Business Report

DATA MINING

Prepared By: ARUNKUMAR S

Date: 05.06.2022

Batch Name: PGPDSBA Online Jan_E 2022

TABLE CONTENTS

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 Dictionary for Market Segmentation:

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).

Sample data of given bank_marketing_part1_Data.csv

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

Table no.1 – Sample data of bank marketing part1.csv (head)

Inferences about the given data base:

Shape
The data set contains rows – 210 and columns - 7

Missing value presence
There is no missing values present in the given data set

Duplicated value presence –
There is no duplicated values present in the given data set

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
```

Figure no.1 – Info of the given data set

Observation:

- 7 variables and 210 records.
- No missing record based on initial analysis.
- All the variables numeric type.

Description of the data

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
count	210.000000	210.000000	210.000000	210.000000	210.000000	210.000000	210.000000
mean	14.847524	14.559286	0.870999	5.628533	3.258605	3.700201	5.408071
std	2.909699	1.305959	0.023629	0.443063	0.377714	1.503557	0.491480
min	10.590000	12.410000	0.808100	4.899000	2.630000	0.765100	4.519000
25%	12.270000	13.450000	0.856900	5.262250	2.944000	2.561500	5.045000
50%	14.355000	14.320000	0.873450	5.523500	3.237000	3.599000	5.223000
75%	17.305000	15.715000	0.887775	5.979750	3.561750	4.768750	5.877000
max	21.180000	17.250000	0.918300	6.675000	4.033000	8.456000	6.550000

Table no.2 – Description of the given data set

Observation:

- Based on summary descriptive, the data looks good.
- We see for most of the variable, mean/medium are nearly equal
- Include a 90% to see variations and it looks distribute evenly
- Std Deviation is high for spending variable

- Spending which is the target variable looks like it's normally distributed as we can see that mean and median are same.
- advance_payments also seems to be normally distributed. This variable might be of use as it shows that customers are paying the amount in advance which is timely payment for the bank.
