Date: 18 November 2021

Great Learning

Authored by: ANIMESH HALDER

Great Learning
POWER AHEAD

Content

List of Tables:

List of Figures:

Problem 1: Clustering

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.

Data Information:

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

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

Libraries which are imported for the given Dataset analysis are as follws:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_samples, silhouette_score
import warnings
warnings.filterwarnings('ignore')
```

The Dataset is tabulated as shown in Table 1.1.

Table 1.1: Dataset used in the present analysis. The Dataset consists of seven columns.

	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

Insights:

1. The Dataset appears to be flawless.
2. The Dataset having 210 rows and 7 columns.
3. The ‘info’ of the Dataset indicates all the variables are float type.
4. There is no Null value in the Dataset.
5. No values are missing in any given columns ofo the Dataset.

The statistical summery of the Dataset is illustrated in Table 1.2

Table 1.2: Screenshot of the raw Dataset used in the present analysis. The Dataset consists of seven columns.

	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

Insights:

1. There are total 7 variables.
2. The mean and median values appear almost equal.
3. The standard deviation for the variable ‘spending’ is compreratively higher than other.
4. There is no duplicate entries in the Dataset.
5. Individual variables range are relatively different from each other, so scaling is required.

Exploratory Data Analysis (**Univariate / Bivariate**) helps to understand the distribution of data in the dataset. With univariate analysis one can find patterns and can summarize the data and acquire understanding about the data to solve the business problem.

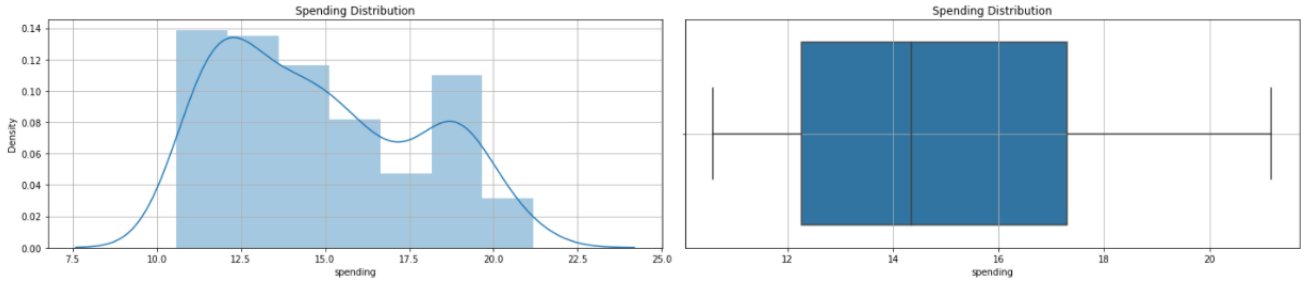


Figure 1.1: Screenshot of the distplot and boxplot distribution for the variable ‘spending’.

Insights:

1. The dist plot shows the distribution of data from 11 to 21.

2. The variable ‘spending’ is positively skewed.
3. The box plot of ‘spending’ shows no outliers.
4. There could be chance of multi modes in the dataset.

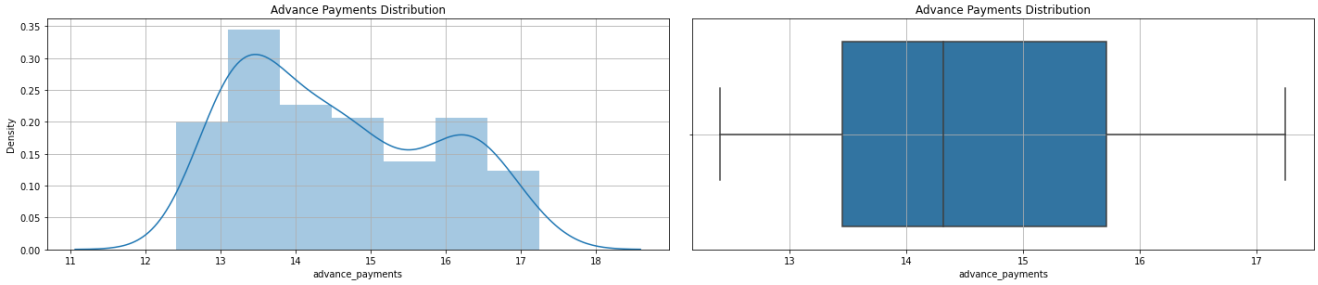


Figure 1.2: Screenshot of the distplot and boxplot distribution for the variable ‘advance_payment’.

Insights:

1. The dist plot shows the distribution of data from 12 to 17.
2. The variable ‘advance_payment’ is positively skewed.
3. The box plot of ‘advance_payment’ shows no outliers.
4. There could be chance of multi modes in the dataset.

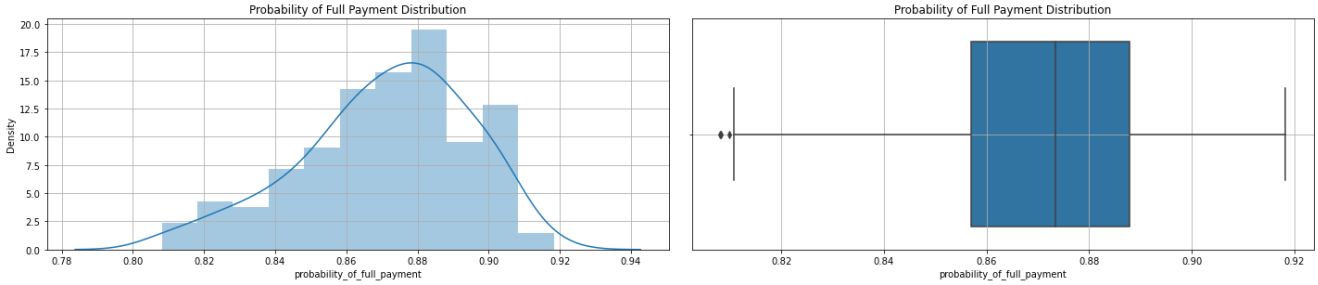


Figure 1.3: Screenshot of the distplot and boxplot distribution for the variable ‘probability_of_full_payment’.

Insights:

1. The dist plot shows the distribution of data from 0.80 to 0.92.
2. The variable is negatively skewed.
3. The box plot of ‘probability_of_full_payment’ shows few outliers.
4. Probability values is good above 80%.

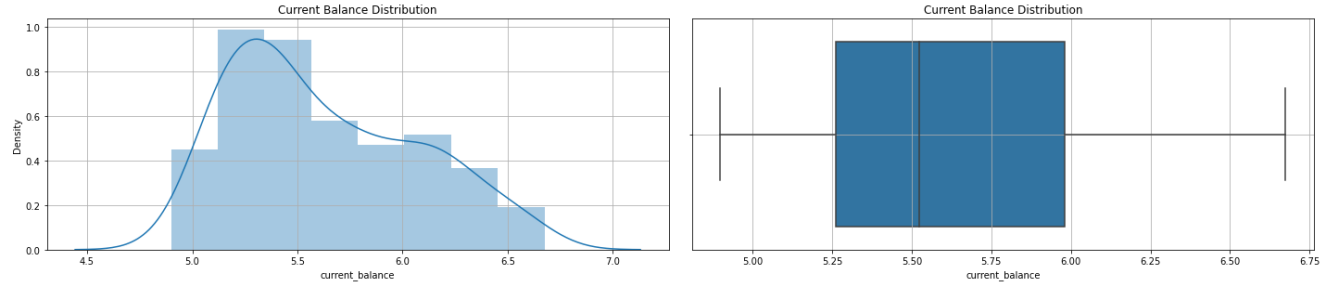


