greatlearning
Learning for Life

Business Report

DSBA

Project - Data Mining

Name: Sandeep Immadi

Date: 25/07/2021

CONTENTS:

Problem 1:

1.1 Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis)…… 6

1.2 Do you think scaling is necessary for clustering in this case? Justify……12

1.3 Apply hierarchical clustering to scaled data. Identify the number of optimum clusters using Dendrogram and briefly describe them……15

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.………19

1.5 Describe cluster profiles for the clusters defined. Recommend different promotional strategies for different clusters.……20

Problem 2:

2.1 Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis) ……25

2.2 Data Split: Split the data into test and train, build classification model CART, Random Forest, Artificial Neural Network………27

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. …… …..36

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

2.5 Inference: Based on the whole Analysis, what are the business insights and recommendations………40

List of Figures

- Figure 1 A-G Boxplot and Histogram …………………………………………………………… … 6
- Figure 2 Scatter plots ………………………………………………………………………………… 6
- Figure 3 Correlation Heatmap …………………………………………………………………… 6
- Figure 4 Pairplot …………………………………………………………………………………… . .6
- Figure 5 Features with boxplot ……………………………………………………………… . 7
- Figure 6 Dendrogram ………………………………………………………………………………… 7
- Figure 7 Truncated Cluster Dendrogram ……………………………………………… .7
- Figure 8 Scatter plots of clusters ……………………………………………………………. 8
- Figure 9 Visualization of raw data …………………………………………………………… 8
- Figure 10 Elbow plot ………………………………………………… ……………………………. 9
- Figure 11 Cluster plot for 10 clusters ………… ………………………………………………… .9
- Figure 12 Cluster plot for 3 clusters……………………………………………………………. 9
- Figure 13 Bar chart of target variable………………………………………………………. 9
- Figure 14 Facet grid of categorical variables w.r.t target variables…………………………… 10
- Figure 15 A-D histogram and boxplot …………………………………………………………… 11
- Figure 16 Frequency distribution of categorical variables ……………………………………… 11
- Figure 17 Scatter plots for numerical variables ………………………………………………… 11
- Figure 18 Countplot for categorical variables ………………………………………………… 12
- Figure 19 Variables with boxplot ………………………………………………………………… 12
- Figure 20 Correlation heatmap …………………………………………………………………… 12
- Figure 21 Pairplot ………………………………………………………………………………… 12
- Figure 22 Feature importances (Decision Tree) ………………………………………………… 13
- Figure 23 Feature importances (Random Forest) ……………………………………………… 13
- Figure 24 ROC Curve for train set (Decision Tree) ……………………………………………… 14
- Figure 25 ROC Curve for test set (Decision Tree) ……………………………………………… 14
- Figure 26 ROC Curve for train set (Random Forest) …………………………………………… …15
- Figure 27 ROC Curve for test set (Random Forest) …………………………………………… 15
- Figure 28 ROC Curve for train set (Neural network) ……………………………………………15
- Figure 29 ROC Curve for test set (Neural network) …………………………………………… 16
- Figure 30 ROC Curve for the 3 models on the Training data . ………………………………………16
- Figure 31 ROC Curve for the 3 models on the Training data ………………………………… 16
- Figure 32 ROC Curve for the 3 models ………………………………………………………… 16
- Figure 33 Description of the data ………………………………………………………………… 17
- Figure 34 Skewness and Kurtosis ……………………………………………………………………17
- Figure 35 Description of features before scaling ……………………………………………………18
- Figure 36 Data after scaling …………………………………………………………………………19
- Figure 37 Description of features after scaling ………………………………………………… 20
- Figure 38 Heirarchical Clusters appended to the original dataframe ………………………… 21
- Figure 39 Within sum of squares ……………………………………………………………………22
- Figure 40 Silhouette Score ……………………………………………………………………… 22
- Figure 41 KMean Clusters appended to the original dataframe…………………………………23
- Figure 42 Number of Heirarchical Clusters …………………………………………………… 23
- Figure 43 Means of Heirarchical Clusters ……………………………………………………… 24
- Figure 44 Number of KMean Clusters …………………………………………………………… 24
- Figure 45 Means of KMean Clusters ……………………………………………………………… 25

- Figure 46 Description of data (Numerical variables) ……………………………………………………26
- Figure 47 Description of data …………………………………………………………………………………….26
- Figure 48 Skewness and Kurtosis ………………………………………………………………………………..27
- Figure 49 Sample of train set ………………………………………………………………………………………28
- Figure 50 Sample of test set ………………………………………………………………………………………..29
- Figure 51 Dimensions of Train and Test set ………………………………………………………………. 30
- Figure 52 Decision Tree model with best parameters …………………………………………………31
- Figure 53 Feature importances ……………………………………………………………………………………31
- Figure 54 Probability of Train and Test set ………………………………………………………………….33
- Figure 55 Random Forest model with best parameters ………………………………………………36
- Figure 56 Feature Importances ……………………………………………………………………………………38
- Figure 57 ANN model with best parameters ……………………………………………………………….38
- Figure 58 Classification report of train set (Decision Tree) ……………………………………… 41
- Figure 59 Classification report of test set (Decision Tree) …………………………………………41
- Figure 60 Classification report of train set (Random Forest) ……………………………………. 42
- Figure 61 Classification report of test set (Random Forest) ………………………………………43
- Figure 62 Classification report of train set (ANN) ……………………………………………………… 46
- Figure 63 Classification report of test set (ANN) ………………………………………………………… 47
- Figure 64 Performance metrics of all the models…………………………………………………………47
- Figure 65 AUC of all the models48……………………………………………… ………………………………48

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. You are given the task to identify the segments based on credit card usage.

Variables:

- 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)

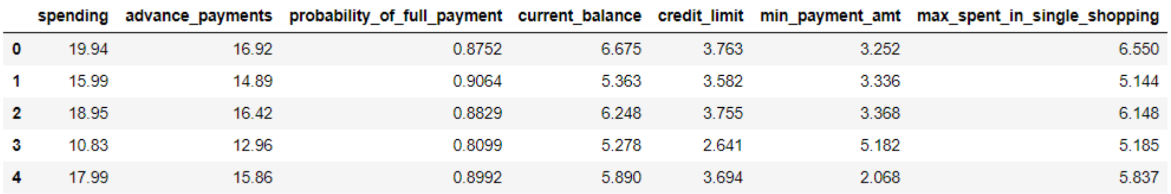
1.1 Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis).

Exploratory Data Analysis:

We will explore the Clustering Data set and perform the exploratory data analysis on the dataset. The major topics to be covered are below:

- Removing duplicates
- Missing value treatment
- Outlier Treatment
- Normalization and Scaling (Numerical Variables)
- Encoding Categorical variables (Dummy Variables)
- Univariate Analysis
- Bivariate Analysis

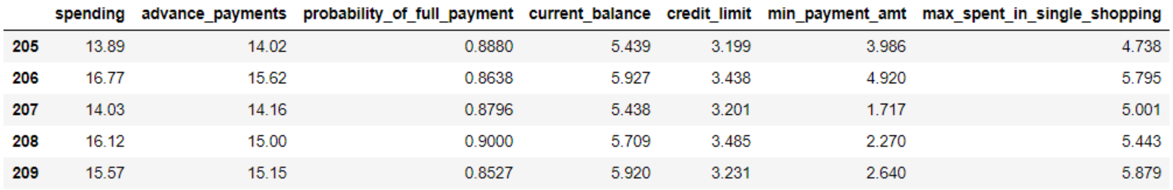
Data Insights of top 5 records:



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

Fig.1

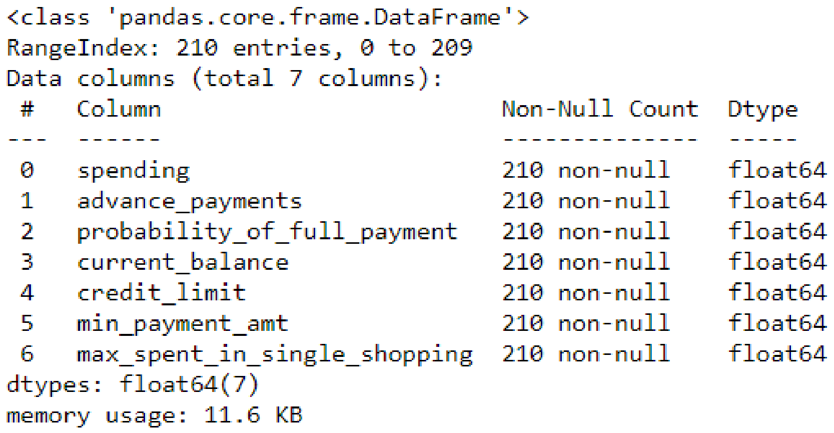
Data Insights of bottom 5 records:



	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
205	13.89	14.02	0.8880	5.439	3.199	3.986	4.738
206	16.77	15.62	0.8638	5.927	3.438	4.920	5.795
207	14.03	14.16	0.8796	5.438	3.201	1.717	5.001
208	16.12	15.00	0.9000	5.709	3.485	2.270	5.443
209	15.57	15.15	0.8527	5.920	3.231	2.640	5.879

Fig.2

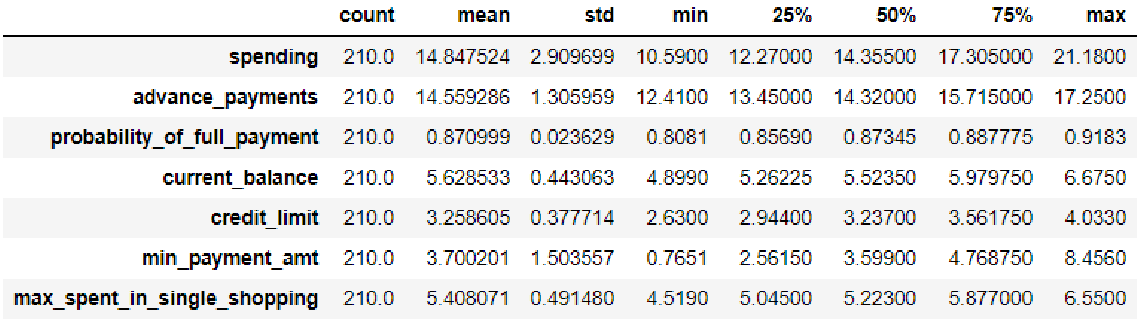
Data Info:



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

Fig.3

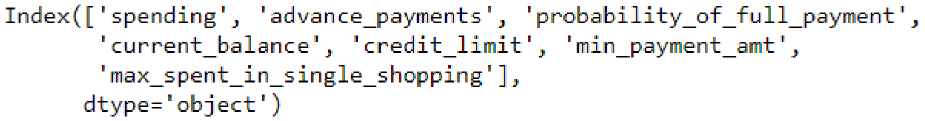
Data Summary:



	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

Fig.4

Total columns:



```
Index(['spending', 'advance_payments', 'probability_of_full_payment',
       'current_balance', 'credit_limit', 'min_payment_amt',
       'max_spent_in_single_shopping'],
      dtype='object')
```

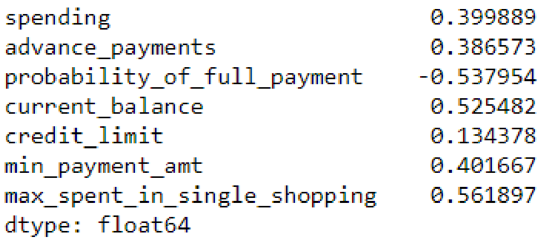
Fig.5

Data Dimensions: (210,7)

Missing values : 0

Duplicates in the data : 0

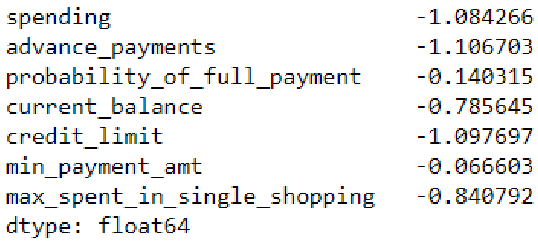
Skewness :



```
spending                        0.399889
advance_payments                0.386573
probability_of_full_payment    -0.537954
current_balance                 0.525482
credit_limit                    0.134378
min_payment_amt                 0.401667
max_spent_in_single_shopping    0.561897
dtype: float64
```

Fig.6

Kurtosis :



```
spending                        -1.084266
advance_payments                -1.106703
probability_of_full_payment     -0.140315
current_balance                 -0.785645
credit_limit                    -1.097697
min_payment_amt                 -0.066603
max_spent_in_single_shopping    -0.840792
dtype: float64
```

Fig.7

Outliers :

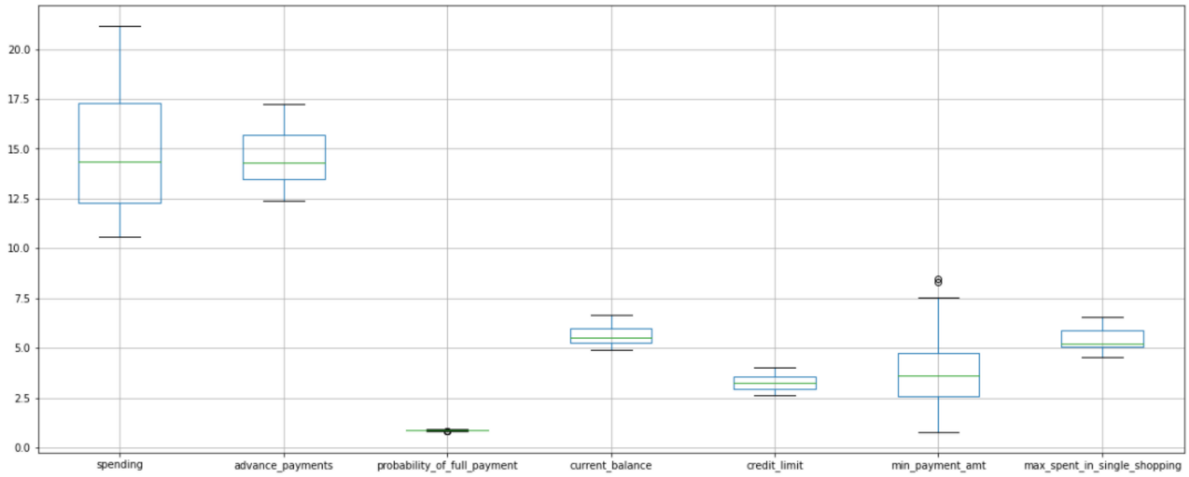
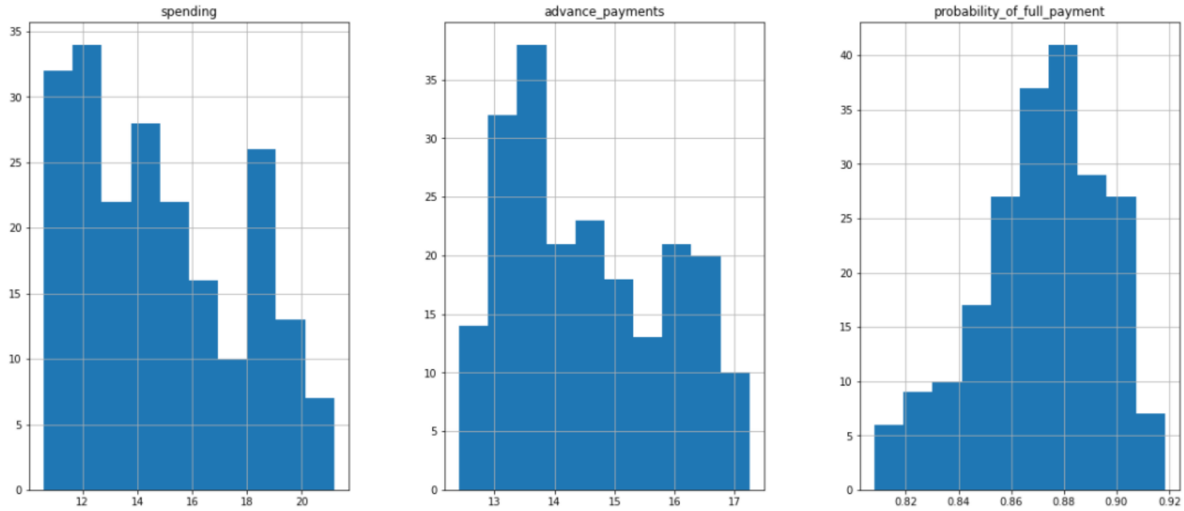