- The average probability_of_full_payment is 87.10%. Hence we need to analyse further to see the rest of the customers who fall under 13% who have not done the payment in full. This variable is normally distributed.
- Minimum current_balance held by customer is 4899.00.
- credit_limit of customers range between 26300.00 to 40330.00. The average credit_limit of customers is 32586.05.
- The minimum of min_payment_amt paid is 76.51. The maximum of min_payment_amt paid is 845.60. This suggests the data is widely spread for this variable and might have outliers. Also looks like normally distributed.
- The average of max_spent_in_single_shopping is 5408.07. The maximum of max_spent_in_single_shopping is 6550.00.

Univariate Analysis

Box plot

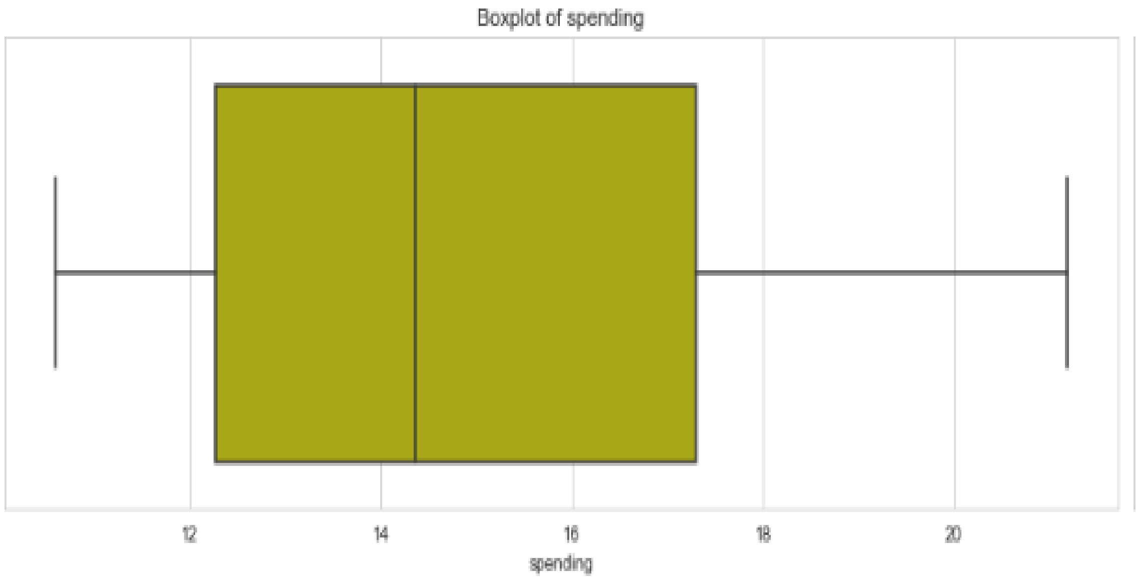


Figure no.2 – Boxplot of spending

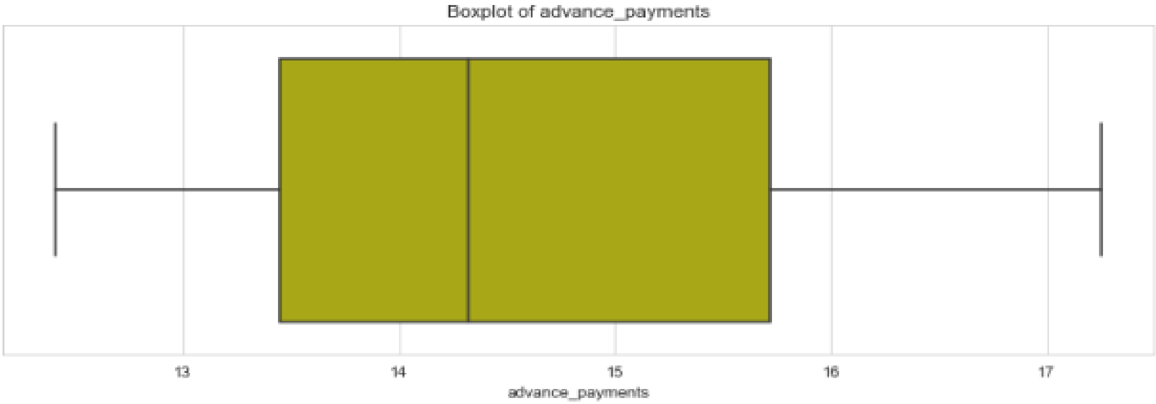


Figure no.3 – Boxplot of advance_payment

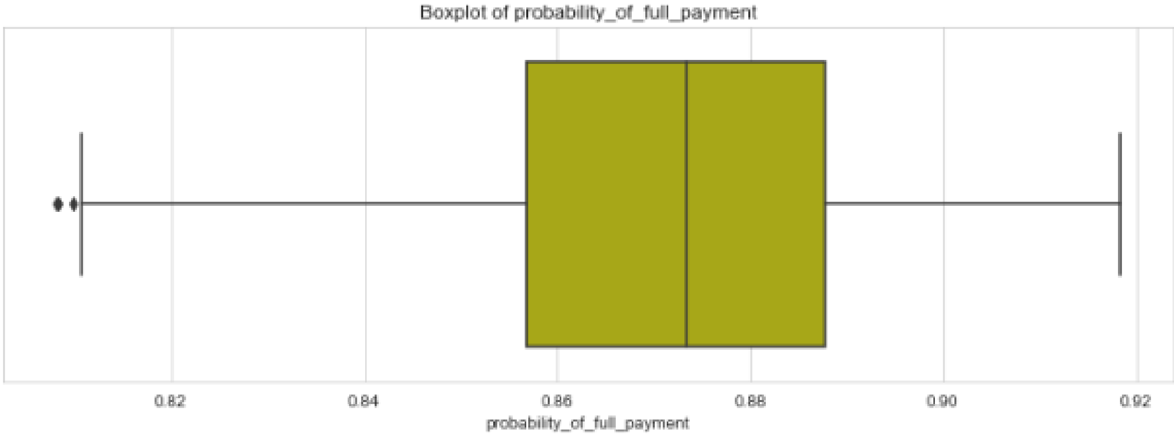


Figure no.4– Boxplot of probability_of_full_payment

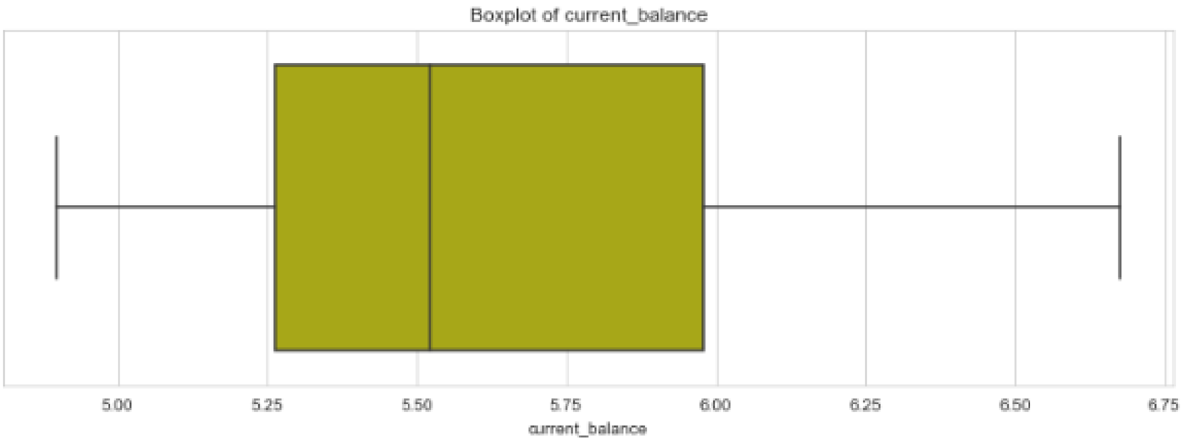


Figure no.5 – Boxplot of current_balance

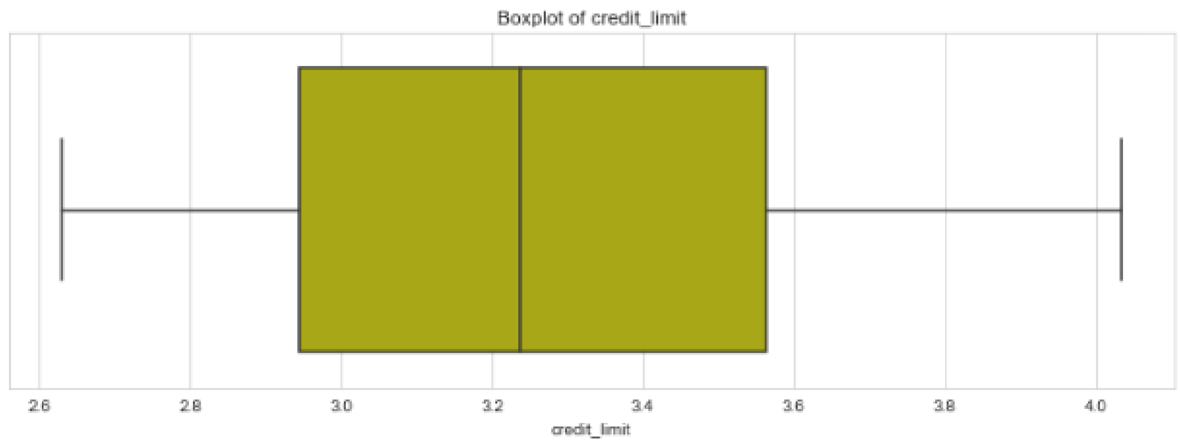


Figure no.6 – Boxplot of credit_limit

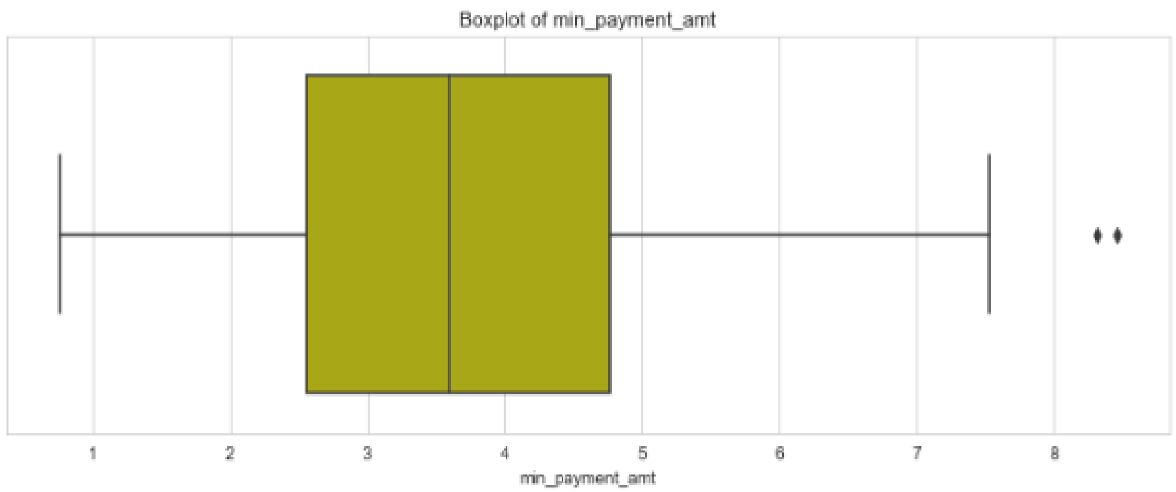


Figure no.7 – Boxplot of min_payment_amt

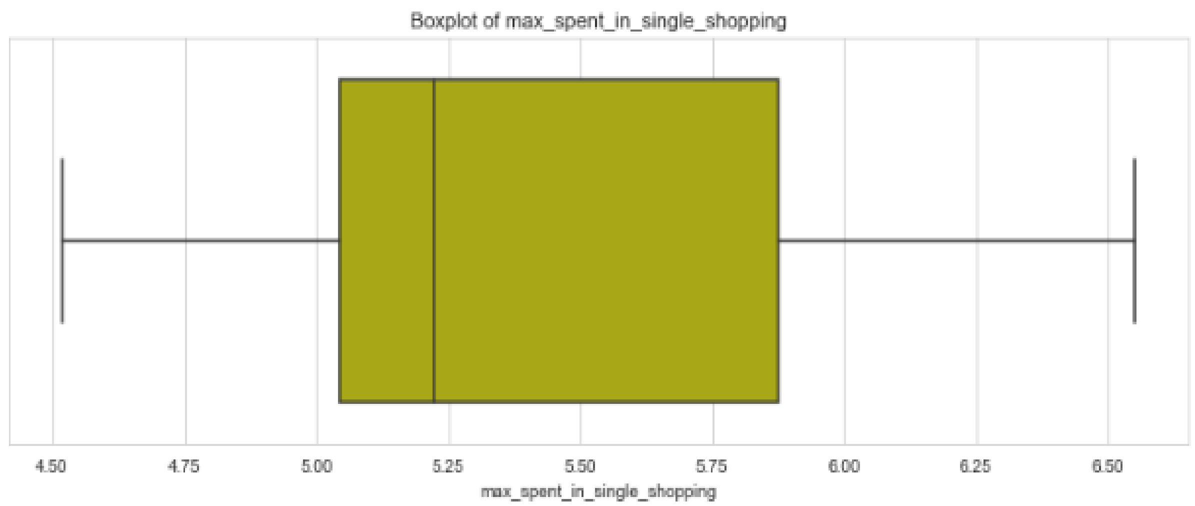


Figure no.8 – Boxplot of max_spent_in_single_shopping

Inferences about the box plot:

spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
No Outliers	No Outliers	Having Outliers	No Outliers	No Outliers	Having Outliers	No Outliers

Table no.3 – Interference about the box plot for outliers presence

Distribution Plots

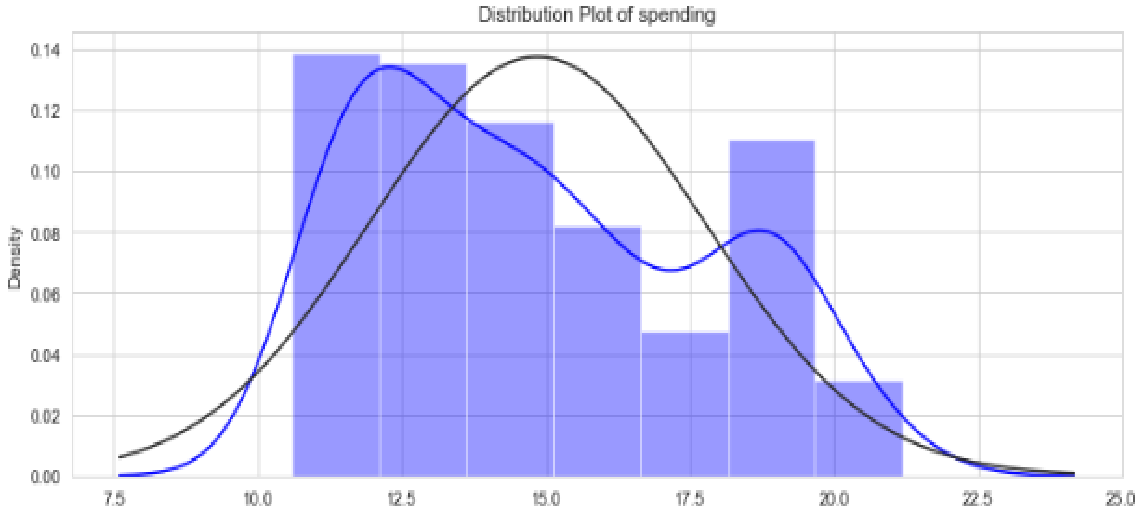


Figure no.9 – Distribution Plot of spending

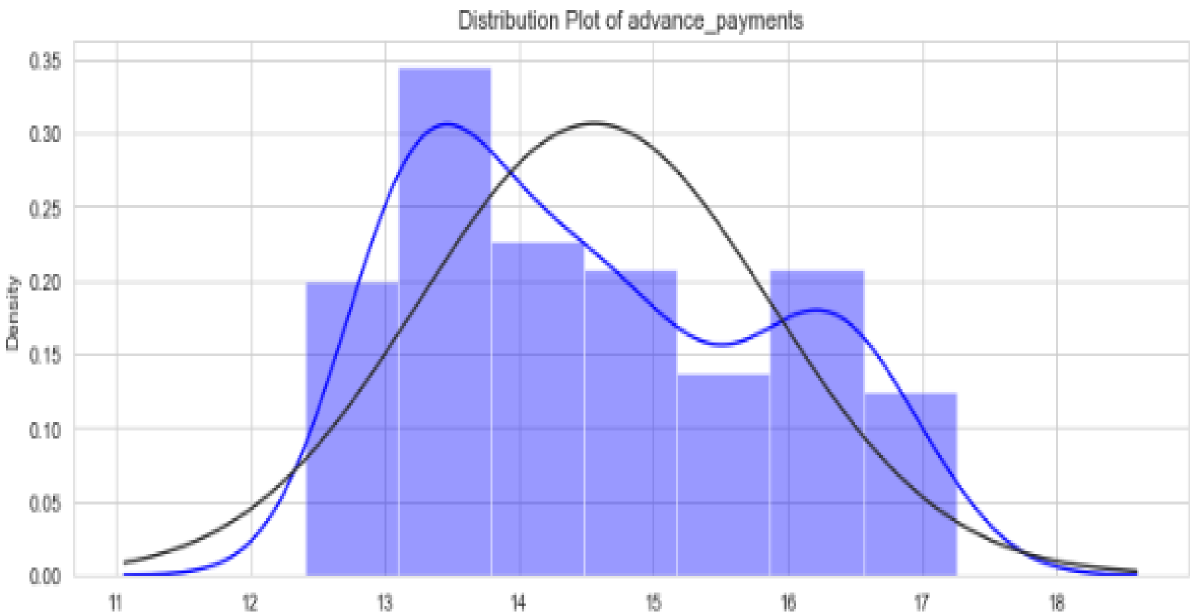


Figure no.10– Distribution Plot of advance_payments

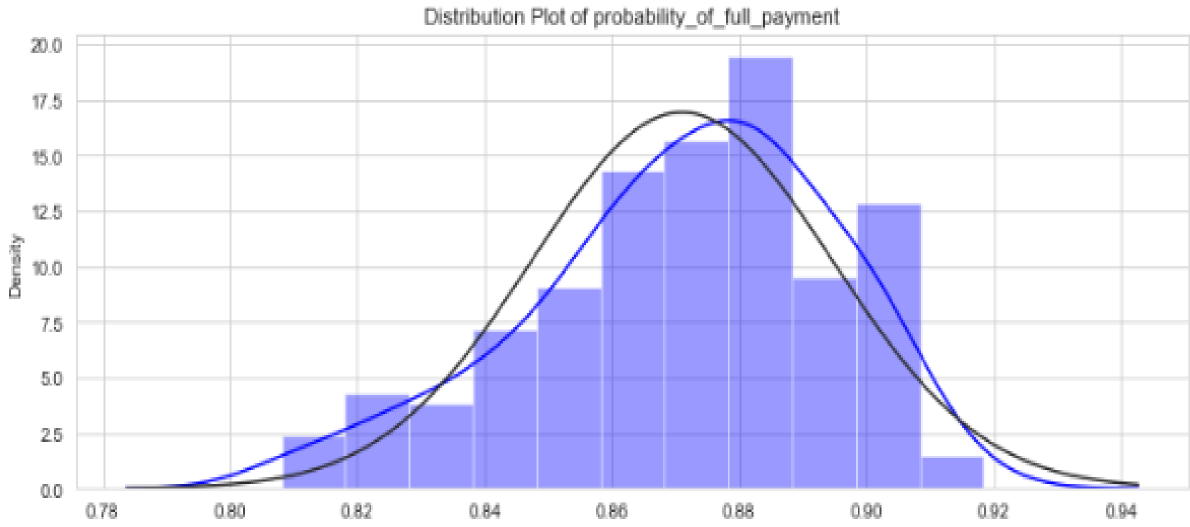


Figure no.11– Distribution Plot of probability_of_full_payment

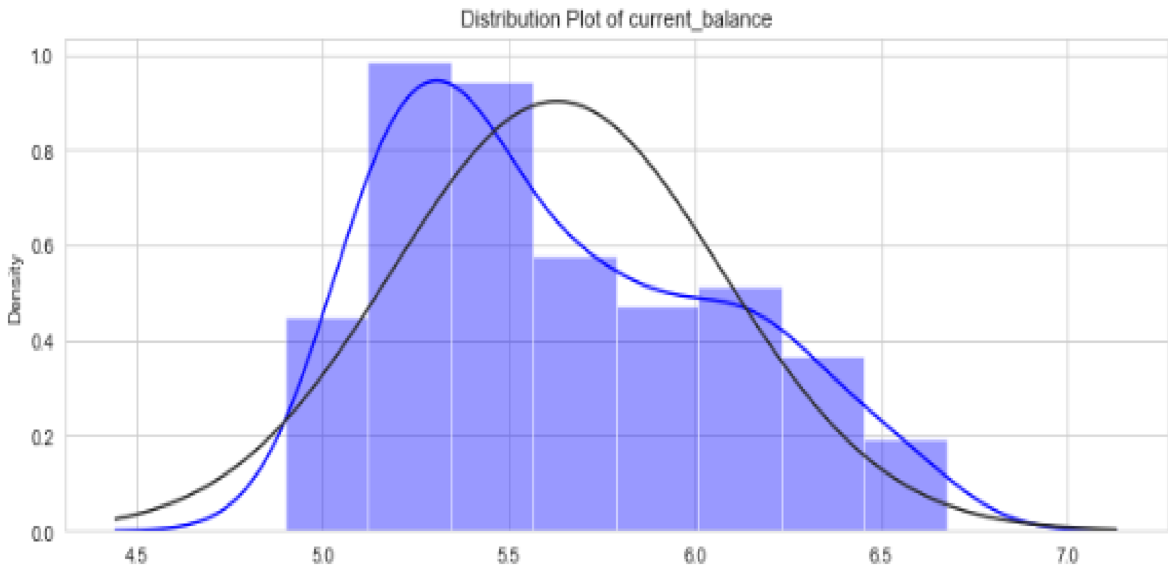


Figure no.12– Distribution Plot of current_balance

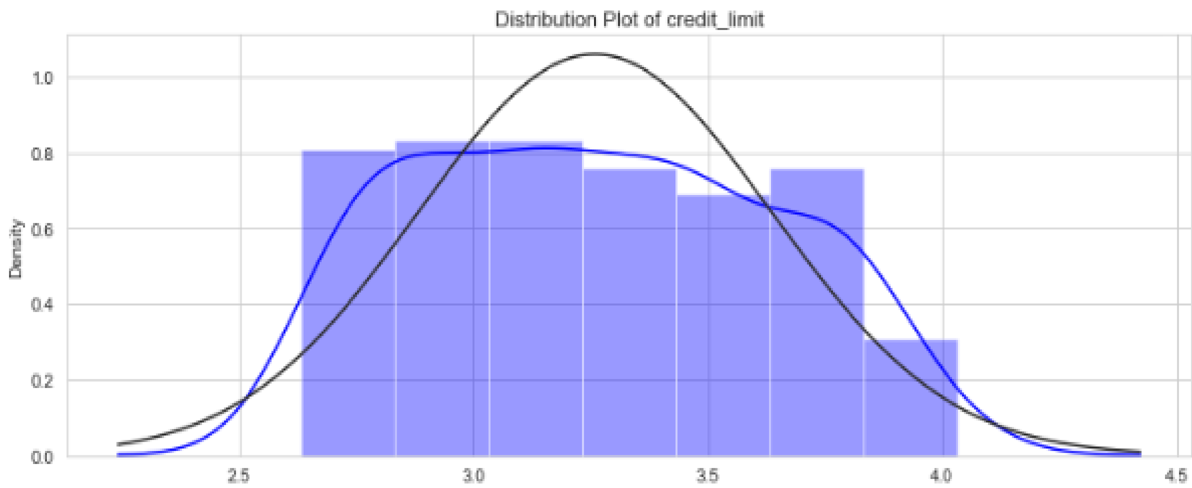


Figure no.13– Distribution Plot of credit_limit

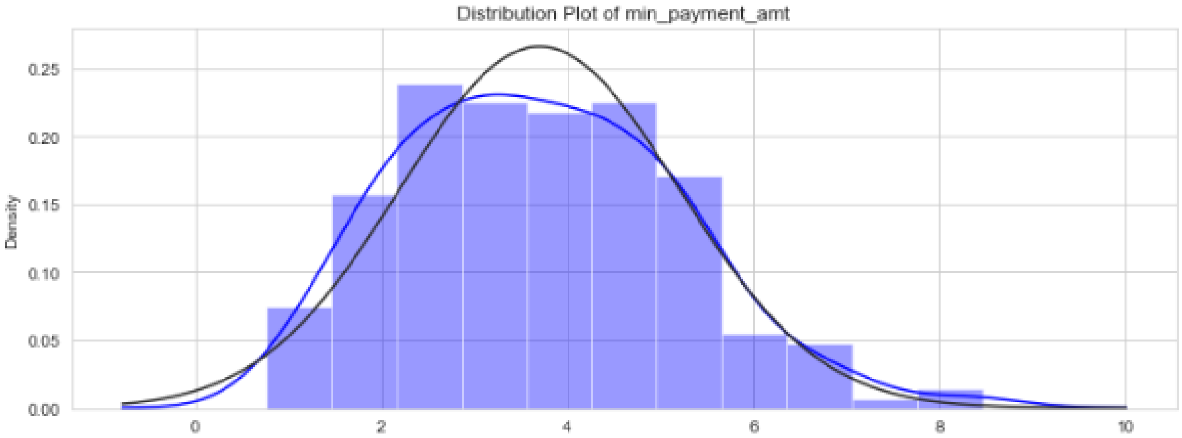


Figure no.14– Distribution Plot of min_payment_amt

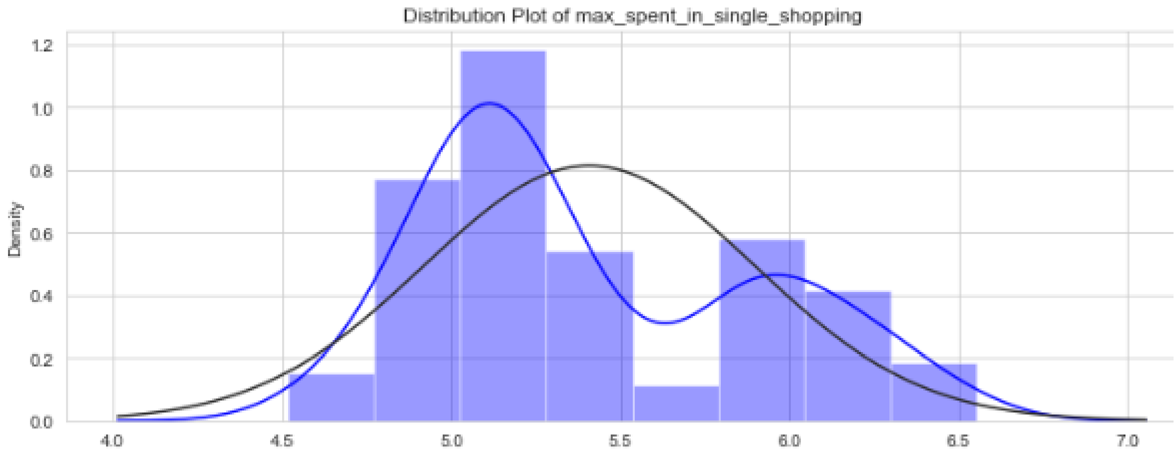


Figure no.15– Distribution Plot of max_spent_in_single_shopping

Skewness and Kurtosis

Skewness of spending is 0.4
