Their remaining balance for shopping is high

### Recommendations –

- ✓ Bank can offer discount on luxury brands to **Group 2** customers to build their habit of spending.
- ✓ Can increase credit limit of **Group 2** customers, so that they can shop more
- ✓ Bank can cut down interest rate on **Group 3** customers to build their habit of spending.
- ✓ Can offer loyalty cards or reward points on consecutive purchases for **Group 3** customers.
- ✓ For **Group 1** customers, bank can offer reward points on utility purchases (like groceries, electrical items etc.)
- ✓ Bank can remind **Group 1** customers frequently and offer discount on early payments.

# CART-RF-ANN

## Executive Summary

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.

## Introduction

Purpose of our exercise would be to use CART, RF & ANN and compare the models' performances in train and test sets.

## 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 in days)
- Destination of the tour (Destination)
- Amount worth of sales per customer in procuring tour insurance policies in rupees (in 100's)
- The commission received for tour insurance firm (Commission is in percentage of sales)
- Age of Insured (Age)

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

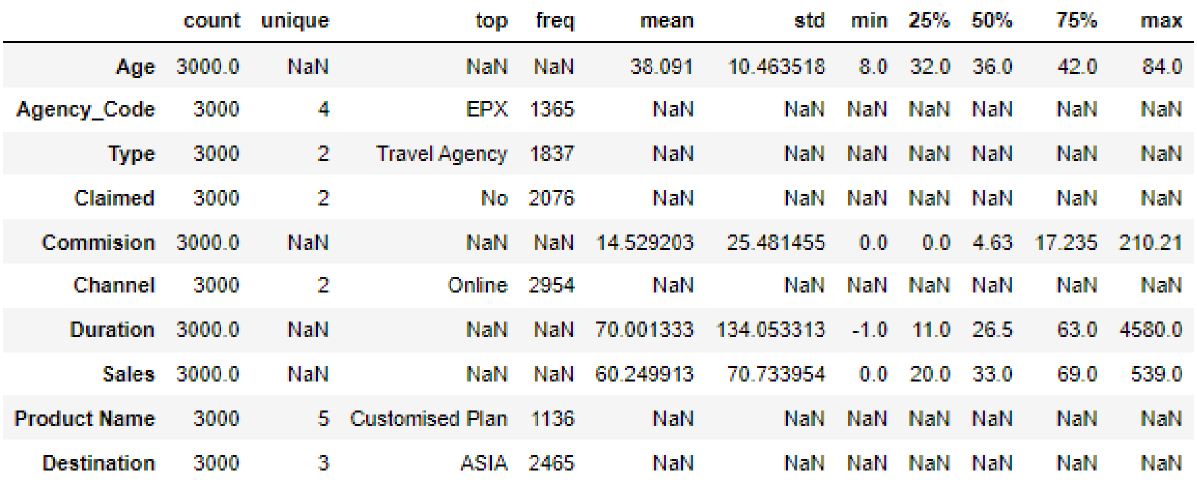
**Basic Data Information –**

| Column | Non-Null Count | Dtype |
|---|---|---|
| Age | 3000 non-null | int64 |
| Agency_Code | 3000 non-null | object |
| Type | 3000 non-null | object |
| Claimed | 3000 non-null | object |
| Commision | 3000 non-null | float64 |
| Channel | 3000 non-null | object |
| Duration | 3000 non-null | int64 |

| | | |
|---|---|---|
| Sales | 3000 non-null | float64 |
| Product Name | 3000 non-null | object |
| Destination | 3000 non-null | object |

*Table 8 - Basic Data Information (Insurance Data)*

- We have 3000 rows and 10 fields of data
- There is no null value in the data frame
- We have 4 fields in numerical type and rest 6 are in object
- There are 139 duplicated records, but there is no unique identifier of customer, we may have those records for different customers



| | 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 9 - General Stats about Insurance Data*

By above table, we can infer that –

- The average age of customers insured is 38, and we have oldest customer of age 84
- The data has 4 different agencies and EPX is the most frequent firm offering insurance
- We have 2 type of tour agencies, and most of the agencies are type of Travel Agency
- Claimed (Target variable) has 2 values either Yes or No
- Average percentage commission on insurance is 14.5% with a highest of 210%
- The mode of insurance (channel) are 2 types and most of the customers were insured using online channel
- Average duration of tour is 70 days with a highest of 4580 days.
- Average sales of insurance are 6k
- There are 5 different plans agencies are offering, and most of the customers opt for customised plan
- Customers are travelling to one of the 3 different destination and most of them travelling to Asia

## Data Visualization

- There is 4 numeric variable and 6 categorical variables, we will analyse numerical variables with distribution and box plot, and for categorical variables we will use count plot for further analysis.

## *Univariate Analysis:*

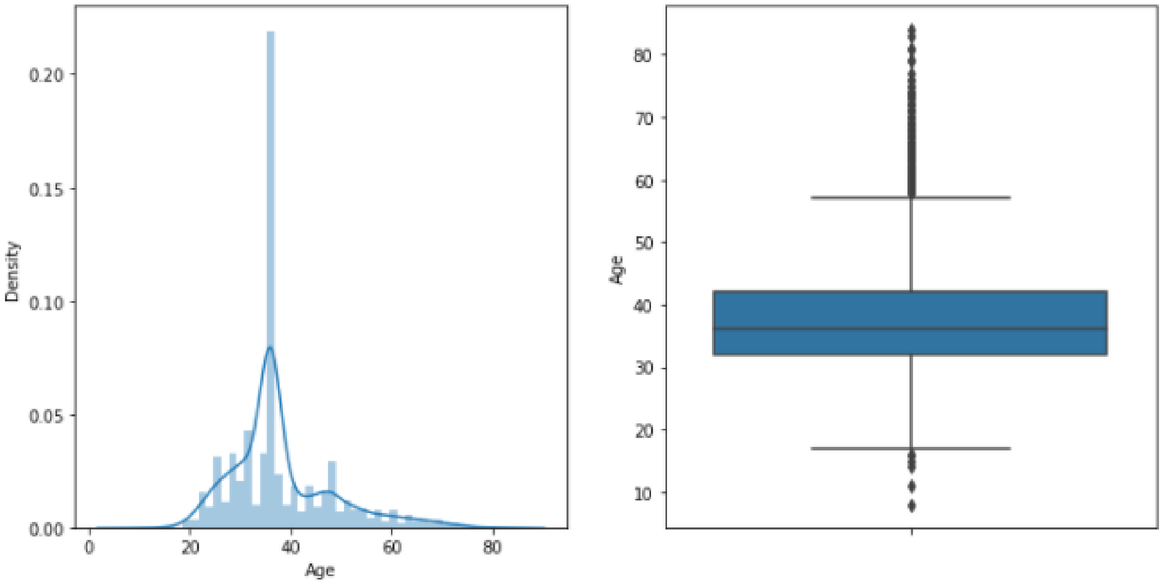


*Figure XIII - Univariate Analysis (Age - Numerical)*

**Age** data have 204 outliers in it (6.8% of total data) and distribution seem positively skewed (value - 1.149). The valid Age values will be 1-100, and we have all data in range of 8 – 84.

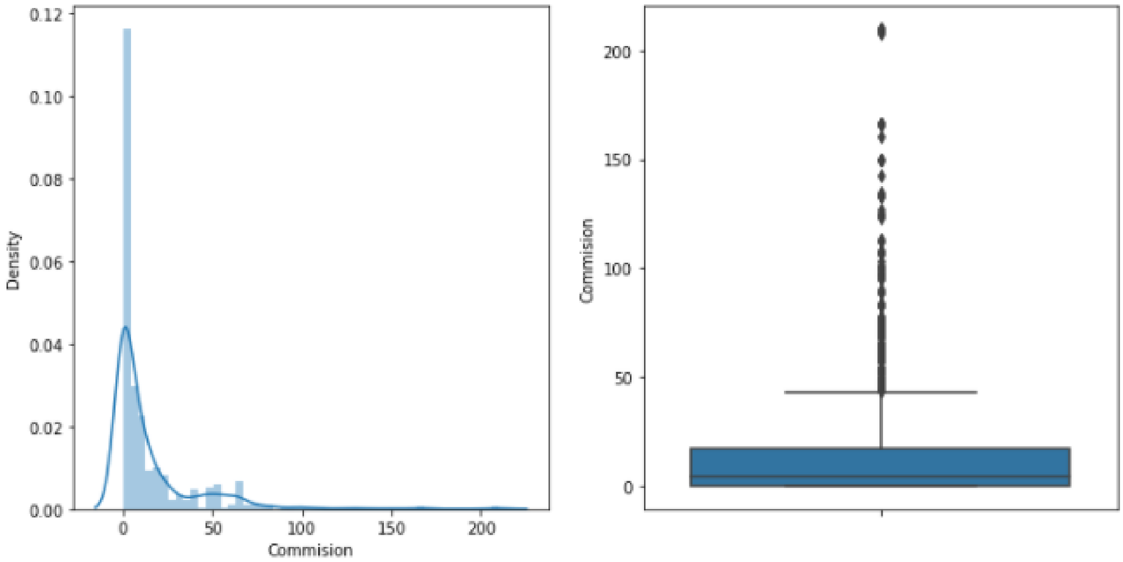


*Figure XIV - Univariate Analysis (Commission - Numerical)*

**Commission** data have 362 outliers (12.06% of total data) and distribution seems well right/positively skewed with a value of 3.14

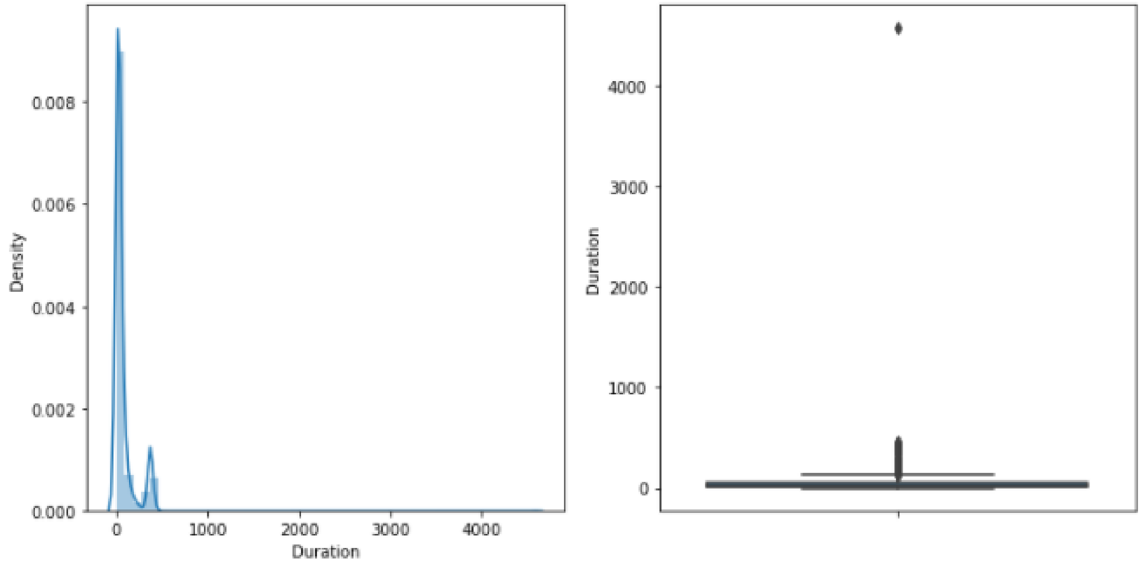


*Figure XV - Univariate Analysis (Duration - Numerical)*

**Duration** data have 382 outliers (12.73% of total data) and one with extreme value >4000. The distribution is very much positively skewed (value – 13.77)