Fig.8

Histograms:



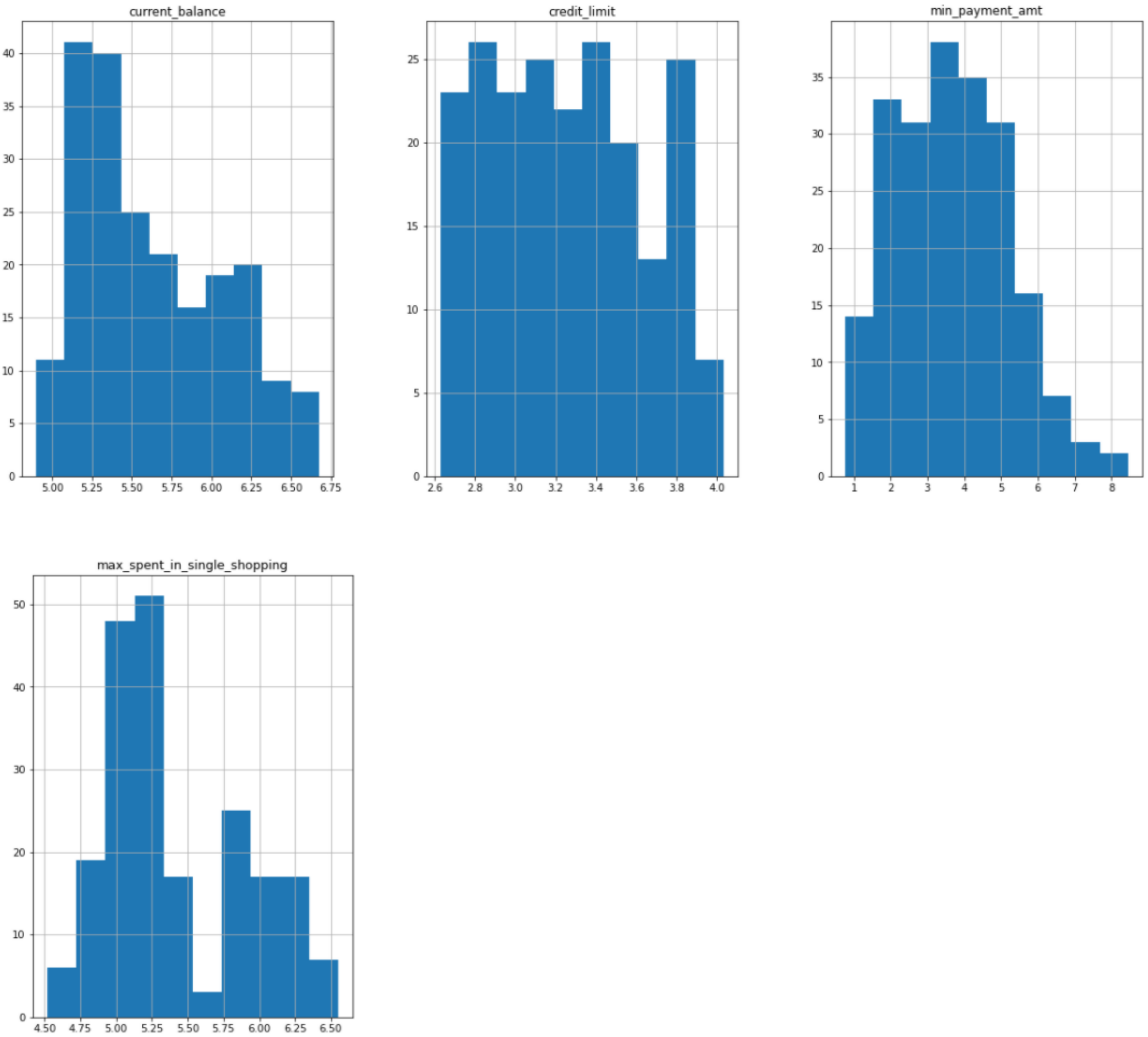


Fig.9

Inferences:

- The data consists of 210 rows and 7 columns
- No missing values found
- Data type is numeric(float64)
- There are no missing values and duplicates present in the data
- Mean and median for all variables are more or less equal
- Apart from probability of full payments (-0.5) the distribution is right skewed (0+)
- Outliers’ present in the data. As we can see their presence in probability of full payments and min payment amount.

Univeriate Analysis:

spending:

- Std in spending (2.9096)
- The maximum and minimum spending is 21.18 and 10.59 respectively
- The mean and median values are 14.8 and 14.3 respectively
- No null values present in the data
- Range (max-min): Distribution of data in spending is 10.59

Outliers proportion in spending :

- Q1 of spending is: 12.27
- Q3 of spending is: 17.305
- IQR of spending is 5.035
- Threshold for lower outliers in spending is: 4.717499999999999
- Threshold for upper outliers in spending is: 24.8575
- No. of outliers in spending upper: 0
- No. of outliers in spending lower: 0

advance_payments:

- Std in spending (1.3059)
- The maximum and minimum spending is 17.25 and 12.41 respectively
- The mean and median values are 14.55 and 14.32 respectively
- No null values present in the data
- Range (max-min): Distribution of data in spending is 4.84

Outliers proportion in advance_payments:

- Q1 of spending is: 13.45
- Q3 of spending is: 15.715
- IQR of spending is 2.2650000000000006
- Threshold for lower outliers in spending is: 10.052499999999998
- Threshold for upper outliers in spending is: 19.1125
- No. of outliers in spending upper: 0
- No. of outliers in spending lower: 0

probability_of_full_payment:

- Std in probability_of_full_payment (0.02)
- The maximum and minimum spending is 0.9 and 0.8 respectively
- The mean and median values are 0.8 and 0.8 respectively
- No null values present in the data
- Range (max-min): Distribution of data in spending is 0.11019999999999996

Outliers proportion in probability_of_full_payment:

- Q1 of probability_of_full_payment is: 0.8569
- Q3 of probability_of_full_payment is: 0.887775
- IQR of probability_of_full_payment is 0.030874999999999986
- Threshold for lower outliers in advance_payments is: 0.8105875
- Threshold for upper outliers in advance_payments is: 0.9340875
- No. of outliers in probability_of_full_payment upper : 0
- No. of outliers in probability_of_full_payment lower : 3

current_balance:

- Std in current_balance (0.4)
- The maximum and minimum current_balance is 6.67 and 4.89 respectively
- The mean and median values are 5.6 and 5.5 respectively
- No null values present in the data
- Range (max-min): Distribution of data in spending is 1.7759999999999998

Outliers proportion in current_balance:

- Q1 of current_balance is: 5.26225
- Q3 of current_balance is: 5.97975
- IQR of current_balance is 0.7175000000000002
- Threshold for lower outliers in current_balance is: 4.186
- Threshold for upper outliers in current_balance is: 7.056000000000001
- No. of outliers in current_balance upper : 0
- No. of outliers in current_balance lower : 0

credit_limit:

- Std in credit_limit (0.3)
- The maximum and minimum credit_limit is 4.03 and 2.6 respectively
- The mean and median values are 3.2 and 3.2 respectively
- No null values present in the data
- Range (max-min): Distribution of data in credit_limit is 1.4030000000000005

Outliers proportion in credit_limit:

- Q1 of credit_limit is: 2.944
- Q3 of credit_limit is: 3.56175
- IQR of credit_limit is 0.61775
- Threshold for lower outliers in credit_limit is: 2.017375
- Threshold for upper outliers in credit_limit is: 4.488375
- No. of outliers in credit_limit upper : 0
- No. of outliers in credit_limit lower : 0

min_payment_amt:

- Std in min_payment_amt (1.5)
- The maximum and minimum min_payment_amt is 8.45 and 0.76 respectively
- The mean and median values are 3.7 and 3.5 respectively
- No null values present in the data

Outliers proportion in min_payment_amt:

- Q1 of min_payment_amt is: 2.5614999999999997
- Q3 of min_payment_amt is: 4.76875
- IQR of min_payment_amt is 2.20725
- Threshold for lower outliers in min_payment_amt is: -0.7493750000000006
- Threshold for upper outliers in min_payment_amt is: 8.079625
- No. of outliers in min_payment_amt upper : 2
- No. of outliers in min_payment_amt lower : 0

max_spent_in_single_shopping:

- Std in max_spent_in_single_shopping (0.49)
- The maximum and minimum max_spent_in_single_shopping is 6.55 and 4.5 respectively
- The mean and median values are 5.4 and 5.2 respectively
- No null values present in the data

Outliers proportion in max_spent_in_single_shopping:

- Q1 of max_spent_in_single_shopping is: 5.045
- Q3 of max_spent_in_single_shopping is: 5.877000000000001
- IQR of max_spent_in_single_shopping   is 0.8320000000000007
- Threshold for lower outliers in max_spent_in_single_shopping   is: 3.796999999999999

- Threshold for upper outliers in max_spent_in_single_shopping is: 7.125000000000002
- No. of outliers in max_spent_in_single_shopping upper : 0
- No. of outliers in max_spent_in_single_shopping lower : 0

<u>Multivariate Analysis:</u>

Correlation:

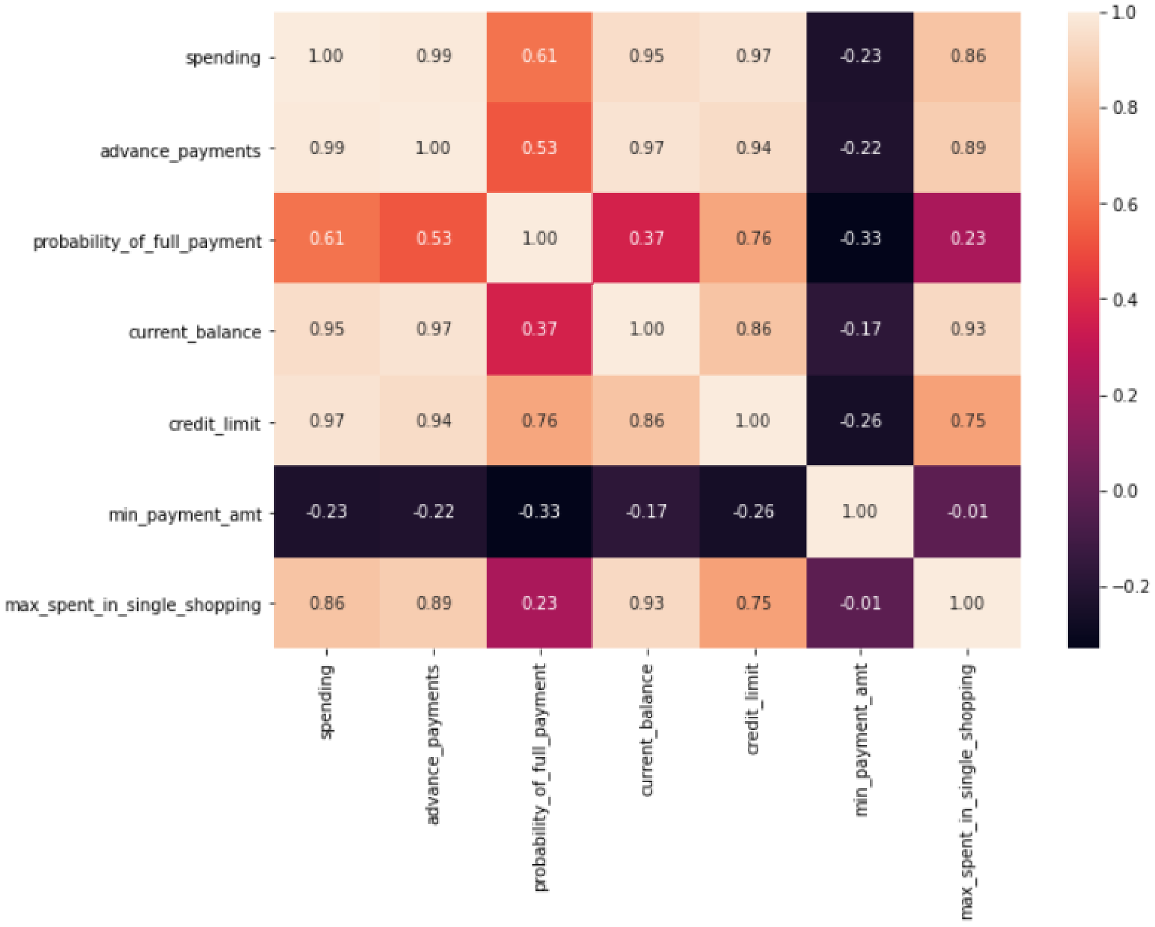
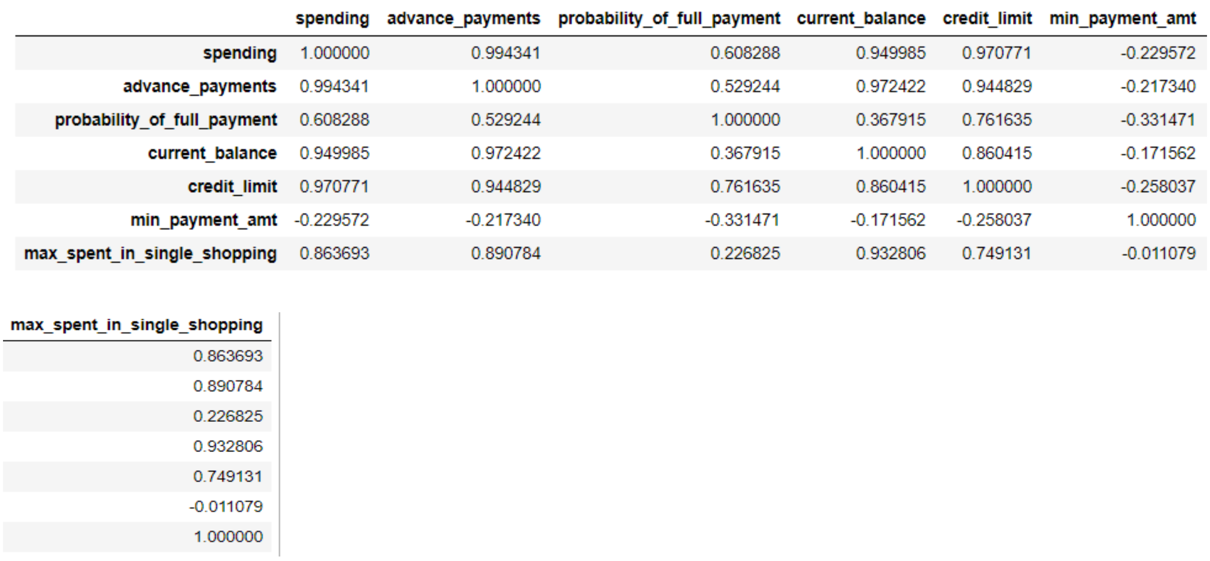