Kurtosis of spending is -1.08
Skewness of advance_payments is 0.39
Kurtosis of advance_payments is -1.11
Skewness of probability_of_full_payment is -0.54
Kurtosis of probability_of_full_payment is -0.14
Skewness of current_balance is 0.53
Kurtosis of current_balance is -0.79
Skewness of credit_limit is 0.13
Kurtosis of credit_limit is -1.1
Skewness of min_payment_amt is 0.4
Kurtosis of min_payment_amt is -0.07
Skewness of max_spent_in_single_shopping is 0.56
Kurtosis of max_spent_in_single_shopping is -0.84

Bi- Variaite & Multivariate Analysis

Pairplots

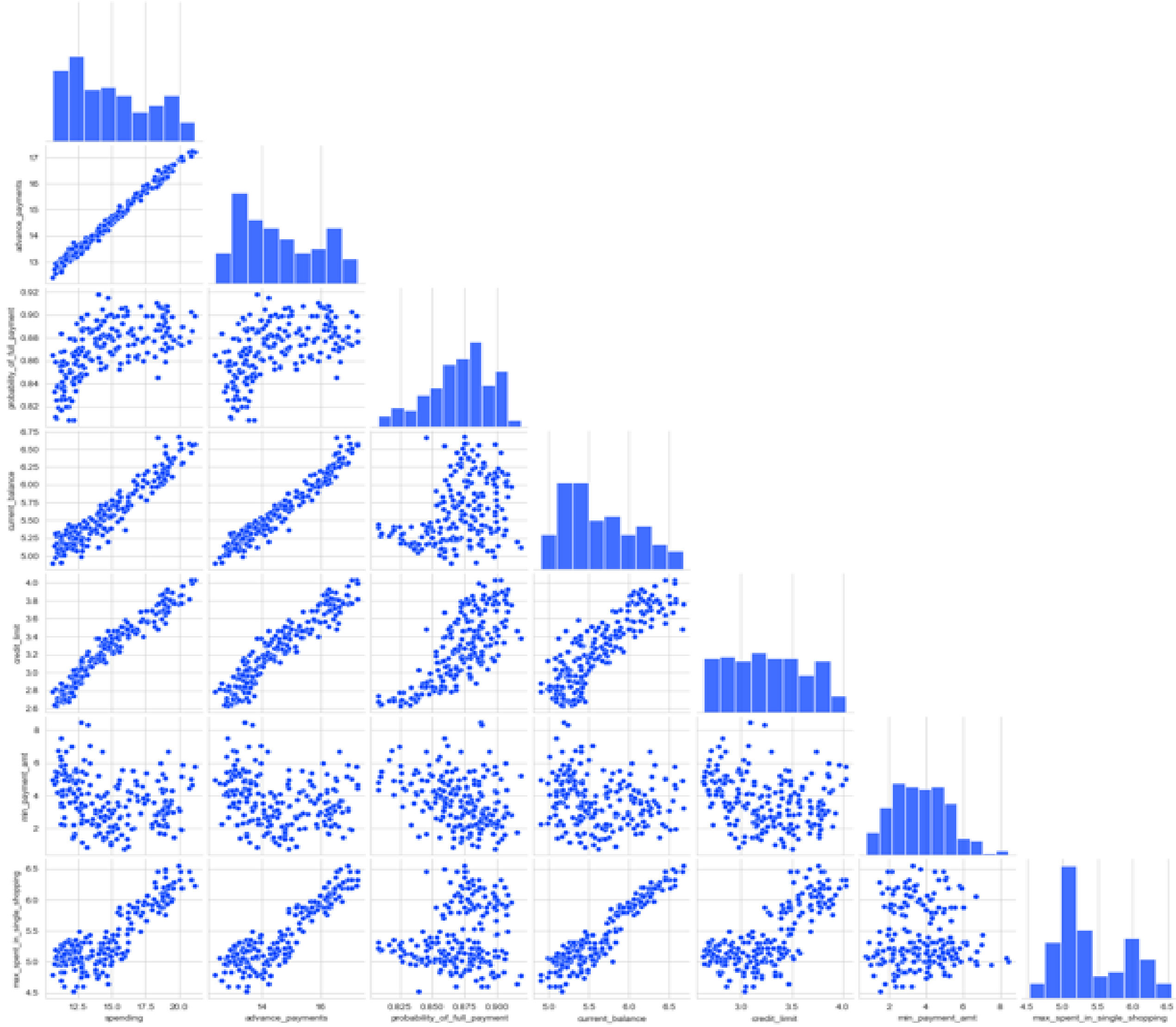


Figure no.16– Pair plot of given data set

Observation - Strong positive correlation between

- spending & advance_payments,
- advance_payments & current_balance,
- credit_limit & spending
- spending & current_balance
- credit_limit & advance_payments
- max_spent_in_single_shopping current_balance

Lmplots

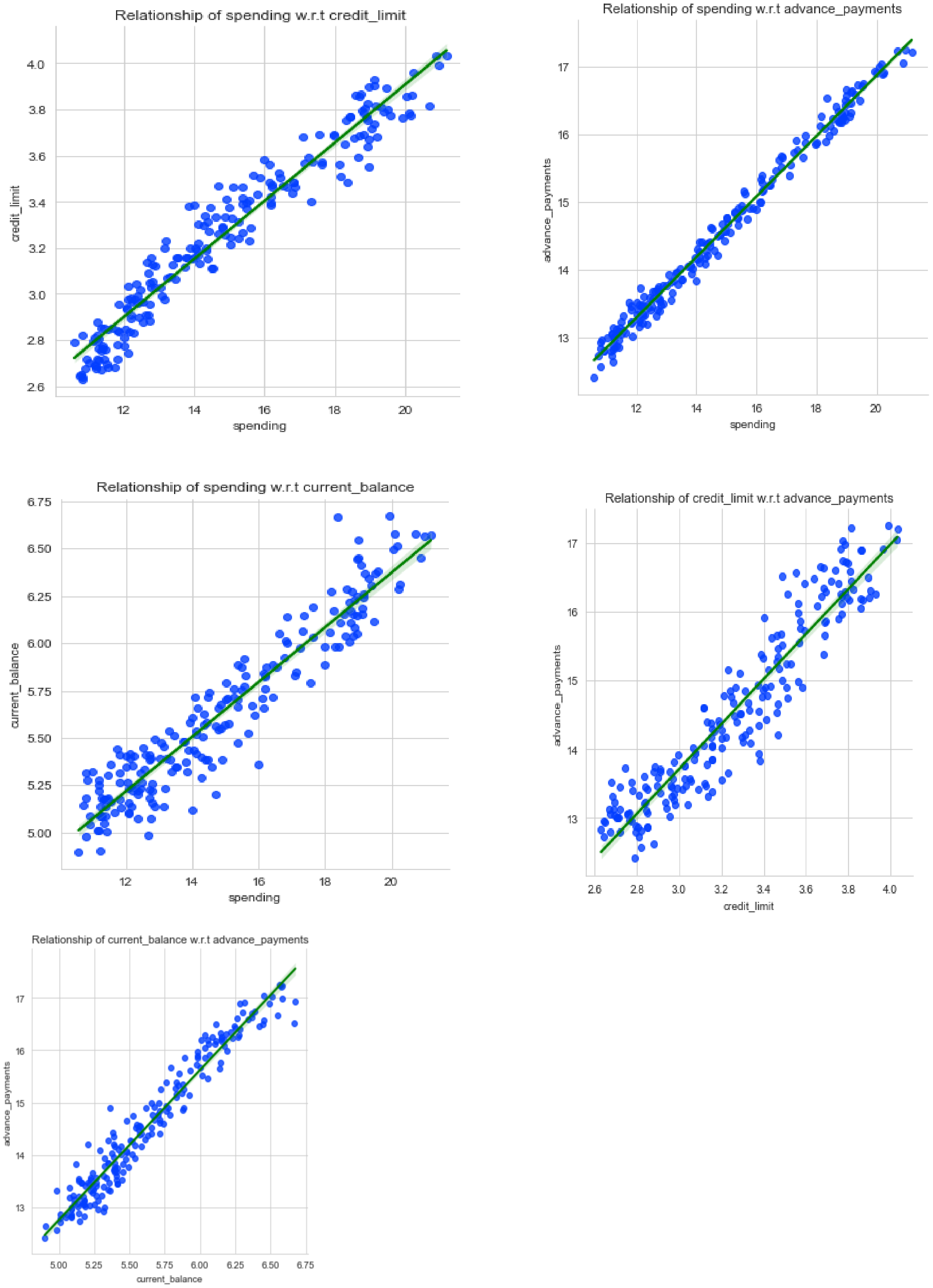


Figure no.17– lmplot of given data set

Correlation Heatmaps

	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

Figure no.18– Correlation Heatmaps of given data set

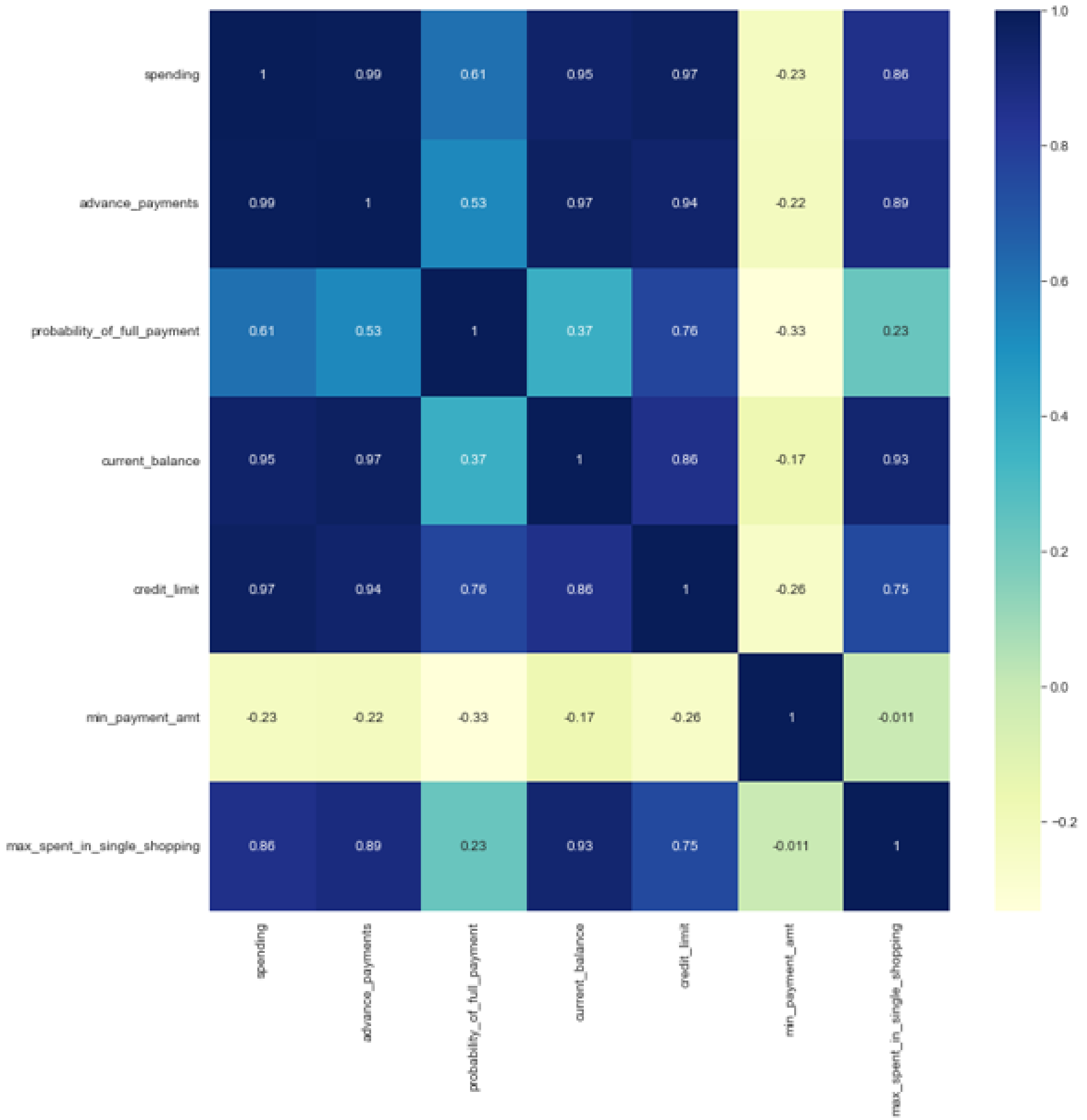


Figure no.19– Correlation Heatmaps plot

Inferences from the above Bivariate & Multivariate Analysis

From the above pairplot and correlation heatmaps, we can see that there is positive linear relationship between advance_payments and spending, current_balance and spending, credit_limit and spending, current_balance and advance_payments, credit_limit and advance_payments, max_spent_in_single_shopping and current_balance. This suggests that there is Multicollinearity between the variables.

Strategy to remove outliers: We choose to replace attribute outlier values by their respective medians, instead of dropping them, as we will lose other column info and also there outlier are present only in two variables and within 5 records.

Replace elements of columns that fall below Q1-1.5*IQR and above Q3+1.5*IQR

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

Scaling or Standardization is an important step in data pre-processing. Most of the machine learning models use scaled data unless the data in hand is naturally scaled.

Let us see the variances between variables in the provided dataset.