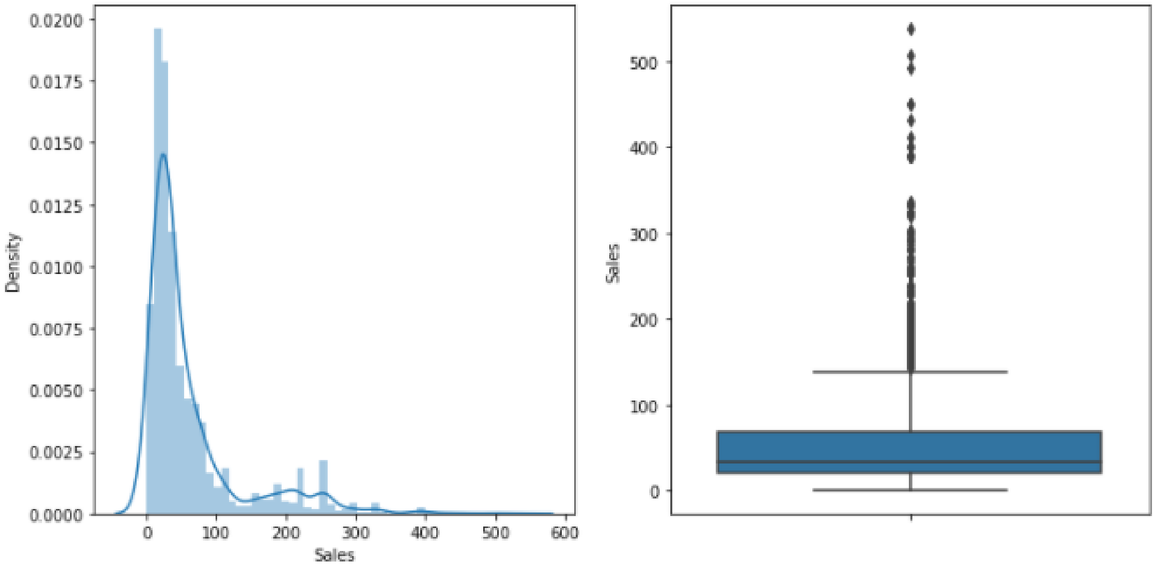


*Figure XVI - Univariate Analysis (Sales - Numerical)*

**Sales** has 353 outliers (11.76% of total data) and the data distribution is rightly skewed with a value 2.37

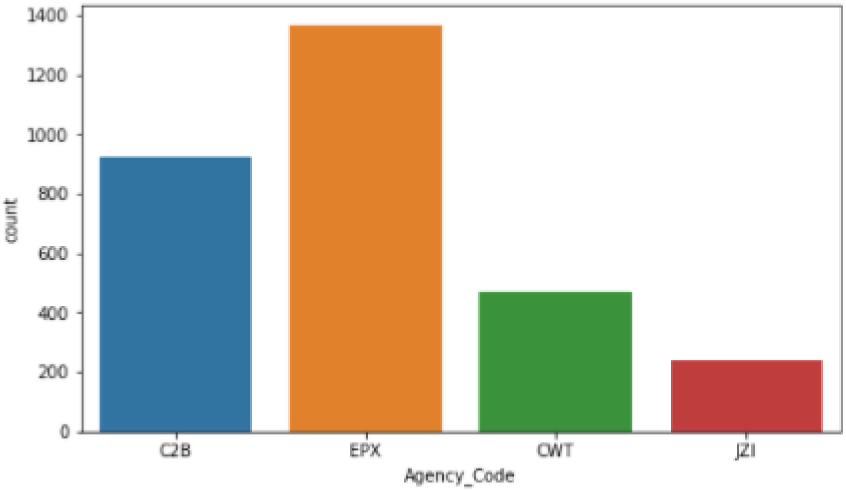


*Figure XVII - Univariate Analysis (Agency Code - Categorical)*

EPX is the most opted tour agency for insurance, the frequency of each agency is –

| | |
|---|---|
| EPX: | 1365 |
| C2B: | 924 |
| CWT: | 472 |
| JZI: | 239 |

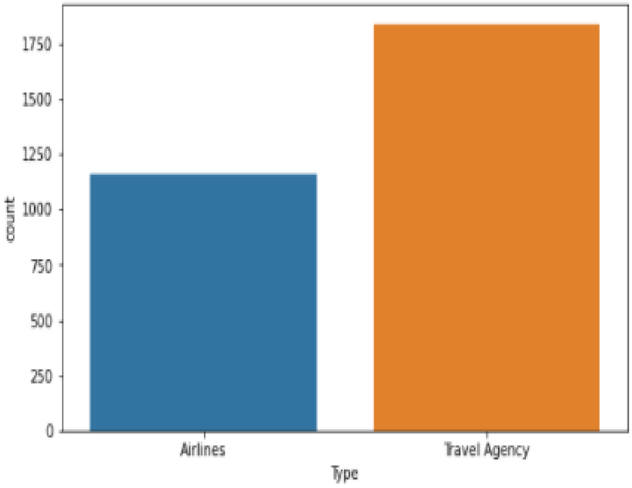


*Figure XVIII - Univariate Analysis (Agency Type - Categorical)*

People insured more through travel agency type; the frequency of agency type is –

| | |
|---|---|
| Travel Agency: | 1837 |
| Airlines: | 1163 |

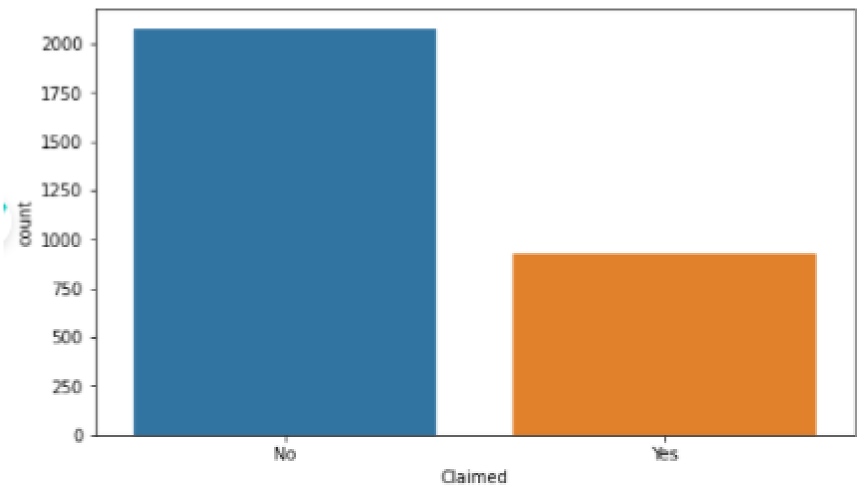


*Figure XIX - Univariate Analysis (Claimed - Categorical)*

**Claimed** status would tell how if customer claimed insurance, we have 2076 customers not claimed on insurance and 924 claimed.

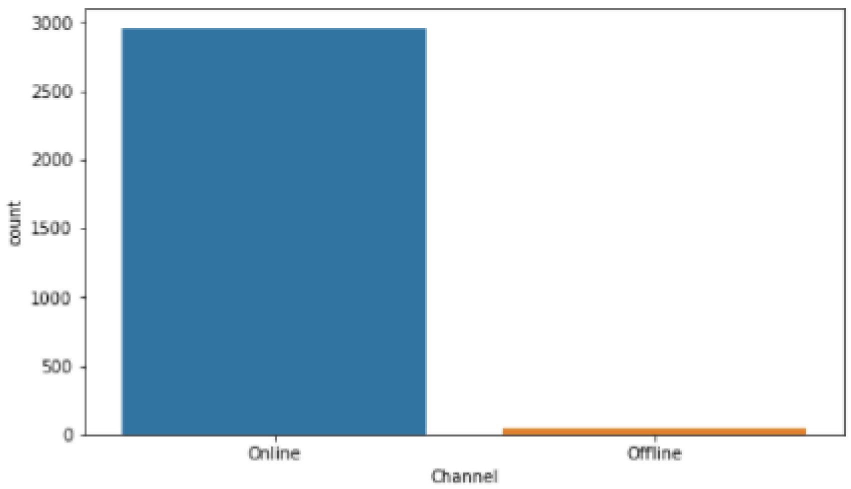


*Figure XX - Univariate Analysis (Channel - Categorical)*

Most of the customers purchased insurance with online mode (2954) and there are few from offline (only 46).

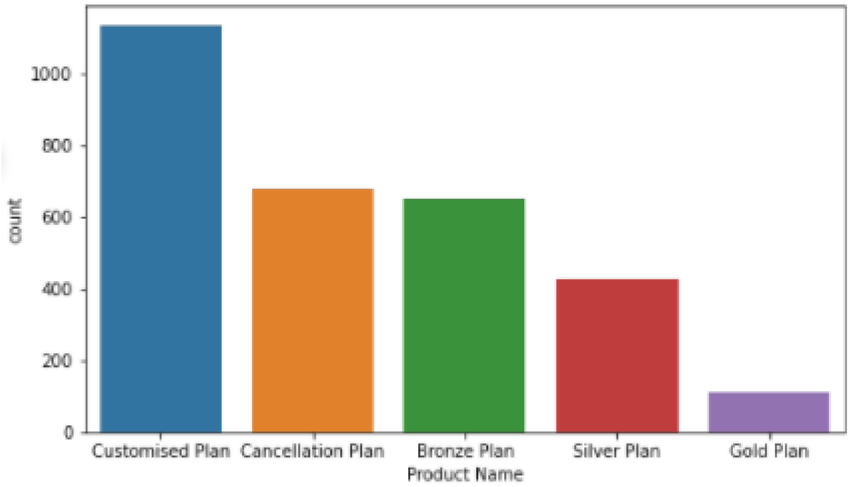


*Figure XXI - Univariate Analysis (Product Name - Categorical)*

Customers mostly opt for customised plan followed by cancellation plan. The frequencies of their choice are –

| | |
|---|---|
| Customised Plan: | 1136 |
| Cancellation Plan: | 678 |
| Bronze Plan: | 650 |
| Silver Plan: | 427 |
| Gold Plan: | 109 |

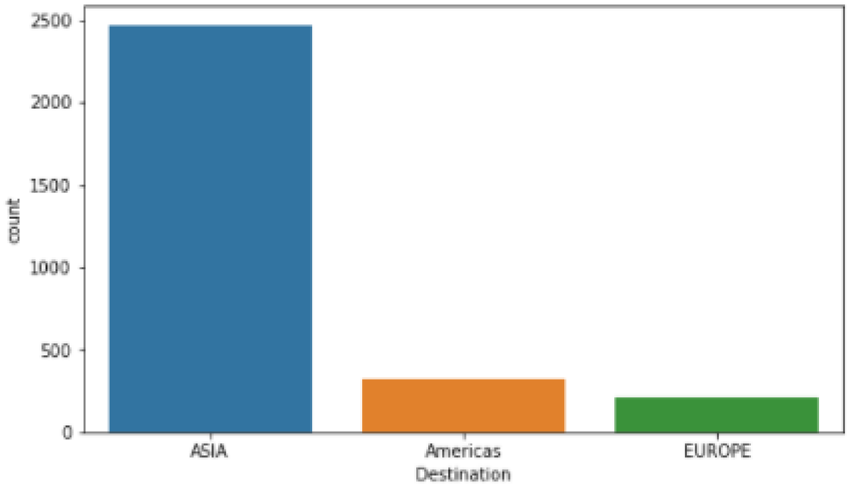


*Figure XXII - Univariate Analysis (Destination - Categorical)*

Most of the customers have destination Asia (2465), followed by America (320) then Europe (215).