Figure 1.4: Screenshot of the distplot and boxplot distribution for the variable ‘current_balance’.

Insights:

1. The dist plot shows the distribution of data from 5.0 to 6.5.
2. The variable is positively skewed.
3. The box plot of ‘current_balance’ shows no outliers.

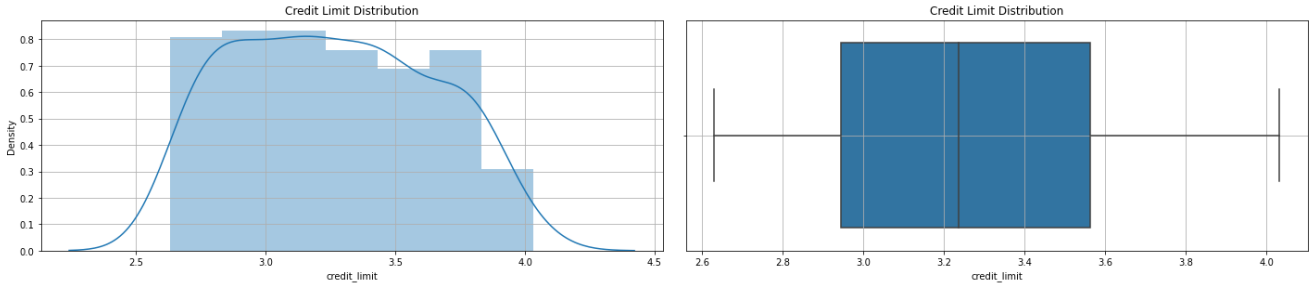


Figure 1.5: Screenshot of the distplot and boxplot distribution for the variable ‘credit_limit’.

Insights:

1. The dist plot shows the distribution of data from 2.5 to 4.0.
2. The variable is positively skewed.
3. The box plot of ‘credit_limit’ shows no outliers.

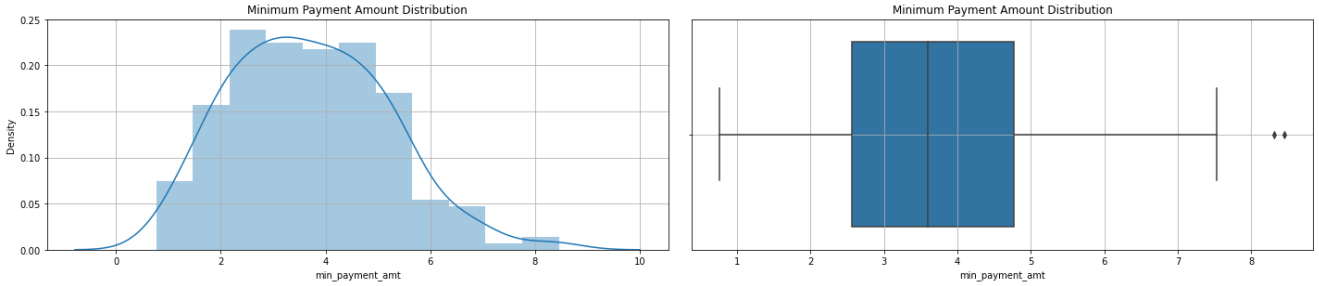


Figure 1.6: Screenshot of the distplot and boxplot distribution for the variable ‘min_payment_amt’.

Insights:

1. The dist plot shows the distribution of data from 2.0 to 8.0.
2. The variable is positively skewed.
3. The box plot of ‘min_payment_amt’ shows few outliers.

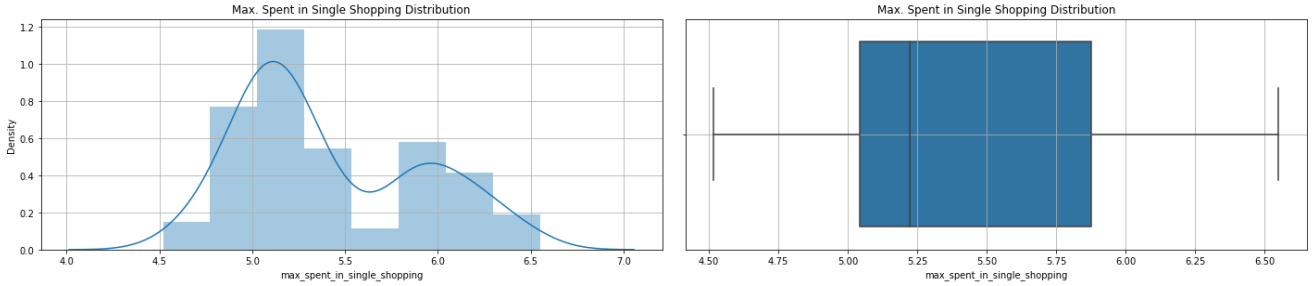


Figure 1.7: Screenshot of the distplot and boxplot distribution for the variable ‘max_spent_in_single_shopping’.

Insights:

1. The dist plot shows the distribution of data from 4.5 to 6.5.

2. The variable is positively skewed.
3. The box plot of ‘max_spent_in_single_shopping’ shows no outliers.

Multivariate analysis is a Statistical procedure for analysis of data involving more than one type of measurement or observation. It may also mean solving problems where more than one dependent variable is analyzed simultaneously with other variables. For multivariate analysis pairplot and heatmap are two useful methods.

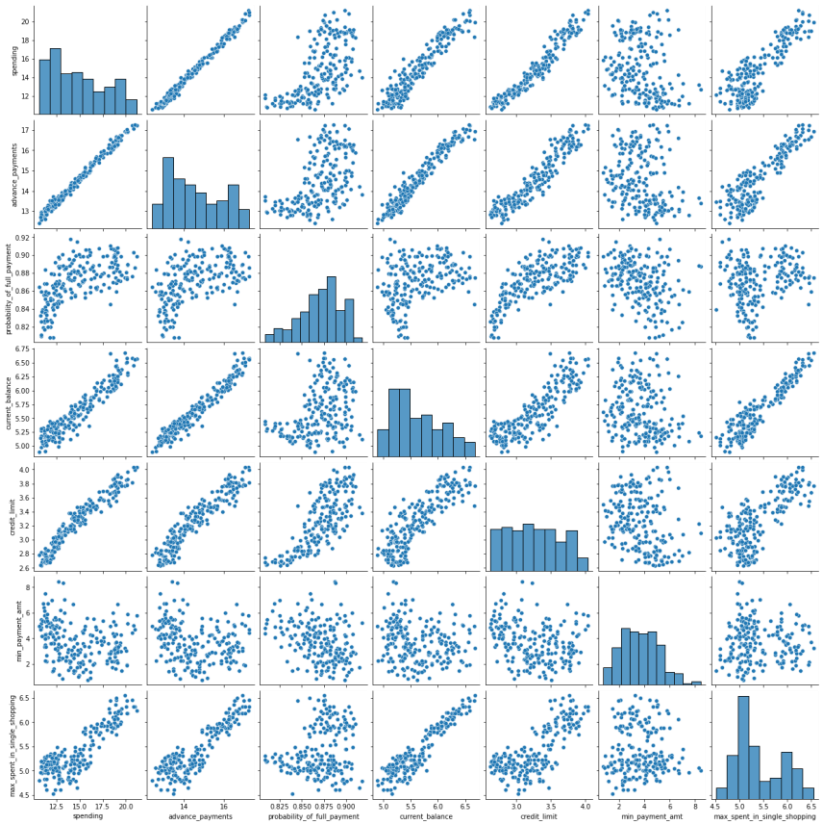


Figure 1.8: Screenshot of the pairplot for the given Dataset. It shows the interrelationship of all the variables available.

Figure 1.9: Screenshot of the heatmap for the given Dataset. It shows the multicollinearity of all the variables.

Insights:

1. Strong positive correlation between the variables:
   a. spending & advance_payments
   b. advance_payments & current_balance
   c. credit_limit & spending
   d. spending & current_balance
   e. credit_limit & advance_payments
   f. max_spent_in_single_shopping & current_balance

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

Yes, scaling is very important as the model works based on the distance based computations scaling is necessary for unscaled data.
Scaling needs to be done as the values of the variables are in different scales. In the Dataset, variables like spending, advance payments are in different values and this may get more weightage, while probability of full payment having least weightage. Scaling will have all the values in the relative same range. The function from 'sklearn' called 'StandardScalar' is used here for scaling and the screenshot of scaled data is shown in Table 1.3.

Table 1.3: Screenshot of the scaled Dataset to get relatively same range for all the variables.

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

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