Fig.10

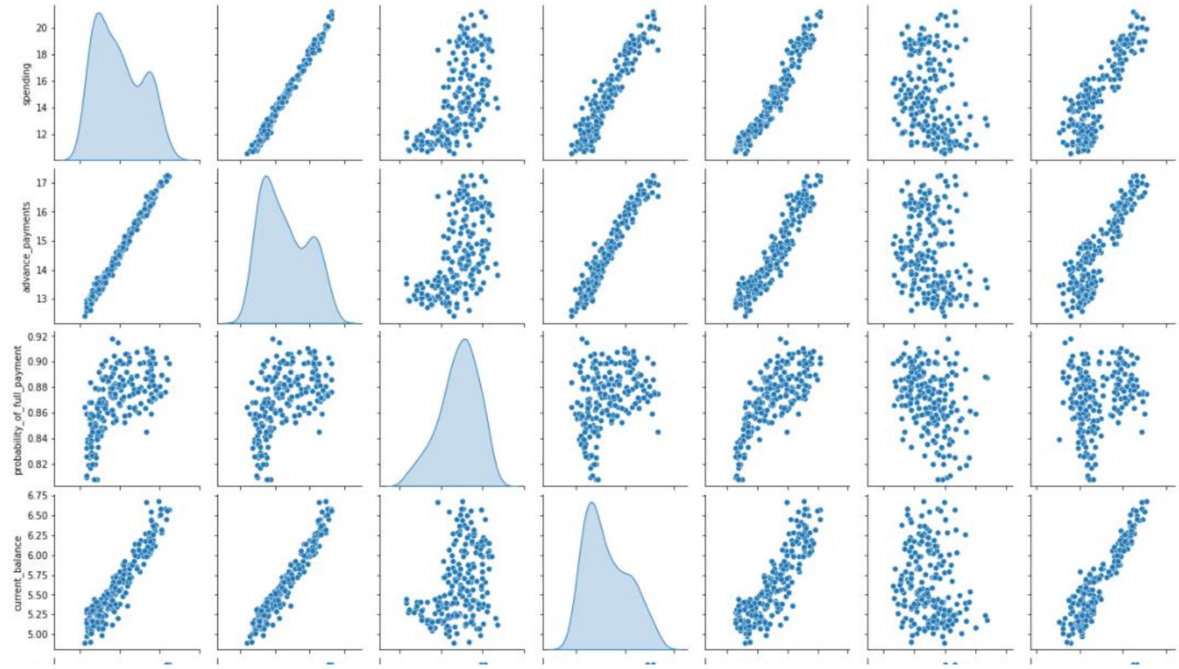
Correlation Data frame:



	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
spending	1.000000	0.994341	0.608288	0.949985	0.970771	-0.229572	0.863693
advance_payments	0.994341	1.000000	0.529244	0.972422	0.944829	-0.217340	0.890784
probability_of_full_payment	0.608288	0.529244	1.000000	0.367915	0.761635	-0.331471	0.226825
current_balance	0.949985	0.972422	0.367915	1.000000	0.860415	-0.171562	0.932806
credit_limit	0.970771	0.944829	0.761635	0.860415	1.000000	-0.258037	0.749131
min_payment_amt	-0.229572	-0.217340	-0.331471	-0.171562	-0.258037	1.000000	-0.011079
max_spent_in_single_shopping	0.863693	0.890784	0.226825	0.932806	0.749131	-0.011079	1.000000

Fig.11

Pair plot:



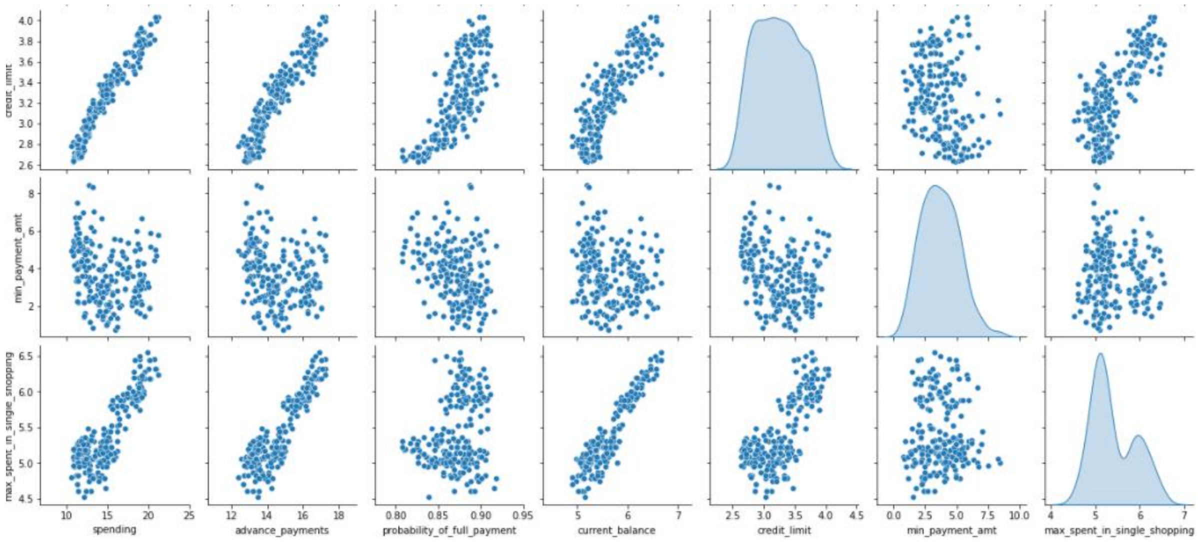


Fig.12

Inferences:

From the above fig we can say that there is a strong positive correlation between:

- spending & advance_payments : 0.994341
- advance_payments & current_balance : 0.972422
- credit_limit & spending : 0.970771
- spending & current_balance : 0.949985
- credit_limit & advance_payments : 0.944829
- max_spent_in_single_shopping & current_balance : 0.932806
- advance_payments & max_spent_in_single_shopping : 0.890784
- spending & max_spent_in_single_shopping : 0.863693
- current_balance & credit_limit : 0.860415

And there is negative correlation associated with min_payment_amt.

Treating the outliers:

Outliers has been treated for probability_of_full_payment,min_payment_amt.

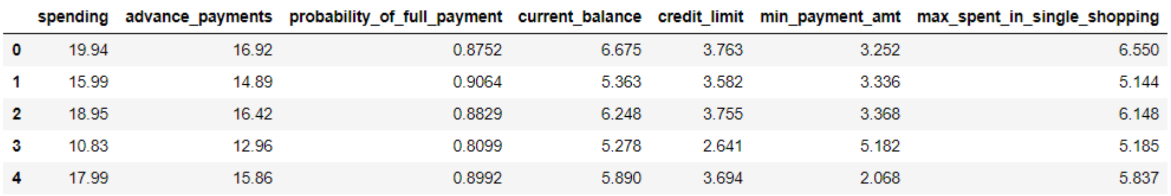
- Q1 of min_payment_amt is: 2.5614999999999997
- Q3 of min_payment_amt is: 4.76875
- IQR of min_payment_amt is 2.20725
- Threshold for lower outliers in min_payment_amt is: -0.7493750000000006
- Threshold for upper outliers in min_payment_amt is: 8.079625
- No. of outliers in min_payment_amt upper : 0
- No. of outliers in min_payment_amt lower : 0

- Q1 of probability_of_full_payment is: 0.8569
- Q3 of probability_of_full_payment is: 0.887775
- IQR of probability_of_full_payment is 0.030874999999999986
- Threshold for lower outliers in probability_of_full_payment is: 0.8105875
- Threshold for upper outliers in probability_of_full_payment is: 0.9340875
- No. of outliers in probability_of_full_payment upper : 0
- No. of outliers in probability_of_full_payment lower : 0

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

- Scaling also called as centring is very important for clustering because of the way that the variables are calculated
- Yes, scaling is necessary for clustering in this case.
- I have used z score to standardized the data.
- Variables in different units must be scaled.
- Variables in the same units but with very different variances are usually scaled.
- Simplest scaling: divide each variable by its standard deviation ⇒ covariances are correlations.
- It calculates a new projection of the dataset and the axis for this is dependent on the standard deviation of the data.
- Scaling helps to standardize the data. The standard deviation of the data becomes 1 after scaling.
- At the end of this process, the data is in the form of an array.
- Hence, it is converted into a data frame and used for further analysis.

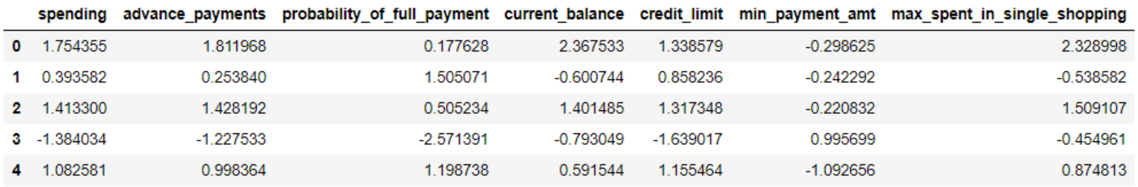
Data before scaling:



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

Fig.13

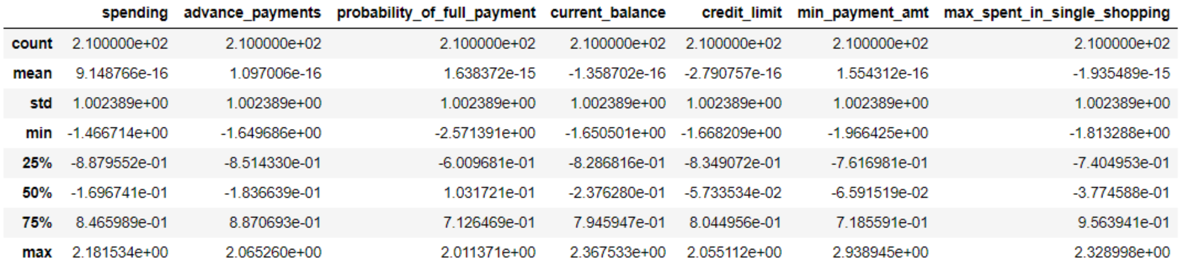
Data after scaling:



	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
0	1.754355	1.811968	0.177628	2.367533	1.338579	-0.298625	2.328998
1	0.393582	0.253840	1.505071	-0.600744	0.858236	-0.242292	-0.538582
2	1.413300	1.428192	0.505234	1.401485	1.317348	-0.220832	1.509107
3	-1.384034	-1.227533	-2.571391	-0.793049	-1.639017	0.995699	-0.454961
4	1.082581	0.998364	1.198738	0.591544	1.155464	-1.092656	0.874813

Fig.14

Scaled data summary :



	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
count	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02	2.100000e+02
mean	9.148766e-16	1.097006e-16	1.638372e-15	-1.358702e-16	-2.790757e-16	1.554312e-16	-1.935489e-15
std	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00	1.002389e+00
min	-1.466714e+00	-1.649686e+00	-2.571391e+00	-1.650501e+00	-1.668209e+00	-1.966425e+00	-1.813288e+00
25%	-8.879552e-01	-8.514330e-01	-6.009681e-01	-8.286816e-01	-8.349072e-01	-7.616981e-01	-7.404953e-01
50%	-1.696741e-01	-1.836639e-01	1.031721e-01	-2.376280e-01	-5.733534e-02	-6.591519e-02	-3.774588e-01
75%	8.465989e-01	8.870693e-01	7.126469e-01	7.945947e-01	8.044956e-01	7.185591e-01	9.563941e-01
max	2.181534e+00	2.065260e+00	2.011371e+00	2.367533e+00	2.055112e+00	2.938945e+00	2.328998e+00

Fig.15

From “std” column we can conclude that the data has been scaled (1.002389).

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

- Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.
- It is basically a collection of objects on the basis of similarity and dissimilarity between them.
- Clustering determines the intrinsic grouping among the unlabelled data present.
- There are no criteria for a good clustering.

Hierarchical Based Methods :

- The clusters formed in this method forms a tree-type structure based on the hierarchy.
- New clusters are formed using the previously formed one. It is divided into two category
- Agglomerative (bottom up approach), Divisive (top down approach)
- In this problem I have used Agglomerative (bottom up approach).

Applying hierarchical clustering to scaled data:

- To show the outcome of hierarchical clustering I have used dendrogram for a pictorial way to visualize hierarchical clustering.
- It's a tree like diagram that records the sequences of merges and splits.
- Used Ward's method where the distance between two clusters is based on the similarity calculated as the sum of square of the of the distances. It works on the sum of squares.

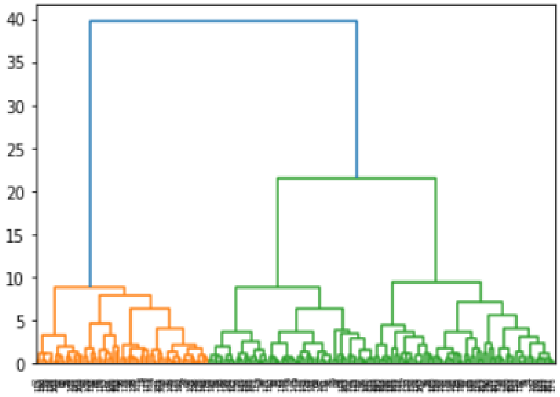


Fig.16

Inferences:

1. Considered each data points as cluster.
2. Calculated the proximity matrix.
3. Joined the two closet clusters based on proximity value and recomputed the proximity matrix.
4. Repeated step 3 until all points are merged into one cluster

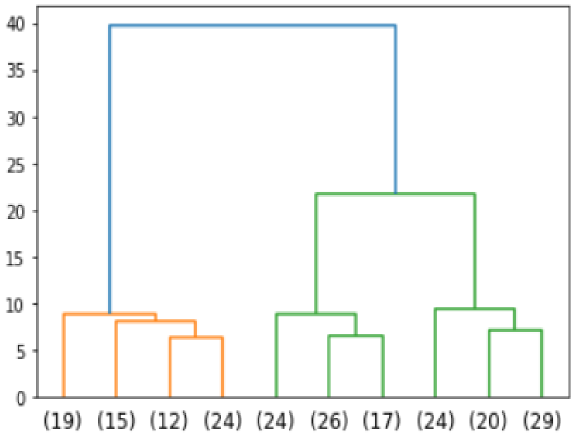
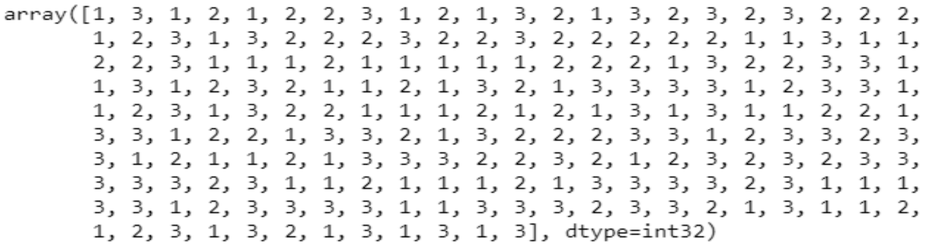


Fig.17

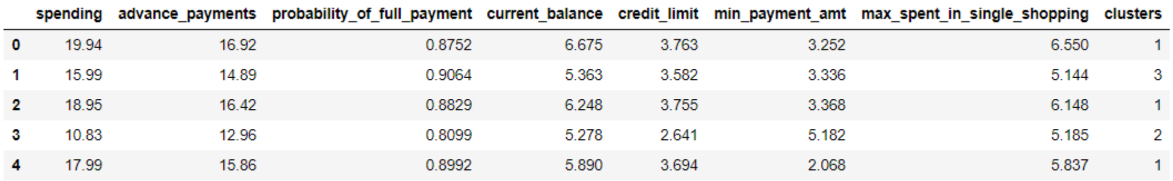
- Used fcluster to flatten the dendrogram and obtained as a result an assignation of the original data points to single clusters



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

Fig.18

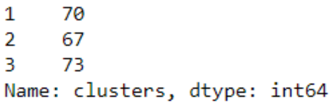
- I have Set criterion as maxclust and then created 3 clusters and stored the result in another object 'clusters'



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

Fig.19

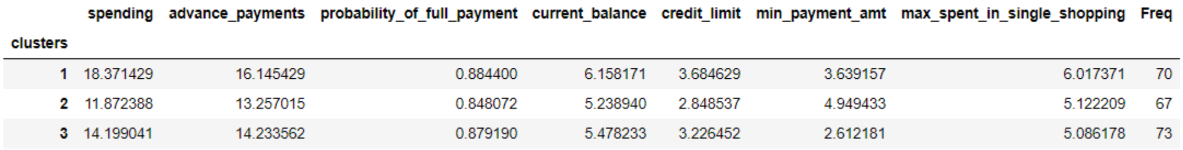
Dividing the clusters (Cluster Frequency)



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

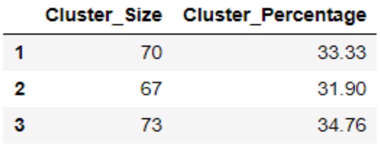
Fig.20

- Cluster profiling: Profiling involves generating descriptions of the clusters with reference to the input variables we used for the cluster analysis.