### *Bi-Variate Analysis:*

**Pair Plot:**

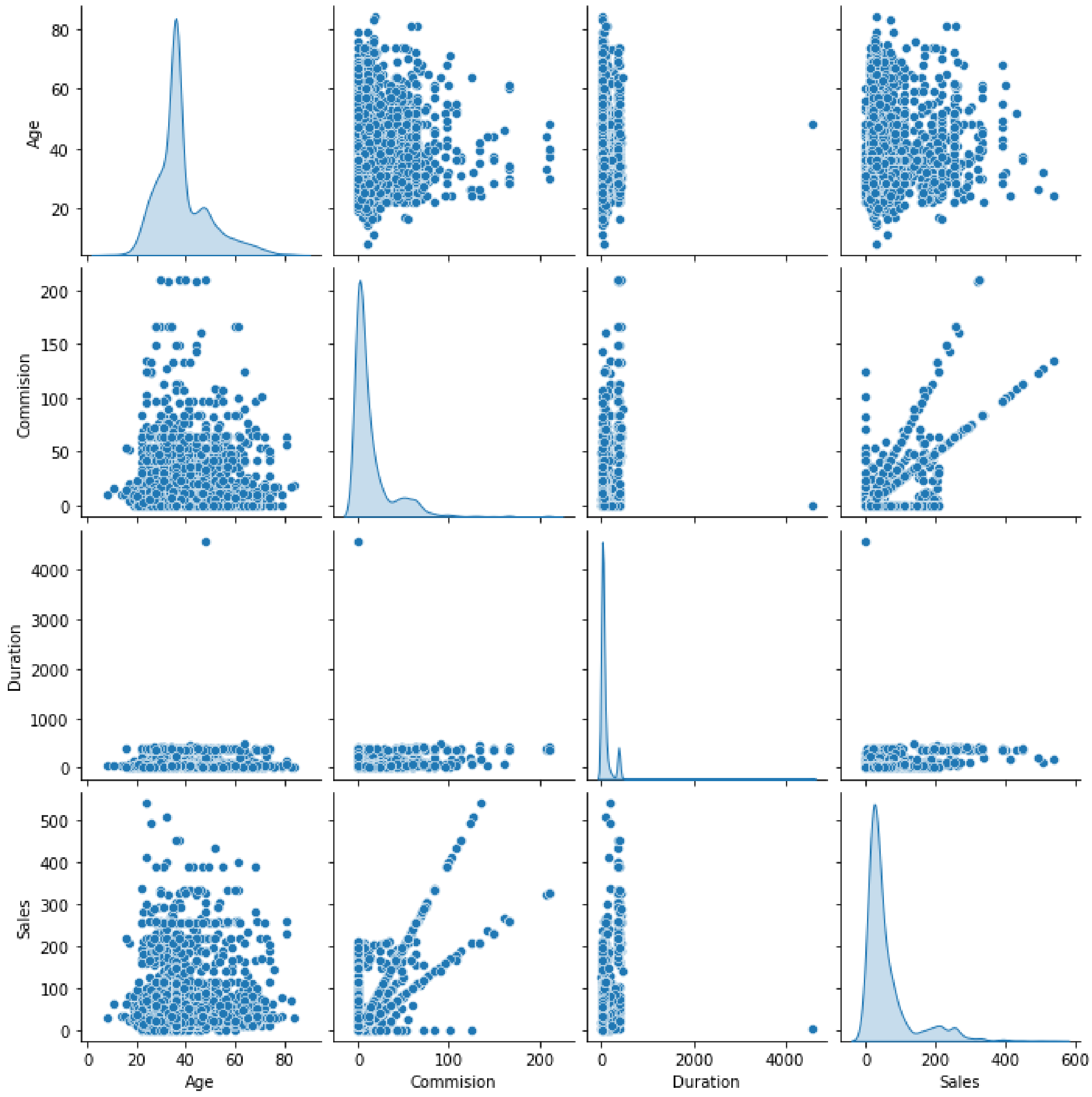


*Figure XXIII - Pair Plot (Numerical Data Points)*

By above figure, we can infer that –

- Commission seems increasing as Sales increases (with couple of exceptions)
- Duration looks within a range but has a record with extreme value > 4000
- Commission rate is high for youths (for older customers commission rate doesn't seem high)

**Correlation Metrix (Heat Map):**

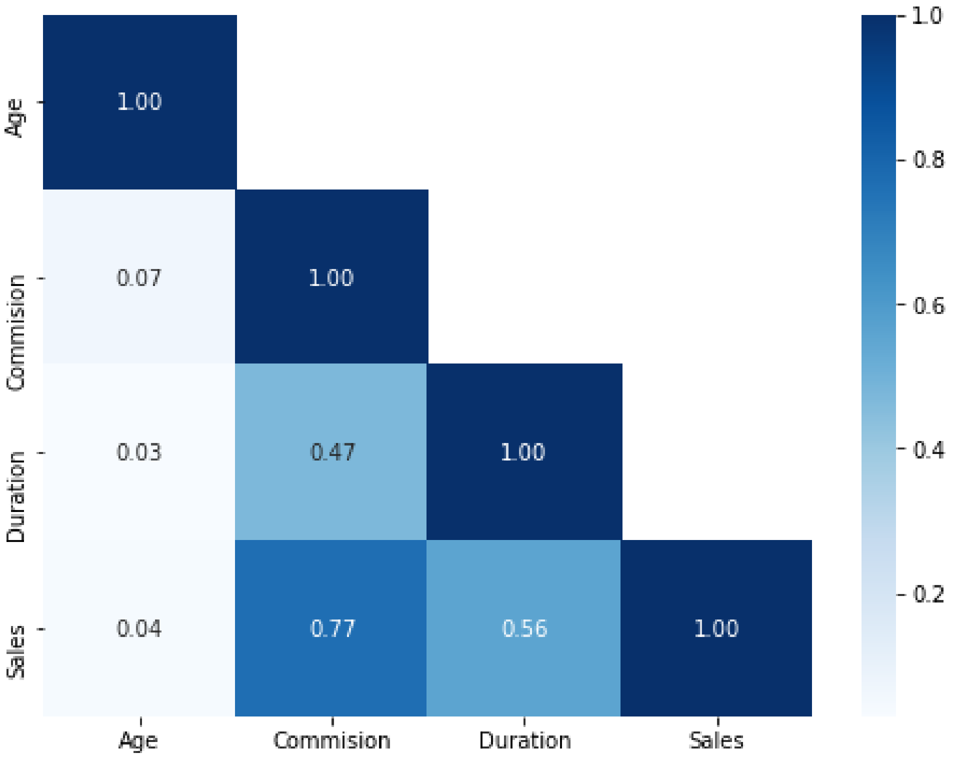


*Figure XXIV - Correlation Matrix*

- There is significant correlation between sales and commission as inferred by pair plot as well
- Age is not correlated with any of other parameter like Sales, Duration, or commission.

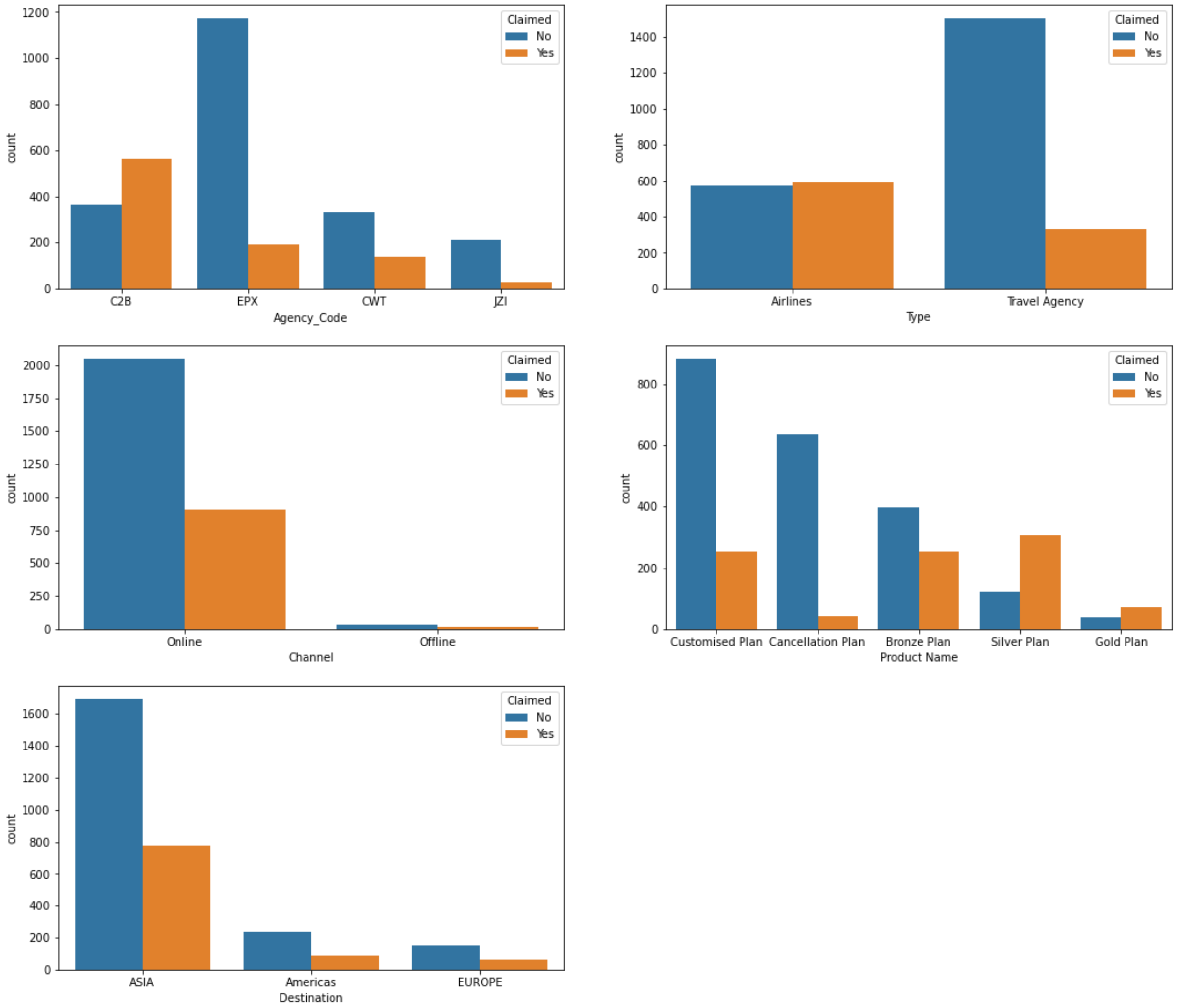


*Figure XXV - Categorical with Claimed*

Graphical representation against claimed status helps us deriving some useful insights out of the data as –

- Claimed rate on C28 agency is high compared to others
- People are claiming more on insurance covered by airline type of agencies
- Customers claim more on Bronze plan of insurance

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

**Categorical to code conversion** –

Feature: Agency_Code

['C2B' 'EPX' 'CWT' 'JZI'] → [0 2 1 3]

Feature: Type

['Airlines' 'Travel gency'] → [0 1]

Feature: Claimed

['No' 'Yes'] → [0 1]

Feature: Channel

['Online' 'Offline'] → [1 0]

Feature: Product Name

['Customised Plan' 'Cancellation Plan' 'Bronze Plan' 'Silver Plan' 'Gold Plan'] → [2 1 0 4 3]

Feature: Destination

['ASIA' 'Americas' 'EUROPE'] → [0 1 2]

**Proportion of Target Variable –**

We have 69.2% no-claim data and 30.8% claim data

## Splitting Train vs. Test Data (For all models)–

We are splitting data in 70 – 30%, our train data will be 70% and test will be 30%

Proportion of Yes/No in Train vs. Test data –

Train data (2100 records) => 68.7% (No) and 31.2% (Yes)

Test Data (900 records) => 70.2% (No) and 29.7% (Yes)

The data doesn't seem bias in any of the bucket, so we are good with building models on top of it.