Ward's method is applied to scaled Dataset for hierarchical clustering. Dendrogram shown in Figure 1.10. indicates all the data points that have clustered to different clusters by wards method.

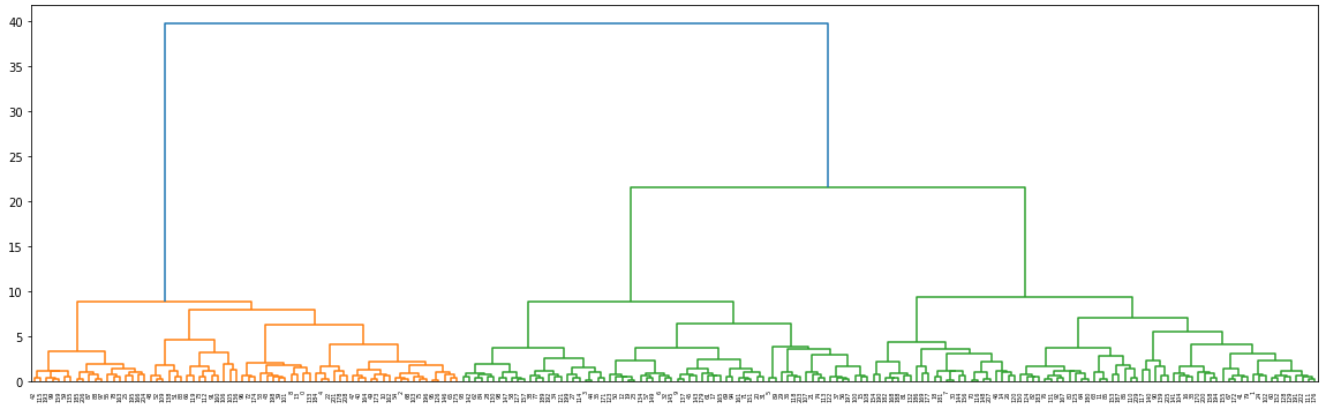


Figure 1.10: Dendrogram indicates all the data points that have clustered to different clusters by wards method.

The function used in the analysis is, ‘from scipy.cluster.hierarchy import dendrogram, linkage’ followed by the commands ‘Hclust = linkage(scaled_DF, method = 'ward')’ and ‘dend = dendrogram(Hclust)’.

To get the optimal number cluster through which for solving business objective ‘truncate_mode = lastp’ is used, with p = 10 according to industry set base value as shown in Figure 1.11. The commands used are as follows:

dend = dendrogram(Hclust, truncate_mode = 'lastp', p = 10)

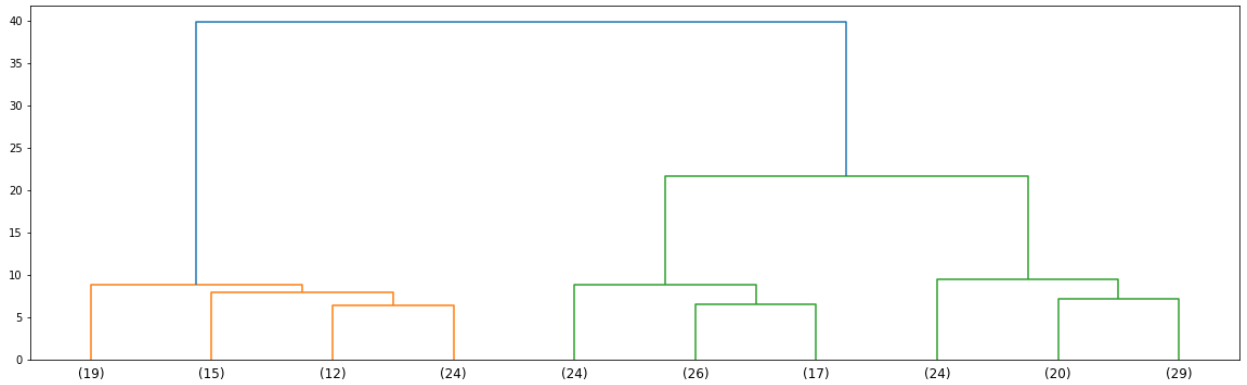


Figure 1.11: Dendrogram with trauncated mode to get optimal number of clusters.

According to Figure 1.11, all the data points have clustered into 3 clusters. Next to map these clusters to given Dataset, ‘fclusters’ function is used with criterion ‘maxclust’. The necessary command lines used are as follows:

‘from scipy.cluster.hierarchy import fcluster’

‘cluster = fcluster(Hclust, 3, criterion = 'maxclust')’

‘DF['H_Cluster'] = cluster’

Table 1.4: Dataset with hierarchical clustering.

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	H_Cluster
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

The cluster profiling to understand the business problem is depicted in Table 1.5.

Table 1.5: Cluster profiling showing the cluster frequency.

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Freq
H_Cluster								
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

Apart from ward’s method, average method is also used to scaled Dataset for hierarchical clustering. The cluster profiling obtained following the same procedure is shown in Table 1.6.

Table 1.6: Cluster profiling showing the cluster frequency.

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	H_Cluster	Freq1
avg									
1	18.129200	16.058000	0.881595	6.135747	3.648120	3.650200	5.987040	1.213333	75
2	11.916857	13.291000	0.846766	5.258300	2.846000	4.619000	5.115071	2.114286	70
3	14.217077	14.195846	0.884869	5.442000	3.253508	2.768418	5.055569	2.830769	65

Insights:

1. Both the method are almost similar means, minor variation.
2. There was not too much variations from both methods Cluster grouping based on the dendrogram, 3 or 4 looks good.
3. Further analysis performed based on the Dataset had gone for 3 group cluster.
4. Three group cluster solution gives a pattern based on high/medium/low spending with max_spent_in_single_shopping (high value item) and probability_of_full_payment (payment made).

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.

Initial 'n_clusters' considered as 2 and then 3 and 4 for Creating Clusters using KMeans and the distribution of clusters according to the n_clusters are observed. Finally the 'silhouette_score' is evaluated (Table 1.7) for 'n_clusters = k', where k = 2,3, and 4 using the following commands.