```
spending                        8.466351
advance_payments                1.705528
probability_of_full_payment     0.000558
current_balance                 0.196305
credit_limit                    0.142668
min_payment_amt                 2.260684
max_spent_in_single_shopping    0.241553
dtype: float64
```

From the above table though there is not much variance between most of the variables, our target variable spending has a variance of 8.46 whereas other variables variance lie between 0 and 2. Hence scaling is necessary.

We will be using the Standard Scaler method for scaling our data. This method will calculate the z-score for each data point and then scale the data such that mean = 0 and variance/standard deviation = 1.

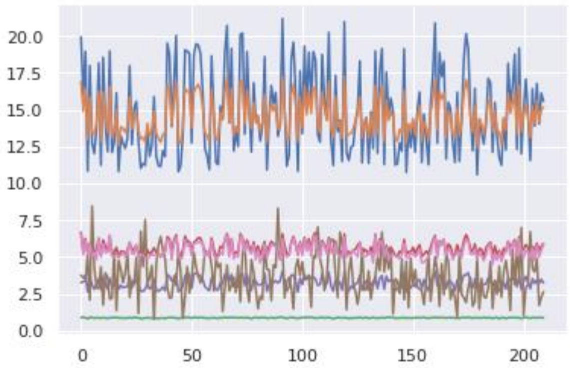
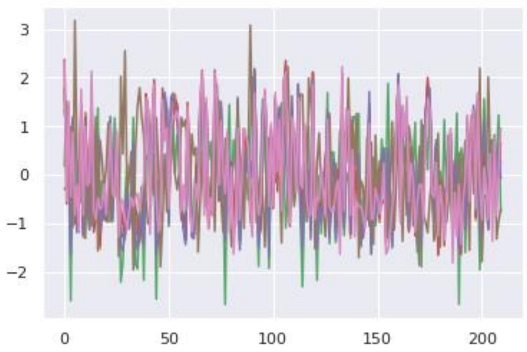


Figure no.20– Before scaling the data



after scaling the data

- Scaling needs to be done as the values of the variables are different.
- Spending, advance_payments are in different values and this may get more weightage.
- Also have shown above the plot of the data before and after scaling.
- Scaling will have all the values in the relative same range.
- I have used zscore to standardize the data to relative same scale -3 to +3.

Scaled Data using StandardScaler function

	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
...	...	...	...	...	...	...	...
205	-0.329866	-0.413929	0.721222	-0.428801	-0.158181	0.190536	-1.366631
206	0.662292	0.814152	-0.305372	0.675253	0.476084	0.813214	0.789153
207	-0.281636	-0.306472	0.364883	-0.431064	-0.152873	-1.322158	-0.830235
208	0.438367	0.338271	1.230277	0.182048	0.600814	-0.953484	0.071238
209	0.248893	0.453403	-0.776248	0.659416	-0.073258	-0.706813	0.960473

210 rows × 7 columns

Figure no.21– After scaling the data of data set

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

Dendrogram of Customers VS Euclidean Distances (Single)

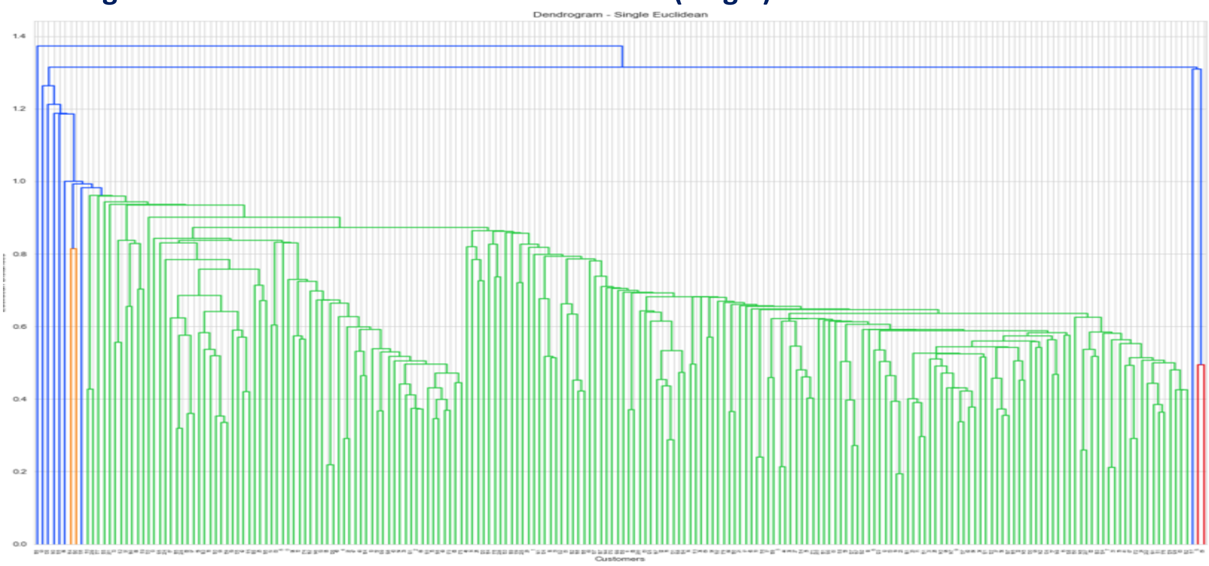


Figure no.22 – Dendrogram single Euclidean

Dendrogram of Customers VS Manhattan Distances (Single)

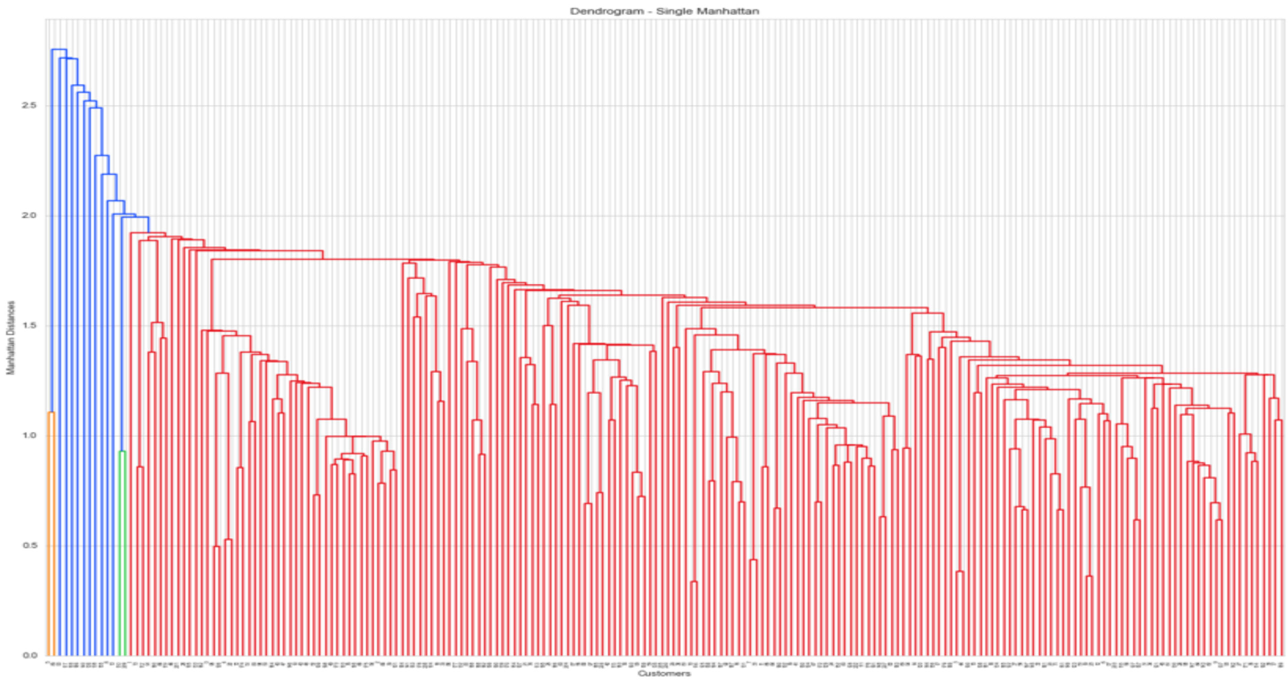


Figure no.23 – Dendrogram single Manhattan

Dendrogram of Customers VS Euclidean Distances (Complete)

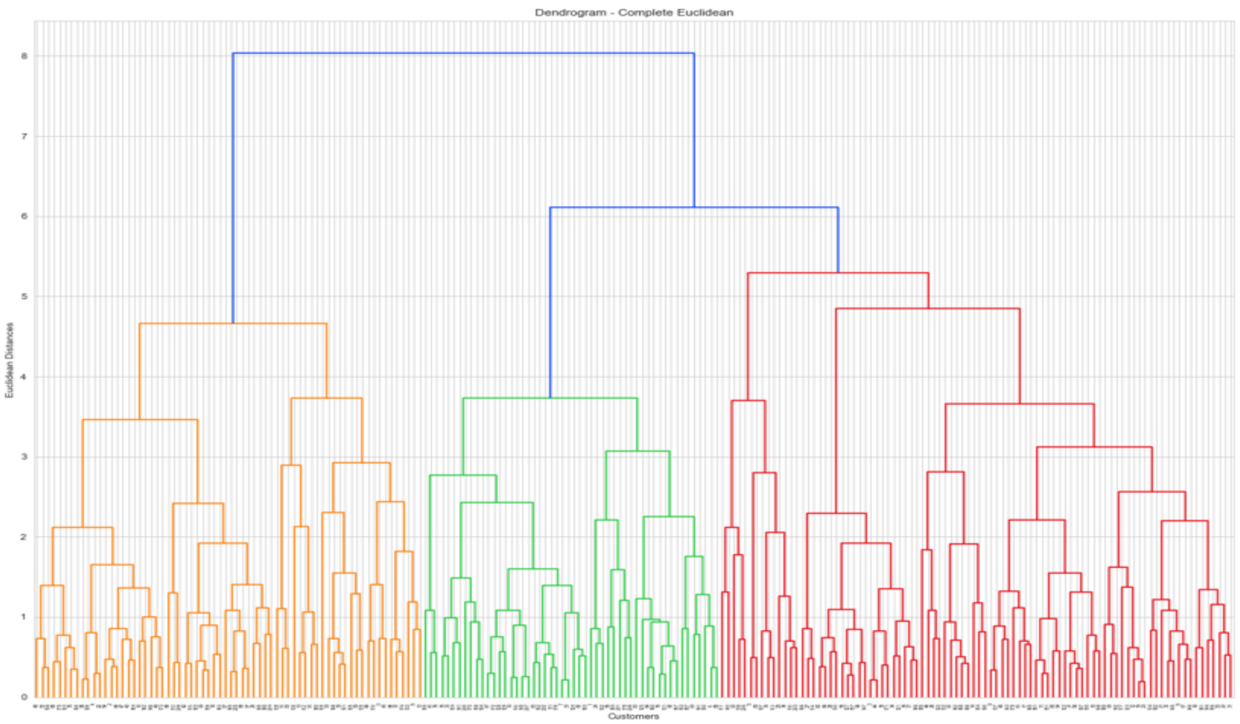


Figure no.24– Dendrogram complete Euclidean

Dendrogram of Customers VS Manhattan Distances (complete)

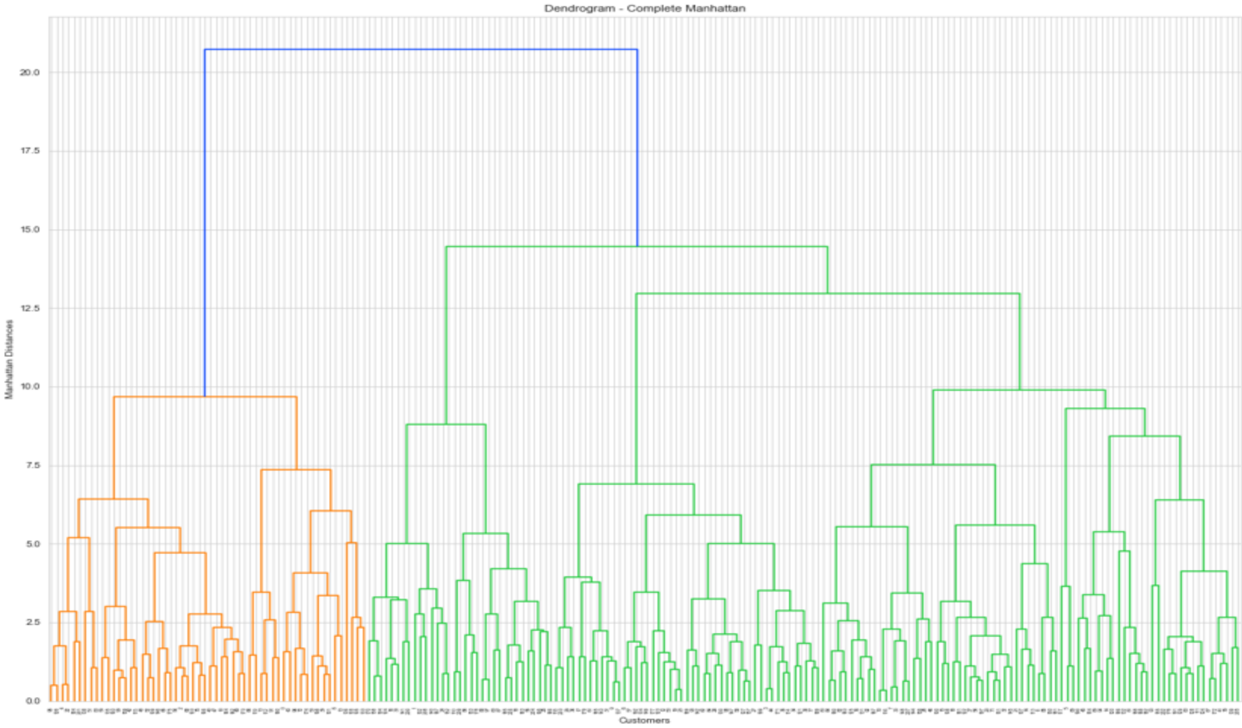


Figure no.25 – Dendrogram complete Manhattan

The number of optimum clusters Dendrogram

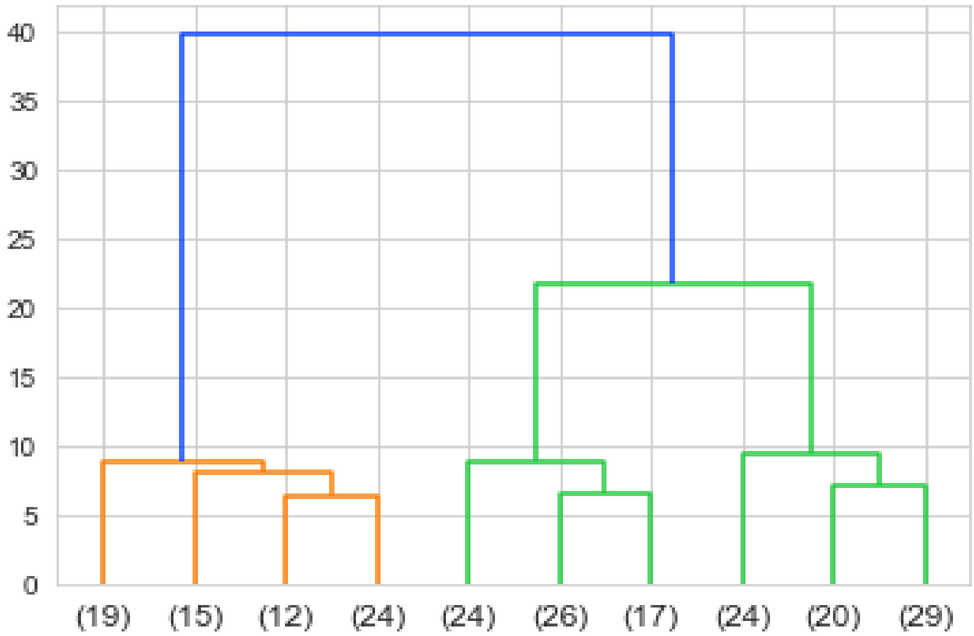


Figure no.26 – Final optimum number of cluster

Hierarchical Clusters visuals using Scatterplot

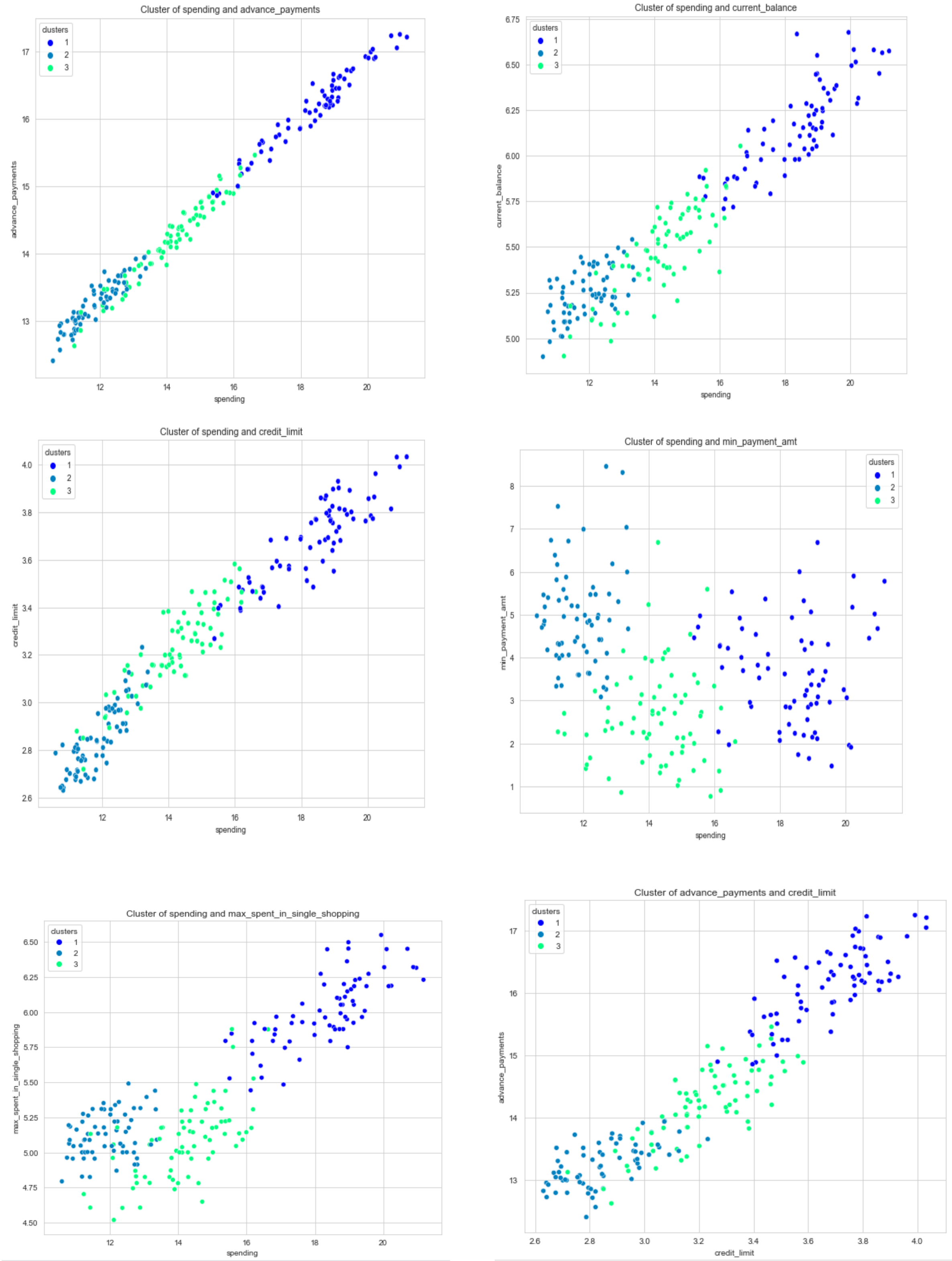


Figure no.27 – Hierarchical Clusters visuals using Scatterplot

Inferences from the above clustering and Dendrogram

- After applied hierarchical clustering to scaled data, we got 3 optimum clustering that we found using 'wardlink' linkage and the criterion used as 'maxclust'

- Clustering obtained via Fclusters function - adding the cluster profiles to the original dataset

	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

Figure no.28 – clusters column added in the data set

- After adding the clustering number into the original data set and we found the clustering frequency as below

Cluster frequency

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

Cluster Profiles

clusters	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	frequency
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

Figure no.29 – clustering profile

- The Dendrogram diagram made using Euclidean and Manhattan distance for both single and complete using ward method.

- Also we see the scatter plot of all three clusters with respect to their spending

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.

- Using sklearn.cluster import KMeans to perform the Kmeans clustering
- After importing the Kmeans then fit the data into Kmeans function
- Iterate the Kmeans inertia_ until we get optimum Kmeans cluster profiles

Let's Calculate WSS for other values of K - Elbow Method

WSS scores keep reducing as we increase the number of clusters

WSS - Values
[1469.9999999999995,
 659.1717544870411,
 430.65897315130064,
 371.2419306631327,
 327.4472622369586,
 289.4975670712945,
 262.8658467199459,
 241.8826310053276,
 223.37789151503583,
 207.33092358250303]

The Elbow Curve for above WSS scores:

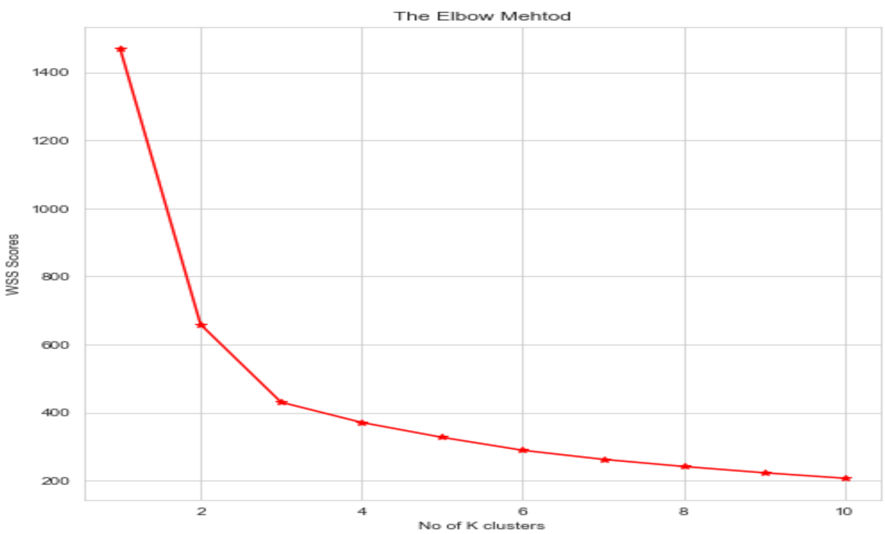


Figure no.30– clustering profile using Elbow method

Let's calculate the silhouette score:

Silhouette value:
i 2 0.46577247686580914
i 3 0.40072705527512986
i 4 0.3347542296283262
i 5 0.28621461554288646
i 6 0.2851581466205877
i 7 0.28238875600233165
i 8 0.25406502102577067
i 9 0.2546806906361287
i 10 0.2562507873415667

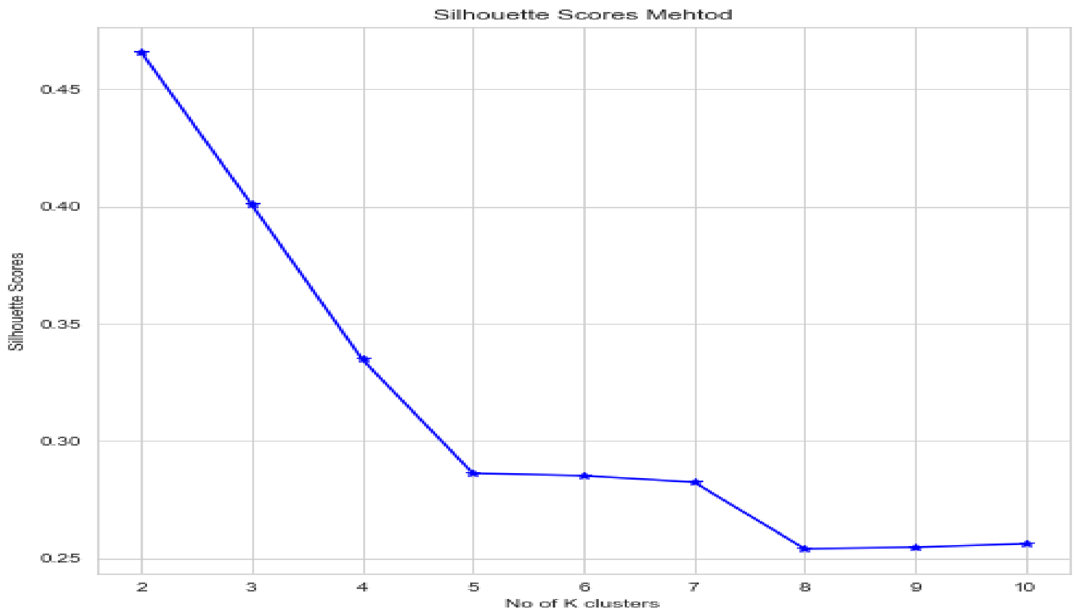


Figure no.31 – clustering profile using Silhouette Scores method

Observation:
Silhouette score is the best for 3 clusters hence we will go with 3 cluster profiling for this dataset

Adding the cluster profiles to the original dataset

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

Figure no.32 – Cluster profiles added data set view

Inference from the above Kmeans Clustering:

- There are three clusters as seen in the above as given in the optimum level

- Cluster frequency

