BUSINESS REPORT

DATA MINING

Created by: Srinidhi Devan

BUSINESS REPORT

Problem 1:

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.

Objective:

To identify the segments based on credit card usage.

About the Data:

Variables	Description
Spending	Amount spent by the customer per month (in '000s)
Advance Payments	Amount paid by the customer in advance by cash (in '00s)
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 '000s)
Credit Limit	Limit of the amount in credit card ('0000s)
Minimum Payment Amount	Minimum paid by the customer while making payments for purchases made monthly (in '00s)
Maximum Spent in Single Shopping	Maximum amount spent in one purchase (in '000s)

1.1. Read the data and do exploratory data analysis. Describe the data briefly.

spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
19.94	16.92	0.8752	6.675	3.763	3.252	6.55
15.99	14.89	0.9064	5.363	3.582	3.336	5.144
18.95	16.42	0.8829	6.248	3.755	3.368	6.148
10.83	12.96	0.8099	5.278	2.641	5.182	5.185
17.99	15.86	0.8992	5.89	3.694	2.068	5.837

The above is the head of the dataset.

Dimension of the Data: 210 Rows and 7 Columns (With No Missing Values and No Duplicate Values)

Measures of Central Tendency:

Variables	Mean	Median	Mode
spending	14.85	14.36	11.23, 14.11, 15.38
advance_payments	14.56	14.32	13.47
probability_of_full_payment	0.87	0.87	0.88
current_balance	5.63	5.52	5.23, 5.39
credit_limit	3.26	3.24	3.03
min_payment_amt	3.70	3.60	2.13
max_spent_in_single_shopping	5.41	5.22	5.00

- The average spending of the customers per month is 14.85 (‘000). However, the advance payments average is 14.56 (’00). If we compare these figures with the current balance credit limit, the average is 5.63 (‘000) and 3.26 (‘0000) respectively. This means people, on an average rely on credit card for their purchases. The average of maximum amount spent in single shopping is 5.41 (’00).

- The variables Spending and Current Balance are bi-/multi-modal.

- All the variables are right-skewed (as Mean > Median), though the magnitude differs. Also, while observing mean and median, probability of full payment seems to have zero skewness.

Measures of Dispersion:

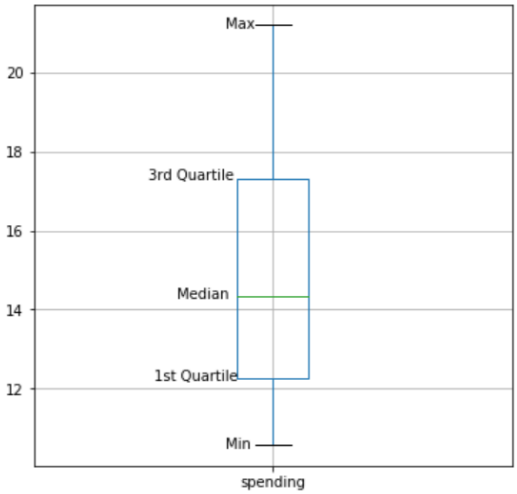
Variables	Variance	Range	IQR	CV
spending	8.466	10.590	5.035	0.196
advance_payments	1.706	4.840	2.265	0.090
probability_of_full_payment	0.001	0.110	0.031	0.027
current_balance	0.196	1.776	0.718	0.079
credit_limit	0.143	1.403	0.618	0.116
min_payment_amt	2.261	7.691	2.207	0.406
max_spent_in_single_shopping	0.242	2.031	0.832	0.091

- Spending has the highest variance amongst the other variables implying that it is highly spread out from the mean. The probability of full payment has the least variance, approximately close to 0, implying the data points are identical to one another.

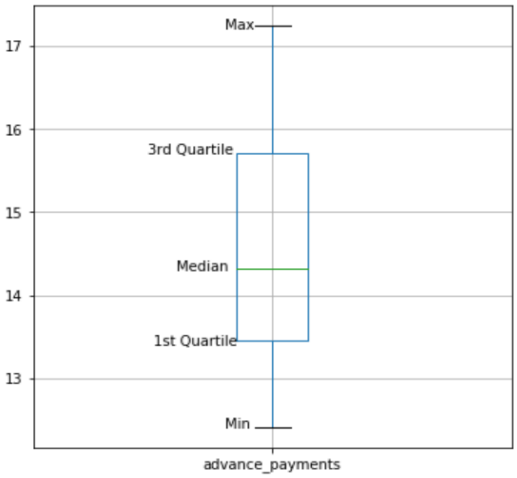
- Since the spread of Spending variable is the highest, it is expected that its range and IQR will also be the highest and on the other hand, probability of full payment is on the lower side.

Five Number Summary: The five number summary helps describe the center, spread and shape of data.

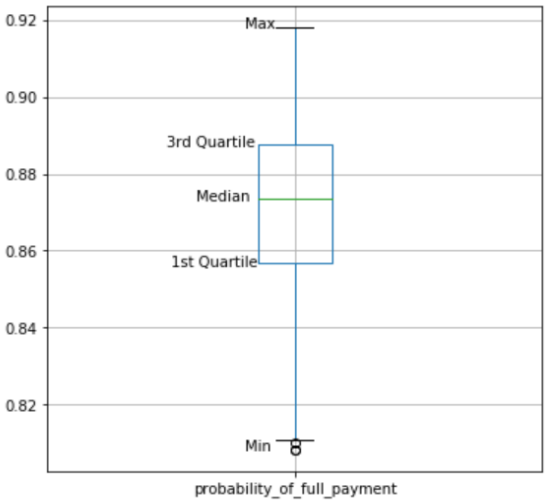
1. *Spending*



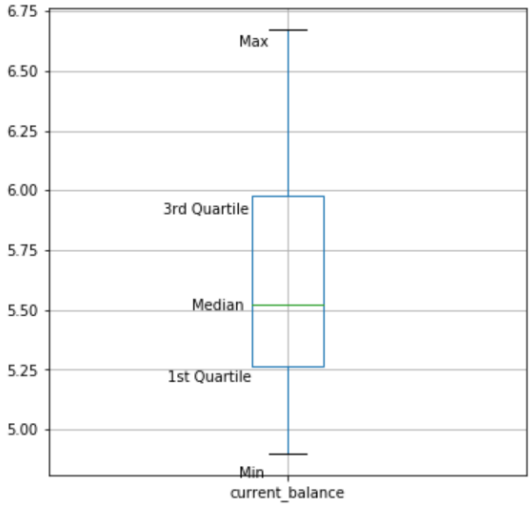
2. *Advance Payments*



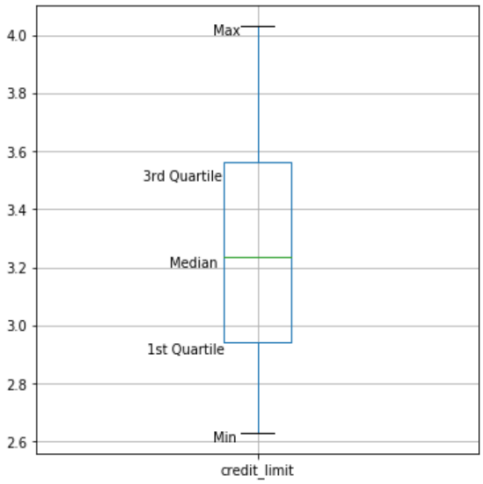
3. *Probability of Full Payment*



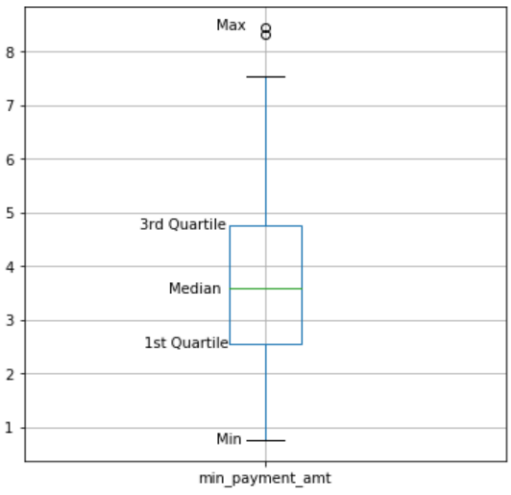
4. *Current Balance*



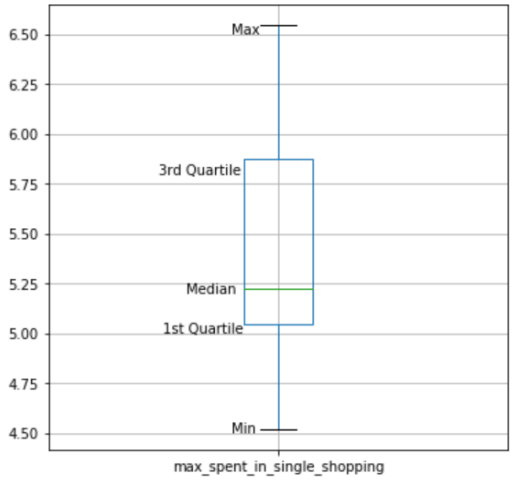
5. *Credit Limit*



6. *Minimum Payment Amount*



7. *Maximum Spent in Single Shopping*



Skewness of the Data:

Variables	Skewness
spending	0.40
advance_payments	0.39
probability_of_full_payment	-0.54
current_balance	0.53
credit_limit	0.13
min_payment_amt	0.40
max_spent_in_single_shopping	0.56

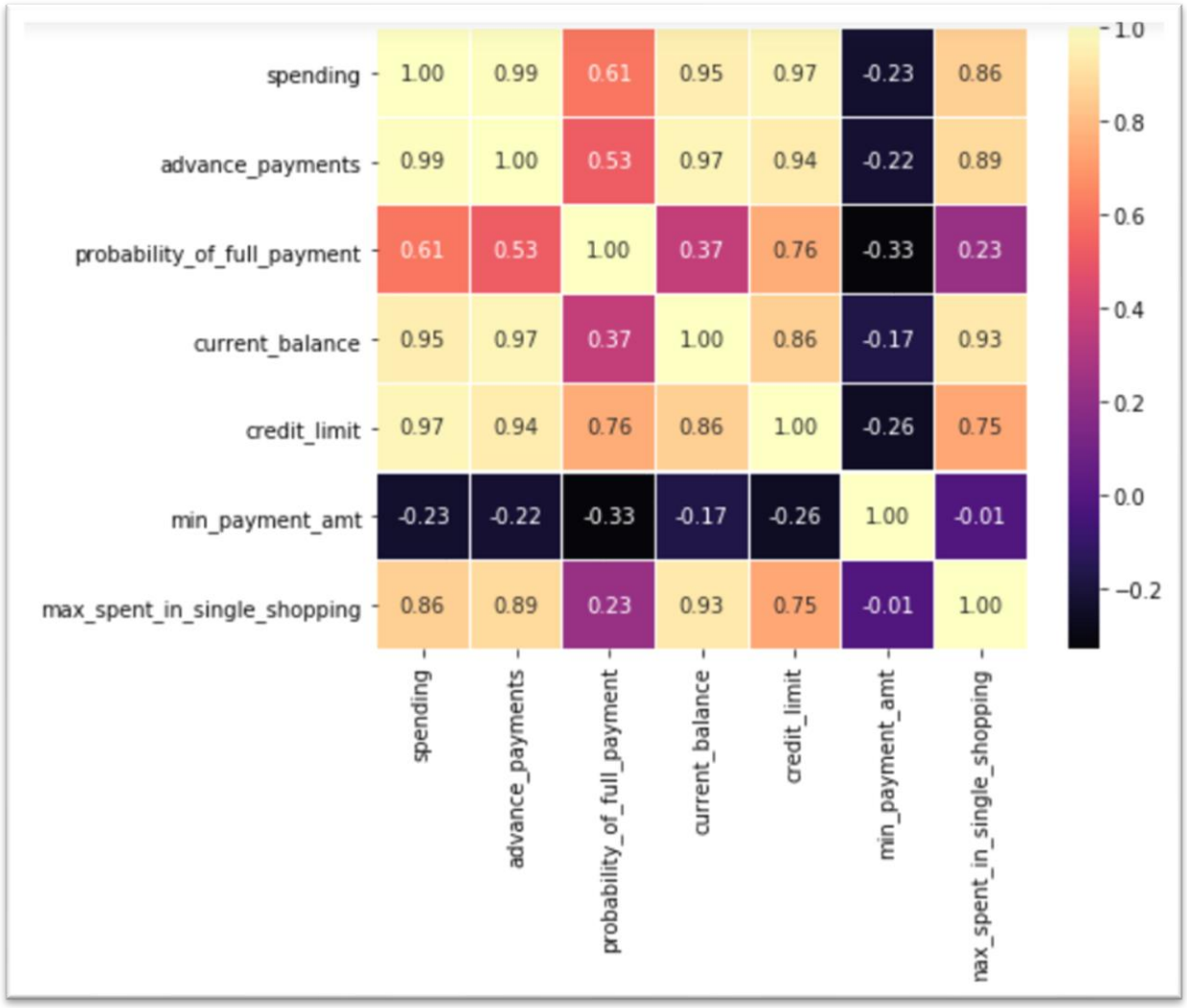
While observing the skewness, probability of full payment seems to have negative skewness and maximum amount spent in single shopping have positive skewness.

Covariance:

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

Covariance tells us to what extent two variables change together. A positive value indicates that the two variables move in the same direction and vice versa: minimum payment amount changes inversely with spending, advance payments, probability of full payments, current balance, credit limit and maximum spent in single shopping. On the other hand, the other variables changes positively with each other.

Correlation:



While covariance can only determine the relationship between the two variables, correlation helps us determine the strength between the variables. In comparison to covariance, correlation is independent of scale. However, correaltion does not imply causation.

As observed from the above heatmap, minimum payment amount has a negative relationship with the other variables. Other variables have positive relation with each other.

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

- Yes, scaling is necessary.
- Scaling, in general, is done so that all the variables under consideration is given the same weightage. In the given dataset, spending is in 1000's, advance payments is in

100's and credit limit is in 10000's- this implies that different variables will be given different weightages.

- Clustering techniques use Distance methods like Euclidean, Manhattan etc., to compute distances between clusters, which is highly affected by the unscaled variables making the models ineffective.

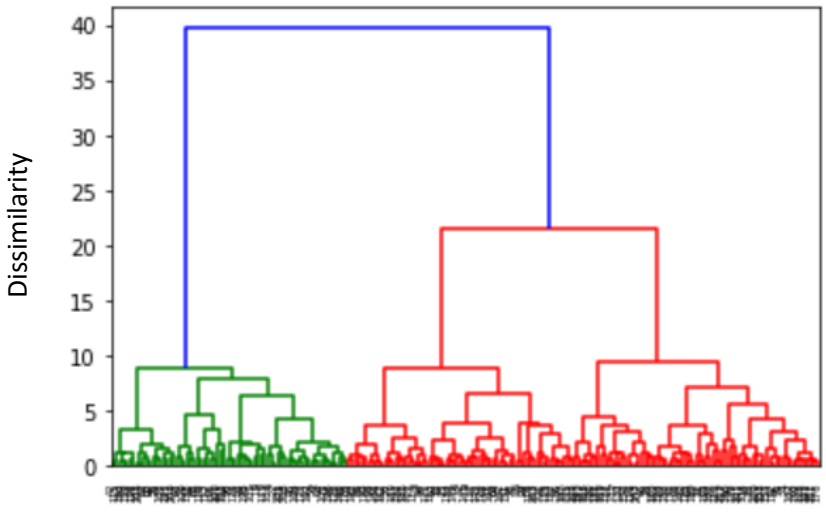
- Hence Scaling is important in case of Clustering.

After scaling the data using z-Score or Standardization method, the values will now range between -3 and +3.