```
k_means = KMeans(n_clusters = 2,random_state=1)
k_means.fit(scaled_DF)
labels = k_means.labels_
silhouette_score(scaled_DF,labels,random_state=1)
```

Table 1.7: Silhouette score for n_clusters = k, where k =2,3,4

k	Silhouette score
2	0.46577247686580914
3	0.40072705527512986
4	0.32757426605518075

To find the optimal number of clusters, k-elbow method can be used.

The command lines for performing such method are like:

```
wss =[]
for i in range(1,11):
    KM = KMeans(n_clusters=i,random_state=1)
    KM.fit(scaled_DF)
```

wss.append(KM.inertia_)

The Within Sum of Squares (WSS) plot using the values of computed 'inertia' , as shown in Figure 1.12. The elbow curve seen here also shows us after 3 clusters there is no huge drop in the values, so 3 is the final cluster number.

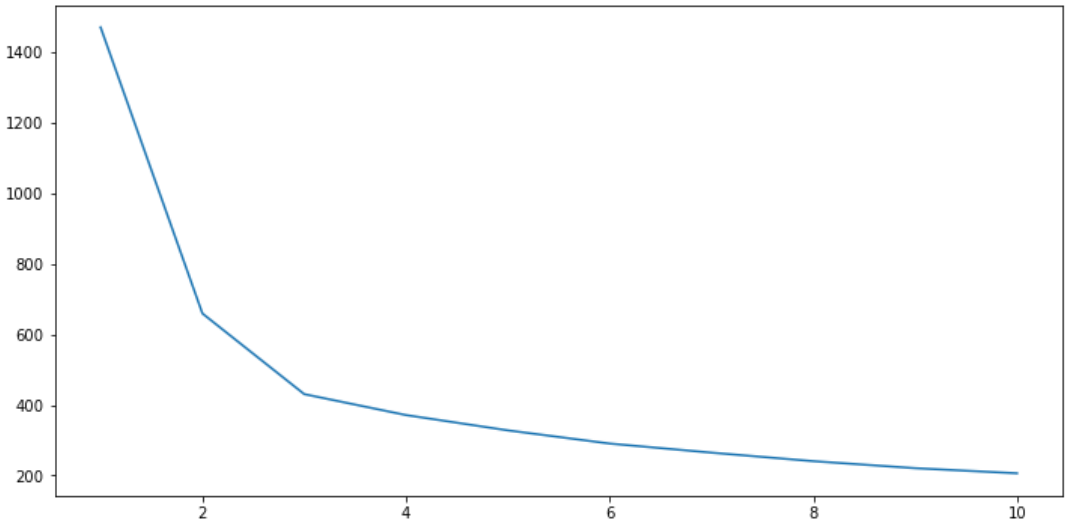


Figure 1.12: WSS plot to find the optimal cluster number.

The cluster results added to Dataset to solve given business objective as shown in Table 1.8.

Table 1.8: Dataset with Kmeans cluster values.

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	Clus_kmeans
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	2
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	0
4	17.99	15.86	0.8992	5.890	3.694	2.068	5.837	1

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

The cluster frequencies are calculated and added to the Dataset as tabulated in Table 1.9

Table 1.9: Cluster profiling showing the cluster frequency.

Clus_kmeans	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	K_Freq
0	11.856944	13.247778	0.848253	5.231750	2.849542	4.742389	5.101722	72
1	18.495373	16.203433	0.884210	6.175687	3.697537	3.632373	6.041701	67
2	14.437887	14.337746	0.881597	5.514577	3.259225	2.707341	5.120803	71

Insights:

1. Cluster 0: Low in spending but high in minimum payment amount.
2. Cluster 1: High in spending, moderate in minimum payment amount.
3. Cluster 2: Moderate in spending, but less in minimum payment amount.

Thus the entire customer community can be segmented into three groups based on their credit card usage during the past few months.

Group 1: High Spending Group:

Giving any reward points might increase their purchases.

Maximum max_spent_in_single_shopping is high for this group, so can be offered discount/offer on next transactions upon full payment

Increase their credit limit and Increase spending habits

Give loan against the credit card, as they are customers with good repayment record.

Tie up with luxury brands, which will drive more one_time_maximun spending

Group 2: Low Spending Group:

Customers should be given remainders for payments.

Offers can be provided on early payments to improve their payment rate.

Increase their spending habits by tying up with grocery stores, utilities (electricity, phone, gas, others)

Group 3: Medium Spending Group:

They are potential target customers who are paying bills and doing purchases and maintaining comparatively good credit score. So we can increase credit limit or can lower down interest rate.

Promote premium cards/loyalty cars to increase transactions.

Increase spending habits by trying with premium ecommerce sites, travel portal, travel airlines/hotel, as this will encourage them to spend more

Problem 2: CART-RF-ANN

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.

Data Information:

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

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

Libraries which are imported for the given Dataset analysis are as follws:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neural_network import MLPClassifier
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, roc_auc_score
from sklearn.preprocessing import StandardScaler
```

```
import warnings
warnings.filterwarnings('ignore')
```

The Dataset is tabulated as shown in Table 2.1.

Table 2.1: Dataset used in the present analysis. The Dataset consists of ten columns.

	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

Insights:

1. The Dataset having 3000 rows and 10 columns.
2. The ‘info’ of the Dataset indicates dataset has object, integer and float. The object data type has to be converted into numeric value.
3. There is no Null value in the Dataset.
4. No values are missing in any given columns ofo the Dataset.

The statistical summery of the Dataset is illustrated in Table 2.2

Table 2.2: Screenshot of the raw Dataset used in the present analysis.

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

Insights:

1. There are total 10 variables.
2. 4 numeric values and 6 categorical values.

3. The standard deviation for the variable ‘Duration’ is compreratively higher than other.
4. There are 139 duplicate entries in the Dataset. Since there is no unique identifier, thus no duplicate removal is performed.
5. Individual variables range are relatively different from each other.
6. Agency code EPX has a frequency of 1365.
7. The most preferred type seems to be travel agency Channel is online
8. Destination ASIA seems to be most preferred destination place by customers.

Checking of outliers for the continuous variables will help to treat the Dataset. According to Figure 2.1, outliers exist in almost all the numeric values. The outliers can be treated using random forest classification.

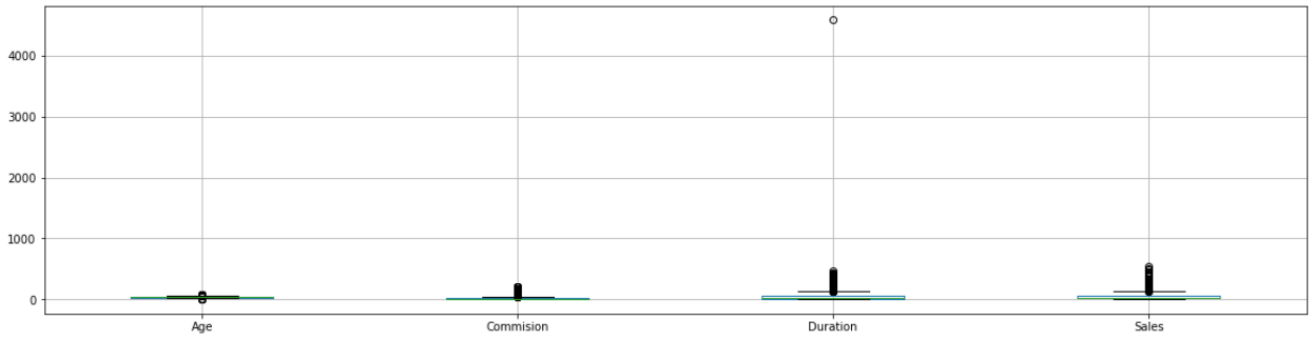


Figure 2.1: Boxplot shows the presence of outliers in four continuous variables.

In this respect the number of numeric counts for all the nominal variables (Table 2.3) will help to understand the Dataset.

Table 2.3: Description of Categorical values present in the raw Dataset used for the analysis.

AGENCY_CODE : 4 C2B 924 CWT 472 EPX 1365 JZI 239	TYPE : 2 Airlines 1163 Travel Agency 1837	CLAIMED : 2 No 2076 Yes 924
CHANNEL : 2 Offline 46 Online 2954	PRODUCT NAME : 5 Bronze Plan 650 Cancellation Plan 678 Customised Plan 1136 Gold Plan 109 Silver Plan 427	DESTINATION : 3 ASIA 2465 Americas 320 EUROPE 215

Univariate/Bivariate Analysis helps to understand the distribution of data in the dataset. With univariate analysis one can find patterns and can summarize the data and acquire understanding about the data to solve the business problem.

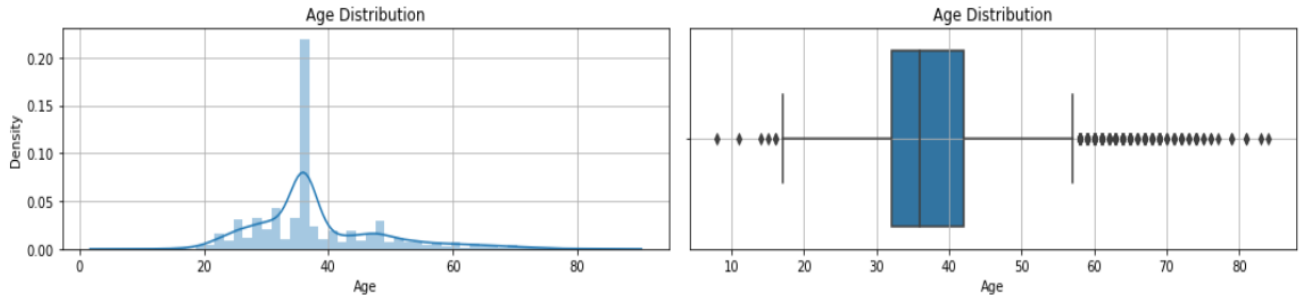


Figure 2.2: Screenshot of the distplot and boxplot distribution for the variable 'Age'.

Insights:

1. The dist plot shows the distribution of data from 20 to 80.
2. In the range of 30 to 40 is where the majority of the distribution lies.
3. The variable is positively skewed.
4. The box plot of 'Age' shows outliers.