```
0     71
1     72
2     67
Name: k_clusters, dtype: int64
```

- We can see the clustering profile in the order of cluster frequency number and it's pr ofiles combined details as shown in the below,

k_clusters	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping	frequency
0	-0.141119	-0.170043	0.449606	-0.257814	0.001647	-0.661919	-0.585893	71
1	-1.030253	-1.006649	-0.964905	-0.897685	-1.085583	0.694804	-0.624809	72
2	1.256682	1.261966	0.560464	1.237883	1.164852	-0.045219	1.292308	67

Figure no.33 – Clustering profiles

- Let's we see the K-means Clusters profiles in the visual form and refer below scatter plot image for better understanding,

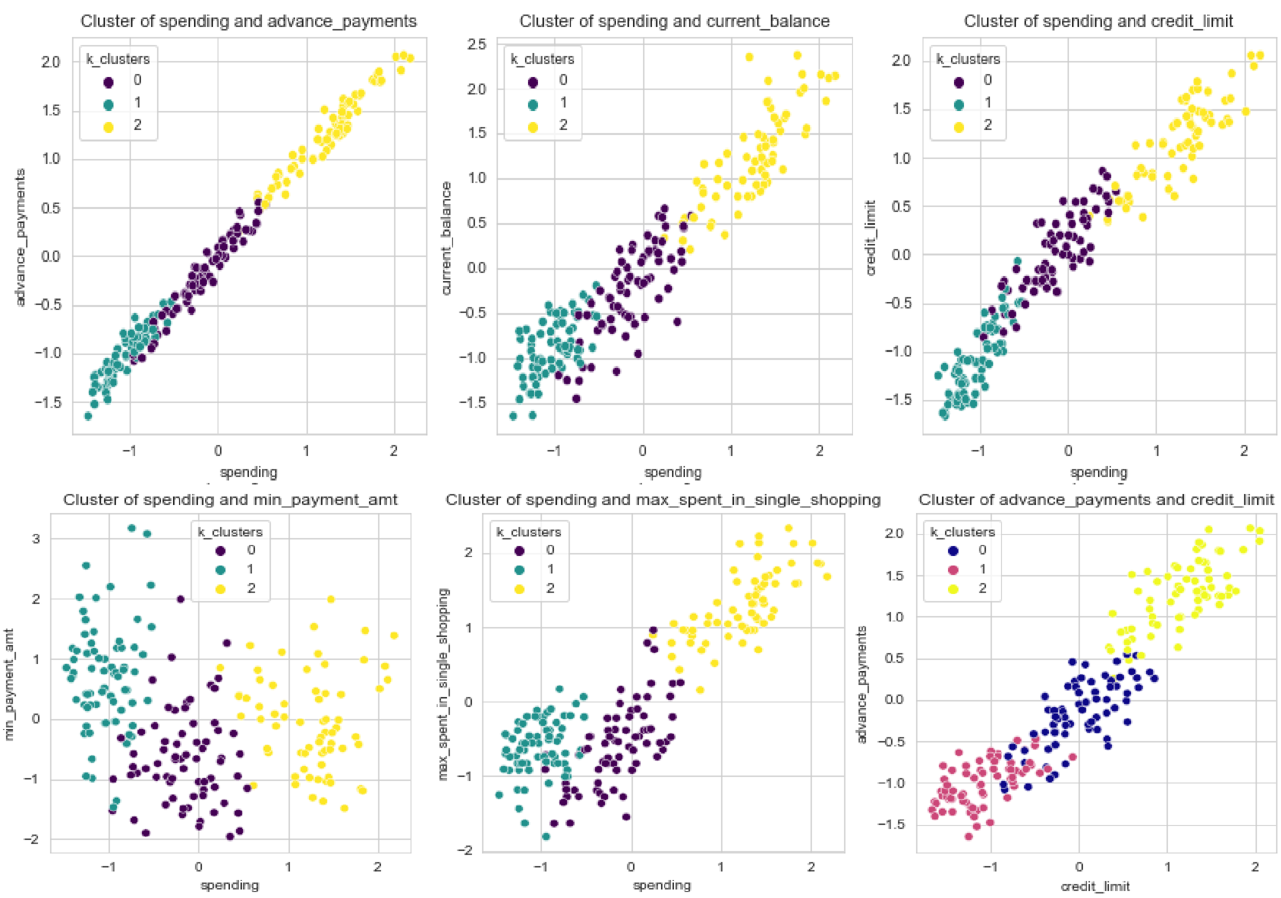


Figure no.34– K-means Clusters Scatterplot of spending VS other variables

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

Three Group cluster via Kmeans

cluster	1	2	3
spending	14.4	11.9	18.5
advance_payments	14.3	13.2	16.2
probability_of_full_payment	0.9	0.8	0.9
current_balance	5.5	5.2	6.2
credit_limit	3.3	2.8	3.7
min_payment_amt	2.7	4.7	3.6
max_spent_in_single_shopping	5.1	5.1	6.0
clusters	2.9	2.1	1.0

Figure no.35– K-means Clusters group

Cluster Group Profiles

- Group 2 : Low Spending / Silver customers
- Group 1 : Medium Spending / Gold customers
- Group 3 : High Spending / Platinum customers

Promotional strategies for each cluster

High Spending Group / Platinum customers

- Giving any reward points might increase their purchases.
- Maximum max_spent_in_single_shopping is high for this group, so can be offered disco unt/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

Medium Spending Group / Gold customers

- 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

Low Spending Group / Silver customers

- Customers should be given remainders for payments. Offers can be provided on earl y payments to improve their payment rate.
- Increase their spending habits by tying up with grocery stores, utilities (electricity, ph one, gas, others)

***** End of Problem 1*****

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.

Attribute 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).

Sample data of given insurance_part2_data.csv

	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

Figure no.36– Sample data of given insurance_part2_data.csv

Info of the given data set