	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Freq
clusters								
1	18.371429	16.145429	0.884400	6.158171	3.684629	3.639157	6.017371	70
2	11.872388	13.257015	0.848072	5.238940	2.848537	4.949433	5.122209	67
3	14.199041	14.233562	0.879190	5.478233	3.226452	2.612181	5.086178	73

Fig.21



	Cluster_Size	Cluster_Percentage
1	70	33.33
2	67	31.90
3	73	34.76

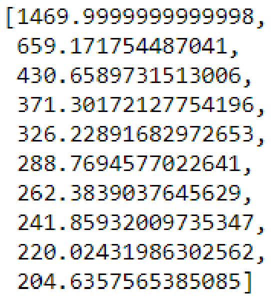
Fig.22

Inferences:

- From the fig.21 we can see that distribution of the clusters of
- spending (18.37 for cluster 1, 11.8 for cluster 2, 14.199 for cluster 3)
- advance_payments(16.14,13.25,14.23 for cluster 1,2,3 respectively)
- Probability of full payment (0.88,0.84,0.87 for cluster 1,2,3 respectively)
- Current balance (6.1,5.2,5.4 for cluster 1,2,3 respectively)
- Credit limit (3.6,2.8,3.2 for cluster 1,2,3 respectively) etc.
- We can see the distribution of 3 clusters follows (From fig.22):
- For the cluster 1 the cluster size is 70 and the percentage of variables is 33.33
- For the cluster 2 the cluster size is 67 and the percentage of variables is 31.90
- For the cluster 3 the cluster size is 73 and the percentage of variables is 34.76
- Hence, from the dendrogram if we look truncated value and draw a line at 18 approx., we can see that 3 vertical lines can be taken into consideration.
- By using distance method, we can say that 3 clusters are sufficient.

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 is one of the simplest unsupervised learning algorithms that solve the well known clustering problem.
- The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori.
- The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result.
- So, the better choice is to place them as much as possible far away from each other.
- The next step is to take each point belonging to a given data set and associate it to the nearest center.
- When no point is pending, the first step is completed and an early group age is done.
- At this point we need to re-calculate k new centroids as barycentre of the clusters resulting from the previous step.
- After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center.
- A loop has been generated.
- As a result of this loop, we may notice that the k centers change their location step by step until no more changes are done.



```
[1469.9999999999998,
 659.171754487041,
 430.6589731513006,
 371.30172127754196,
 326.22891682972653,
 288.7694577022641,
 262.3839037645629,
 241.85932009735347,
 220.02431986302562,
 204.6357565385085]
```

Fig.23

The working of k-means clustering:

- Initialized the K random centroids or k points.
- Picked random data points and considered those as starting points.
- Took random values for each particular variable.
- For each data point calculated the distance of it from randomly chosen K centroid and assign each point to minimum distance cluster.
- Updated the centroid by using newly assigned data points to the cluster by calculating the average of data points
- Repeated the above process for a given no. of iterations or until the centroid allocation no longer changes.
- The algorithm is said to be converged once there are no more changes in the values of centroids.
- The objective of clustering is to minimize the distance between data points and its centroid, which can also be expressed as square error term.
- Choosing value of k:
- Basically, there is no such method for define the exact value of k,but we can be used as heuristic to get the maximum number of clusters as k=Square root n/2.
- Elbow Method,Silhouette method used to find the approximate or optimal value of k.
- Imported KMeans from sklearn
- Performed KMeans algorithm for range(1,11)
- For KMeans(n_clusters=1) the value is 1469.99
- For KMeans(n_clusters=2) the value is 659.17
- For KMeans(n_clusters=3) the value is 430.65
- For KMeans(n_clusters=11) the value is 204
- Found KMeans inertia as 430.65

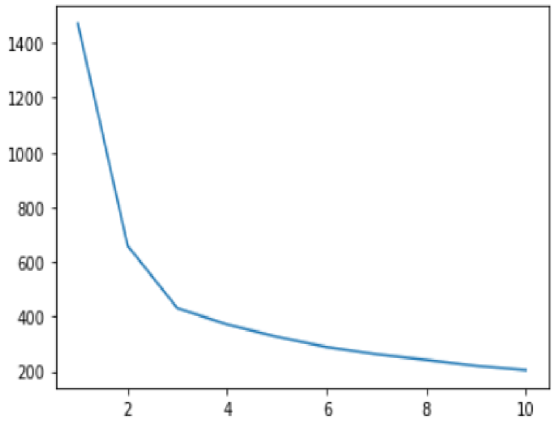
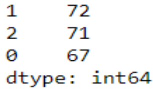


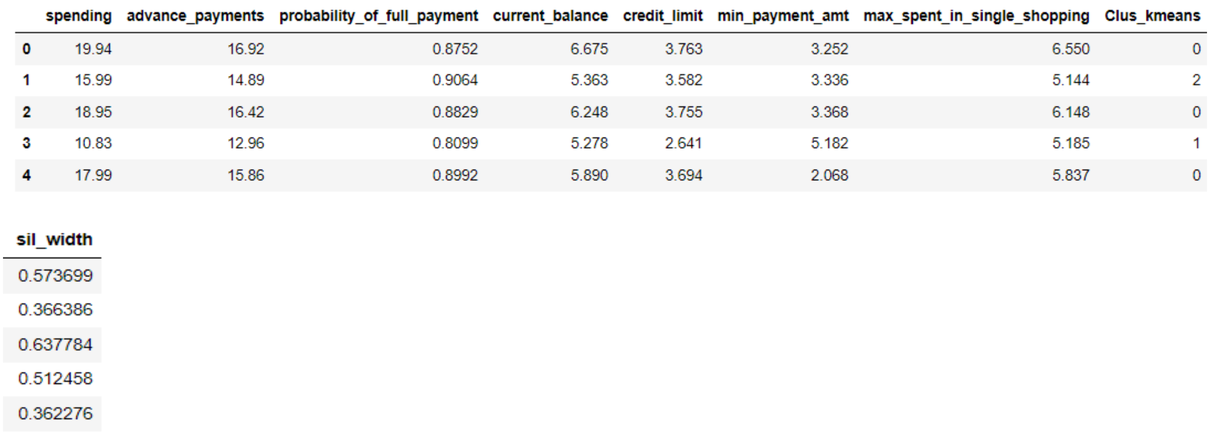
Fig.24

Dividing the clusters (Cluster Frequency)



```
1    72
2    71
0    67
dtype: int64
```

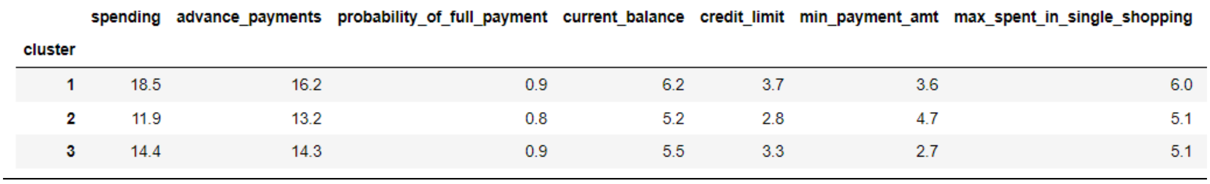
Fig.25



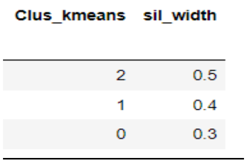
	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_kmeans	sil_width
0	19.94	16.92	0.8752	6.675	3.763	3.252	6.550	0	0.573699
1	15.99	14.89	0.9064	5.363	3.582	3.336	5.144	2	0.366386
2	18.95	16.42	0.8829	6.248	3.755	3.368	6.148	0	0.637784
3	10.83	12.96	0.8099	5.278	2.641	5.182	5.185	1	0.512458
4	17.99	15.86	0.8992	5.890	3.694	2.068	5.837	0	0.362276

Fig.26

Cluster profiling:



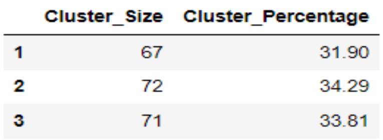
	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
cluster							
1	18.5	16.2	0.9	6.2	3.7	3.6	6.0
2	11.9	13.2	0.8	5.2	2.8	4.7	5.1
3	14.4	14.3	0.9	5.5	3.3	2.7	5.1



Clus_kmeans	sil_width
2	0.5
1	0.4
0	0.3

Fig.27

Cluster size and distribution percentage:



	Cluster_Size	Cluster_Percentage
1	67	31.90
2	72	34.29
3	71	33.81

Fig.28

silhouette_samples, silhouette_score : 0.4007270552751299,0.002713089347678533 respectively

Elbow method :

- This method is based of plotting the value of cost function against different values of k.
- As the number of clusters (k) increase lesser number of points fall with in clusters or around the centroids.
- Hence the average distortion decreases with the increase of number of clusters.
- The point where the distortion declines most is said to be the elbow point and define the optimal number of clusters for dataset.

Silhouette Method:

- Silhouette is a different method to determine optimal number of clusters for given dataset.
- It defines as a coefficient of measure of how similar an observation to its own cluster compared to that of other clusters.
- The range of silhouette coefficient varies between -1 to 1.1 value indicate that an observation is far from its neighbouring cluster and close to its own whereas -1 denotes that an observation is close to neighbouring cluster than its own cluster.
- The 0 value indicate the presence of observation on boundary of two clusters.
- We built the K mean model using K means cluster.
- We fitted the scaled data into K means model.
- We are able to see that the cluster mapping for the variables.
- 0 – cluster 1
- 1 – cluster 2
- 2 – cluster 3

- If we look at the K Mean inertia, it’s the total WSS when K = 3 it gives the value 430.658
- We can try for different clusters and find the inertia as given below. The larger the drop in WSS it better.
- If the drop is not significant the additional cluster is not useful for us.
- From 1 cluster to 2 cluster we have a significant drop close to 900 points
- From 2 cluster to 3 cluster we do have a good drop close to 240 points
- From 3 cluster to 4 cluster it’s not a significant only 50 points drop
- From 4 cluster to 5 cluster it’s not a significant only 50 points drop
- By looking at the graph it’s also evident that drop is significant for 1 ,2 and 3.
- post that the drop is very minimal.
- By using silhouette score and Silhouette analysis we can find that the mapping of each variable to the specific cluster is valid or invalid.
- The smallest value of the silhouette width is 0.002 this indicates that no observation are wrongly mapped to a cluster. SW is a validation index to estimate the number of clusters. Shows the optimal one.
- We can see the distribution of 3 clusters follows (From fig.28):
- For the cluster 1 the cluster size is 67 and the percentage of variables is 31.90
- For the cluster 2 the cluster size is 72 and the percentage of variables is 34.29
- For the cluster 3 the cluster size is 71 and the percentage of variables is 33.81
- spending (18.5 for cluster 1, 11.9 for cluster 2, 14.4 for cluster 3)
- advance_payments(16.2,13.2,14.3 for cluster 1,2,3 respectively)
- Probability of full payment (0.9,0.8,0.9 for cluster 1,2,3 respectively)
- Current balance (6.2,5.2,5.5 for cluster 1,2,3 respectively)
- Credit limit (3.7,2.8,3.3 for cluster 1,2,3 respectively)
- min payment amount (3.6,4.7,2.7 for cluster 1,2,3 respectively)
- max spent in single shopping (6.0,5.1,5.1 for cluster 1,2,3 respectively)
- The main Advantages of K-means clustering is:
- Ease of implementation. It works great on large scale data. Results guarantees convergence. Easily works with new examples.

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

Inferences from Hierarchical Clustering:

- For Cluster 1, Spending is on higher side, averaging 18.371.
- Probability of Full Payment is very high, averaging around 0.8844.
- It has the Highest advance payments around 16.14.
- Current Balance is around 6.158 which is highest among three clusters.
- Credit Limit is also high for this cluster ranging around 3.6846.
- max_spent_in_single_shopping is around 6.017. Which is higher comparatively to other clusters.

- For Cluster 2, this is average Spending cluster is on lower side, averaging 11.872.
- Probability of Full Payment is the least amongst other clusters, averaging around 0.848.
- Current Balance is around 5.238 which is least among three clusters.
- Credit Limit is least for this cluster ranging around 2.8485.
- minimum payment amount is found max in this cluster, it is around 4.94.
- Max_spent_in_single_shopping is around 5.122.

- For Cluster 3, Spending is 14.199.
- Probability of Full Payment is little on the higher side, averaging around 0.879.
- Current Balance is around 5.478 which is average among three clusters.
- Credit Limit is ranging in between for this cluster ranging around 3.2264.
- minimum payment amount is found least in this cluster, it is around 2.61.
- Max_spent_in_single_shopping is the least around 5.086.

Inferences from K-Means Clustering:

- For Cluster 1, Spending is highest amongst other clusters, averaging 18.5.
- Probability of Full Payment is little on the higher side, averaging around 0.9.
- Current Balance is around 6.2 which is highest among three clusters.
- Credit Limit is ranging in between for this cluster ranging around 3.7.
- It is highest amongst three clusters.
- minimum payment amount is found least in this cluster, it is around 3.6.
- Max_spent_in_single_shopping is the highest around 6.0.
- For Cluster 2, this is least Spending cluster, averaging 11.9.
- Probability of Full Payment is the least amongst other clusters, averaging around 0.8.
- Current Balance is around 5.2 which is least among three clusters.
- Credit Limit is least for this cluster ranging around 2.8.