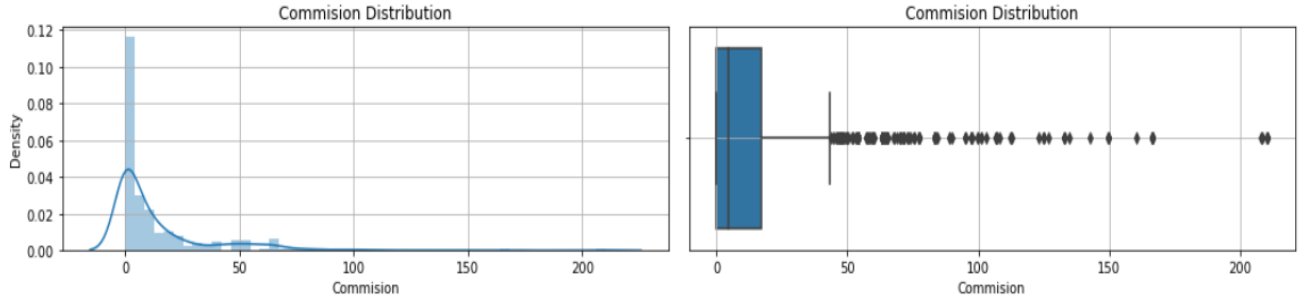


Figure 2.3: Screenshot of the distplot and boxplot distribution for the variable 'Commision'.

Insights:

1. The dist plot shows the distribution of data from 0 to 30.
2. The variable is positively skewed.
3. The box plot of 'Commision' shows outliers.

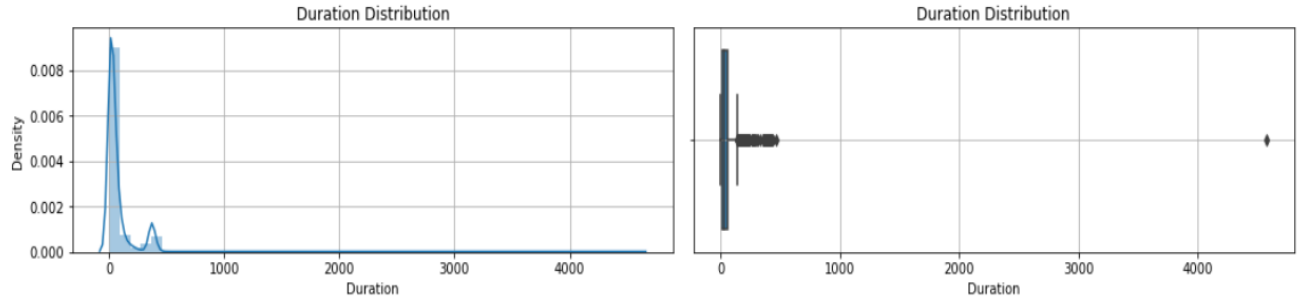


Figure 2.4: Screenshot of the distplot and boxplot distribution for the variable 'Duration'.

Insights:

1. The dist plot shows the distribution of data from 0 to 100.
2. The variable is positively skewed.
3. The box plot of 'Duration' shows few outliers.

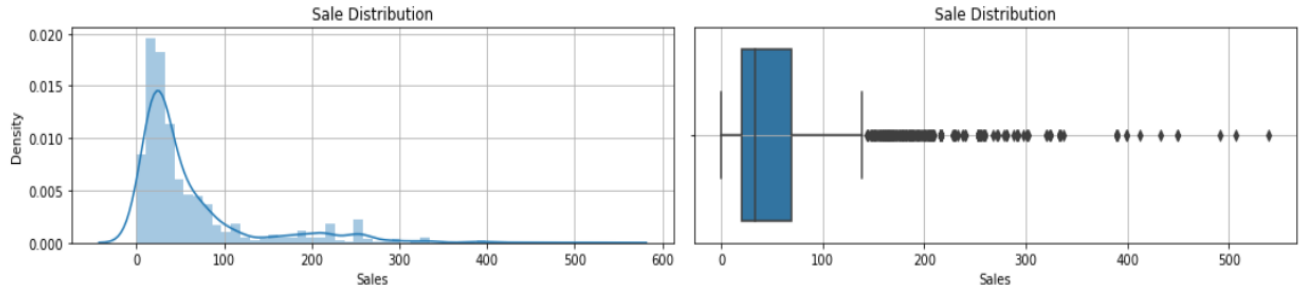


Figure 2.5: Screenshot of the distplot and boxplot distribution for the variable 'Sale'.

Insights:

1. The dist plot shows the distribution of data from 0 to 300.
2. The variable is positively skewed.
3. The box plot of 'Sales' shows outliers.

The distribution of the categorical variables are shown below:

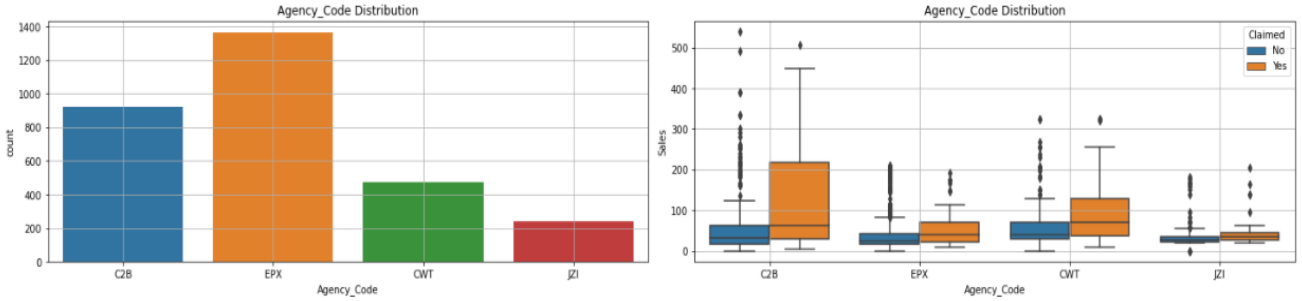


Figure 2.6: Screenshot of the countplot and boxplot distribution for the variable 'Agency_Code'.

Insights:

1. The distribution of the agency code, shows us EPX with maximum frequency.
2. The box plot shows the split of sales with different agency code and also hue having claimed
   column.
3. It seems that C2B have claimed more than other agency.
4. The box plot of 'Agency_Code' shows outliers.

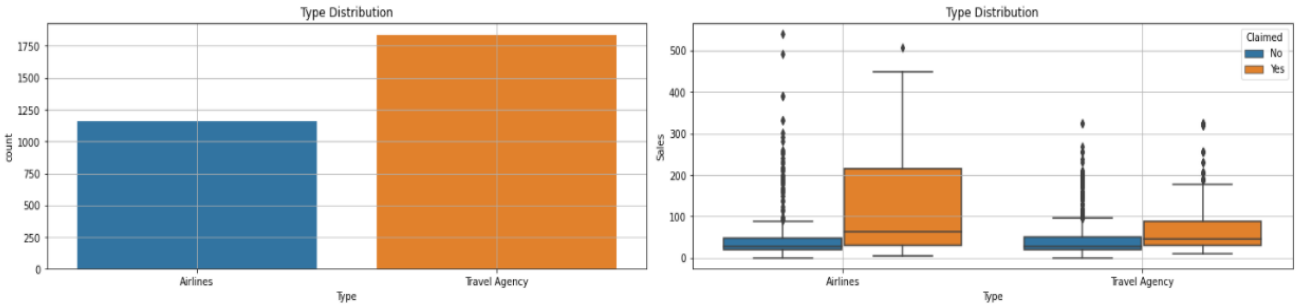


Figure 2.7: Screenshot of the countplot and boxplot distribution for the variable 'Type'.

Insights:

1. The distribution of the ‘Type’, shows travel agency with maximum frequency.
2. The box plot shows the split of sales with different types and also hue having claimed column.
3. It seems that airlines type has more claims.
4. The box plot of ‘Type’ shows outliers.

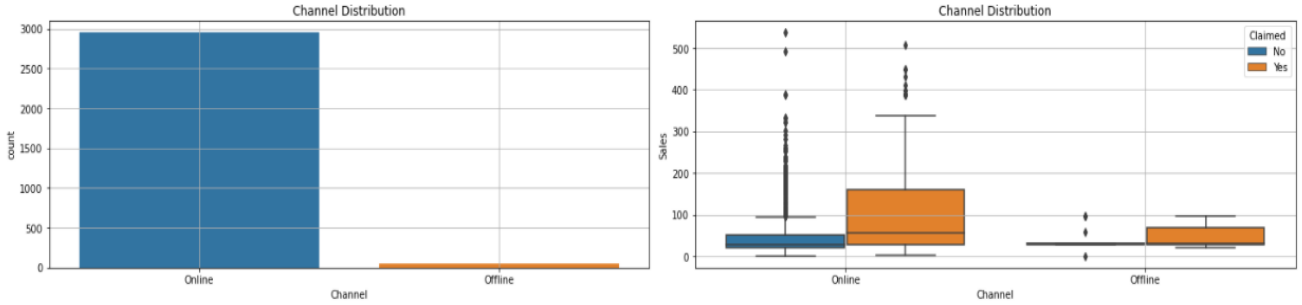


Figure 2.8: Screenshot of the countplot and boxplot distribution for the variable ‘Channel’.