** Random State will be taken as **123** at all places

## Decision Tree Classifier

We are building the model based on **gini** criterion, and with help of grid search fetching best parameters for pruned decision tree.

Grid Search Param –

Criterion – Gini

Max Depth – 10, 20 or 30 (Depth of the tree)

Min Samples Leaf - 50, 100 or 150 (Minimum samples available in leaf nodes)

Max Samples Split – 150, 300 or 450 (Maximum samples in a split)

Since we have 2100 records in Train data, tree depth from 10 to 30 with leaf node samples 50 to 150 and max samples in split from 150 to 450 items would be good to cater all records evenly.

After fitting the Train data, we have fetched out with best parameters as –

Max Depth – 10

Min Samples Leaf – 50

And max samples split – 150

And build the **Decision Tree model** with the same best parameters…

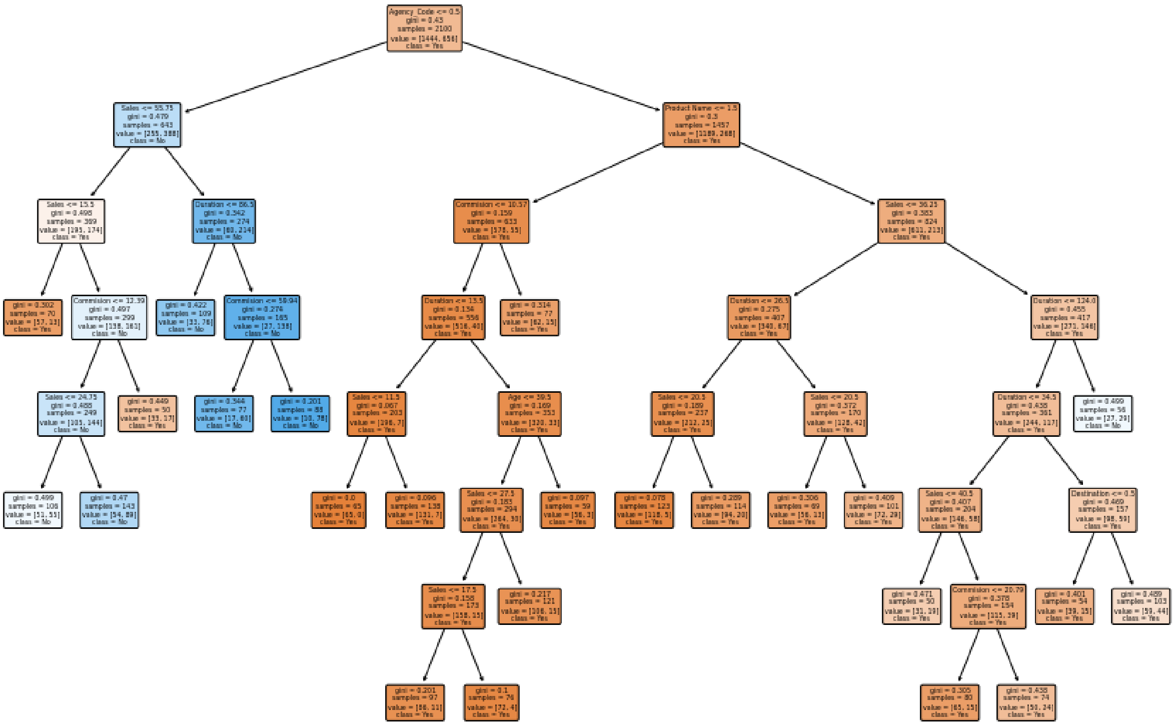

Agency_Code <= 0.5 gini = 0.43 samples = 2100 value = [1444, 656] class = Yes
Sales <= 55.75 gini = 0.479 samples = 643 value = [255, 388] class = No
Product Name <= 1.5 gini = 0.3 samples = 1457 value = [1189, 268] class = Yes
Sales <= 15.5 gini = 0.498 samples = 369 value = [195, 174] class = Yes
Duration <= 86.5 gini = 0.342 samples = 274 value = [60, 214] class = No
Commision <= 10.57 gini = 0.159 samples = 633 value = [578, 55] class = Yes
Sales <= 36.25 gini = 0.383 samples = 824 value = [611, 213] class = Yes
gini = 0.302 samples = 70 value = [57, 13] class = Yes
Commision <= 12.39 gini = 0.497 samples = 299 value = [138, 161] class = No
gini = 0.422 samples = 109 value = [33, 76] class = No
Commision <= 59.94 gini = 0.274 samples = 165 value = [27, 138] class = No
Duration <= 13.5 gini = 0.134 samples = 556 value = [516, 40] class = Yes
gini = 0.314 samples = 77 value = [62, 15] class = Yes
Duration <= 26.5 gini = 0.275 samples = 407 value = [340, 67] class = Yes
Duration <= 124.0 gini = 0.455 samples = 417 value = [271, 146] class = Yes
Sales <= 24.75 gini = 0.488 samples = 249 value = [105, 144] class = No
gini = 0.449 samples = 50 value = [33, 17] class = Yes
gini = 0.344 samples = 77 value = [17, 60] class = No
gini = 0.201 samples = 88 value = [10, 78] class = No
Sales <= 11.5 gini = 0.067 samples = 203 value = [196, 7] class = Yes
Age <= 39.5 gini = 0.169 samples = 353 value = [320, 33] class = Yes
Sales <= 20.5 gini = 0.189 samples = 237 value = [212, 25] class = Yes
Sales <= 20.5 gini = 0.372 samples = 170 value = [128, 42] class = Yes
Duration <= 34.5 gini = 0.438 samples = 361 value = [244, 117] class = Yes
gini = 0.499 samples = 56 value = [27, 29] class = No
gini = 0.499 samples = 108 value = [51, 57] class = No
gini = 0.47 samples = 143 value = [54, 89] class = No
gini = 0.0 samples = 65 value = [65, 0] class = Yes
gini = 0.096 samples = 138 value = [131, 7] class = Yes
Sales <= 27.5 gini = 0.183 samples = 294 value = [264, 30] class = Yes
gini = 0.097 samples = 59 value = [56, 3] class = Yes
gini = 0.078 samples = 123 value = [118, 5] class = Yes
gini = 0.289 samples = 114 value = [94, 20] class = Yes
gini = 0.306 samples = 69 value = [56, 13] class = Yes
gini = 0.409 samples = 101 value = [72, 29] class = Yes
Sales <= 40.5 gini = 0.407 samples = 204 value = [146, 58] class = Yes
Destination <= 0.5 gini = 0.469 samples = 157 value = [98, 59] class = Yes
Sales <= 17.5 gini = 0.158 samples = 173 value = [158, 15] class = Yes
gini = 0.217 samples = 121 value = [106, 15] class = Yes
gini = 0.471 samples = 50 value = [31, 19] class = Yes
Commision <= 20.79 gini = 0.378 samples = 154 value = [115, 39] class = Yes
gini = 0.401 samples = 54 value = [39, 15] class = Yes
gini = 0.489 samples = 103 value = [59, 44] class = Yes
gini = 0.201 samples = 97 value = [86, 11] class = Yes
gini = 0.1 samples = 76 value = [72, 4] class = Yes
gini = 0.305 samples = 80 value = [65, 15] class = Yes
gini = 0.438 samples = 74 value = [50, 24] class = Yes

*Figure XXVI - Decision Tree Model Visual*

**Feature importance –**

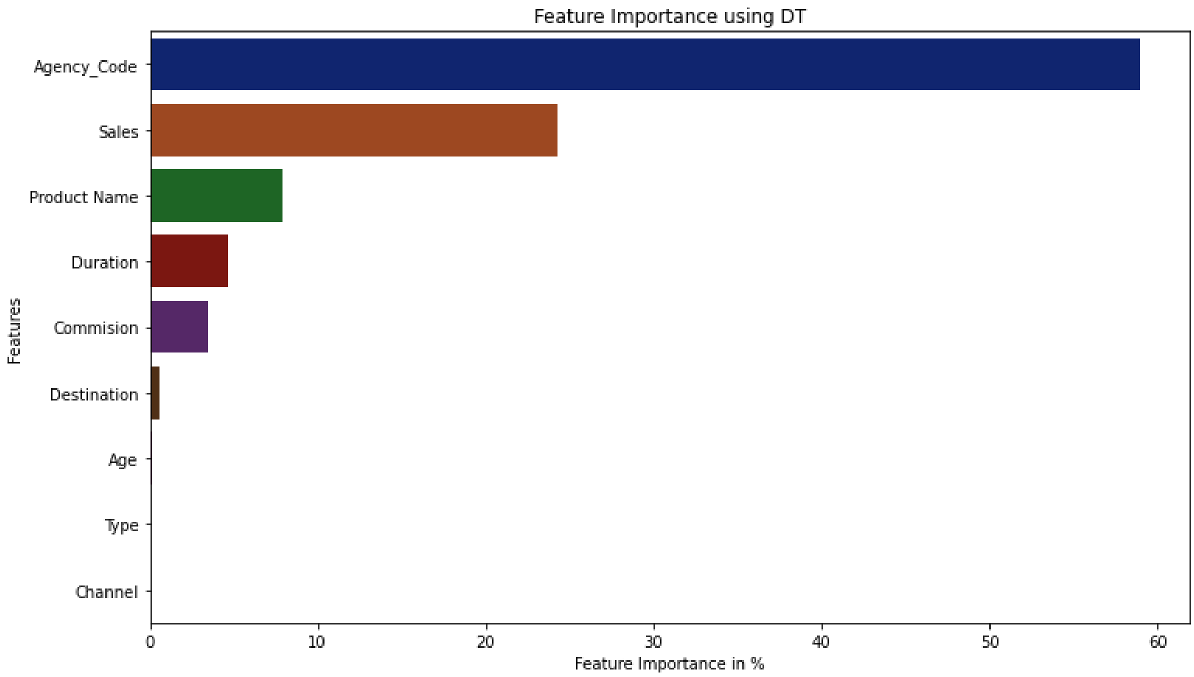
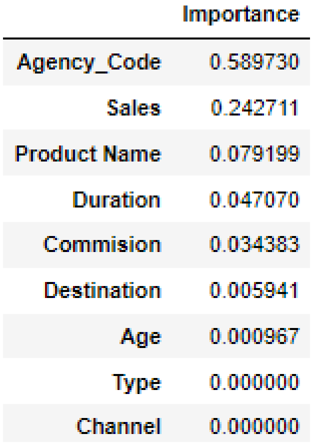


*Figure XXVII - Feature Importance (DT)*



| | Importance |
|---|---|
| Agency_Code | 0.589730 |
| Sales | 0.242711 |
| Product Name | 0.079199 |
| Duration | 0.047070 |
| Commision | 0.034383 |
| Destination | 0.005941 |
| Age | 0.000967 |
| Type | 0.000000 |
| Channel | 0.000000 |

*Table 10 - Feature Importance (Decision tree)*

With above importance metric, we can see that Agency code has highest importance in deciding the nodes and Type & Channel doesn't play crucial role.