- minimum payment amount is found max in this cluster, it is around 4.7.
- Max_spent_in_single_shopping is around 5.1.

- For Cluster 3, Spending is average spending cluster, averaging 14.4.
- Probability of Full Payment is least among other Clusters, averaging around 0.9.
- It has the Highest advance payments around 14.3.
- Current Balance is around 5.5.
- It is average when compared to other two clusters.
- Credit Limit is for this cluster ranging around 3.3.
- Minimum payment is averaging 2.7.
- max_spent_in_single_shopping is around 5.1.

Promotions for Clustering:

- Bank should focus on Cluster 1 as the customers in this cluster have higher spending.
- Bank can think of providing them offers like, for shopping if they spend more than their current maximum spending in single shopping. And also giving Personalized services,Rewards Programs, Lifestyle benefits by providing coupons etc
- Cluster 2 spends the least reason might be less Credit limit.
- There are probabilities that these customers may increase spending if their Credit limit increases by providing No annual fee,Merchant Category Offers,Cashback offers,Balance Transfer Offers etc.
- Cluster 3 has average spending. Bank should give customers in this Cluster more promotional offers because there are more chances that these customers may move to Cluster 1 of high spending.
- Cluster 2 and 3 have lesser maximum one-time spending compared to Cluster 1, which has highest max one-time spending.
- Hence Bank should promote customers in Cluster 2 & 3 for more effect of the promotional offers.
- Bank should focus on Cluster 3 as the customers in this cluster have higher spending.
- Bank can think of providing them offers like, for shopping if they spend more than their current maximum spending in single shopping, Use and get offers,Sign-up rewards,Reward accelerators etc
- Cluster 2 spends the least reason might be less Credit limit.
- There are probabilities that these customers may increase spending if their Credit limit increases.
- Cluster 3 has average spending.
- Bank should give customers in this Cluster more promotional offers because there are more chances that these customers may move to Cluster 1 of high spending.
- Cluster 1 and 2 have lesser maximum one-time spending compared to Cluster 3, which has highest max one-time spending.
- Hence Bank should promote customers in Cluster 1 & 2 for more effect of the promotional offers.

Problem 2:

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

Variables:

1. Target: Claim Status (Claimed)
2. Code of tour firm (Agency_Code)
3. Type of tour insurance firms (Type)
4. Distribution channel of tour insurance agencies (Channel)
5. Name of the tour insurance products (Product)
6. Duration of the tour (Duration)
7. Destination of the tour (Destination)
8. Amount of sales of tour insurance policies (Sales)
9. The commission received for tour insurance firm (Commission)
10. Age of insured (Age)

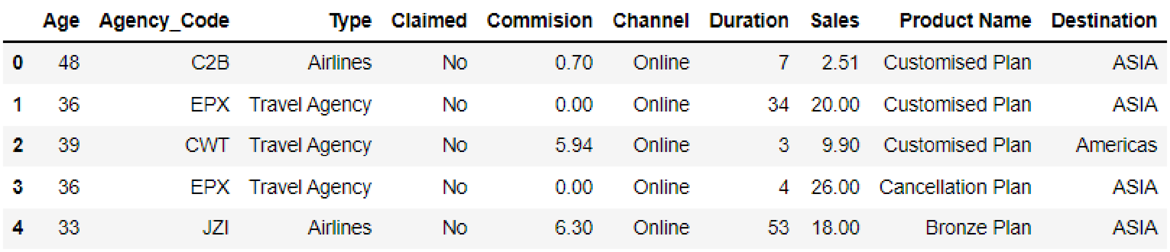
2.1 Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis).

Exploratory Data Analysis:

We will explore the Clustering Data set and perform the exploratory data analysis on the dataset. The major topics to be covered are below:

- Removing duplicates
- Missing value treatment
- Outlier Treatment
- Normalization and Scaling (Numerical Variables)
- Encoding Categorical variables (Dummy Variables)
- Univariate Analysis
- Bivariate Analysis

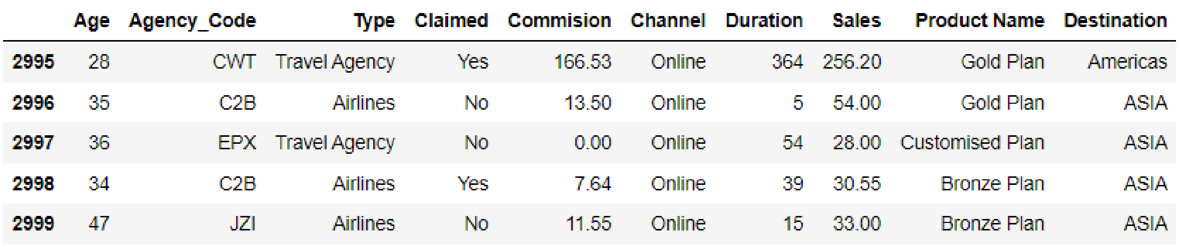
Data Insights of top 5 records:



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

Fig.29

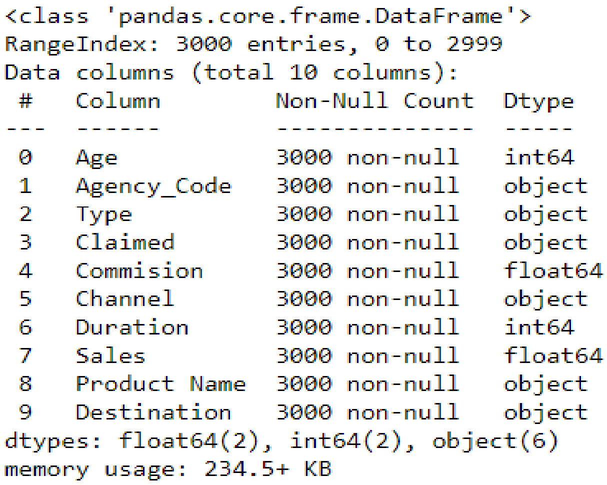
Data Insights of bottom 5 records:



	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
2995	28	CWT	Travel Agency	Yes	166.53	Online	364	256.20	Gold Plan	Americas
2996	35	C2B	Airlines	No	13.50	Online	5	54.00	Gold Plan	ASIA
2997	36	EPX	Travel Agency	No	0.00	Online	54	28.00	Customised Plan	ASIA
2998	34	C2B	Airlines	Yes	7.64	Online	39	30.55	Bronze Plan	ASIA
2999	47	JZI	Airlines	No	11.55	Online	15	33.00	Bronze Plan	ASIA

Fig.30

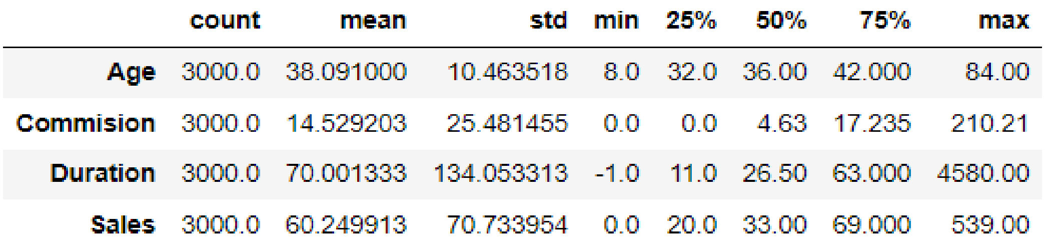
Data Info:



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

Fig.31

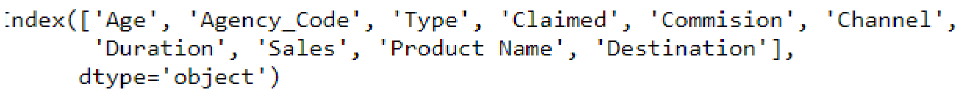
Data Summary:



	count	mean	std	min	25%	50%	75%	max
Age	3000.0	38.091000	10.463518	8.0	32.0	36.00	42.000	84.00
Commision	3000.0	14.529203	25.481455	0.0	0.0	4.63	17.235	210.21
Duration	3000.0	70.001333	134.053313	-1.0	11.0	26.50	63.000	4580.00
Sales	3000.0	60.249913	70.733954	0.0	20.0	33.00	69.000	539.00

Fig.32

Total columns:



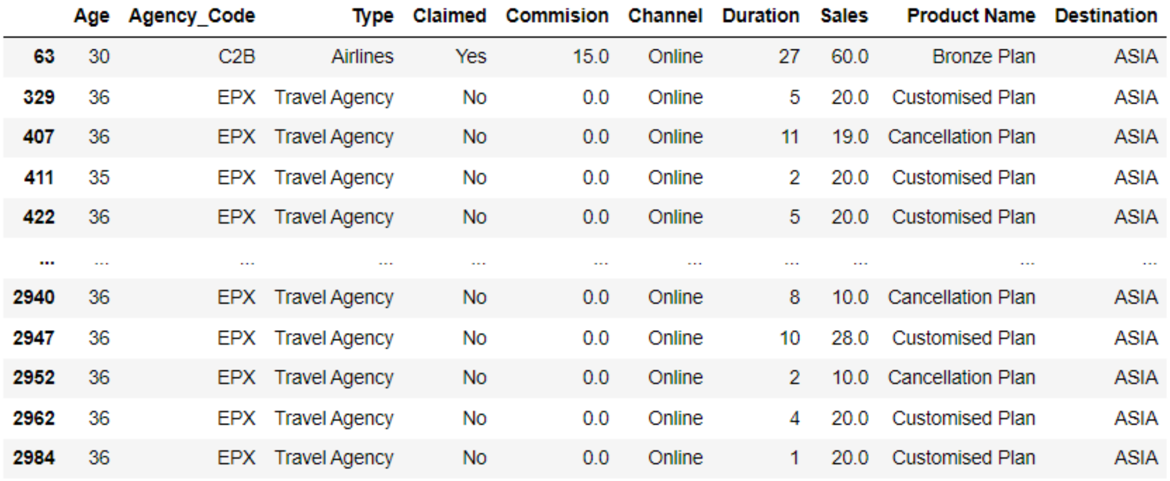
```
Index(['Age', 'Agency_Code', 'Type', 'Claimed', 'Commision', 'Channel',
       'Duration', 'Sales', 'Product Name', 'Destination'],
      dtype='object')
```

Fig.33

Data Dimensions: (3000,10)

Missing values : 0

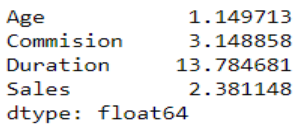
Duplicates in the data : 139



	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
63	30	C2B	Airlines	Yes	15.0	Online	27	60.0	Bronze Plan	ASIA
329	36	EPX	Travel Agency	No	0.0	Online	5	20.0	Customised Plan	ASIA
407	36	EPX	Travel Agency	No	0.0	Online	11	19.0	Cancellation Plan	ASIA
411	35	EPX	Travel Agency	No	0.0	Online	2	20.0	Customised Plan	ASIA
422	36	EPX	Travel Agency	No	0.0	Online	5	20.0	Customised Plan	ASIA
...	...	...	...	...	...	...	...	...	...	...
2940	36	EPX	Travel Agency	No	0.0	Online	8	10.0	Cancellation Plan	ASIA
2947	36	EPX	Travel Agency	No	0.0	Online	10	28.0	Customised Plan	ASIA
2952	36	EPX	Travel Agency	No	0.0	Online	2	10.0	Cancellation Plan	ASIA
2962	36	EPX	Travel Agency	No	0.0	Online	4	20.0	Customised Plan	ASIA
2984	36	EPX	Travel Agency	No	0.0	Online	1	20.0	Customised Plan	ASIA

Fig.34

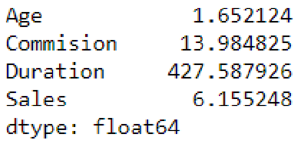
Skewness :



```
Age            1.149713
Commision      3.148858
Duration      13.784681
Sales          2.381148
dtype: float64
```

Fig.35

Kurtosis :



```
Age             1.652124
Commision      13.984825
Duration      427.587926
Sales           6.155248
dtype: float64
```

Fig.36

Outliers :

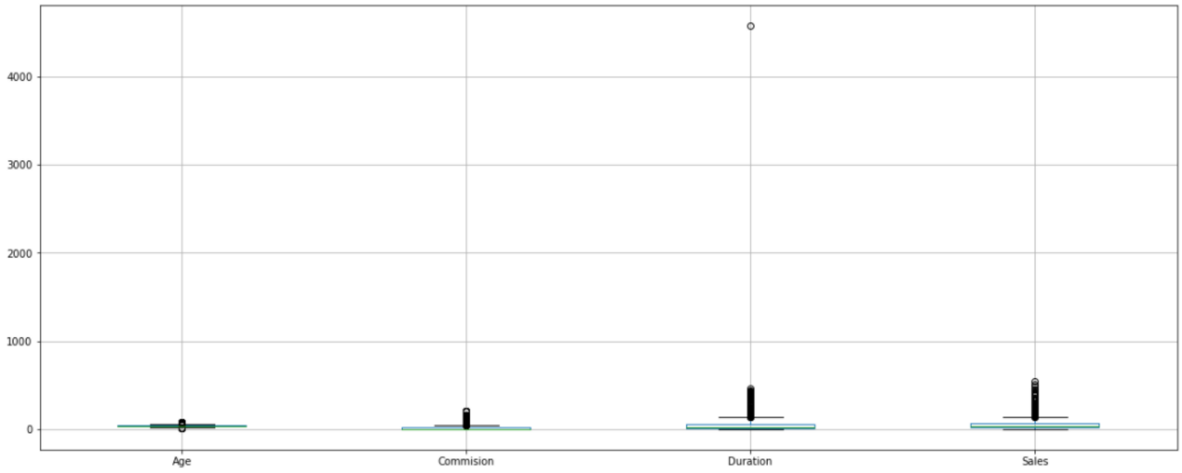


Fig.37

Histograms:

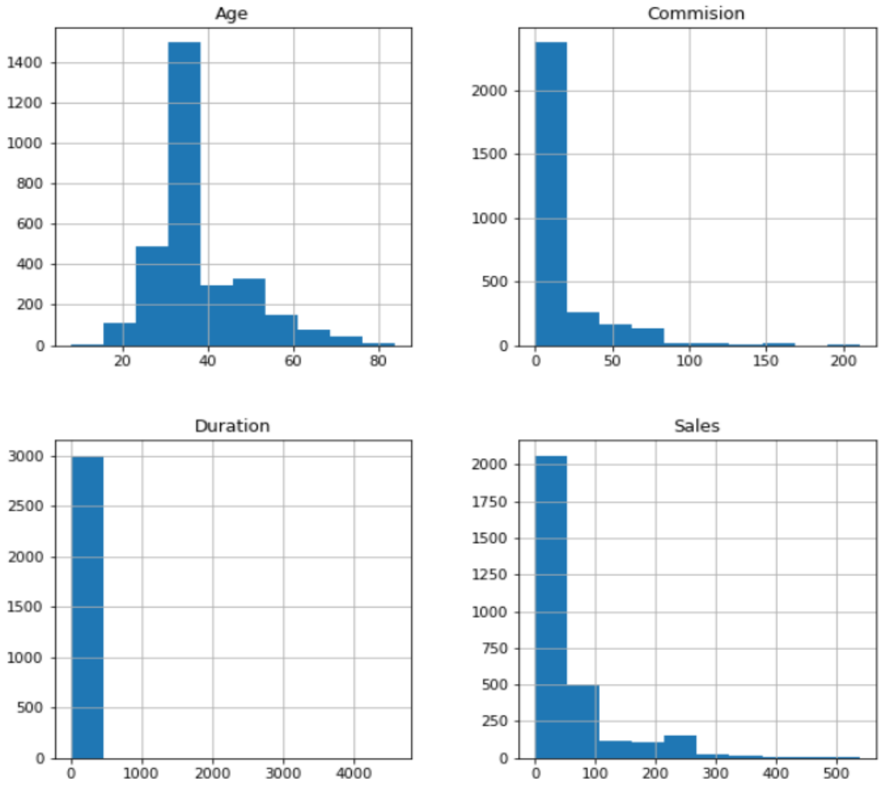


Fig.38

Inferences:

- The data consists of 3000 rows and 10 columns
- No missing values found
- Data type contains numeric(float64), categorical (object)
- There are no missing values
- Most of the rightly skewed.
- Duplicates present in the data (139)
- Manipulating the data in any way is necessary only when it is an absolute certainty that the data given contains some kind of erroneous observations.
- Here, there can be any duplicate observations as the travel company can sell the same kind of tour package to similar demography.
- 139 duplicates is huge. If we see the data set, There the columns are age,agency code,type,commission,channel etc etc. There is no info provided regarding the customer (Name/Id/unique codes etc). There can be customers of same age to book the travel package. Because there are different customers with same age.Hence, we are not sure in targeting the particular category.If we remove the duplicates we may loose the valuable data.