```
<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
```

Figure no.37 – info of given insurance_part2_data.csv

Observation:

- 10 variables
- Age, Commision, Duration, Sales are numeric variable
- rest are categorical variables
- 3000 records, no missing one
- 9 independent variable and one target variable – Claimed
- The shape of the data is 3000 rows and 10 columns

Missing Value checking

```
Age             0
Agency_Code     0
Type            0
Claimed         0
Commision       0
Channel         0
Duration        0
Sales           0
Product Name    0
Destination     0
dtype: int64
```

Figure no.38 – Missing value result

Observation:

There is no missing values presence in the data set

Description of the data

	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

Figure no.39– Description of the given data

Observation

- Duration has negative value, it is not possible. Wrong entry.
- Commision & Sales- mean and median varies significantly
- Categorical code variable maximum unique count is 5

Head of the given data

	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
5	45	JZI	Airlines	Yes	15.75	Online	8	45.00	Bronze Plan	ASIA
6	61	CWT	Travel Agency	No	35.64	Online	30	59.40	Customised Plan	Americas
7	36	EPX	Travel Agency	No	0.00	Online	16	80.00	Cancellation Plan	ASIA
8	36	EPX	Travel Agency	No	0.00	Online	19	14.00	Cancellation Plan	ASIA
9	36	EPX	Travel Agency	No	0.00	Online	42	43.00	Cancellation Plan	ASIA

Figure no.40 – Head (1st 10 rows) of the given data

Tail of the given data

	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product Name	Destination
2990	51	EPX	Travel Agency	No	0.00	Online	2	20.00	Customised Plan	ASIA
2991	29	C2B	Airlines	Yes	48.30	Online	381	193.20	Silver Plan	ASIA
2992	28	CWT	Travel Agency	No	11.88	Online	389	19.80	Customised Plan	ASIA
2993	36	EPX	Travel Agency	No	0.00	Online	234	10.00	Cancellation Plan	ASIA
2994	27	C2B	Airlines	Yes	71.85	Online	416	287.40	Gold Plan	ASIA
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

Figure no.41 – Tail (last 10 rows) of the given data

Observation

- Data looks good at first glance

Presence of Duplicated values

Number of duplicate rows = 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

139 rows × 10 columns

Figure no.42– presence of duplicated values in the given data set

Observation

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

Univariate Analysis

Age variable

Central values

```
Minimum Age:  8
Maximum Age:  84
Mean value:  38.091
Median value:  36.0
Standard deviation:  10.463518245377944
Null values:  False
```

Quartiles

```
spending - 1st Quartile (Q1) is:  32.0
spending - 3st Quartile (Q3) is:  42.0
Interquartile range (IQR) of Age is  10.0
```

Outlier detection from Interquartile range (IQR) in original data

```
Lower outliers in Age:  17.0
Upper outliers in Age:  57.0
```

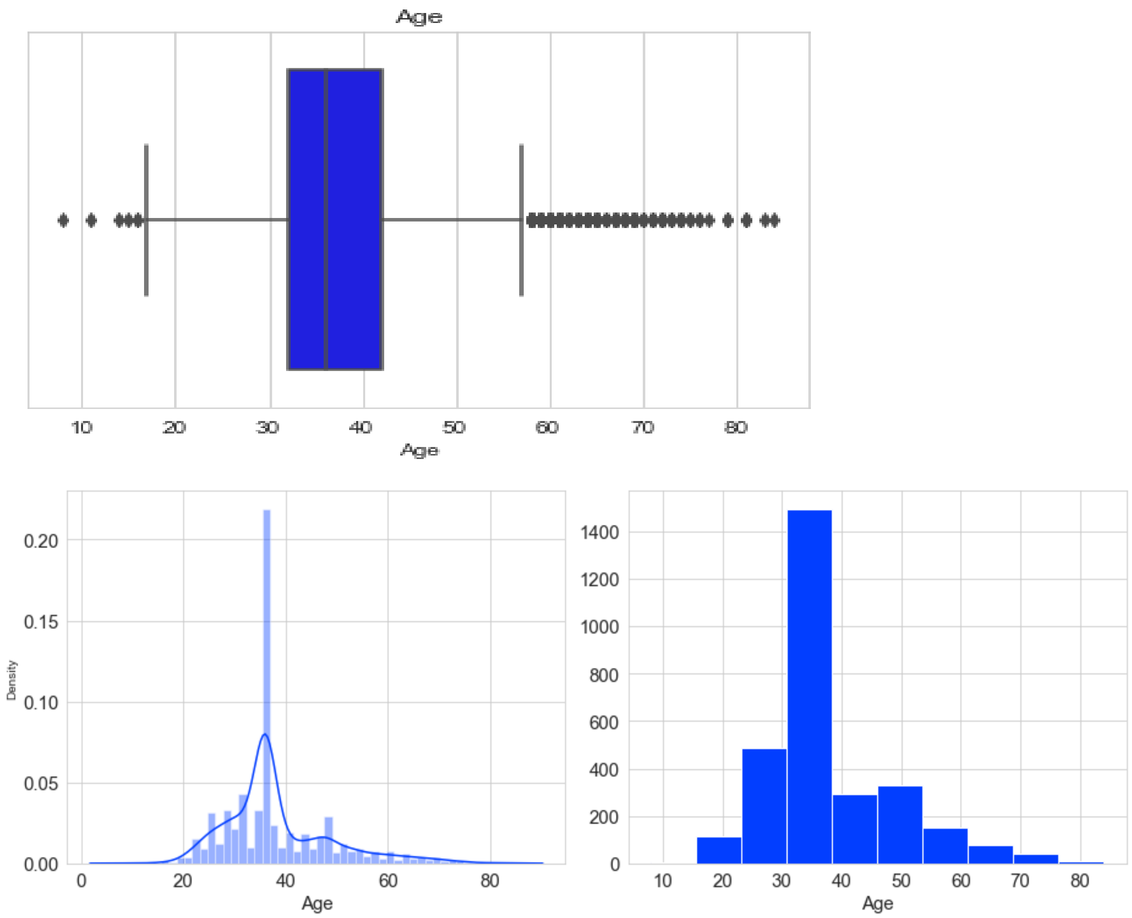


Figure no.43– Boxplot, distplot and histplot of the Age variable

Commision variable

Central values

```
Minimum Commision:  0.0
Maximum Commision:  210.21
Mean value:  14.529203333333266
Median value:  4.63
Standard deviation:  25.48145450662553
Null values:  False
```

Quartiles

```
Commision - 1st Quartile (Q1) is:  0.0
Commision - 3st Quartile (Q3) is:  17.235
Interquartile range (IQR) of Commision is  17.235
```

Outlier detection from Interquartile range (IQR) in original data