## Random Forest Classifier

For sub-trees in Random forest, we will use gini criterion by default. And search for best grid among below parameters in Grid Search –

Max depth of sub-tree: 6, 8, 10 or 12 (With limited data 6-12 depth tree should be able to make decision at good accuracy)

Max Features in a sub-tree: 5, 6 or 7 (Since we have 9 features in total)

Min Samples Leaf: 50, 75, 100 or 125

Min Samples Split: 30, 50 or 70 (Minimum items requires for new split)

N Estimators: 100, 200 or 300 (# of Trees should be formed in the forest, 100-300 trees are significantly good to predict target variable)

After fitting the Train data, we have fetched out with best parameters as –

Max Depth – 6

Max Features – 5

Min Samples Leaf – 125

Min samples split – 30

N Estimators – 100

And build the **Random Forest model** with the same best parameters…

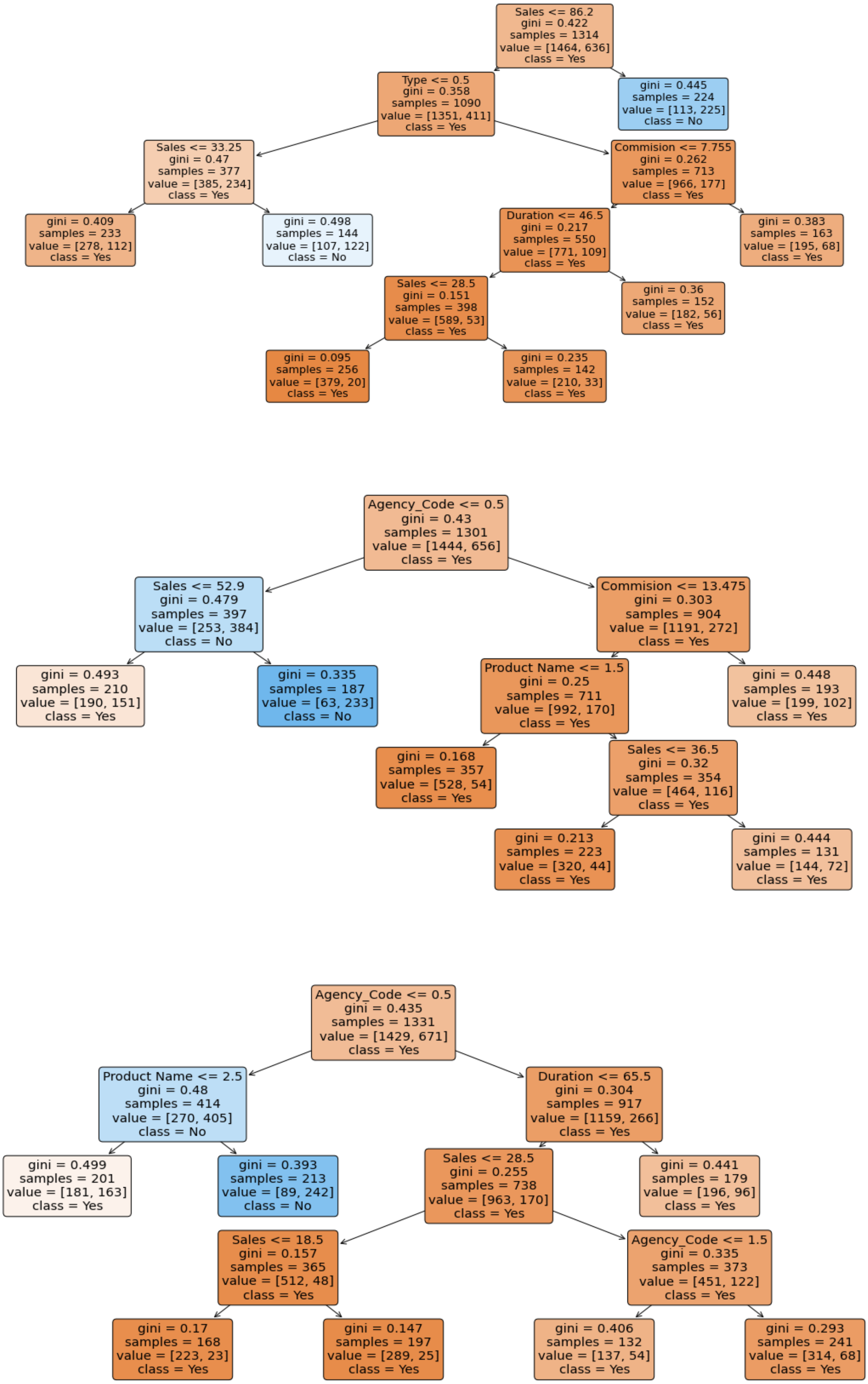


*Figure XXVIII - Decision Trees of Random Forest*

**Feature importance –**

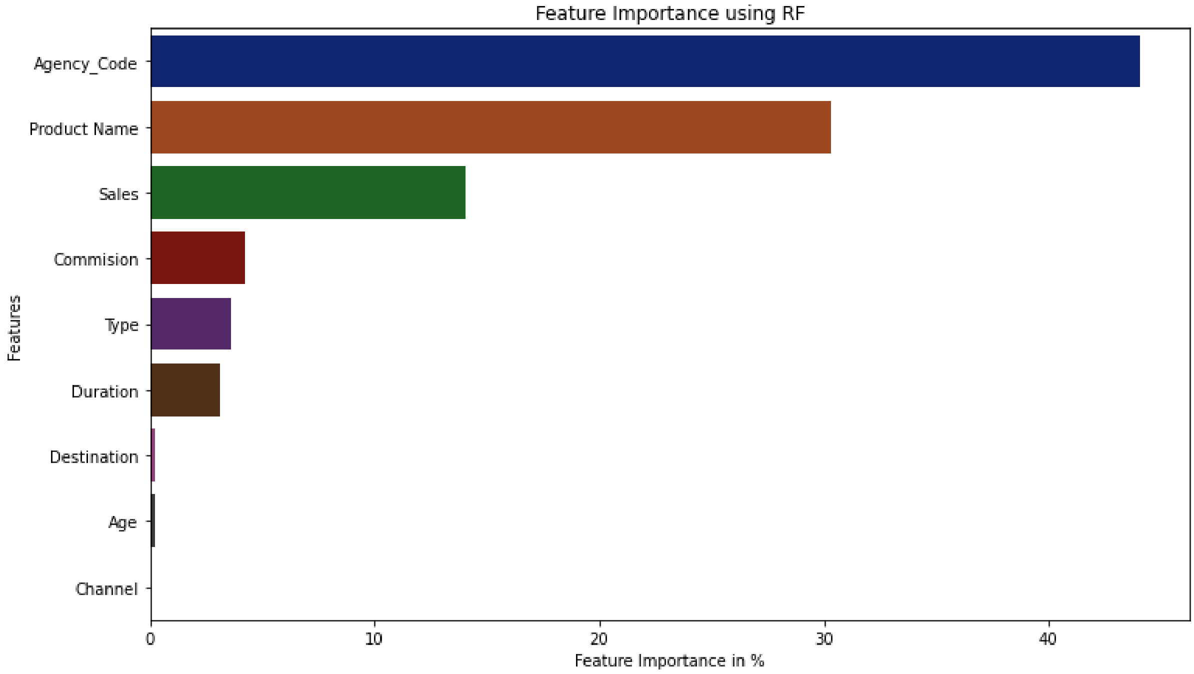
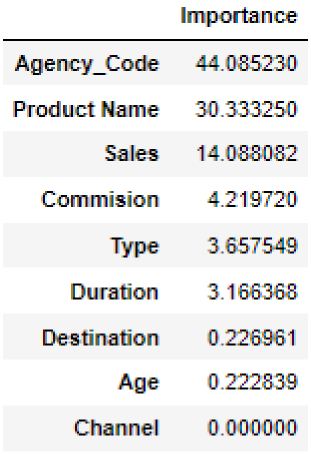


*Figure XXIX - Feature Importance (RF)*



| | Importance |
|---|---|
| Agency_Code | 44.085230 |
| Product Name | 30.333250 |
| Sales | 14.088082 |
| Commision | 4.219720 |
| Type | 3.657549 |
| Duration | 3.166368 |
| Destination | 0.226961 |
| Age | 0.222839 |
| Channel | 0.000000 |

*Table 11 - Feature Importance (Random Forest)*

As infer from above, Agency code feature has highest importance among decision trees followed by product name. The channel feature doesn’t play crucial role in deciding nodes for decision trees.

## MLP Classifier (Artificial Neural Network)

In artificial neural network works with assigning weightage to features and derive best weightage possible to minimize loss.

If we do not normalize our inputs the large values become dominating in ANN training.

Hence, we have scaled the train and test data.

In search of best grid for ANN, we have passed array of below parameters –

*Hidden Layer Sizes* – Taken 3 hidden layers for neurons 100 neurons in each

*Activation* – Selecting either logistic or relu

*Solver* – Out of sgd and adam, we will be selecting for best ANN model

*Tolerance (Tol)* – Decrease in loss after each adjustment, taking 0.01 along with 0.1 in case the ANN model works more efficiently with low tolerance

*Max Iterations* – To coverage the ANN model, how much iteration model should take.

Best parameters came out from Grid search –

Activation – Relu

Hidden layers sized – 3 layers of 100 neurons in each

Max iteration – 10000

Solver – Adam

Tolerance – 0.1

And build the **MLP Classifier ANN model** with the same best parameters…

The graphical representation of model would look like –

**Features (9) → HL1 (100 Neuron) → HL2(100 Neuron) → HL3 (100 Neuron) → Output**

## 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.

### Accuracy Score –

| Model | Train Data | Test Data |
|---|---|---|
| CART | 0.78 | 0.781 |
| RF | 0.772 | 0.784 |
| ANN | 0.78 | 0.791 |

*Table 12 - Accuracy Score of Models*

ANN model has best accuracy on test data (79.1%), and random forest model is least accurate for train data (77.2%). We can adjust the parameters further to increase accuracy of models.

## Confusion Matrix –

**Decision Tree:**

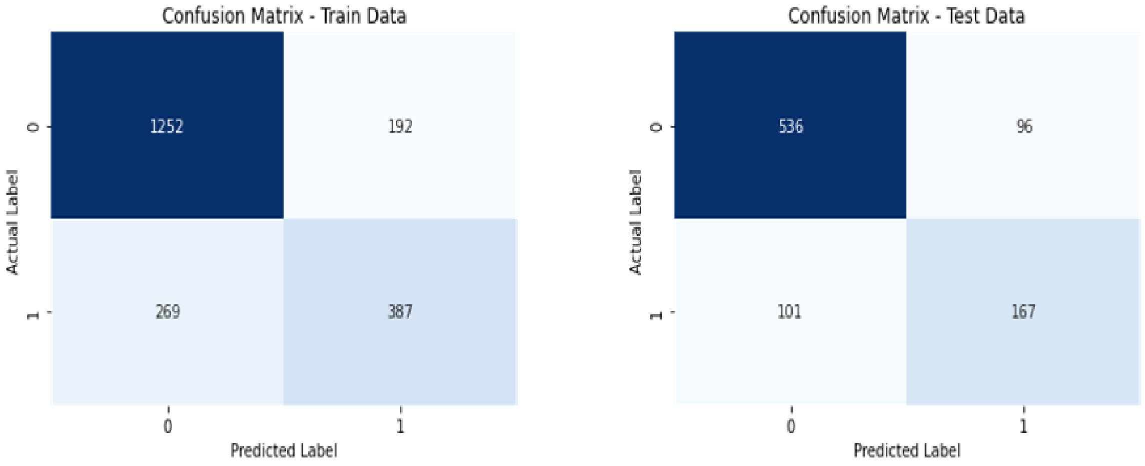


*Table 13 - Confusion Matrix (DT)*

CART Model (Decision Tree) correctly predicts 1639 records out of 2100 in train data (78% accuracy) and 703 out of 900 in test data (78.1% accuracy)

**Random Forest:**

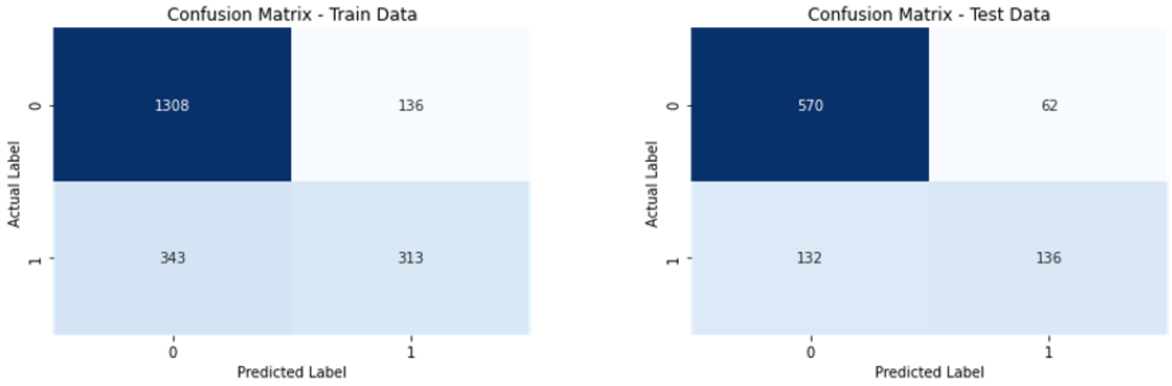


*Table 14 - Confusion Matrix (RF)*

RF Model correctly predicts 1621 records out of 2100 (77.2%) in train data and 706 out of 900 (78.4%) of test data.