- There is large difference between the 75% and the Max values and also the mean and median.
- Outliers' present in the data. Treating outliers sometimes results in the models having better performance but the models lose out on the generalization. Hence, not treating the outliers

Univeriate Analysis:

Age:

- Std in age (10.46)
- The maximum and minimum age is 84 and 8 respectively
- The mean and median values are 38 and 36 respectively
- No null values present in the data

Outliers proportion in Age :

- Q1 of Age is: 32.0
- Q3 of Age is: 42.0
- IQR of Age is 10.0
- Threshold for lower outliers in Age is: 17.0
- Threshold for upper outliers in Age is: 57.0
- No. of outliers in Age upper : 198
- No. of outliers in Age lower : 6

Commission:

- Std in Commission (25)
- The maximum and minimum Commission is 210 and 0 respectively
- The mean and median values are 14 and 4 respectively
- No null values present in the data

Outliers proportion in Commission:

- Q1 of Commision is: 0.0
- Q3 of Commision is: 17.235
- IQR of Commision is 17.235
- Threshold for lower outliers in Commision is: -25.8525
- Threshold for upper outliers in Commision is: 43.0875
- No. of outliers in Commision upper : 362
- No. of outliers in Commision lower : 0

Duration:

- Std in duration (134)
- The maximum and minimum duration is 4580 and -1.0 respectively
- The mean and median values are 70 and 26 respectively
- No null values present in the data

Outliers proportion in Duration:

- Q1 of Duration is: 11.0
- Q3 of Duration is: 63.0
- IQR of Duration is 52.0
- Threshold for lower outliers in Duration is: -67.0
- Threshold for upper outliers in Duration is: 141.0
- No. of outliers in Duration upper : 382
- No. of outliers in Duration lower : 0

Sales:

- Std in sales (70)
- The maximum and minimum sales is 539 and 0 respectively
- The mean and median values are 60 and 33 respectively
- No null values present in the data

Outliers proportion in Sales:

- Q1 of Sales is: 20.0
- Q3 of Sales is: 69.0
- IQR of Sales is 49.0
- Threshold for lower outliers in Sales is: -53.5
- Threshold for upper outliers in Sales is: 142.5
- No. of outliers in Sales upper : 353
- No. of outliers in Sales lower : 0

Countplot/Boxplot for Categorical variables :

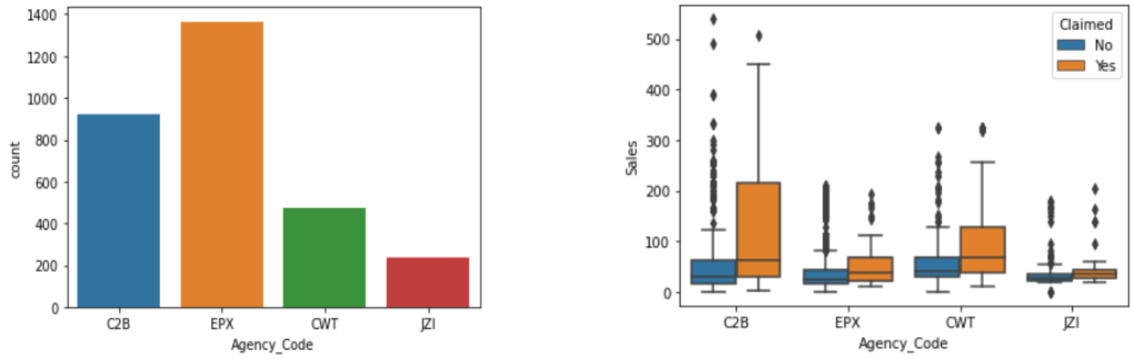


Fig:39

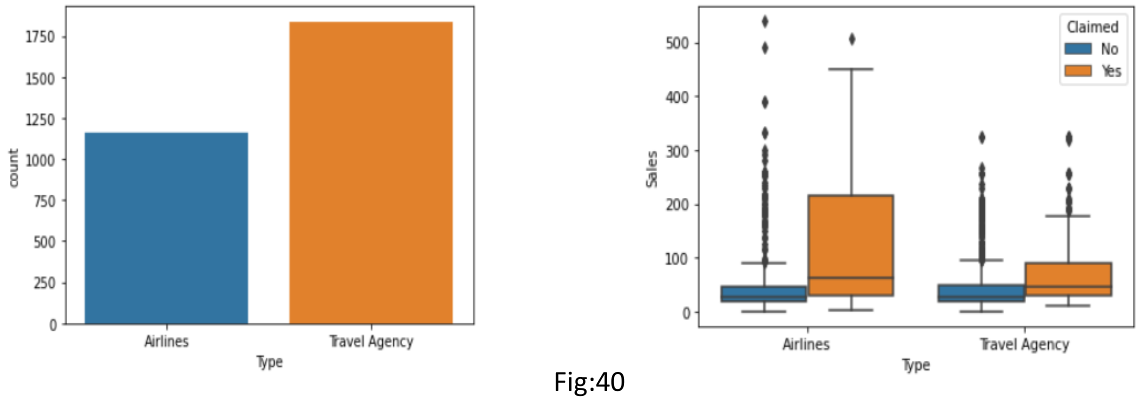


Fig:40

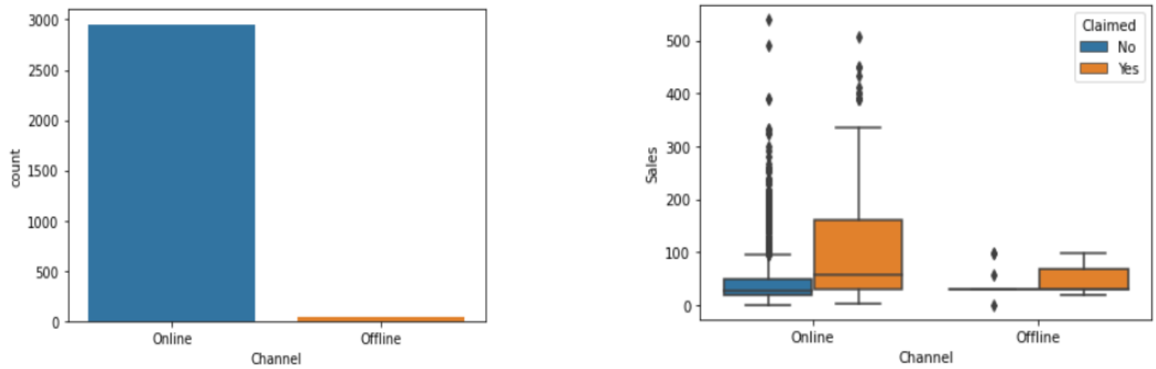


Fig:41

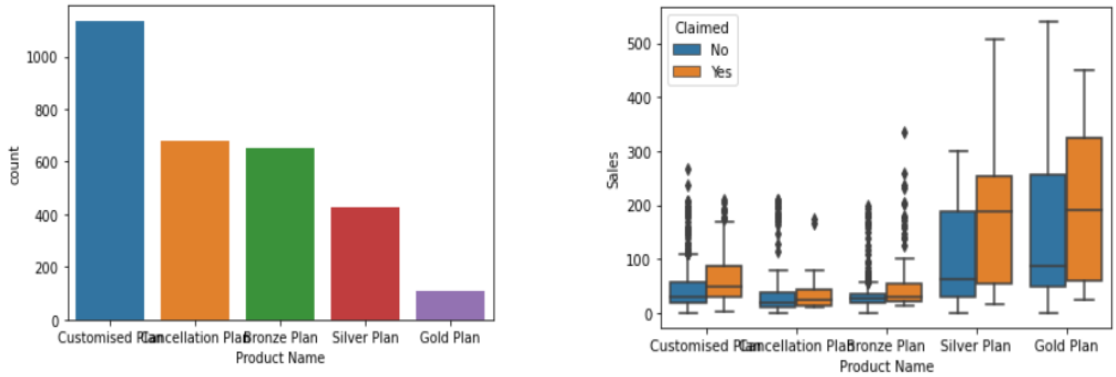


Fig:42

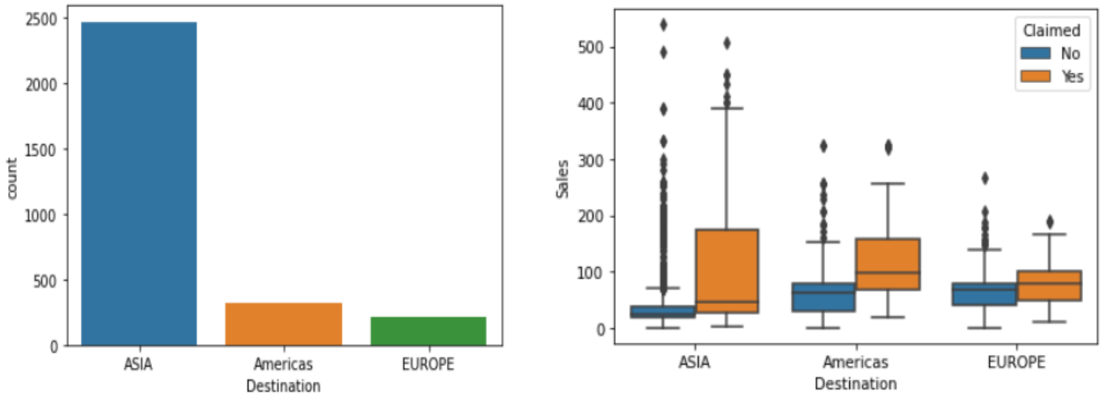


Fig:43

- From the Agency_Code : the following are the categories from descending order EPX 1365, CWT 472, C2B 924, JZI 239
- Type: travel agency 1837(claimed and unclaimed ratio is more or less equal), Airlines 1163 (claimed percent is high)
- Claimed : Yes 924, No 2076
- Channel : Online 2954, Offline 46
- Product Name : Customised Plan 1136, Cancellation Plan 678, Bronze Plan 650, Silver Plan 427 , Gold Plan 109
- Destination : ASIA 2465, Americas 320, EUROPE 215

Multivariate Analysis :

Pair plot :

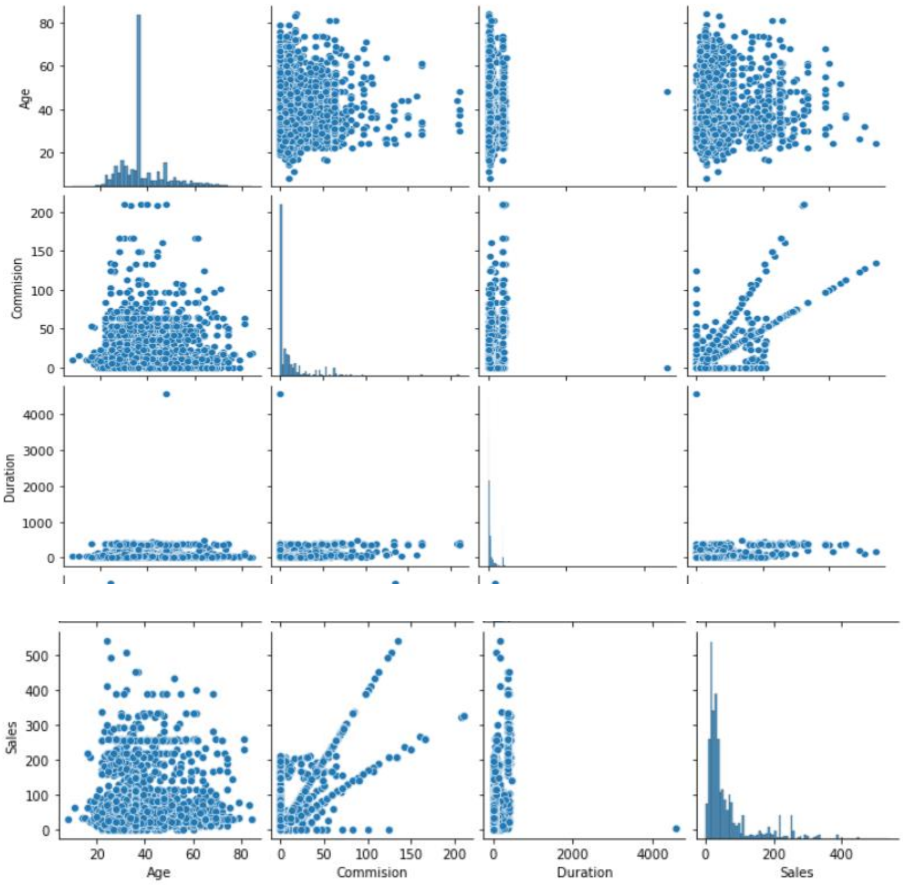


Fig.44

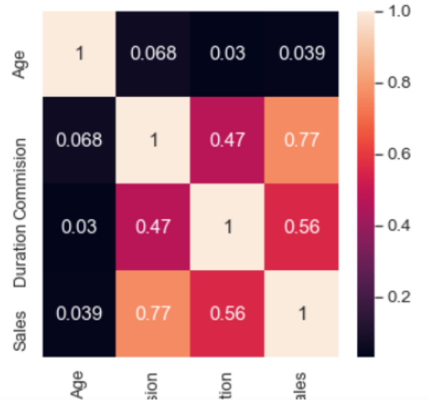


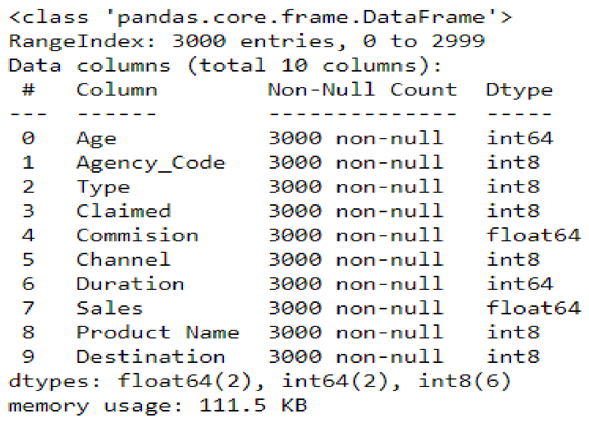
Fig.45

- There is no extreme low correlation between the variables. Sales and commission are highly correlated (0.77)