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

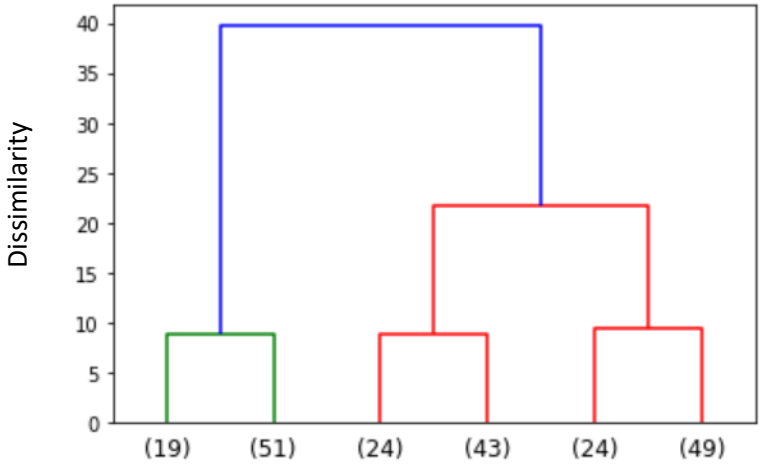
In Hierarchical Clustering, the records are subsequentially grouped to create clusters based on the distances between records and distances between clusters.
The advantage of using Hierarchical Clustering is that there is no assumptions on the number of clusters and any number of clusters can be obtained by cutting the dendrogram at the precise level.
It uses various linkage method to calculate distance between clusters like Single Linkage, Complete Linkage, Average Linkage, Centroid Linkage and Ward's Linkage techniques.

Here, the linkage method used is Ward's linkage method. The advantage of this is that the sum of squares within group is minimized.



The x-axis of the dendrogram represents the distance between clusters (or dissimilarities) and the y-axis represents the clusters/objects. The vertical position of the split, displayed by the short horizontal bar, gives the distance dissimilarity between the two clusters.

The original dendrogram can be hard to read/interpret, to condense it, we here use truncate mode as 'lastp' such that the leaf nodes contains all the other non-singleton clusters. The parameter (p) for truncate mode is taken as 6.



The above is the condensed dendrogram.

To know how many categories the data needs to be clustered into, we use 'fcluster' with 'Ward' linkage method (as mentioned above) by setting the 'maxclust' (the max number of clusters) required as 3.

The alternative method used here is 'distance' which requires us to specify the distance at which the dendrogram is needed to be cut in accordance with our desired numbers of clusters selected.

THE OPTIMUM NUMBER OF CLUSTERS (FROM HIERARCHICAL CLUSTERING): '3'.

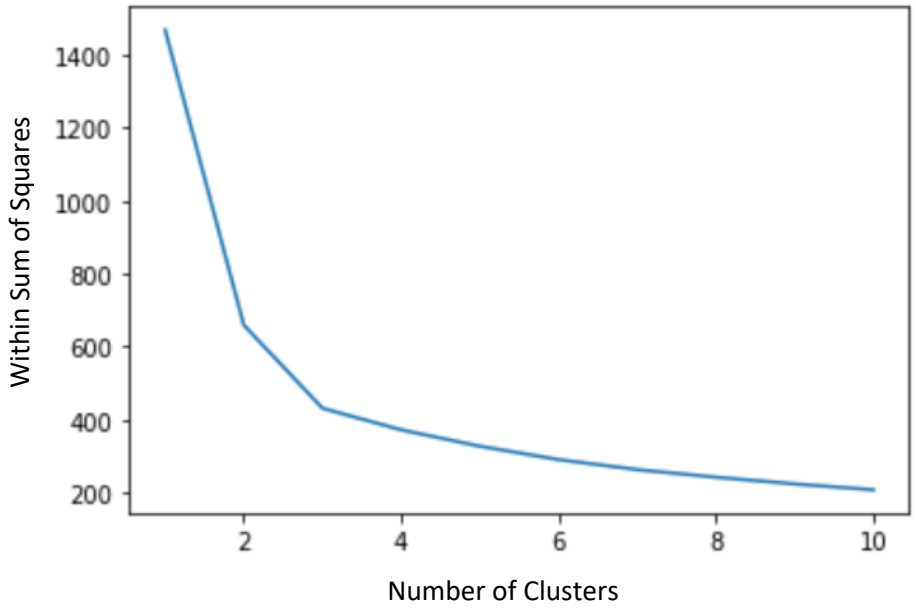
spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	clusters
19.94	16.92	0.8752	6.675	3.763	3.252	6.55	1
15.99	14.89	0.9064	5.363	3.582	3.336	5.144	3
18.95	16.42	0.8829	6.248	3.755	3.368	6.148	1
10.83	12.96	0.8099	5.278	2.641	5.182	5.185	2
17.99	15.86	0.8992	5.89	3.694	2.068	5.837	1
...	...	...	...	...	...	...	...
13.89	14.02	0.888	5.439	3.199	3.986	4.738	3
16.77	15.62	0.8638	5.927	3.438	4.92	5.795	1
14.03	14.16	0.8796	5.438	3.201	1.717	5.001	3
16.12	15	0.9	5.709	3.485	2.27	5.443	1
15.57	15.15	0.8527	5.92	3.231	2.64	5.879	3

The above table shows us which cluster each row has been classified into.

1.4. Apply K-Means clustering on scaled data and determine optimum clusters. Apply elbow curve and silhouette score.

In K-Means, each record is assigned to one of the K-clusters according to their distance from each cluster.
Unlike Hierarchical Clustering, to form good cluster, in K-Means we have to pre-specify a desired number of clusters (K) which is the biggest challenge. To overcome this, we can use Within Sum of Squares/ Distortion/ Error Plot.
The x-axis of WSS plot represent the number of clusters and the y-axis represents the within sum of squares. The location with a bend is using considered as the optimal number of clusters.
Note: Inertia is the Error Sum of Squares for each cluster under consideration. Larger the value of inertia, the more is the distance between all the points in the cluster and vice versa.

No of Clusters	WSS
1	1469.99
2	659.17
3	430.65
4	371.3
5	326.55
6	289.42
7	262.34
8	243.29
9	224.17
10	205.13



FROM THE ABOVE ELBOW CURVE, THE OPTIMAL NUMBER OF CLUSTERS: '3'.

spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_kmeans
19.94	16.92	0.8752	6.675	3.763	3.252	6.55	0
15.99	14.89	0.9064	5.363	3.582	3.336	5.144	1
18.95	16.42	0.8829	6.248	3.755	3.368	6.148	0
10.83	12.96	0.8099	5.278	2.641	5.182	5.185	2
17.99	15.86	0.8992	5.89	3.694	2.068	5.837	0
...	...	...	...	...	...	...	...
13.89	14.02	0.888	5.439	3.199	3.986	4.738	1
16.77	15.62	0.8638	5.927	3.438	4.92	5.795	0
14.03	14.16	0.8796	5.438	3.201	1.717	5.001	1
16.12	15	0.9	5.709	3.485	2.27	5.443	1
15.57	15.15	0.8527	5.92	3.231	2.64	5.879	1

The above table shows us which cluster each row has been classified into.

Now that we have completed clustering using K-Means, to check whether the observations are correctly clustered according to the distance method or not can be verified using 'Silhouette Score'.

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

If the Silhouette Width id positive, mapping to its centroid is correct. Otherwise, not.

The average of Silhouette Width for all observations is then computed which gives us the Silhouette Score.

If Silhouette Score is close to 1, on an average, the clusters are well separated. Otherwise, not.

The Silhouette Score is 0.40072.

In this case, the clusters, on an average, are *well separated*.

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

For Hierarchical Clustering:

Cluster	No of Rows
1	70
2	67
3	73
Total	210

Cluster	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
1	18.37	16.15	0.88	6.16	3.68	3.64	6.02
2	11.87	13.26	0.85	5.24	2.85	4.95	5.12
3	14.20	14.23	0.88	5.48	3.23	2.61	5.09

For k-Means Clustering:

Cluster	No of Rows
0	67
1	71
2	72
Total	210

Cluster	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
0	18.50	16.20	0.88	6.18	3.70	3.63	6.04
1	14.44	14.34	0.88	5.51	3.26	2.71	5.12
2	11.86	13.25	0.85	5.23	2.85	4.74	5.10

Note:

- *Cluster 1 in Hierarchical Clustering is mapped to Cluster 0 in K-Means.*
- *Cluster 2 in Hierarchical Clustering is mapped to Cluster 2 in K-Means.*
- *Cluster 3 in Hierarchical Clustering is mapped to Cluster 1 in K-Means.*

A. *Promotional Strategies for Cluster 1 in Hierarchical Clustering & Cluster 0 in K-Means:*

This cluster is the richer category observing their 'spending', 'advance payments', 'credit limit', 'minimum payment amount' and 'maximum spent in single shopping' which is the highest amongst the others.