```
Lower outliers in Commision:  -25.8525
Upper outliers in Commision:  43.0875
```

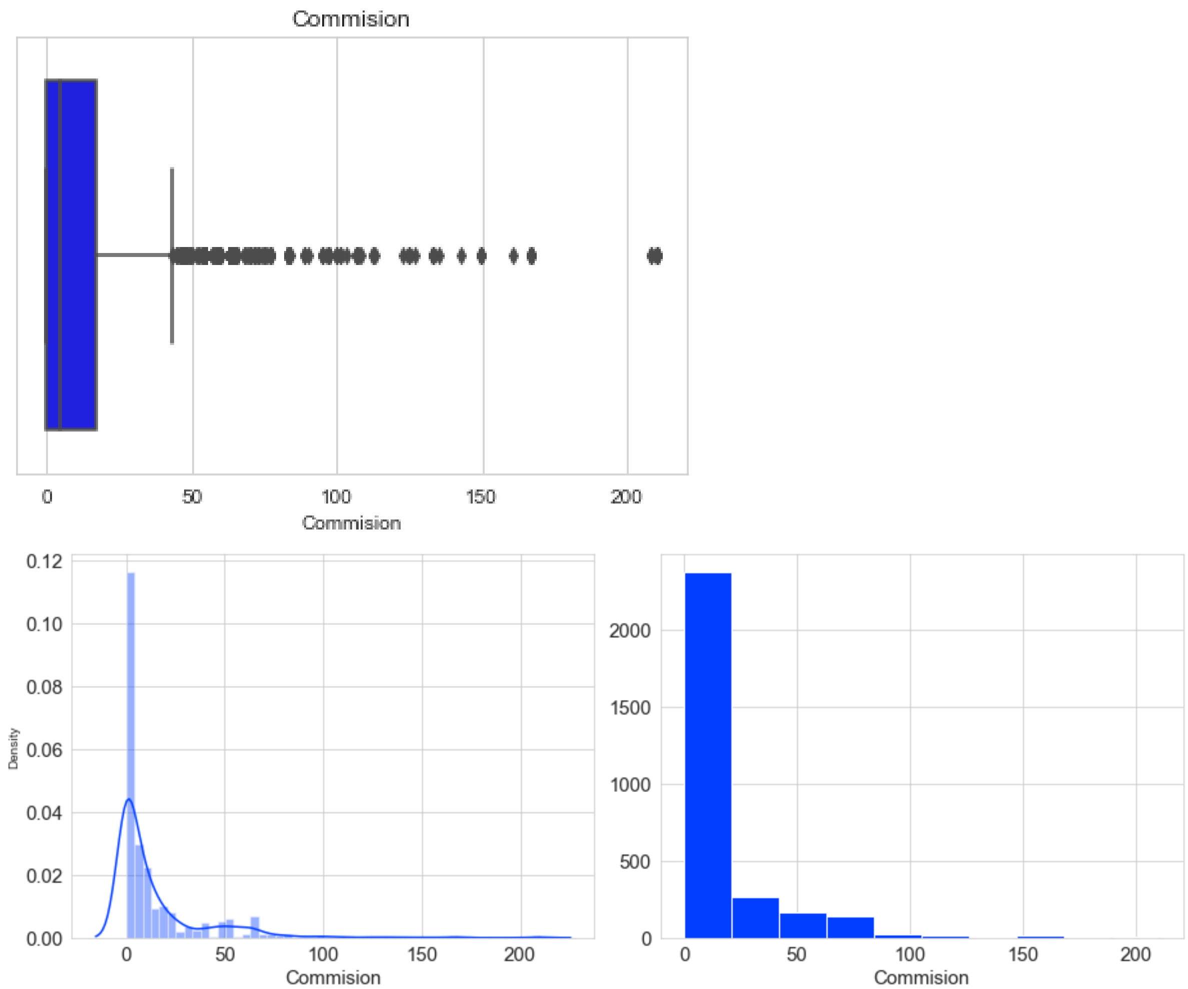


Figure no.44– Boxplot, distplot and histplot of the commision variable

Duration variable

Central values

```
Minimum Duration:  -1
Maximum Duration:  4580
Mean value:  70.00133333333333
Median value:  26.5
Standard deviation:  134.05331313253495
Null values:  False
```

Quartiles

```
Duration - 1st Quartile (Q1) is:  11.0
Duration - 3st Quartile (Q3) is:  63.0
Interquartile range (IQR) of Duration is  52.0
```

Outlier detection from Interquartile range (IQR) in original data

```
Lower outliers in Duration:  -67.0
Upper outliers in Duration:  141.0
```

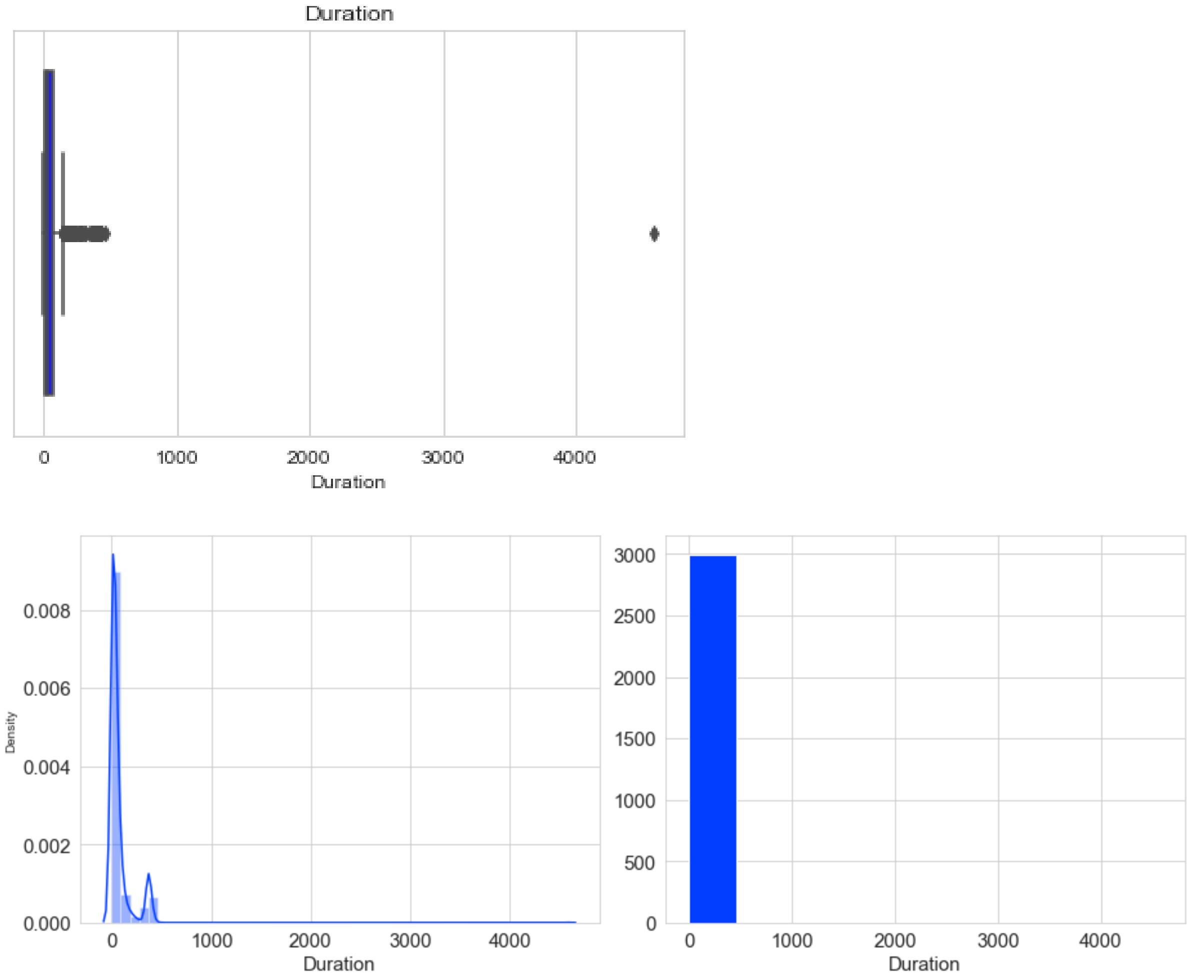


Figure no.45– Boxplot, distplot and histplot of the Duration variable

Sales variable

Central values

```
Minimum Sales:  0.0
Maximum Sales:  539.0
Mean value:  60.24991333333344
Median value:  33.0
Standard deviation:  70.73395353143047
Null values:  False
```

Quartiles

```
Sales - 1st Quartile (Q1) is:  20.0
Sales - 3st Quartile (Q3) is:  69.0
Interquartile range (IQR) of Sales is  49.0
```

Outlier detection from Interquartile range (IQR) in original data