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

- The train-test split is a technique for evaluating the performance of a machine learning algorithm.
- It can be used for classification or regression problems and can be used for any supervised learning algorithm.
- The procedure involves taking a dataset and dividing it into two subsets.
- The first subset is used to fit the model and is referred to as the training dataset.
- The second subset is not used to train the model; instead, the input element of the dataset is provided to the model, then predictions are made and compared to the expected values.
- This second dataset is referred to as the test dataset.
- Train Dataset: Used to fit the machine learning model.
- Test Dataset: Used to evaluate the fit machine learning model.
- Hence, estimated the performance of the machine learning model on new data.
- The train-test procedure is appropriate when there is a sufficiently large dataset available.
- Splitting the data into test and train In the ratio of 30 and 70 for test and training respectively.
- We are using "gini" as the criterion.
- The GridSearchCV function in python helps us to determine optimum number of nodes,minimum sample leaf size, minimum sample split size.
- Converted categorical variables to numerical datatypes

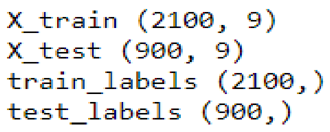
Converted datatypes:



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Age           3000 non-null   int64  
 1   Agency_Code   3000 non-null   int8   
 2   Type          3000 non-null   int8   
 3   Claimed       3000 non-null   int8   
 4   Commision     3000 non-null   float64
 5   Channel       3000 non-null   int8   
 6   Duration      3000 non-null   int64  
 7   Sales         3000 non-null   float64
 8   Product Name  3000 non-null   int8   
 9   Destination   3000 non-null   int8   
dtypes: float64(2), int64(2), int8(6)
memory usage: 111.5 KB
```

Fig.46

Dimensions :



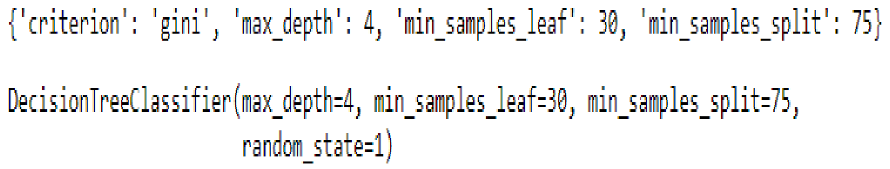
```
X_train (2100, 9)
X_test (900, 9)
train_labels (2100,)
test_labels (900,)
```

Fig.47

DT Classifier:

- Decision Trees are a non-parametric supervised learning method used for both classification and regression tasks.
- Tree models where the target variable can take a discrete set of values are called classification trees.
- Decision trees use multiple algorithms to decide to split a node into two or more sub-nodes.
- The creation of sub-nodes increases the homogeneity of resultant sub-nodes.
- The decision tree splits the nodes on all available variables and then selects the split which results in most homogeneous sub-nodes.

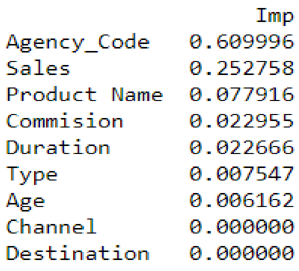
○ Taken max depth 4,5,6; min samples leaf 30,35,40; min samples split 75,90,105:



```
{'criterion': 'gini', 'max_depth': 4, 'min_samples_leaf': 30, 'min_samples_split': 75}

DecisionTreeClassifier(max_depth=4, min_samples_leaf=30, min_samples_split=75,
                       random_state=1)
```

Fig.48

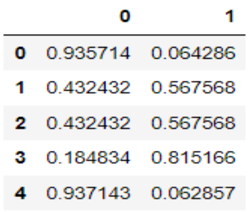
variable importance values or the feature importance to build the tree :



```
                   Imp
Agency_Code   0.609996
Sales         0.252758
Product Name  0.077916
Commision     0.022955
Duration      0.022666
Type          0.007547
Age           0.006162
Channel       0.000000
Destination   0.000000
```

Fig.49

Predicting on training and testing data and getting the predicted classes and probabilities :

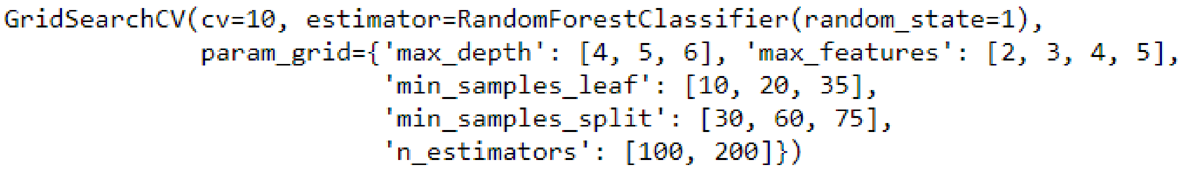


	0	1
0	0.935714	0.064286
1	0.432432	0.567568
2	0.432432	0.567568
3	0.184834	0.815166
4	0.937143	0.062857

Fig.50

Random Forest:

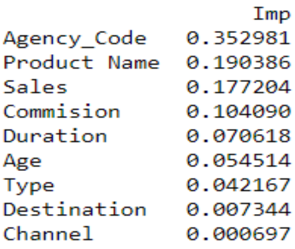
- A random forest is a machine learning technique that's used to solve regression and classification problems.
- It utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems.
- This algorithm consists of many decision trees
- Taken max depth 4,5,6; min samples leaf 10,20,35; min samples split 30,60,75:



```
GridSearchCV(cv=10, estimator=RandomForestClassifier(random_state=1),
             param_grid={'max_depth': [4, 5, 6], 'max_features': [2, 3, 4, 5],
                         'min_samples_leaf': [10, 20, 35],
                         'min_samples_split': [30, 60, 75],
                         'n_estimators': [100, 200]})
```

Fig.51

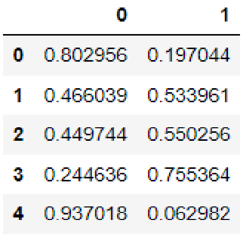
variable importance values or the feature importance to build the tree:



```
                   Imp
Agency_Code   0.352981
Product Name  0.190386
Sales         0.177204
Commision     0.104090
Duration      0.070618
Age           0.054514
Type          0.042167
Destination   0.007344
Channel       0.000697
```

Fig.52

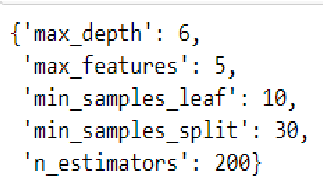
Predicting on training and testing data and getting the predicted classes and probabilities :



	0	1
0	0.802956	0.197044
1	0.466039	0.533961
2	0.449744	0.550256
3	0.244636	0.755364
4	0.937018	0.062982

Fig.53

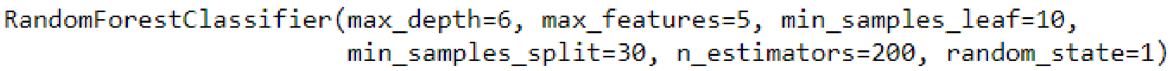
Grid search parameters :



```
{'max_depth': 6,
 'max_features': 5,
 'min_samples_leaf': 10,
 'min_samples_split': 30,
 'n_estimators': 200}
```

Fig.54

Best grid :

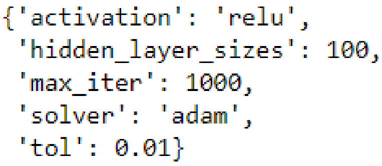


```
RandomForestClassifier(max_depth=6, max_features=5, min_samples_leaf=10,
                       min_samples_split=30, n_estimators=200, random_state=1)
```

Fig.55

Neural Network Classifier:

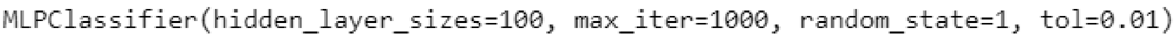
- An artificial neural network learning algorithm is a computational learning system that uses a network of functions to understand and translate a data input of one form into a desired output.
- Scaling the data
- Taken grid parameters as:



```
{'activation': 'relu',
 'hidden_layer_sizes': 100,
 'max_iter': 1000,
 'solver': 'adam',
 'tol': 0.01}
```

Fig.56

- Best grid estimator:



```
MLPClassifier(hidden_layer_sizes=100, max_iter=1000, random_state=1, tol=0.01)
```

Fig.57

criterion:

- The function to measure the quality of a split. Supported criteria are “gini” for the Gini impurity and “entropy” for the information gain.

splitter:

- The strategy used to choose the split at each node. Supported strategies are “best” to choose the best split and “random” to choose the best random split.

max_depth:

- The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.

min_samples_split:

- The minimum number of samples required to split an internal node.

min_samples_leaf:

- The minimum number of samples required to be at a leaf node.
- A split point at any depth will only be considered if it leaves at least min_samples_leaf training samples in each of the left and right branches.
- This may have the effect of smoothing the model, especially in regression.

random_state:

- Controls the randomness of the estimator.

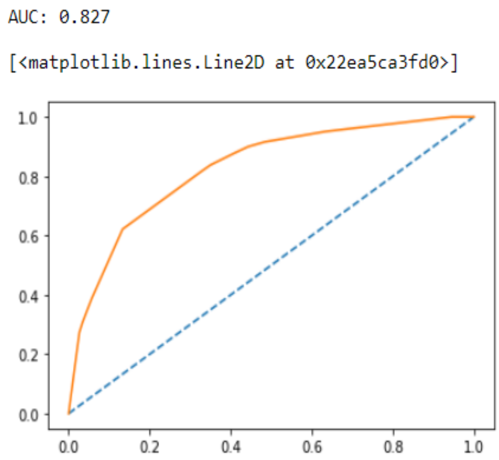
max_leaf_nodes:

- Best nodes are defined as relative reduction in impurity

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.

CART :

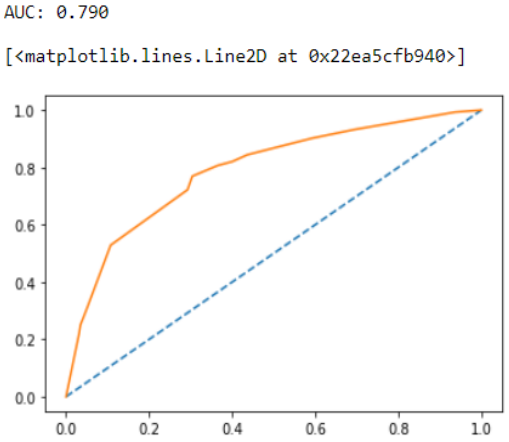
Area under the curve for training data :



```
AUC: 0.827

[<matplotlib.lines.Line2D at 0x22ea5ca3fd0>]
```

Fig.57

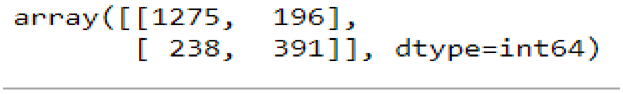
Area under the curve for testing data :



```
AUC: 0.790

[<matplotlib.lines.Line2D at 0x22ea5cfb940>]
```

Fig.58

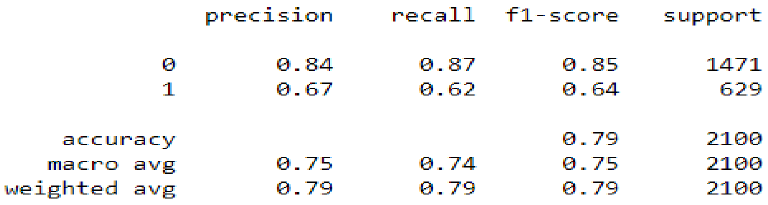
Confusion Matrix training data :



```
array([[1275,  196],
       [ 238,  391]], dtype=int64)
```

Train data accuracy check: 0.7933333333333333

Classification report for train :



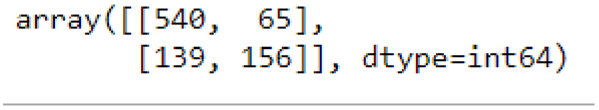
```
              precision    recall  f1-score   support

           0       0.84      0.87      0.85      1471
           1       0.67      0.62      0.64       629

    accuracy                           0.79      2100
   macro avg       0.75      0.74      0.75      2100
weighted avg       0.79      0.79      0.79      2100
```

Fig.59

CART metrics for train:

- cart_train_precision 0.67
- cart_train_recall 0.62
- cart_train_f1 0.64

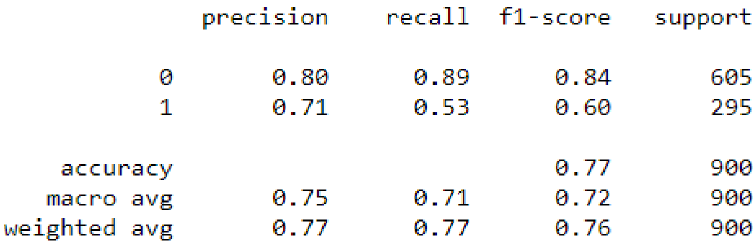
Confusion Matrix test data :



```
array([[540,  65],
       [139, 156]], dtype=int64)
```

test data accuracy check: 0.7733333333333333

Classification report for test :



```
              precision    recall  f1-score   support

           0       0.80      0.89      0.84       605
           1       0.71      0.53      0.60       295

    accuracy                           0.77       900
   macro avg       0.75      0.71      0.72       900
weighted avg       0.77      0.77      0.76       900
```

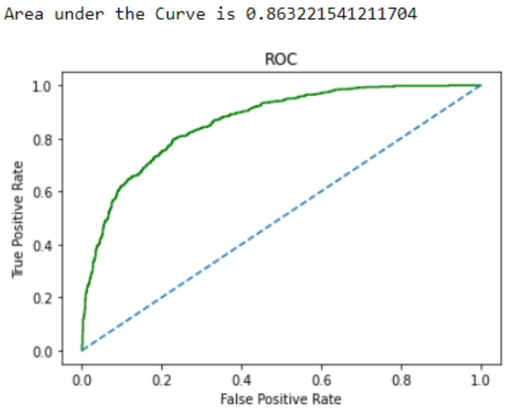
Fig.60

CART metrics for test:

- cart_test_precision 0.71
- cart_test_recall 0.53
- cart_test_f1 0.6

Random forest :

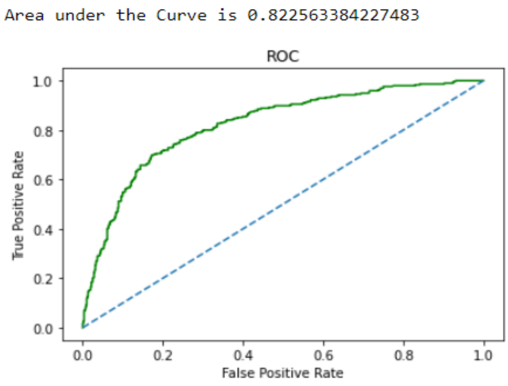
Area under the curve for training data :



```
Area under the Curve is 0.863221541211704
```

Fig.61

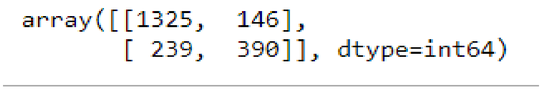
Area under the curve for testing data :