**Artificial Neural Network (MLP Classifier):**

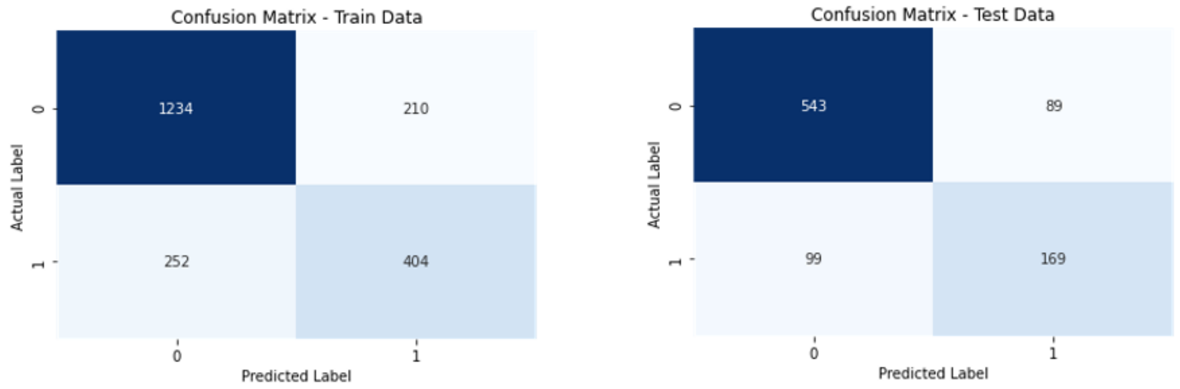


*Table 15 - Confusion Matrix (ANN)*

ANN model correctly predicts 1638 records out of 2100 (78%) in train data and 712 out of 900 (79.1%) in test data.

## ROC AUC Score and ROC Curve –

**Decision Tree:**

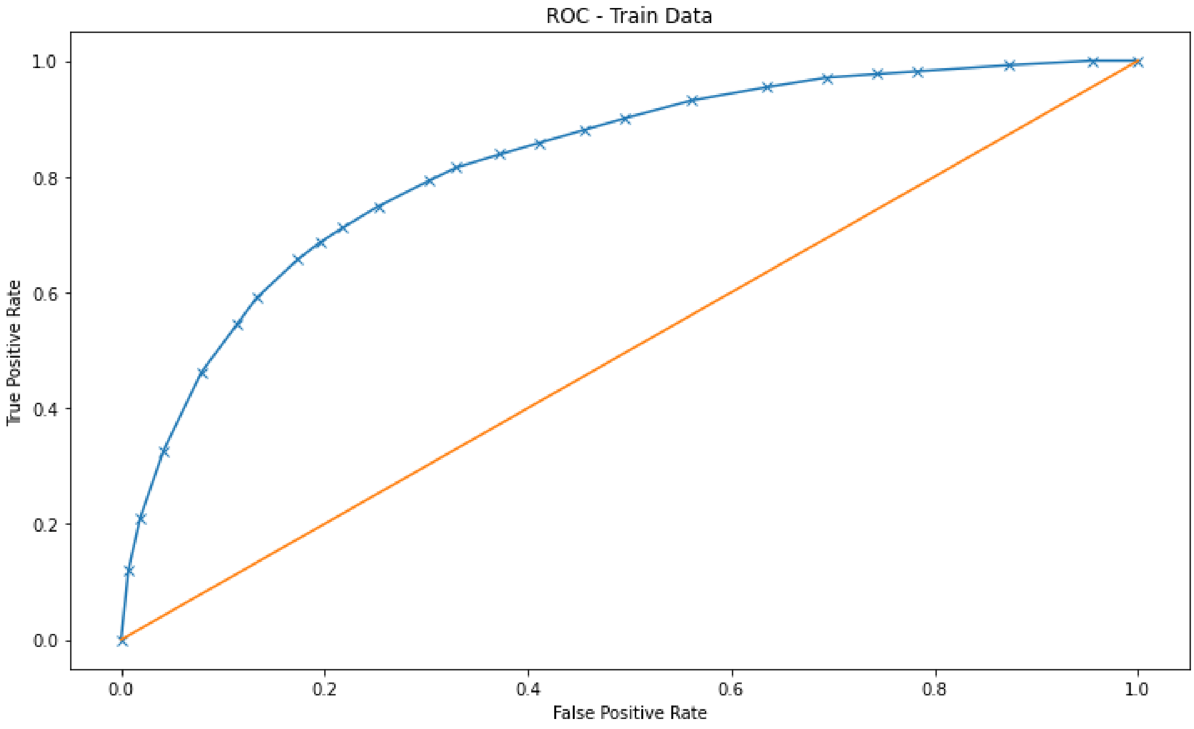


*Figure XXX - ROC Curve - Train Data (DT)*

TP Rate seems reaching to 0.8 with FP rate 0.35, the ***ROC AUC score (Area under the curve) is 0.824***