Insights:

1. The majority of customers have used online medium, very less with offline medium.
2. The box plot shows the split of sales with different channel and also hue having claimed column.
3. It seems that online have claimed more than other channel.
4. The box plot of ‘Channel’ shows outliers.

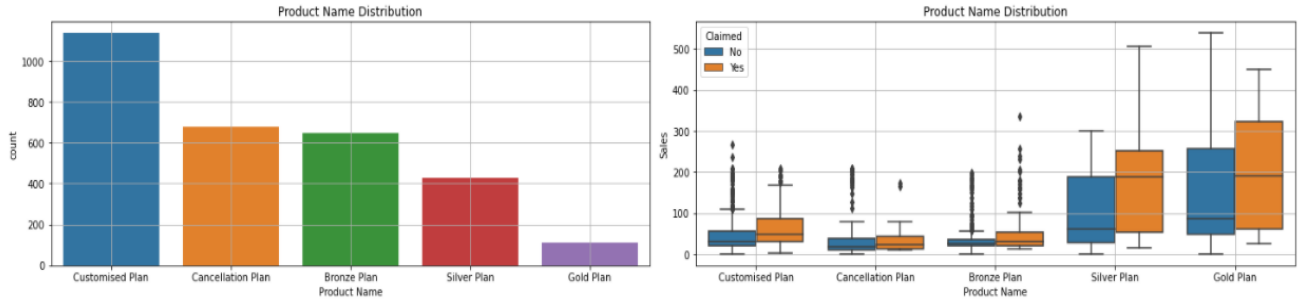


Figure 2.9: Screenshot of the countplot and boxplot distribution for the variable ‘Product Name.

Insights:

1. Customized plan seems to be most liked plan by customers when compared to all other plans.
2. The box plot shows the split of sales with different product name and also hue having claimed
   column.
3. It seems that gold plan have claimed more than other product name.
4. The box plot of ‘Product Name’ shows outliers.

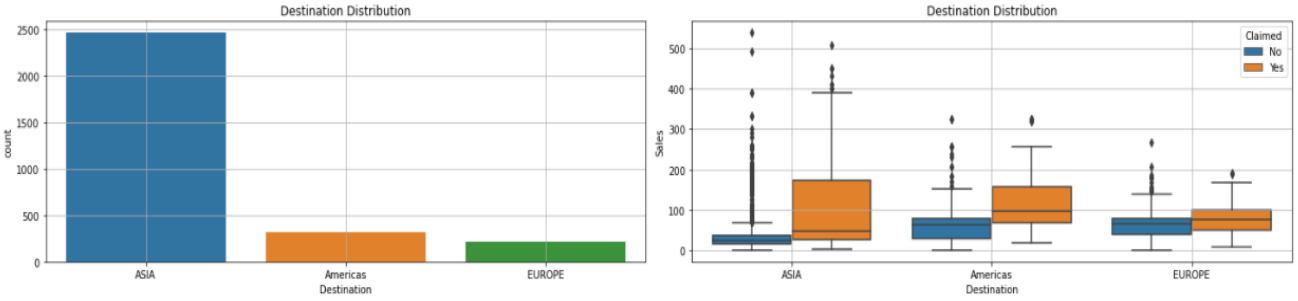


Figure 2.10: Screenshot of the countplot and boxplot distribution for the variable 'Destination'.

Insights:

1. Asia is where customers choose when compared with other destination places.
2. The box plot shows the split of sales with different destination and also hue having claimed column.
3. It seems that ASIA have claimed more than other destination.
4. The box plot of 'Destination' shows outliers.

Multivariate analysis is a Statistical procedure for analysis of data involving more than one type of measurement or observation. It may also mean solving problems where more than one dependent variable is analyzed simultaneously with other variables. For multivariate analysis pairplot and heatmap are two useful methods.

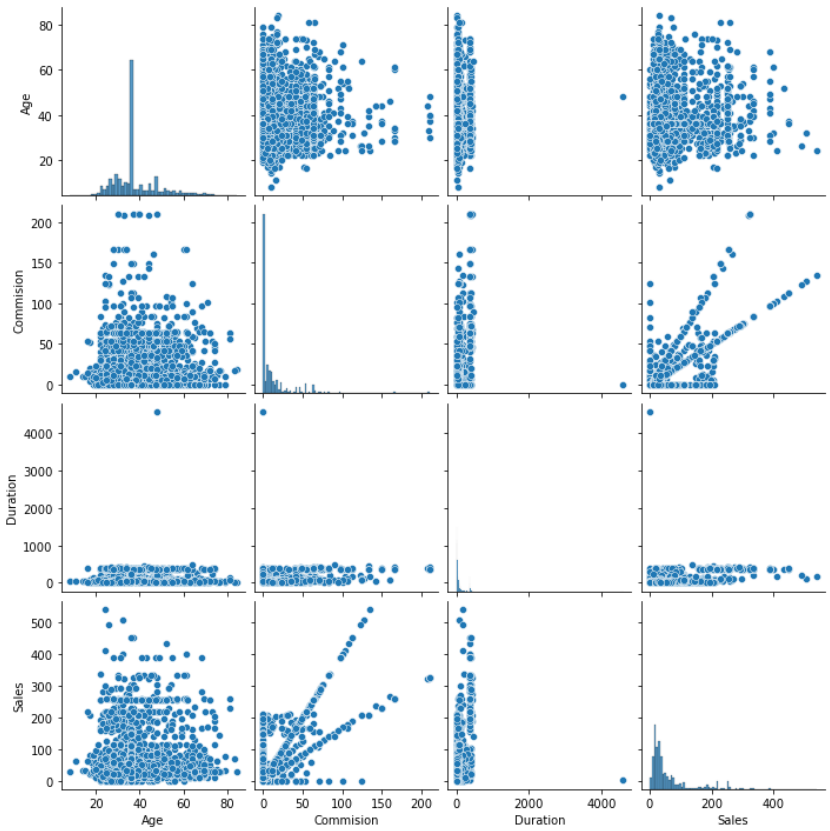


Figure 2.11: Screenshot of the pairplot for the given Dataset. It shows the interrelationship of all the variables available.

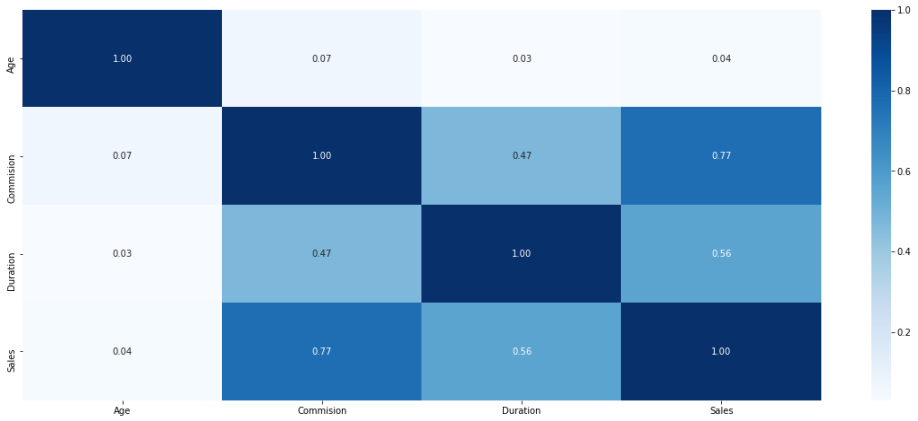


Figure 2.12: Screenshot of the heatmap for the given Dataset. Not much of multicollinearity is observed.