To convert them into more profitable customers:

i. Increase their credit limit, so that they use the credit cards more frequently since the chances of them being converted into bad customers are very less.
ii. Since they are the highest spenders of the three clusters, they can be offered a few loyalty rewards to hold them back as our customers. This may include a round-trip to various countries, cashback/reward points to the highest spender(s) of the week.

B. *Promotional Strategies for Cluster 2 in Hierarchical Clustering & Cluster 2 in K-Means:*

This cluster comes under the 'aspiring spenders' (middle income) category as almost all the parameters under consideration are closer to the previous category.

Since their aspirations to spend is high, the chances that they will churn out is also high. To hold them as a loyal customers, instead of increasing their credit limit (as this can prove to be a risky category of customer as well), the bank should offer more attractive promotional strategy to this group than the remaining group of customers.

C. *Promotional Strategies for Cluster 3 in Hierarchical Clustering & Cluster 1 in K-Means:*

This is the category consisting of low-income people with lowest spending and advance payments. However, the minimum payment amount is the highest.

The best promotional strategies for this group would be:

i. Providing an EMI option on purchases (say, up to maximum of 6 months) which might sound attractive to them and they can easily purchase now and pay later. This is also one of the ways to generate some interest to the banks from this category.
ii. Instead of providing cashbacks/other promotional offers, the best suited promotional strategies would be to provide product discounts (say 2% to 5% on particular essential products). This way, the chances that these customer will churn out would reduce.

Note:

Even after treating outliers for this particular problem, there is not much of a difference in the performance of the model.

Problem 2:

An Insurance firm providing tour insurance is facing higher claim frequency. The management decides to collect data from the past few years.

Objective:

To make a model which predicts the claim status and provide recommendations to management using CART, Random Forest, and Artificial Neural Network.

About the Data:

Variables	Description	Data Type
Claimed	Claim Status	Categorical
Agency_Code	Code of tour firm	Categorical
Type	Type of tour insurance firms	Categorical
Channel	Distribution channel of tour insurance agencies	Categorical
Product	Name of the tour insurance products	Categorical
Duration	Duration of the tour	Numerical
Destination	Destination of the tour	Categorical
Sales	Amount of sales of tour insurance policies	Numerical
Commission	The commission received for tour insurance firm	Numerical
Age	Age of insured	Numerical

2.1. Data Ingestion: Read the dataset. Do the descriptive statistics and do null value condition check, write an inference on it.

Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product_Name	Destination
48	C2B	Airlines	No	0.7	Online	7	2.51	Customised Plan	ASIA
36	EPX	Travel Agency	No	0	Online	34	20	Customised Plan	ASIA
39	CWT	Travel Agency	No	5.94	Online	3	9.9	Customised Plan	Americas
36	EPX	Travel Agency	No	0	Online	4	26	Cancellation Plan	ASIA
33	JZI	Airlines	No	6.3	Online	53	18	Bronze Plan	ASIA

The above is the head of the dataset.

Dimension of the Data: 3000 Rows and 10 Columns

Measures of Central Tendency:

Variables	Mean	Median	Mode
Age	38.09	36.00	36.00
Commision	14.53	4.63	0.00
Duration	70.00	26.50	8.00
Sales	60.25	33.00	20.00

The average age of the customers/ insured is 38. On an average, the commission received by the firm stood around 14.53.

However, mean is highly affected by the outliers. Hence it is important to check that if there are any outlier(s) in any of the above variables.

Also, in all the numerical variables under consideration, mean > median, clearly indicating it is right-skewed.

Measures of Dispersion:

Variables	Variance	IQR	CV
Age	109.49	10.00	0.27
Commision	649.30	17.24	1.75
Duration	17970.29	52.00	1.92
Sales	5003.29	49.00	1.17

Duration has an extremely high variance followed by Sales. We know that both variance and standard deviation are not robust to outliers. Such high values may be an indicative of outliers.

IQR helps us look at spread of the data and is robust to outliers. This tells us the Duration has the highest spread of the data closely followed by the Sales.

Skewness of the Data:

Variables	Skewness
Age	1.149713
Commision	3.148858
Duration	13.784681
Sales	2.381148

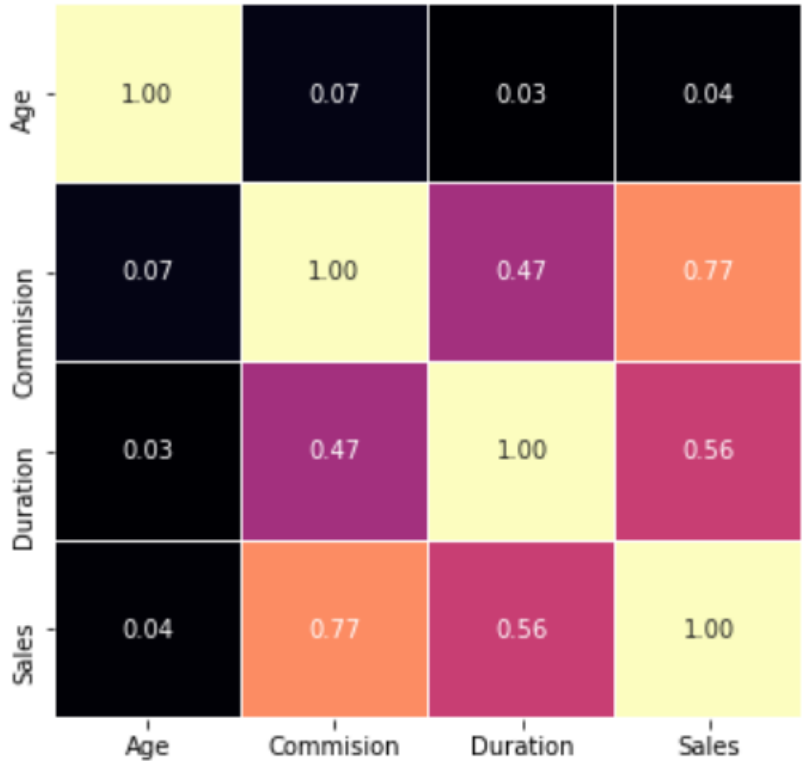
In general, skewness tends to be large and highly positive when there are many outliers present in the variables. Such is the case of Duration which is suggestive of outliers.

Covariance:

Variables	Age	Commision	Duration	Sales
Age	109.49	18.06	42.68	29.20
Commision	18.06	649.30	1610.20	1381.55
Duration	42.68	1610.20	17970.29	5299.84
Sales	29.20	1381.55	5299.84	5003.29

A positive correlation value indicates that the two variables move in the same direction. All the variables are positively related.

Correlation:



Correlation helps us determine the strength between the variables and is independent of scale. The highest correlation is observed between Commission and Sales which is very intuitive, followed Duration and Sales; Commission and Duration.

Checking for Missing Values:

Variables	Missing Values
Age	0
Agency_Code	0
Type	0
Claimed	0
Commision	0
Channel	0
Duration	0
Sales	0
Product_Name	0
Destination	0

There are no missing values in the dataset.