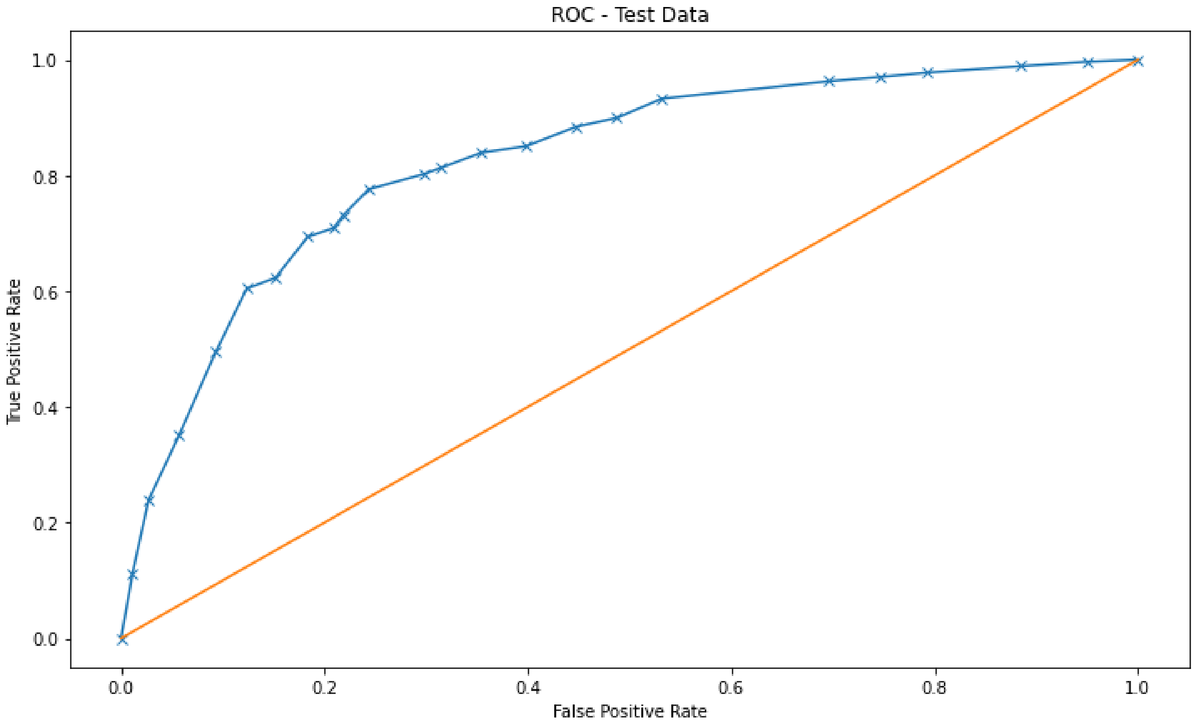


*Figure XXXI - ROC Curve - Test Data (DT)*

***Area under ROC Curve (ROC AUC Score) is 0.826***

**Random Forest:**

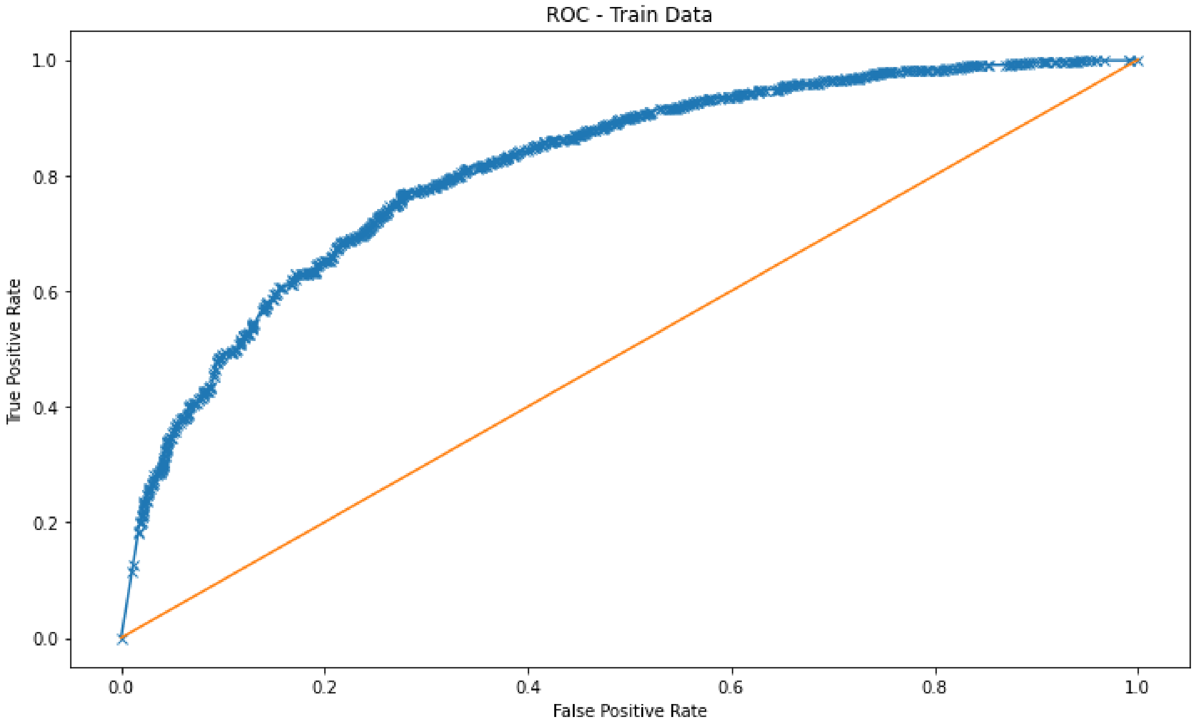


*Figure XXXII - ROC Curve - Train Data (RF)*

***ROC AUC Score – 0.812***

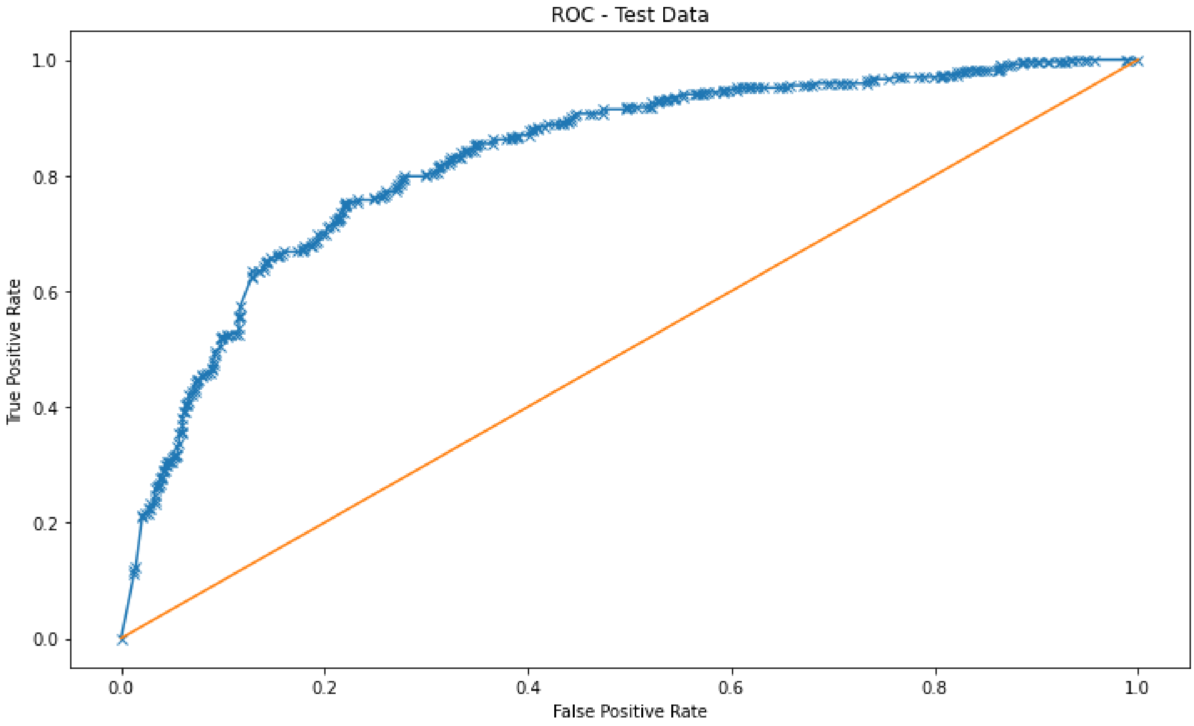


*Figure XXXIII - ROC Curve - Test Data (RF)*

***ROC AUC Score – 0.828***

**Artificial Neural Network:**

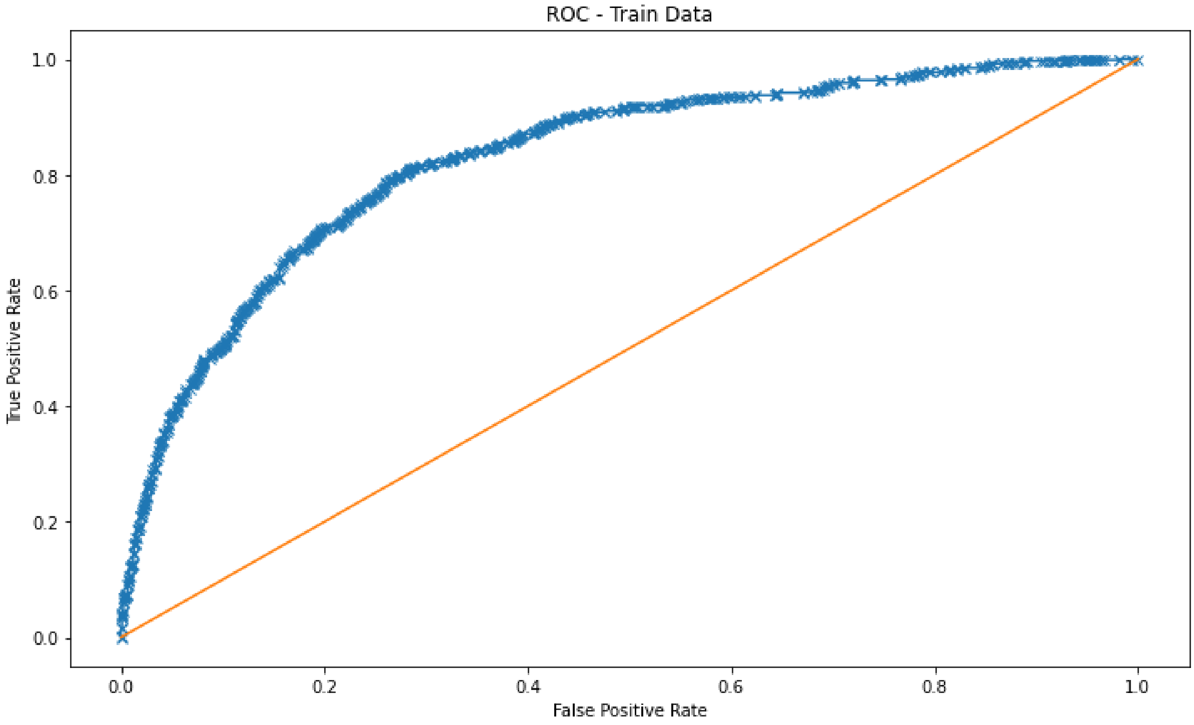


*Figure XXXIV - ROC Curve - Train Data (ANN)*

***ROC AUC Score – 0.828***

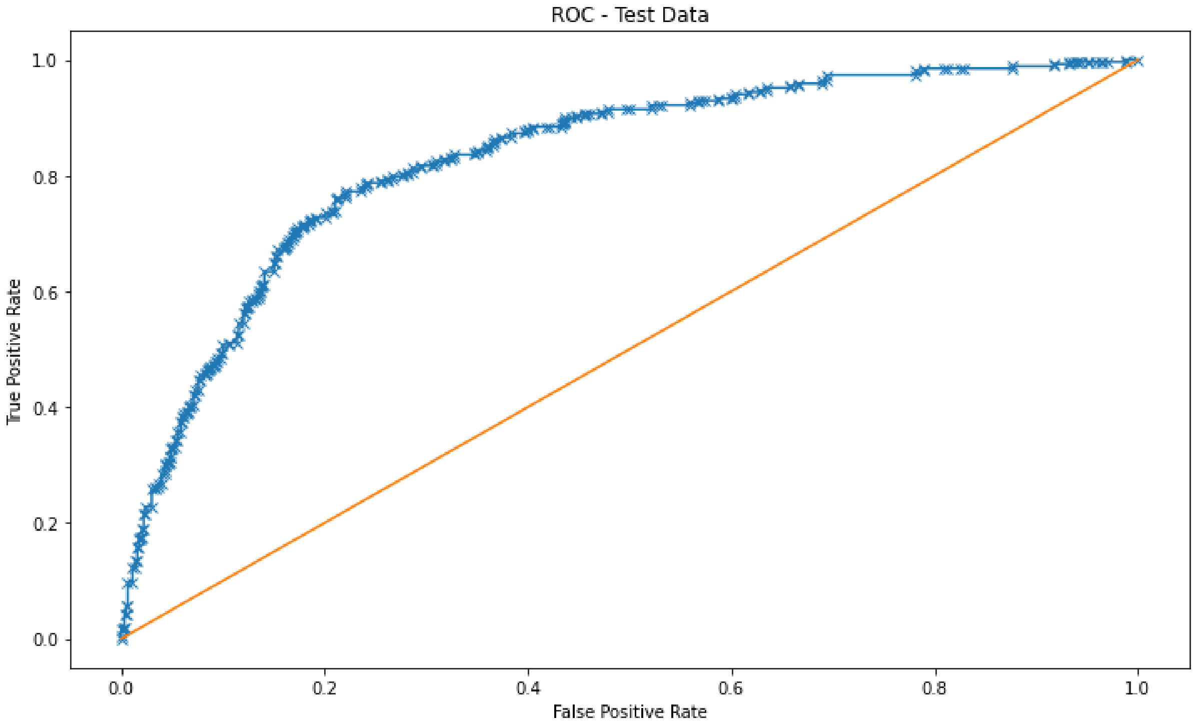
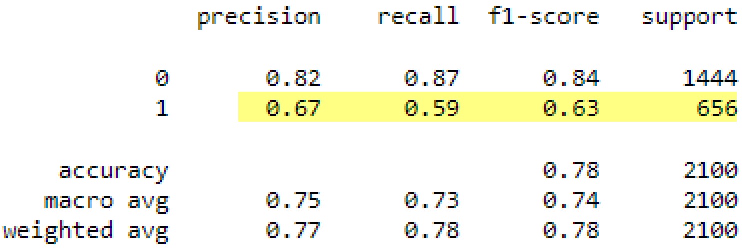


*Figure XXXV - ROC Curve - Test Data (ANN)*

***ROC AUC Score – 0.830***

## Classification Reports –

**Decision Tree:**



```
              precision    recall  f1-score   support

           0       0.82      0.87      0.84      1444
           1       0.67      0.59      0.63       656

    accuracy                           0.78      2100
   macro avg       0.75      0.73      0.74      2100
weighted avg       0.77      0.78      0.78      2100
```

*Table 16 - Classification Report - Train Data (DT)*

With above classification report, we can comment on positive as well as negative class identification.

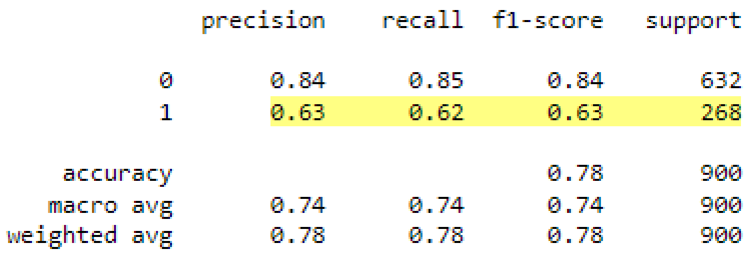
Positive class metrices –

***Precision*** (ratio of true positives to the sum of true and false positives) for DT model on train data is 0.67

***Recall*** (ratio of true positives to the sum of true positives and false negatives) for DT model on train data is 0.59

***F1-Score*** (The F1 is the weighted harmonic mean of precision and recall. The closer the value of the F1 score is to 1.0, the better the expected performance of the model is) – 0.63

***Support*** is the number of actual occurrences of the class in the dataset. It doesn't vary between models



```
              precision    recall  f1-score   support

           0       0.84      0.85      0.84       632
           1       0.63      0.62      0.63       268

    accuracy                           0.78       900
   macro avg       0.74      0.74      0.74       900
weighted avg       0.78      0.78      0.78       900
```

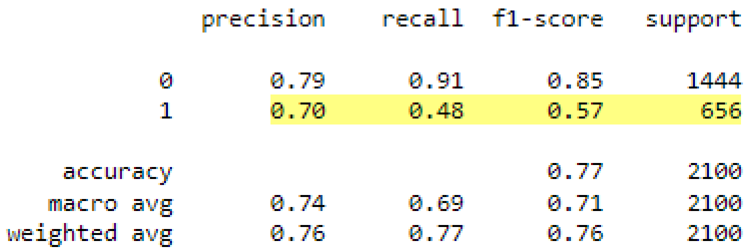
*Table 17 - Classification Report - Test Data (DT)*

On test data we have,

Precision – 0.63, Recall – 0.62 and F1-Score – 0.63

With accuracy around 0.78, train and test data show similar results. The CART model looks good and not overfitted.