Insights:

1. Not much of multi collinearity observed
2. No negative correlation
3. Only positive correlation

In order to build the models, the object data type must be changed to numeric values as shown in Table 2.4 and the description of the categorical values are tabulated in Table 2.5.

Table 2.4: Screenshot of the modified Dataset. All the categorical values updated to numerical.

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

Table 2.5: Description of Categorical values present in the raw Dataset and their conversion to numeric.

Feature: Agency_Code ['C2B', 'EPX', 'CWT', 'JZI'] Categories (4, object): ['C2B', 'CWT', 'EPX', 'JZI'] [0 2 1 3]	Feature: Type ['Airlines', 'Travel Agency'] Categories (2, object): ['Airlines', 'Travel Agency'] [0 1]	Feature: Claimed ['No', 'Yes'] Categories (2, object): ['No', 'Yes'] [0 1]
Feature: Channel ['Online', 'Offline'] Categories (2, object): ['Offline', 'Online'] [1 0]	Feature: Product Name ['Customised Plan', 'Cancellation Plan', 'Bronze Plan', 'Silver Plan', 'Gold Plan'] Categories (5, object): ['Bronze Plan', 'Cancellation Plan', 'Customised Plan', 'Gold Plan', 'Silver Plan'] [2 1 0 4 3]	Feature: Destination ['ASIA', 'Americas', 'EUROPE'] Categories (3, object): ['ASIA', 'Americas', 'EUROPE'] [0 1 2]

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

To split the data into test and train, the following commands are suitable:

X = df.drop('Claimed', axis = 1)

Y = df.pop('Claimed')

For training and testing purpose the dataset can be splitted into train and test data in the ratio 70:30 follwing the 'X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = 0.30, random_state = 1)' command.

The dimensions of the train and test data are obtained as follows:

X_train: (2100, 9)

X_test: (900, 9)

Y_train: (2100,)

Y_train: (900,)

Classification Model using **CART**

In this model 'criterion = 'gini'' is used, and the features tabulated (Table 2.4) according to the importance as obtained using the following command line.

'print(pd.DataFrame(dt_model.feature_importances_,columns=['Importance'],index=X_train.columns). sort_values('Importance', ascending= False))'

To get the optimal values for Decision tree, Grid Search is used with cross validation = 10, as shown below. With the obtained tuning parameters the Decision Tree is regularized as shown in Figure 2.13.

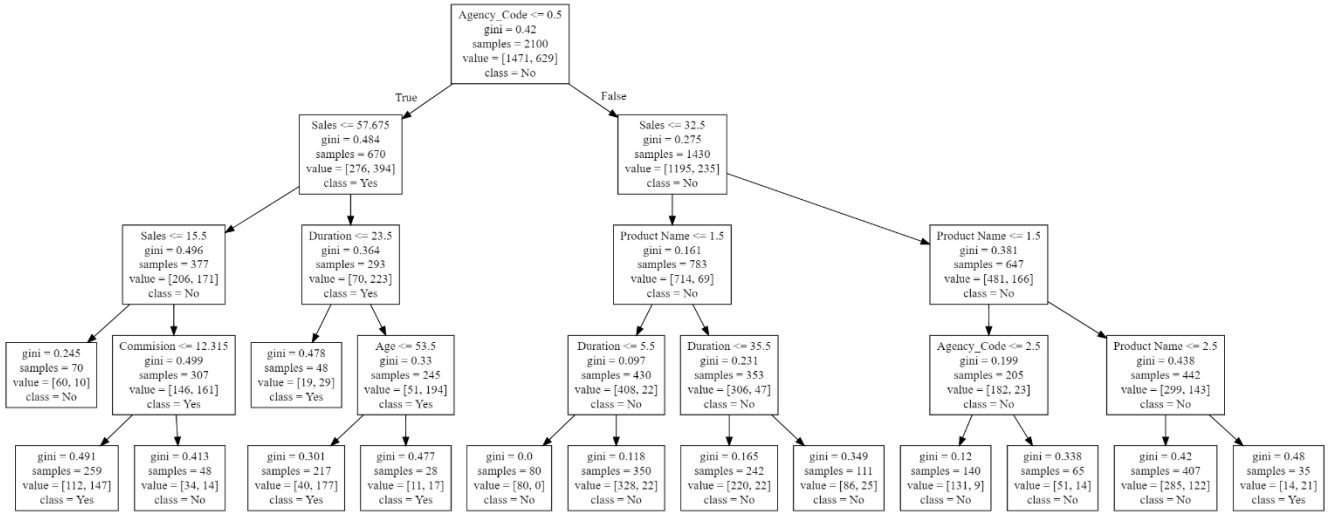


Figure 2.13: Screenshot of the decision tree after tuning the given Dataset. It shows the interrelationship of all the variables available.

The importance of the features also changes once a new tree is generated (Table 2.6).

Table 2.6: Comperative analysis of the important features obtained from decision tree.

	Importance	
	Before Tuning	After Tuning
Agency_Code	0.256154	0.616392
Sales	0.214714	0.252286
Product Name	0.193590	0.077771
Commision	0.176494	0.022912
Duration	0.086691	0.022624
Age	0.037279	0.008015
Type	0.023956	0.000000
Channel	0.007645	0.000000
Destination	0.003478	0.000000

Classification Model using **Random Forest**

The outlier present in four variables namely ‘Age’, ‘Commision’, ‘Sales’, and ‘Duration’. For improving the data quality the outliers are required to be imputed, and Figure 2.14 illustrates the event.

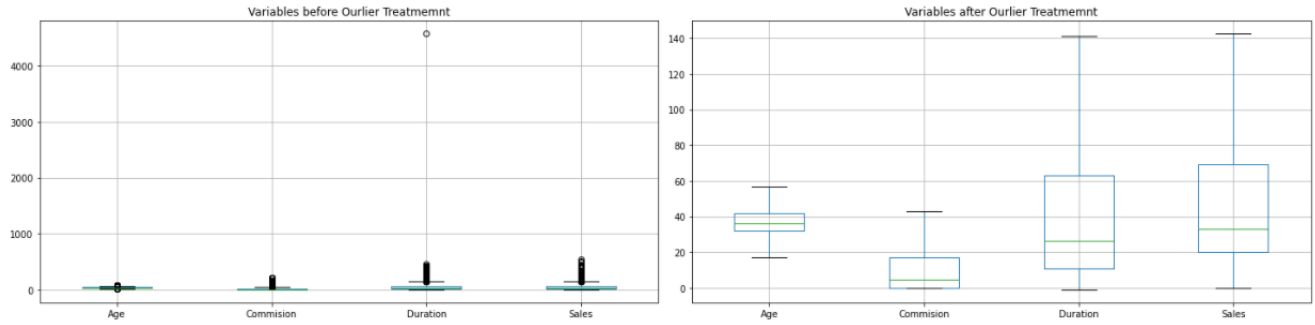


Figure 2.14: Comperative illustration of the four variables with resoect to the outliers.

To get the optimal values for Random Forest, Grid Search is used with cross validation = 10.

The best parameters for the model using Random Forest, are obtained as:
RandomForestClassifier(max_depth=6,max_features=6,min_samples_leaf=8,min_samples_split=60,n_estimators=150)

Classification Model using **Artificial Neural Network**

Using MLPClassifier, the model is trained, and observed that after the iteration 30 the training loss did not improve more than tol=0.010000 for 10 consecutive epochs. Hence terminated the iteration loop.

Like previous, to get the optimal values for Artifician Neural Network, Grid Search is used with cross validation = 10.