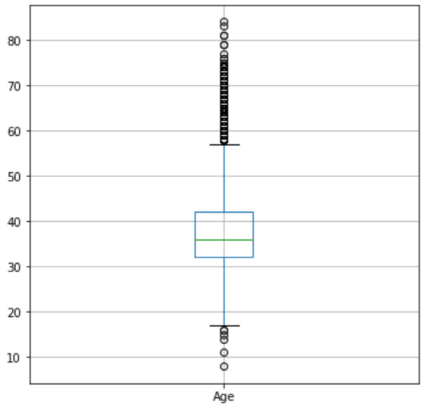
Checking for Duplicate Rows:

Number of Duplicated Rows = 139

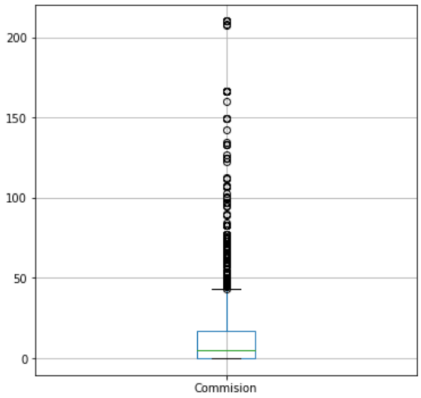
Row No.	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product_Name	Destination
63	30	C2B	Airlines	Yes	15	Online	27	60	Bronze Plan	ASIA
329	36	EPX	Travel Agency	No	0	Online	5	20	Customised Plan	ASIA
407	36	EPX	Travel Agency	No	0	Online	11	19	Cancellation Plan	ASIA
411	35	EPX	Travel Agency	No	0	Online	2	20	Customised Plan	ASIA
422	36	EPX	Travel Agency	No	0	Online	5	20	Customised Plan	ASIA

Box Plot:

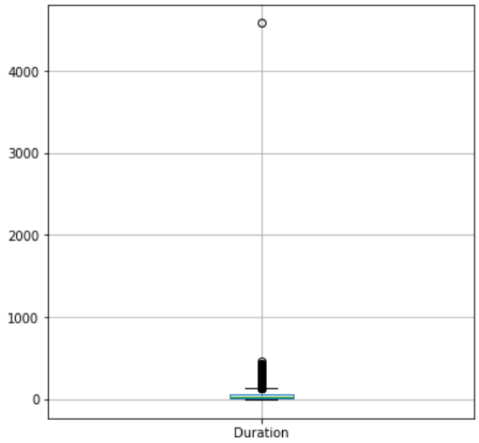
1. AGE



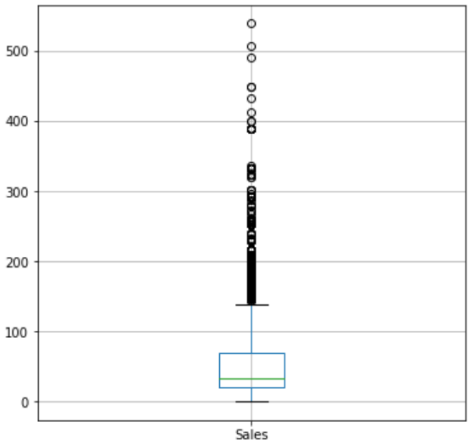
2. COMMISSION



3. DURATION

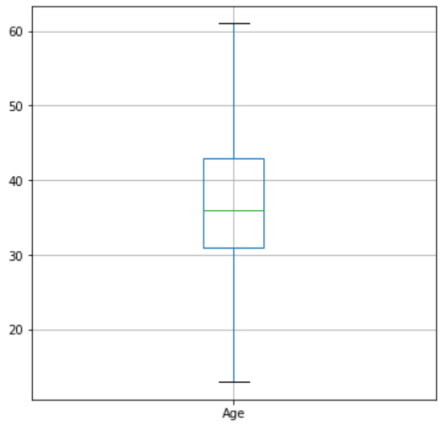


4. SALES

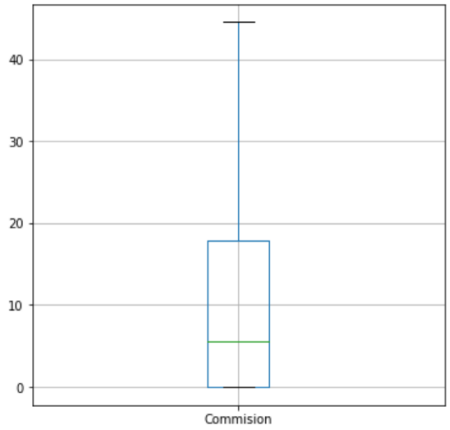


Since there are Outliers in the variables: Sales, Duration, Age and Commission, we will treat them using the IQR (Inter Quartile Range) Method.

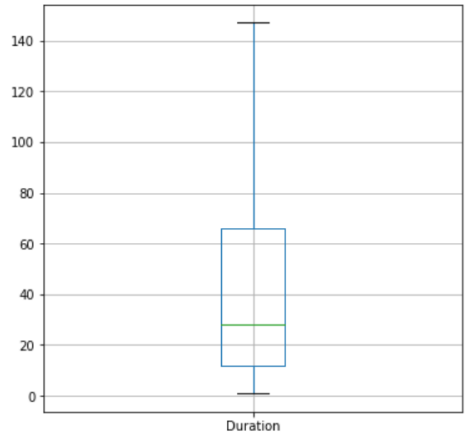
1. AGE



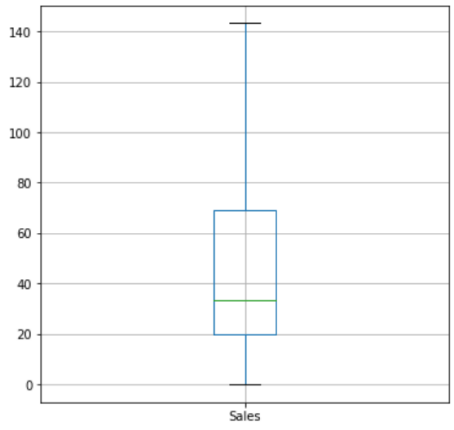
2. COMMISSION



3. DURATION



4. SALES



Most of the Machine Learning techniques cannot understand Categorical data. Hence it is essential that we convert the categorical variable in the given dataset to numerical variables.

The value counts of the Categorical Data are as follows:

Product Name	Value Counts	Agency Code	Value Count	Destination	Value Counts	Type	Value Counts	Channel	Value Counts
Customised Plan	1071	EPX	1238	ASIA	2327	Travel Agency	1709	Online	2815
Bronze Plan	645	C2B	913	Americas	319	Airlines	1152	Offline	46
Cancellation Plan	615	CWT	471	EUROPE	215				
Silver Plan	421	JZI	239						
Gold Plan	109								

The value counts and proportion of the target variable (Categorical) viz 'Claimed' is:

Claimed	Value Counts		Claimed	Proportion
No	1947		0	0.68
Yes	914		1	0.31

The above table shows us that there is no problem of Class Imbalance and we can proceed to our next step which is Splitting the Data into Train and Test set.

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

Before even splitting the data into Train and Test data, it is important that we define the Independent (X) and Dependent (y) variables.

Here the independent variables are: Agency Code, Type, Channel, Product, Duration, Destination, Sales, Commission and Age.

The dependent variable is: Claimed.

The data is then split in 70:30 ratio i.e., 70% of the data goes to the Train set and the remaining 30% to the Test set (where rows are selected randomly).
Random State is set to 1 (this ensures the generated splits are reproducible).

	Rows	Variables
X_train	2002	9
X_test	859	9
y_train	2002	1
y_test	859	1

After the above important step, we can now proceed to our Model Building process.

I. DECISION TREE: A Supervised Learning technique which can be used both in the case of Classification and Regression.

The splitting criteria used here is 'Gini'. According to the Gini criteria, the independent variable(s) which has the highest Gini Gain with respect to the dependent variable is used as the best variable for splitting the data. The Gini calculation is easy for binary variable. On the other hand, for continuous variable, various binary cut-off are chosen and the best Gini cut-off of those are then chosen.

The next step is to build the Decision Tree and visualizing it. It is done in two ways: One, create a dot file and visualize it using WebGraphViz; Second is using 'plot_tree' and 'tree' packages from 'sklearn'. The advantage of the second method is that we can directly visualize it in our Jupyter notebook, however, if the tree is too large in size, this method might not be useful.
Hence here, we have created a 'dot file' to visualize it.
After visualizing the tree, we see that there is a need to prune the tree. This ensures that the tree is not overgrown.

So, the next step is setting the parameter for Decision Tree. The parameters are as follows:

a. Max Depth: The maximum depth of the tree.

In our case, it is chosen to be 4, 6 & 7.

b. Min Samples Split: To split an internal node, this help us specify the minimum number of samples required.

In our case, it is: 30, 35 & 40.

c. Min Samples Leaf: Minimum number of sample to be at a leaf node.

In our case, it is: 60, 80 & 120.

Again the 'random state' is set to 1.