**Random Forest:**



```
              precision    recall  f1-score   support

           0       0.79      0.91      0.85      1444
           1       0.70      0.48      0.57       656

    accuracy                           0.77      2100
   macro avg       0.74      0.69      0.71      2100
weighted avg       0.76      0.77      0.76      2100
```

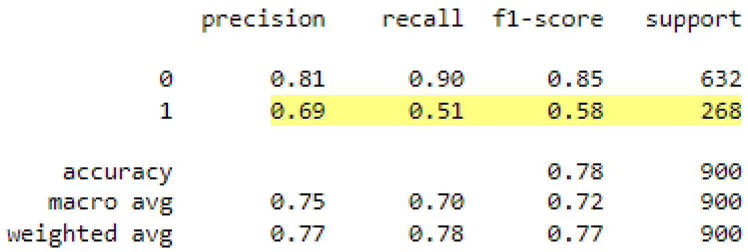
*Table 18 - Classification Report - Train Data (RF)*

Train data metrices for positive class –

Precision – 0.70

Recall – 0.48

F1-Score – 0.57



```
              precision    recall  f1-score   support

           0       0.81      0.90      0.85       632
           1       0.69      0.51      0.58       268

    accuracy                           0.78       900
   macro avg       0.75      0.70      0.72       900
weighted avg       0.77      0.78      0.77       900
```

*Table 19 - Classification Report - Test Data (RF)*

Test data metrices for positive class –

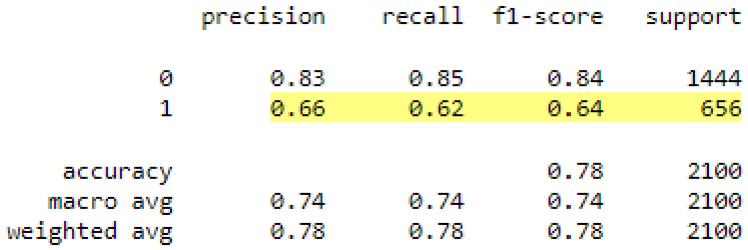
Precision – 0.69

Recall – 0.51

F1-Score – 0.58

With accuracy of 77-78% on train and test data, the RF model looks good and doesn't seem overfitted.

**Artificial Neural Network –**



```
              precision    recall  f1-score   support

           0       0.83      0.85      0.84      1444
           1       0.66      0.62      0.64       656

    accuracy                           0.78      2100
   macro avg       0.74      0.74      0.74      2100
weighted avg       0.78      0.78      0.78      2100
```

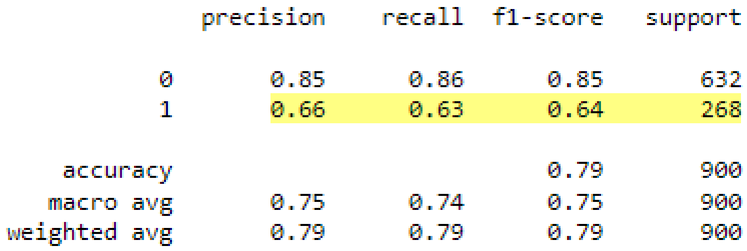
*Table 20 - Classification Report - Train Data (ANN)*

Train data metrices for positive class –

Precision – 0.66

Recall – 0.62

F1-Score – 0.64



```
              precision    recall  f1-score   support

           0       0.85      0.86      0.85       632
           1       0.66      0.63      0.64       268

    accuracy                           0.79       900
   macro avg       0.75      0.74      0.75       900
weighted avg       0.79      0.79      0.79       900
```

*Table 21 - Classification Report - Test Data (ANN)*

Test data metrices for positive class –

Precision – 0.66

Recall – 0.63

F1-Score – 0.64

With accuracy of 78-79% on train and test data, the ANN model looks good and doesn't seem overfitted.

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

| Metrices | CART | | Random Forest | | ANN | |
|---|---|---|---|---|---|---|
| | Train Data | Test Data | Train Data | Test Data | Train Data | Test Data |
| Accuracy | 0.78 | 0.781 | 0.772 | 0.784 | 0.78 | **0.791** |
| AUC Score | 0.824 | 0.826 | 0.812 | 0.828 | 0.828 | **0.83** |
| Recall | 0.59 | 0.62 | 0.48 | 0.51 | 0.62 | **0.63** |
| Precision | 0.67 | 0.63 | 0.7 | 0.69 | 0.66 | **0.66** |
| F1 Score | 0.63 | 0.63 | 0.57 | 0.58 | 0.64 | **0.64** |

*Table 22 - Value Comparison Among Models*

With help of above table, we can compare the metrices' value for each model –

- **ANN model looks better** in comparison to other 2, as it has better accuracy, AUC Score and recall, has a good precision and F1 Score is also high.
- All models have slightly better accuracy on test data.
- RF model has highest precision in comparison to other models.
- RF model has lowest recall compared to others, which makes its' F1-Score low
- CART and ANN has good AUC score
- ANN has similar metrices values among train and test data.

I will select **ANN model** for our problem as it has better accuracy, AUC score, Recall and F1 Score. For Precision RF model works slightly better but it gets penalized because of bad recall.

**ROC Curve Comparison –**

Train data:

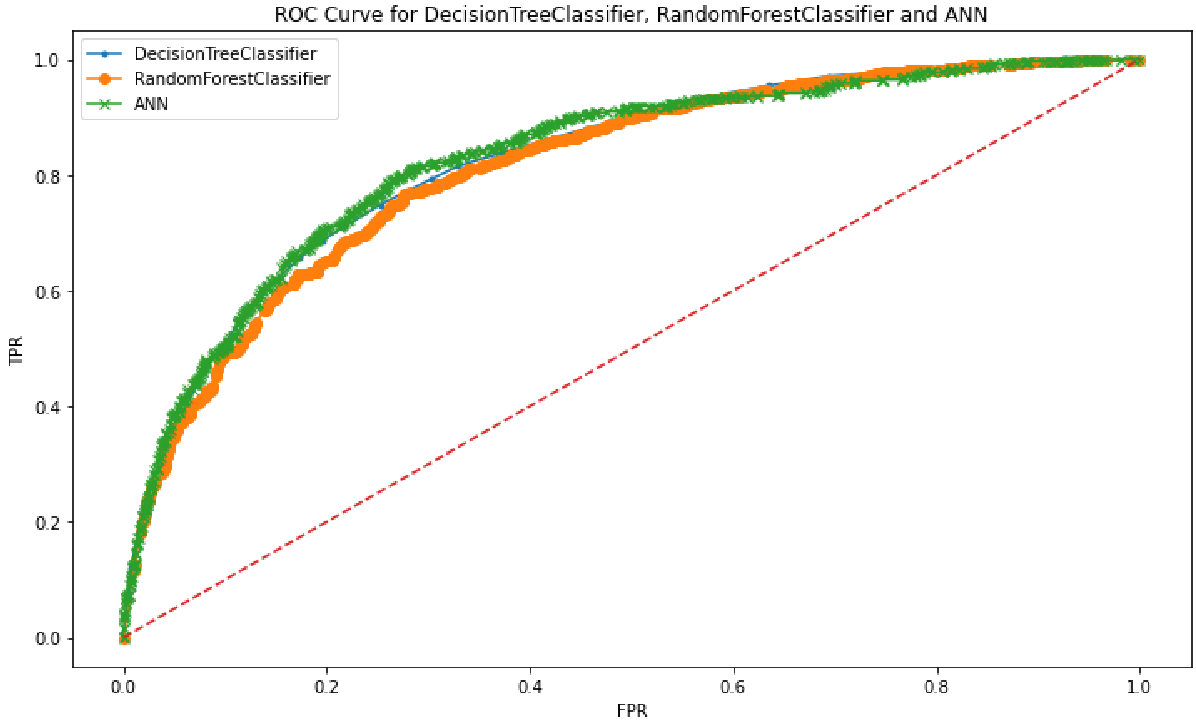


*Table 23 - Model Comparison in ROC Curve (Train Data)*

ANN seem to have better AUC, as it covers more area under the curve.

Test Data:

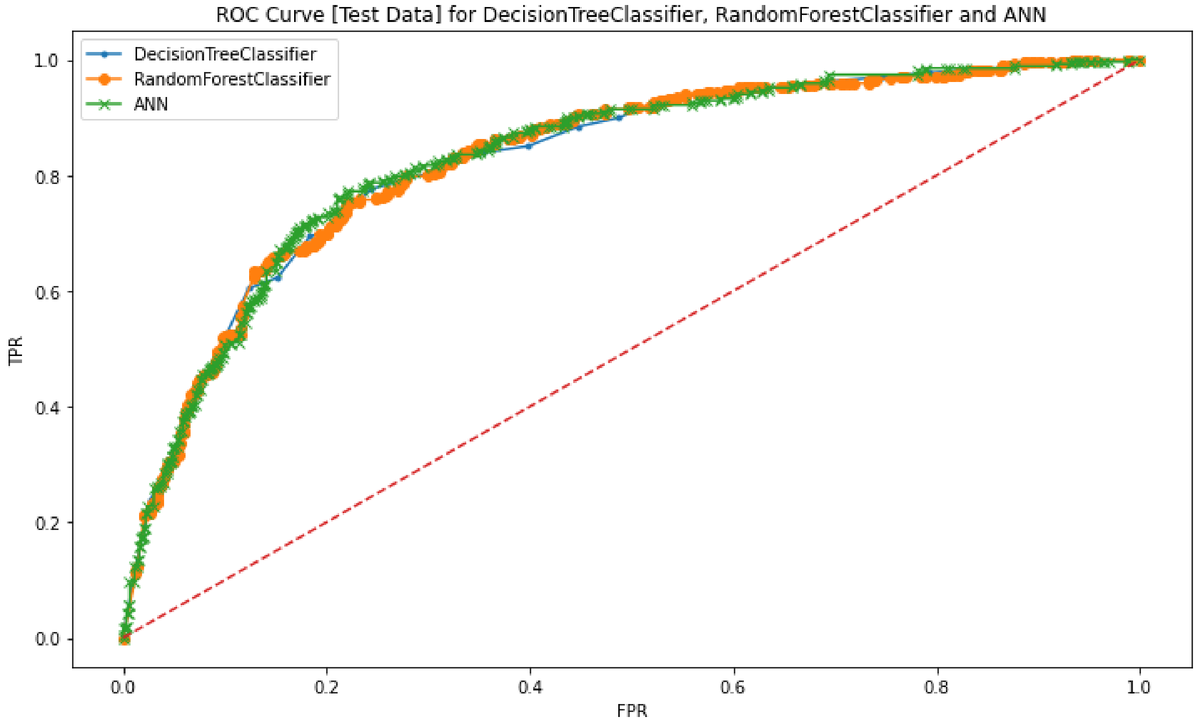


*Table 24 - Model Comparison in ROC Curve (Test Data)*

ANN seem to have better AUC, as it covers more area under the curve.

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

- As per data 98% insurance was done using online channel.
- More customers are opting for customized plan rather than agency offered.
- JZI has less insurance plans sold; they need to come up with marketing strategy to increase their sales.
- Airline agency type has a greater number of claims, we need to deep dive on customer’s data and reason why they are claiming more.
- Offline channel comparatively has more claims than online channel, there is more possibility of having agent influence there resulting false claims, need to analyse further.
- C2B has more insurance claimed percentage, need to deep dive in that data.
- On model preparation, Agency seem to play import role in deciding claim status, we need to be sure all agencies are providing correct information and not leading customers for fraudulent direction
- Travel agency is selling more insurance than Airline, we can come up with offers and marketing strategies to increase sales in airlines as well.

` *****END*****