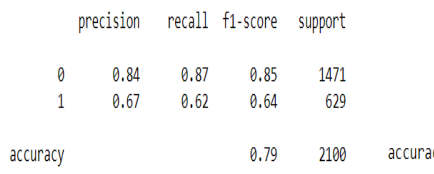
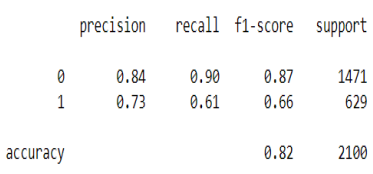
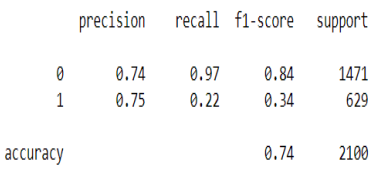
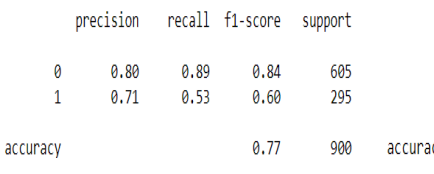
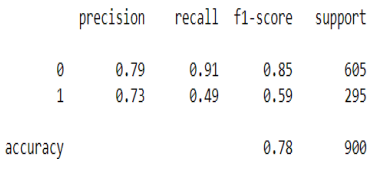
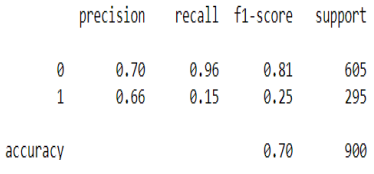
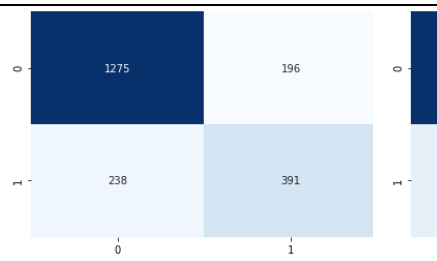
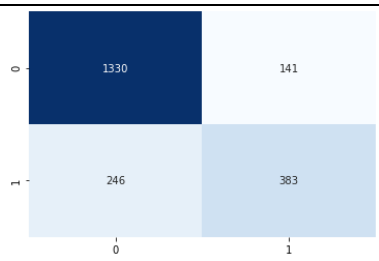
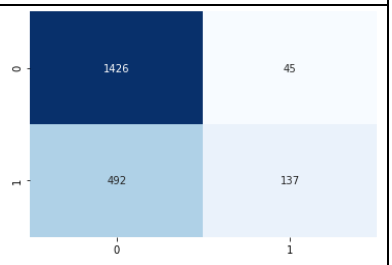
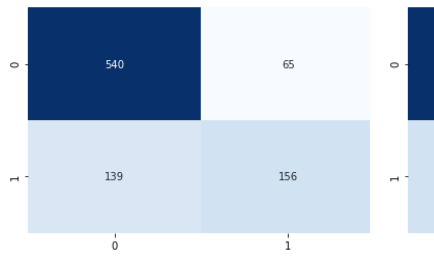
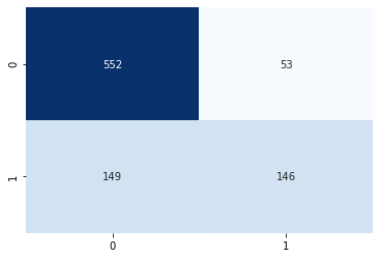
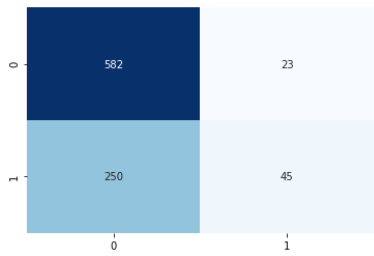
The best parameters for the model using Artificial Neural Network, are obtained as:

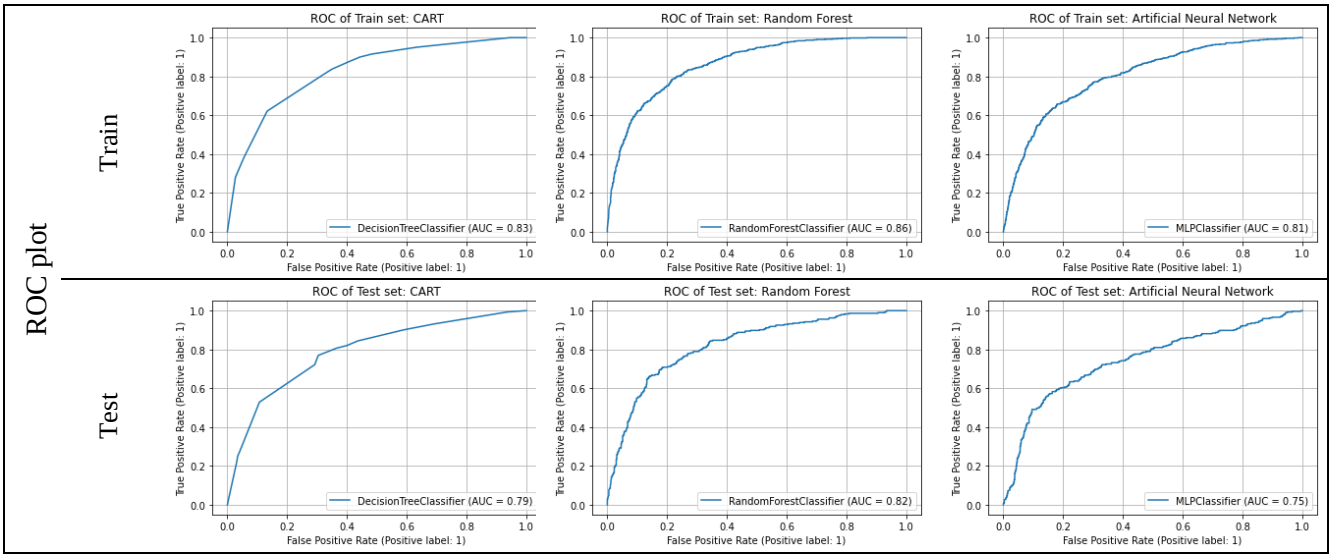
MLPClassifier(hidden_layer_sizes=200, max_iter=2500, solver='sgd', tol=0.01)

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.

The performance metrics of the three models are illustrated in Table 2.7 based on features like accuracy, confusion matrix, ROC plot, area under curve value and classification report.

Table 2.7: Performance of Predictions on Train & Test sets

Model / Feature		CART	Random Forest	Artificial Neural Network
Accuracy	Train	0.79	0.82	0.74
	Test	0.77	0.76	0.70
AUC Score	Train	0.83	0.86	0.81
	Test	0.79	0.82	0.75
Classification Report	Train			
	Test			
Confusion Matrix	Train			
	Test			



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

Using the test set information obtained from three different models (Table 2.8), it is possible to summarise the following observations.

1. The accuracy of the Ramdom Forest model is best than other.
2. The AUC or the area under the curve for the Random Forest model is highest among the three, which inturn offer best performance at distinguishing between the positive and negative classes.
3. Precision is a metric that quantifies the number of correct positive predictions made. The Random Forest model offers the best in comparison to other models.
4. Recall is a metric that quantifies the number of correct positive predictions made out of all positive predictions that could have been made. It is connected to the false negative error or type II error. The CART model produce least false negative error, while the Artificial Neural Network model produce maximum type II error.
5. The same F1 score offered by the Random Forest and CART models, while the Artificial Neural Network model offer least F1 score, which signifies either presicion or recall is very low.

Table 2.8: Performance of Predictions on Train & Test sets

	CART Train	CART Test	RF Train	RF Test	ANN Train	ANN Test
Accuracy	0.79	0.77	0.82	0.78	0.74	0.70
AUC	0.83	0.79	0.86	0.82	0.81	0.75
Precision	0.67	0.71	0.73	0.73	0.75	0.66
Recall	0.62	0.53	0.61	0.49	0.22	0.15
F1 Score	0.64	0.60	0.66	0.59	0.34	0.25

Thus from the above observations, the Random Forest model can be declared as the best model for solving the present Dataset, as this model has best accuracy, area under curve score, precision, F1 score, and good recall value in comparison to the CART and Artificial Neural Network models.

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

In terms of the model, more data will help us understand and predict models better.

As a result of streamlining online experiences, customer satisfaction increased, leading to higher conversion rates, which improved profit.

1. Data shows that 98% of insurance is done online.
2. The other interesting fact is that the majority of offline businesses have an associated website.
3. The JZI resources need to be trained since they are at the bottom and need to be managed. JZI should consider partnering with an agency like EPX to run a promotional marketing campaign.
4. Using ticket or plan data to cross-sell insurance.
5. Agency sales are higher than airline sales, and the agency claim process is more frequent.
6. More revenue as a result of good customer service.
7. Keep fraudulent transactions to a minimum.
8. Improve the claims recovery process.
9. Improve the claim processing process.