As seen above, the Gini criteria chooses different binary cut-off for the continuous variables and the best Gini gain cut-off is shortlisted, this, however, leads to a problem of Greedy Algorithm (in other words, sub-optimal decision). To overcome this, we set CV (Cross Validation) to 10 in this case.

The best parameters chose are: Max Depth = 4
Min Samples Split = 30
Min Samples Leaf = 120

After this step, the tree looks neat and easy to interpret.

The variable importance is shown below:

Variables	Importance
Agency_Code	0.5812
Sales	0.2952
Product_Name	0.0458
Commision	0.0346
Duration	0.0230
Destination	0.0201
Age	0.0000
Type	0.0000
Channel	0.0000

From the above table, we can see that Agency code has the highest importance followed by Sales, Product Name and so on.

II. RANDOM FOREST CLASSIFIER: An ensembling technique. A collection of many decision tree, which can also be used in the case of classification and regression. It randomly chooses different rows with replacement and random set of variables.

Starting with the parameter setting:

a. Max Depth**:** The maximum depth of the tree.

In our case, it is 6, 7 & 8.

b. Max Features: When splitting a node, it helps us to set the maximum size of random subset of features.

In our case, it is 5, 6, 7 & 8.

c. Min Samples Leaf: The minimum number of samples required to be at a leaf node.

In our case, it is 20, 30 & 40.

d. Min Samples Split: To split an internal node, this help us specify the minimum number of samples required.

In our case, it is 40, 45 & 50.

e. n estimators: Number of trees required to be built before averaging out the prediction.

In our case, it is 100, 200 & 300.

Here, again the random state is set to 1 and CV (Cross Validation) is set to 10.

The best parameters turned out to be: Max Depth = 7

Max Features = 7

Min Samples Leaf = 20

Min Samples Split = 50

n_estimators = 200

After this step, we can now move on to calculate the 'Out-Of-Bag' Score which measures how accurate the random forest is.

OOB Score = 0.7667

It shows us that our Random Forest model is 76.7% accurate, which is a relatively better result.

The variable importance is shown below:

Variables	Importance
Agency_Code	0.3920
Sales	0.2204
Product_Name	0.1643
Duration	0.0813
Commision	0.0660
Age	0.0600
Destination	0.0100
Type	0.0060
Channel	0.0000

The above table shows us the 'Agency Code' is the most important variable followed by 'Sales', 'Product Name' and so on.

III. ARTIFICIAL NEURAL NETWORK: It is a black box technique and an extended deep learning technique. This an attempt of how brain works when used for dataset. It has ability to learn, generalize and adapt to the changing environmental conditions.

Here, Scaling of Independent and Dependent variables is a crucial step. If not, the weights calculated may give us a misleading picture in the case of ANN.

Once this step is done, we can move on to parameter setting:

a. Hidden Layer Sizes: Number of neurons in the i^{th} hidden layer

In our case, it is: (100, 100, 100)

b. Activation: Hidden layer's activation function

In our case, we have chosen: Logistic & Relu

c. Solver: For weight optimization

In our case, we have chosen: sgd & adam

d. Tol (Tolerance): Optimization tolerance level

In our case, it is 0.1, 0.01 & 0.001

e. Max Iteration: Number of maximum iterations

In our case, it is 150 & 300

The next step is to build the model with the best parameters chosen:

Hidden Layer Sizes = (100, 100, 100)

Activation = relu

Max iter = 150

Solver = adam

Verbose = True

Tol = 0.001

Random state = 1

After iteration = 57, the loss is brought down to 0.4596.

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

Performance metrics helps us to evaluate how the models under consideration has performed overall using different criteria.

I. ACCURACY: Number of correct predictions made by the model divided by the total number of predictions. Lesser the false prediction, more the accuracy.

II. CONFUSION MATRIX: A 2 X 2 matrix reflecting the performance of the model in 4 blocks and is specifically designed for Classification techniques.
The y-axis represent Actual Label and x-axis represent Predicted Label.

Here,

- True Negative: The ‘no’ claimed data which is classified as ‘no’ claimed.
- False Positive: The ’no’ claimed data which is classified as ‘yes’. Also known as Type I error.
- True Positive: The claimed data actually classified as claimed.
- False Negative: The claimed data that is classified as ‘no’. Also known as a "Type II error.")

a. Accuracy: How accurately does the model classify the data points.
b. Sensitivity: How many of the actual true data points are identified as true data points by the model.
c. Specificity: How many of the actual negative data points are identified as negative by the model.
d. Precision: Among the positives identified as positives by the model, how many are really positive.

III. CLASSIFICATION REPORT: This is also specifically used for Classification algorithm. This helps us identifying the quality of the model using: Precision, Recall, f1-Score, Support and Accuracy.

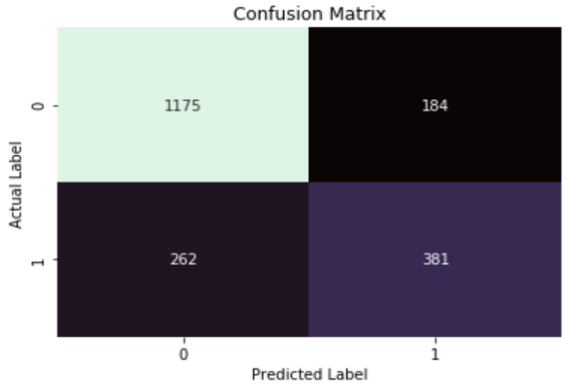
IV. AREA UNDER THE R.O.C. CURVE: At various threshold levels, how does the model perform is evaluated by AUC ROC Curve as well as AUC Score are useful to compare the performance of classifier method. The y-axis represents True Positive Rate and x-axis represents False Positive Rate.
Graph is a trade-off between Benefits (True Positive) and Costs (False Positive). Larger the area under the curve, better is the model.

DECISION TREE

Train Dataset:

I. Accuracy: 79.37%

II. Confusion Matrix:



Metrics	Value
True Negative	1175
False Positive	184
True Positive	381
False Negative	262

III. Classification Report:

	precision	recall	f1-score	support
0	0.82	0.86	0.84	1359
1	0.67	0.59	0.63	643
accuracy			0.78	2002
macro avg	0.75	0.73	0.74	2002
weighted avg	0.77	0.78	0.77	2002

a. PRECISION: Percentage of 0s correctly predicted is 82%

Percentage of 1s correctly predicted is 67%

b. RECALL: Percentage of positive cases in 0s is 86%

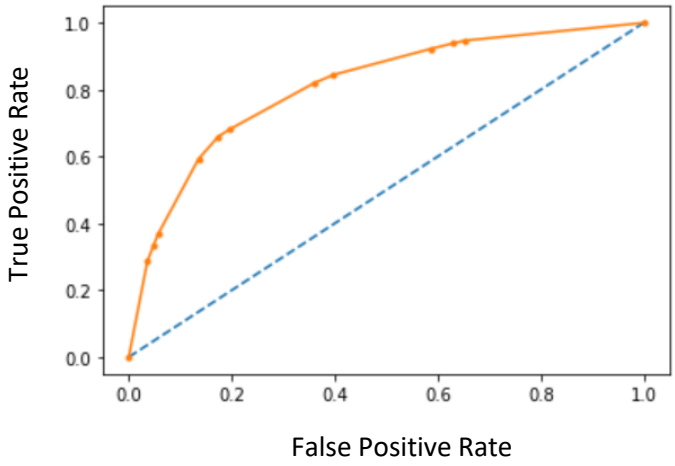
Percentage of positive cases in 1s is 59%

c. F1-SCORE: Percentage of positive predictions in 0s which were correct is 84%

Percentage of positive predictions in 1s which were correct is 63%

d. ACCURACY: 78%

IV. Area Under Curve/Receiver Operating Characteristics Curve:

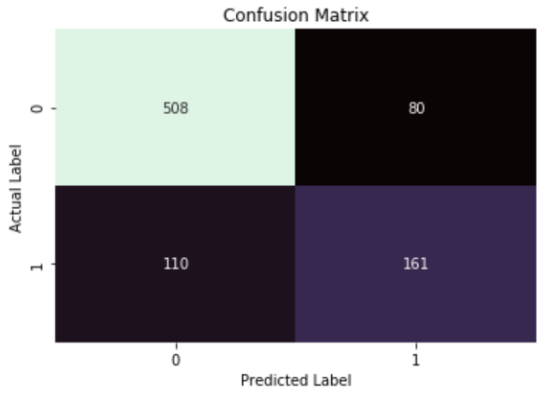


AUC Score = 0.81

Test Dataset:

I. Accuracy: 78.46%

II. Confusion Matrix:



Metrics	Value
True Negative	508
False Positive	80
True Positive	161
False Negative	110

III. Classification Report:

	precision	recall	f1-score	support
0	0.82	0.86	0.84	588
1	0.67	0.59	0.63	271
accuracy			0.78	859
macro avg	0.75	0.73	0.74	859
weighted avg	0.77	0.78	0.78	859

a. PRECISION: Percentage of 0s correctly predicted is 82%

Percentage of 1s correctly predicted is 67%

b. RECALL: Percentage of positive cases in 0s is 86%

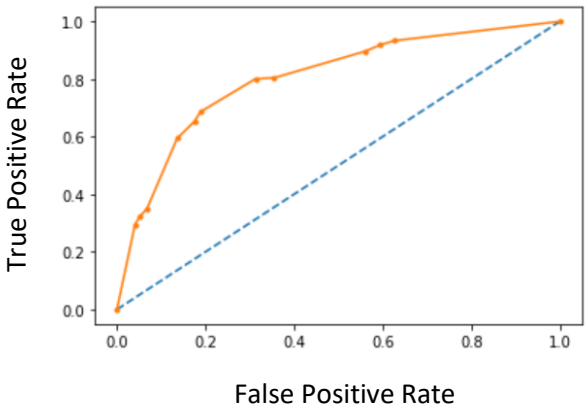
Percentage of positive cases in 1s is 59%

c. F1-SCORE: Percentage of positive predictions in 0s which were correct is 84%

Percentage of positive predictions in 1s which were correct is 63%

d. ACCURACY: 78%

IV. Area Under Curve/Receiver Operating Characteristics Curve:



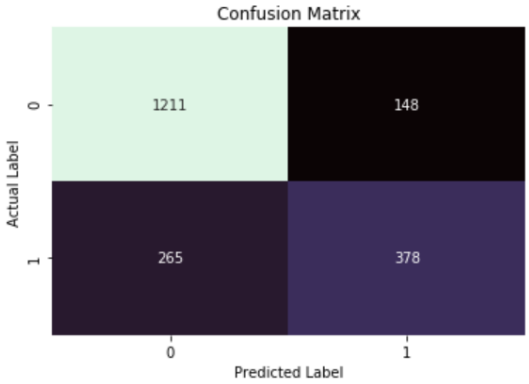
AUC Score = 0.803

RANDOM FOREST

Train Dataset:

I. Accuracy: 79.37%

II. Confusion Matrix:



Metrics	Value
True Negative	1211
False Positive	148
True Positive	378
False Negative	265

III. Classification Report:

	precision	recall	f1-score	support
0	0.82	0.89	0.85	1359
1	0.72	0.59	0.65	643
accuracy			0.79	2002
macro avg	0.77	0.74	0.75	2002
weighted avg	0.79	0.79	0.79	2002

a. PRECISION: Percentage of 0s correctly predicted is 82%

Percentage of 1s correctly predicted is 72%

b. RECALL: Percentage of positive cases in 0s is 89%

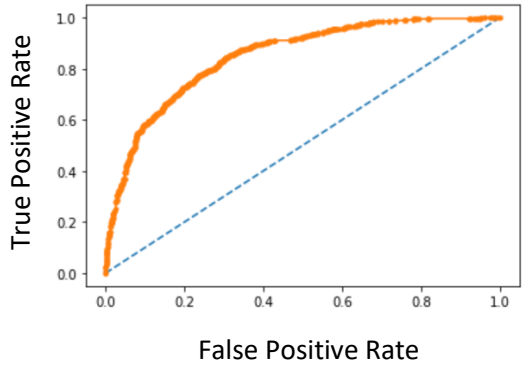
Percentage of positive cases in 1s is 59%

c. F1-SCORE: Percentage of positive predictions in 0s which were correct is 85%

Percentage of positive predictions in 1s which were correct is 65%

d. ACCURACY: 79%

IV. Area Under Curve/Receiver Operating Characteristics Curve:

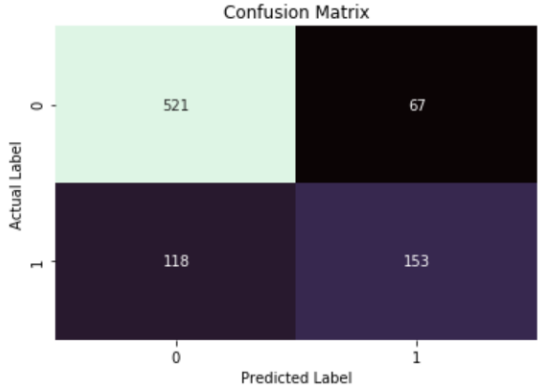


AUC Score = 0.85

Test Dataset:

I. Accuracy: 78.46%

II. Confusion Matrix:



Metrics	Value
True Negative	521
False Positive	67
True Positive	153
False Negative	118

III. Classification Report:

	precision	recall	f1-score	support
0	0.82	0.89	0.85	588
1	0.7	0.56	0.62	271
accuracy			0.78	859
macro avg	0.76	0.73	0.74	859
weighted avg	0.78	0.78	0.78	859

a. PRECISION: Percentage of 0s correctly predicted is 82%

Percentage of 1s correctly predicted is 70%

b. RECALL: Percentage of positive cases in 0s is 89%

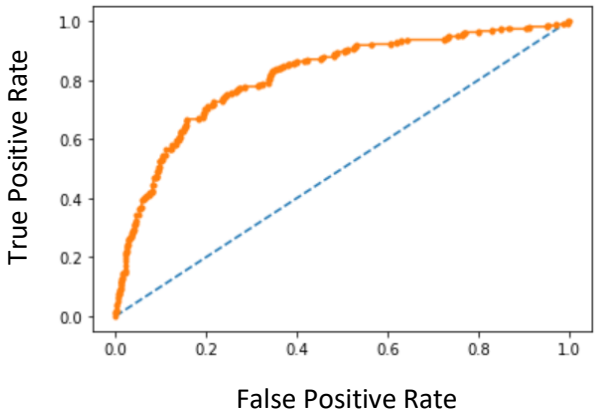
Percentage of positive cases in 1s is 56%

c. F1-SCORE: Percentage of positive predictions in 0s which were correct is 85%

Percentage of positive predictions in 1s which were correct is 62%

d. ACCURACY: 78%

IV. Area Under Curve/Receiver Operating Characteristics Curve:



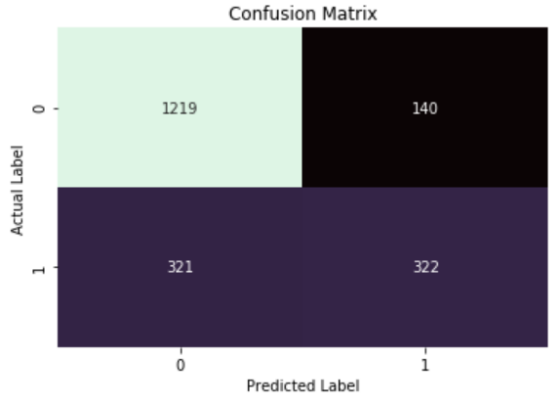
AUC Score = 0.814

ARTIFICIAL NEURAL NETWORK

Train Dataset:

I. Accuracy: 76.97%

II. Confusion Matrix:



Metrics	Value
True Negative	1219
False Positive	140
True Positive	322
False Negative	321