```
Area under the Curve is 0.822563384227483
```

Fig.62

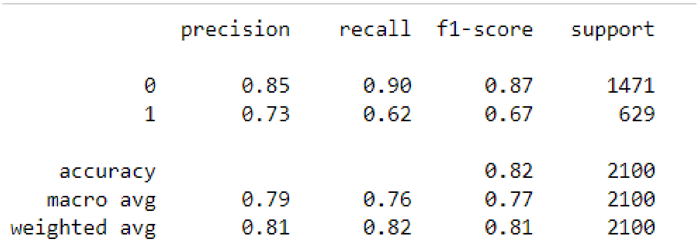
Confusion Matrix training data :



```
array([[1325,  146],
       [ 239,  390]], dtype=int64)
```

Train data accuracy check: 0.8166666666666667

Classification report for train :



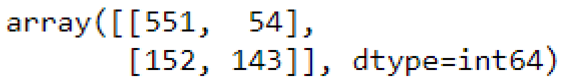
```
              precision    recall  f1-score   support

           0       0.85      0.90      0.87      1471
           1       0.73      0.62      0.67       629

    accuracy                           0.82      2100
   macro avg       0.79      0.76      0.77      2100
weighted avg       0.81      0.82      0.81      2100
```

Fig.63

Random forest metrics for train:

- rf_train_precision 0.73
- rf_train_recall 0.62
- rf_train_f1 0.67

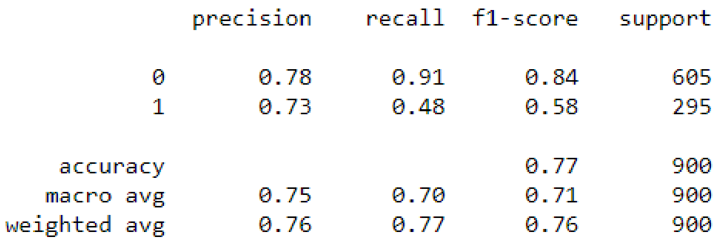
Confusion Matrix test data :



```
array([[551,  54],
       [152, 143]], dtype=int64)
```

test data accuracy check: 0.7711111111111111

Classification report for test :



```
              precision    recall  f1-score   support

           0       0.78      0.91      0.84       605
           1       0.73      0.48      0.58       295

    accuracy                           0.77       900
   macro avg       0.75      0.70      0.71       900
weighted avg       0.76      0.77      0.76       900
```

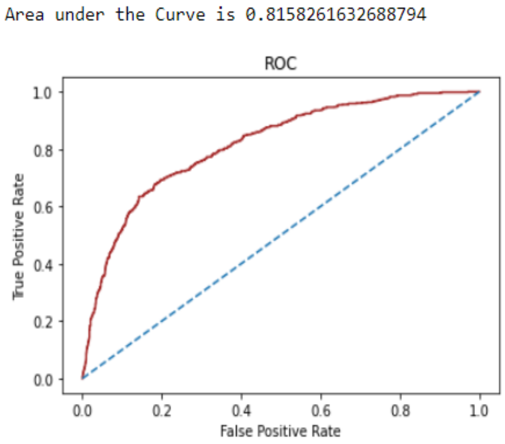
Fig.64

Random forest metrics for test:

- rf_test_precision 0.73
- rf_test_recall 0.48
- rf_test_f1 0.58

Neural Network:

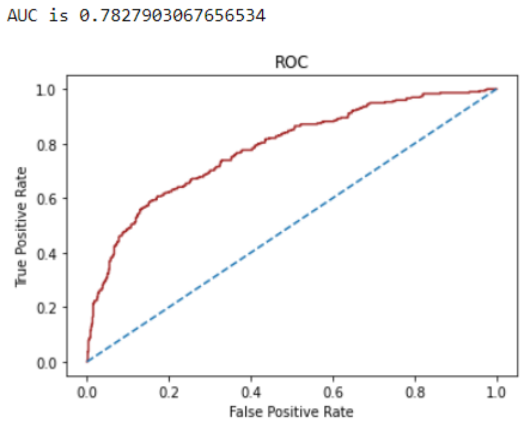
Area under the curve for training data :



```
Area under the Curve is 0.8158261632688794
```

Fig.65

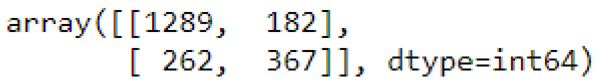
Area under the curve for testing data :



```
AUC is 0.7827903067656534
```

Fig.66

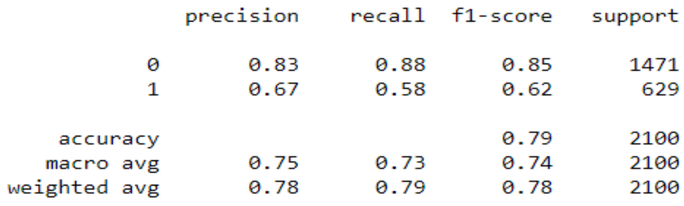
Confusion Matrix training data :



```
array([[1289,  182],
       [ 262,  367]], dtype=int64)
```

Train data accuracy check: 0.7885714285714286

Classification report for train :



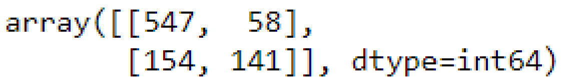
```
              precision    recall  f1-score   support

           0       0.83      0.88      0.85      1471
           1       0.67      0.58      0.62       629

    accuracy                           0.79      2100
   macro avg       0.75      0.73      0.74      2100
weighted avg       0.78      0.79      0.78      2100
```

Fig.67

Neural Network metrics for train:

- nn_train_precision 0.67
- nn_train_recall 0.58
- nn_train_f1 0.62

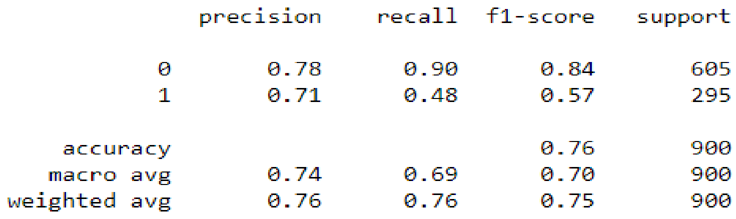
Confusion Matrix test data :



```
array([[547,  58],
       [154, 141]], dtype=int64)
```

test data accuracy check: 0.7644444444444445

Classification report for test :



```
              precision    recall  f1-score   support

           0       0.78      0.90      0.84       605
           1       0.71      0.48      0.57       295

    accuracy                           0.76       900
   macro avg       0.74      0.69      0.70       900
weighted avg       0.76      0.76      0.75       900
```

Fig.68

Neural Network metrics for test:

- nn_test_precision 0.71
- nn_test_recall 0.48
- nn_test_f1 0.57

Confusion Matrix:

- The confusion matrix provides more insight into not only the performance of a predictive model, but also which classes are being predicted correctly, which incorrectly, and what type of errors are being made.
- The simplest confusion matrix is for a two-class classification problem, with negative (class 0) and positive (class 1) classes.

ROC Curves and ROC AUC:

- An ROC curve (or receiver operating characteristic curve) is a plot that summarizes the performance of a binary classification model on the positive class.
- Ideally, we want the fraction of correct positive class predictions to be 1 (top of the plot) and the fraction of incorrect negative class predictions to be 0 (left of the plot). This highlights that the best possible classifier that achieves perfect skill is the top-left of the plot (coordinate 0,1)
- The curve provides a convenient diagnostic tool to investigate one classifier with different threshold values and the effect on the TruePositiveRate and FalsePositiveRate. One might choose a threshold in order to bias the predictive behavior of a classification model.

ROC Area Under Curve (AUC) Score:

- Instead, the area under the curve can be calculated to give a single score for a classifier model across all threshold values. This is called the ROC area under curve or ROC AUC or sometimes ROCAUC.
- The score is a value between 0.0 and 1.0 for a perfect classifier.

Precision-Recall Curves and AUC:

- Precision is a metric that quantifies the number of correct positive predictions made.

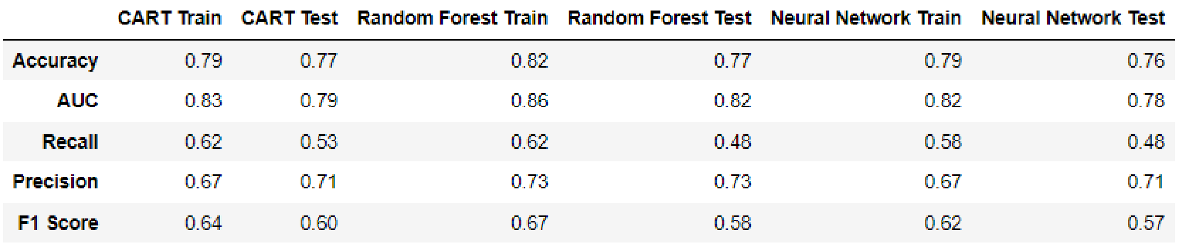
- It is calculated as the number of true positives divided by the total number of true positives and false positives.

Precision-Recall Area Under Curve (AUC) Score:

- The Precision-Recall AUC is just like the ROC AUC, in that it summarizes the curve with a range of threshold values as a single score.
- The score can then be used as a point of comparison between different models on a binary classification problem where a score of 1.0 represents a model with perfect skill.

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

Comparing all models on the basis of the performance metrics in a structured tabular manner:



	CART Train	CART Test	Random Forest Train	Random Forest Test	Neural Network Train	Neural Network Test
Accuracy	0.79	0.77	0.82	0.77	0.79	0.76
AUC	0.83	0.79	0.86	0.82	0.82	0.78
Recall	0.62	0.53	0.62	0.48	0.58	0.48
Precision	0.67	0.71	0.73	0.73	0.67	0.71
F1 Score	0.64	0.60	0.67	0.58	0.62	0.57

Fig.69

Accuracy:

- Accuracy can be defined as the percentage of correct predictions made by our classification model.
- The formula is:
- Accuracy = Number of Correct predictions/number of rows in data
- Which can also be written as:
- Accuracy = (TP+TN)/number of rows in data

Precision:

- Precision indicates out of all positive predictions, how many are actually positive. It is defined as a ratio of correct positive predictions to overall positive predictions.
- Precision = Predictions actually positive/Total predicted positive.
- Precision = TP/TP+FP

Recall:

- Recall indicates out of all actually positive values, how many are predicted positive. It is a ratio of correct positive predictions to the overall number of positive instances in the dataset.

- Recall = Predictions actually positive/Actual positive values in the dataset.
- Recall = TP/TP+FN

F1 score:

- An f1 score is defined as the harmonic mean of precision and recall.
- Used to predict if a particular employee has to be promoted or not and promotion is the positive outcome.
- In this case, promoting an incompetent employee(false positive) and not promoting a deserving candidate(false negative) can both be equally risky for the company.
- When avoiding both false positives and false negatives are equally important. we need a trade-off between precision and recall. This is when we use the f1 score as a metric.

Threshold:

- Any machine learning algorithm for classification gives output in the probability format,
- In order to assign a class to an instance for binary classification, we compare the probability value to the threshold, i.e if the value is greater than or less than the threshold.

AUC-ROC:

- We use the receiver operating curve to check model performance.
- Wikipedia defines ROC as: “A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied”.

- The Accuracy, AUC, Precision and F-1 scores are highest in Random forest train set.
- We can come to a conclusion on plotting the ROC curve for all Testing data and Training data.
- As we can see when we compare Accuracy from three models:
- CART : 0.79/0.77 - train/test respectively
- RF : 0.82/0.77 - train/test respectively
- NN : 0.79/0.76 - train/test respectively
- As we can see when we compare AUC from three models:
- CART : 0.83/0.79 - train/test respectively
- RF : 0.86/0.82 - train/test respectively
- NN : 0.82/0.78 - train/test respectively
- As we can see when we compare Precision from three models:
- CART : 0.67/0.71 - train/test respectively
- RF : 0.73/0.73 - train/test respectively
- NN : 0.67/0.71 - train/test respectively
- As we can see when we compare F1 score from three models:
- CART : 0.64/0.60 - train/test respectively

- RF : 0.67/0.58 - train/test respectively
- NN : 0.62/0.57 - train/test respectively

RF model, as it has better accuracy, precision, recall, f1 score better than other two CART & NN.

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

- The whole business insights and recommendations understood by looking at the insurance data by drawing relations between different variables such as day of the incident, time, age group, and associating it with other external information such as location, behavior patterns, weather information, airline/vehicle types, etc.
- The Accuracy, Precision, F1 Score are computed using Classification Report.
- The confusion matrix, AUC_ROC Scores and ROC plot are computed for each model separately and compared.
- All the three models have performed well but to increase our accuracy in determining the claims made by the customers we can choose the Random Forest Model.
- Instead of creating a single Decision Tree it can create a multiple decision trees and hence can provide the best claim status from the data.
- For all the models i.e.CART, Random Forest and ANN have performed exceptionally well.
- Hence, we can choose either of the models but choosing Random Forest Model is a great option as even though they exhibit the
- same accuracy but choosing Random Forest over Cart model is way better as they have much less variance than a single decision tree.
- By performing the 3 models, we can conclude that:
  - the data set is well balanced to conduct the modelling.
  - The data set is containing significant outliers
  - The Agency code has significant importance
- Claims are Higher for Online Distribution channel of tour insurance agencies.
- Claims are very low for Offline Distribution channel of tour insurance agencies.
- Reason might be, in recent time many people are preferring Online purchase which is very easier.
- So, Management can think of promoting offline Distribution channel of tour insurance agencies in order to reduce claims.
- Offline Purchase can be made more attractive by offering extra discounts or additional benefits.
- Higher Claims are observed for Agency Code C2B.

- So, Management need to check why claim state is high for this agency.
- Reason might be lack of knowledge to the insurance representative on insurance policies which leads loop holes.
- This might be leading to high claims.
- There is also a possibility that people might be purposely taking insurance from this agency for the reason of easy claims.
- This agency services can be compared with other agencies with are leading to fewer number of claims.
- Claims are higher for Airlines as Type of tour insurance firms.
- There are many factors involved in this like flight delay, baggage delay poor service recorded by the Airline Service or connection flight missing.
- These are usual; hence terms and conditions can be added or delay time frame can be increased.
- For baggage loss or any other delays, we can increase the premium Or claim value can be reduced by some percentage.
- Claims are higher for Silver plan which is one of the Name of the tour insurance products.
- We need to recheck for which reason are we getting claims from this Silver plan, or this plan can be stopped for sometime to confirm if this reduces the claims.
- Other plans can be promoted in order to reduce the claims.
- Claims are higher for Destination of the tour as ASIA. We need to dig out the reason why there are high claims for ASIA. Several reasons are present for it. People are negligent may be in ASIA.
- Terms and Conditions need to be increased for ASIA.
- There is also possibility that we can increase the premium for insurances related to ASIA in order to compensate for the claims.
- If customers are checking for ASIAN travels, we can give them more interesting options of other places to travel.
- The category for which company is getting more claims can be added as addons at extra cost.
- And the category for which company is getting less claims can be kept as basic facilities.
- And this can be varied based on location of travel.
- Management can also think of keeping the purchase procedure simple but we need to increase the complexity of the Claim Procedure.
- Claiming can be done only by offline mode, so as to verify the genuineness of the claim.
- Hence, we can say that RF model made easy to us for predicting the data by showing better accuracy, precision, recall, f1 score better than other two CART & NN.