III. Classification Report:

	precision	recall	f1-score	support
0	0.79	0.9	0.84	1359
1	0.7	0.5	0.58	643
accuracy			0.77	2002
macro avg	0.74	0.7	0.71	2002
weighted avg	0.76	0.77	0.76	2002

a. PRECISION: Percentage of 0s correctly predicted is 79%

Percentage of 1s correctly predicted is 70%

b. RECALL: Percentage of positive cases in 0s is 90%

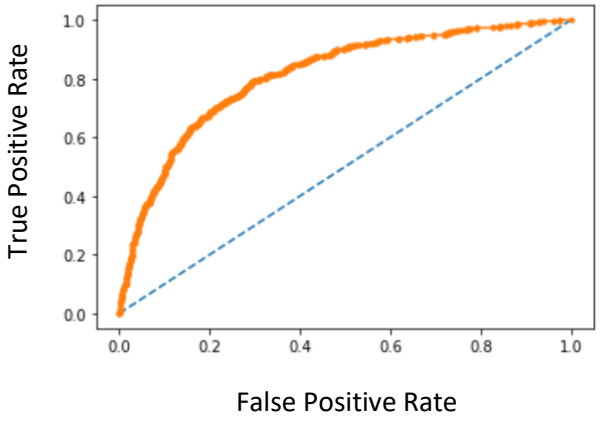
Percentage of positive cases in 1s is 50%

c. F1-SCORE: Percentage of positive predictions in 0s which were correct is 84%

Percentage of positive predictions in 1s which were correct is 58%

d. ACCURACY: 77%

IV. Area Under Curve/Receiver Operating Characteristics Curve:

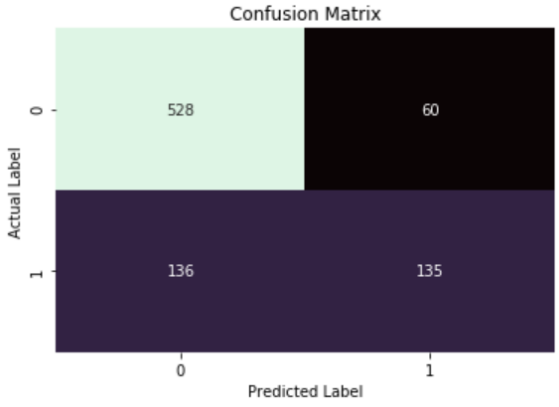


AUC Score = 0.813

Test Dataset:

I. Accuracy: 77.18%

II. Confusion Matrix:



Metrics	Value
True Negative	528
False Positive	60
True Positive	135
False Negative	136

III. Classification Report:

	precision	recall	f1-score	support
0	0.8	0.9	0.84	588
1	0.69	0.5	0.58	271
accuracy			0.77	859
macro avg	0.74	0.7	0.71	859
weighted avg	0.76	0.77	0.76	859

a. PRECISION: Percentage of 0s correctly predicted is 80%

Percentage of 1s correctly predicted is 69%

b. RECALL: Percentage of positive cases in 0s is 90%

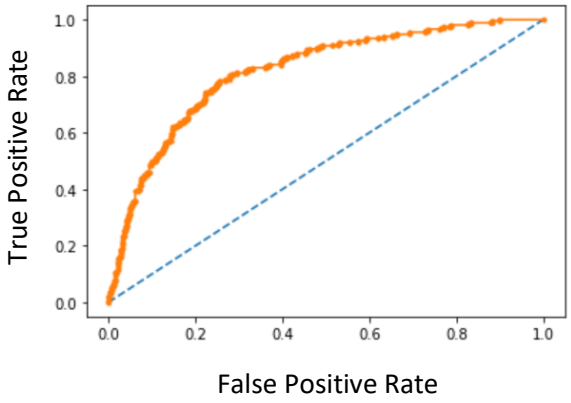
Percentage of positive cases in 1s is 50%

c. F1-SCORE: Percentage of positive predictions in 0s which were correct is 84%

Percentage of positive predictions in 1s which were correct is 58%

d. ACCURACY: 77%

IV. Area Under Curve/Receiver Operating Characteristics Curve:

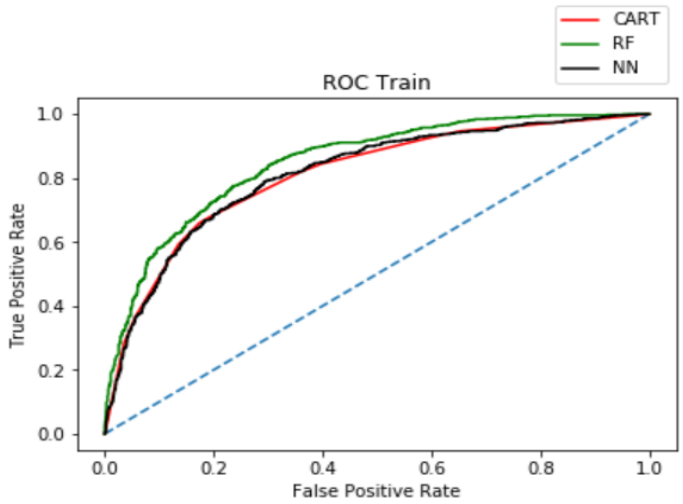


AUC Score = 0.819

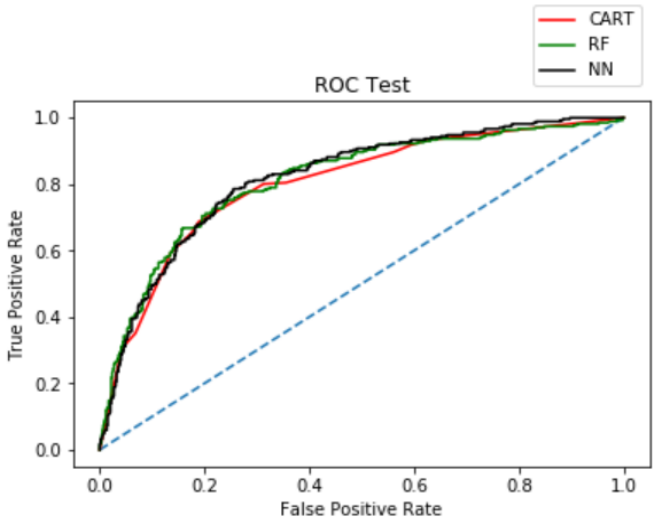
2.4. Final Model: Compare all the model and write an inference which model is best/optimized.

	Models					
Metrics	CART Train	CART Test	Random Forest Train	Random Forest Test	Neural Network Train	Neural Network Test
Accuracy	0.79	0.78	0.79	0.78	0.77	0.77
AUC	0.81	0.8	0.85	0.81	0.81	0.82
Recall	0.59	0.59	0.59	0.56	0.5	0.5
Precision	0.67	0.67	0.72	0.7	0.7	0.69
F1 Score	0.63	0.63	0.65	0.62	0.58	0.58

Train Dataset:



Test Dataset:



- Comparing the Train and Test Dataset, the accuracy for all the three models under consideration is not showing much of a difference. Also, considering only accuracy i.e., how accurately does the model classify the data points may not give us a full picture of the solution.

- Looking at the AUC Score, the obvious preference would be Random Forest.

- Further, from Precision, which helps us identify how many are really positive among the positives identified as positives by the model again Random Forest is preferred and is closely followed by Neural Network. This is important from an insurance company perspective whose is particularly facing a higher claim for the tour insurance.

- It is strenuous to compare two models if there is low recall and high precision or high recall and low precision value. For this purpose, we can use f1-Score to compare the models. The highest f1-Score is observed for Random Forest model which is followed by Decision Tree.

In conclusion, all the models have performed more or less in a similar fashion when compared with respect to all the metrics considered. In this case, out of all, Random Forest has performed slightly better than the rest (with a small difference of +0.05).

2.5. Inference: Basis on these predictions, what are the business insights and recommendations.

- Of the three models, Random Forest has performed slightly better than the other two models: Decision Tree and Neural Network.

- It is of prime importance that we use build a model which is efficient in predicting the claim status for us to provide valuable insights to the business.

- If the model is a black-box technique wherein setting the hyper-parameters is crucial, it can prove to be difficult to rely on that particular model to make a business acumen.

- However, in this particular case, all the three models have shown as a stable performance considering all the metrics of comparison.

- One useful metrics from Decision Tree and Random Forest is the variable importance which stands crucial for the business. From both the models, we saw that 'Agency Code' was the most important variable followed by 'Sales' and 'Product Name'. These variables are potential identifiers for whether the insured claimed or not.

- On the other hand, the variables that are of least importance are: 'Type' and 'Channel'.