```
Lower outliers in Sales:  -53.5
Upper outliers in Sales:  142.5
```

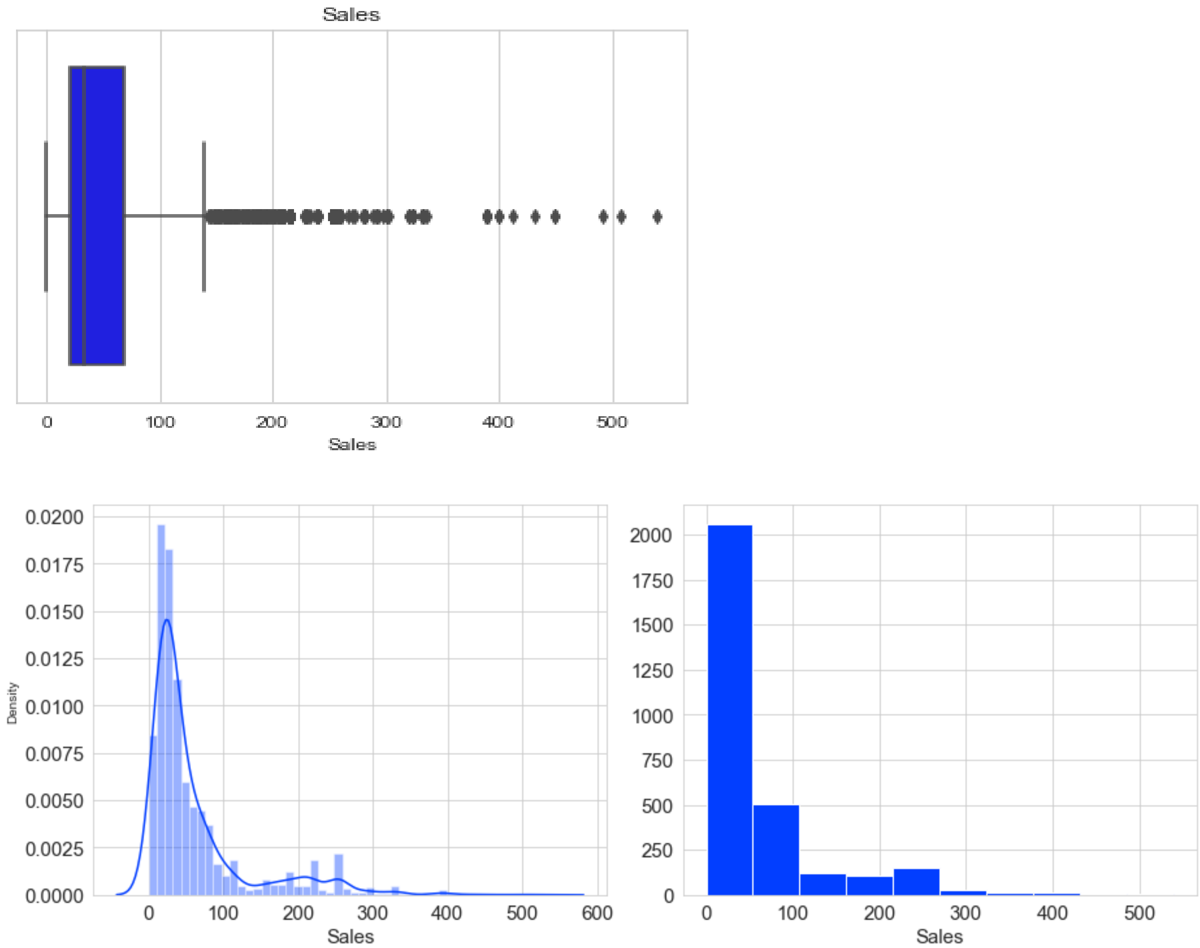


Figure no.46– Boxplot, distplot and histplot of the Sales variable

Inference from the Univariate Analysis

- There are outliers in all the variables, but the sales and commision can be a genuine business value. Random Forest and CART can handle the outliers. Hence, Outliers are not treated for now, we will keep the data as it is.
- We will treat the outliers for the ANN model to compare the same after the all the steps just for comparison.

Categorical Variables

Agency_Code

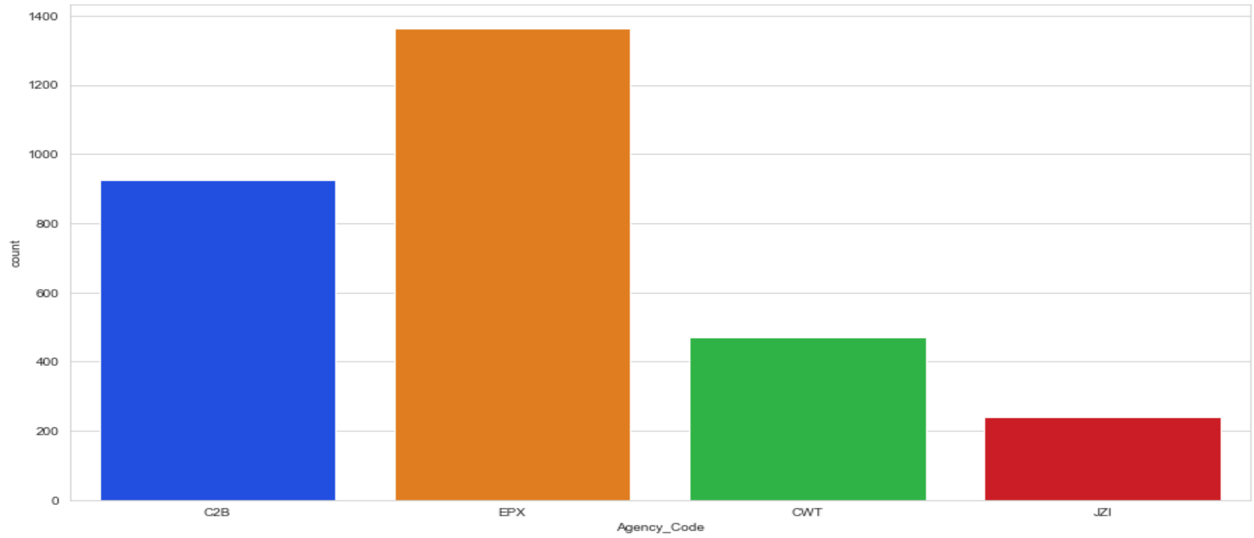


Figure no.47– Count plot for Agency_Code of the Categorical variable

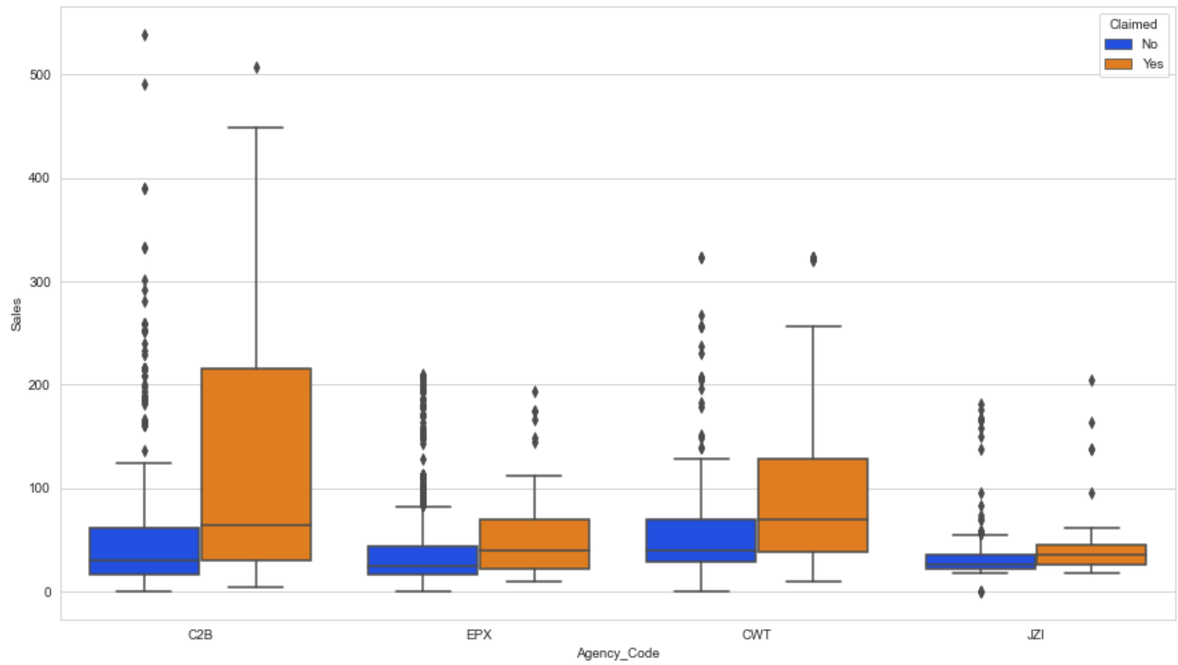


Figure no.48– Boxplot for Agency_Code of the Categorical variable

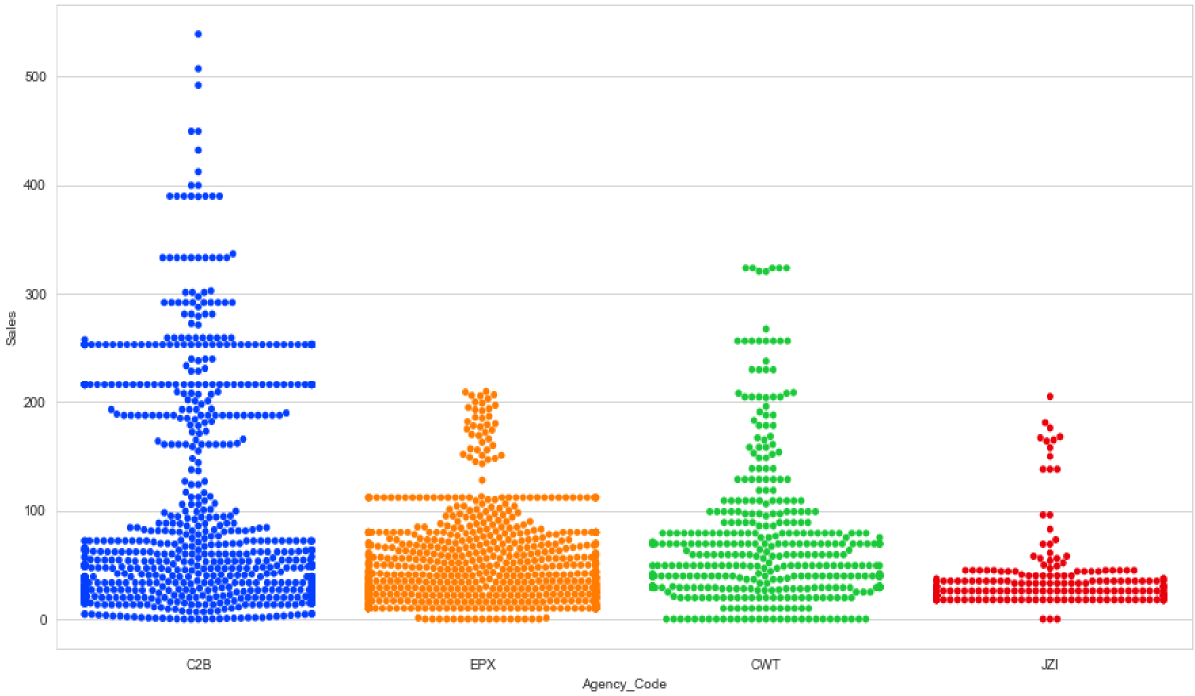


Figure no.49– Swarmplot for Agency_Code of the Categorical variable

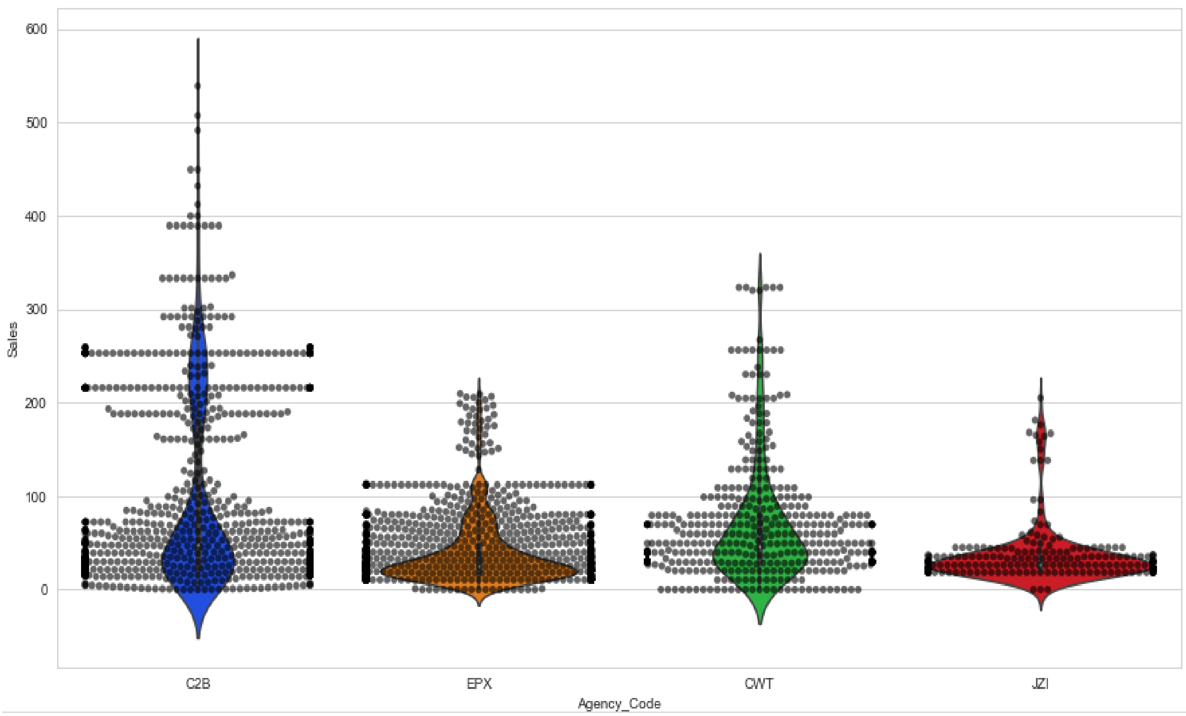


Figure no.50– Combine Violin plot and Swarmplot for Agency_Code of the Categorical variable

Type

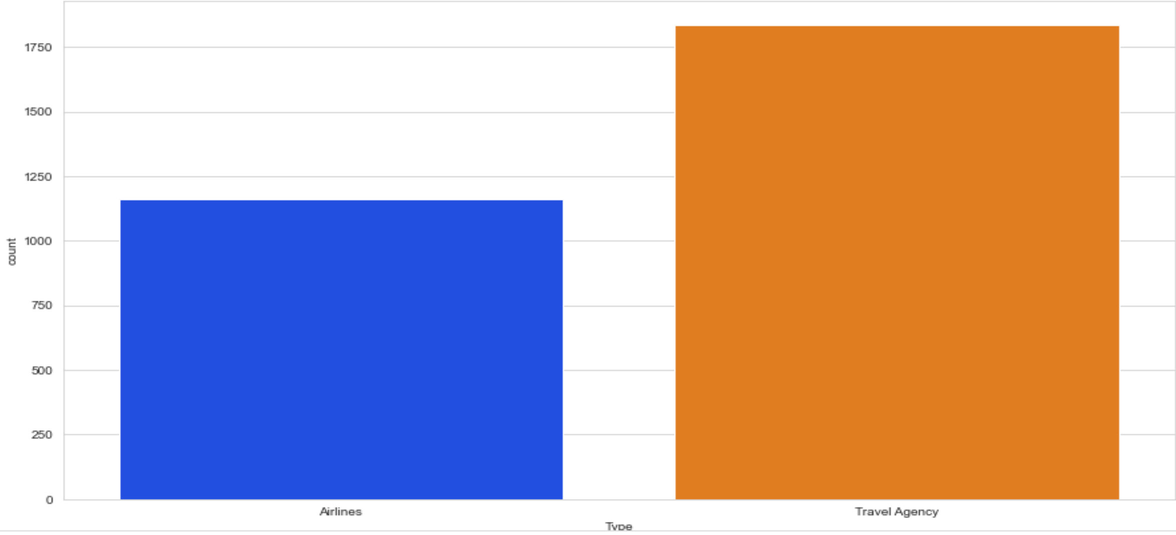


Figure no.51– Count plot of type In the categorical variable

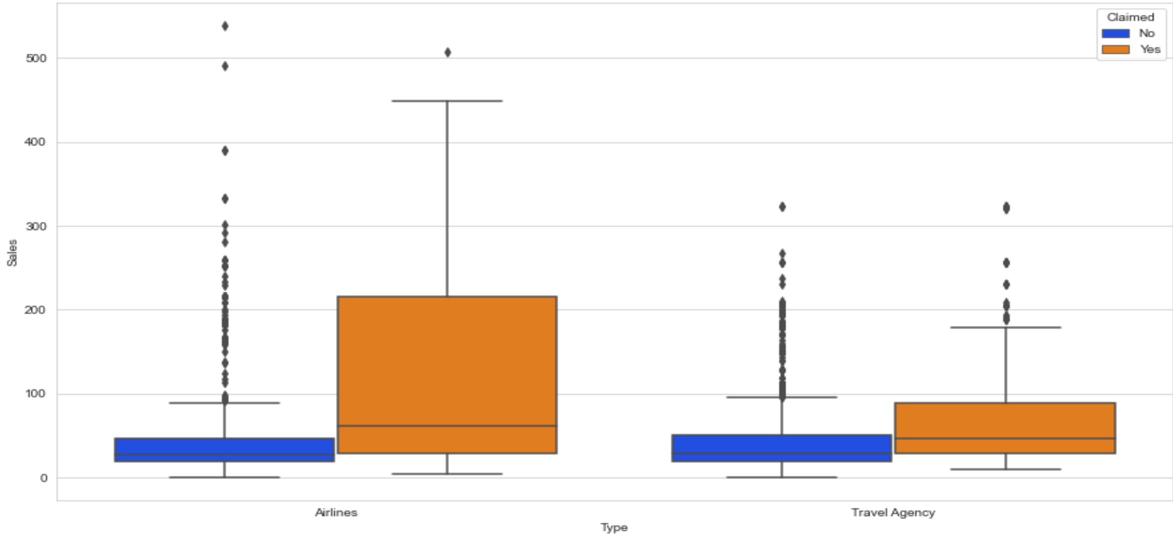


Figure no.52– Box plot of type in the categorical variable

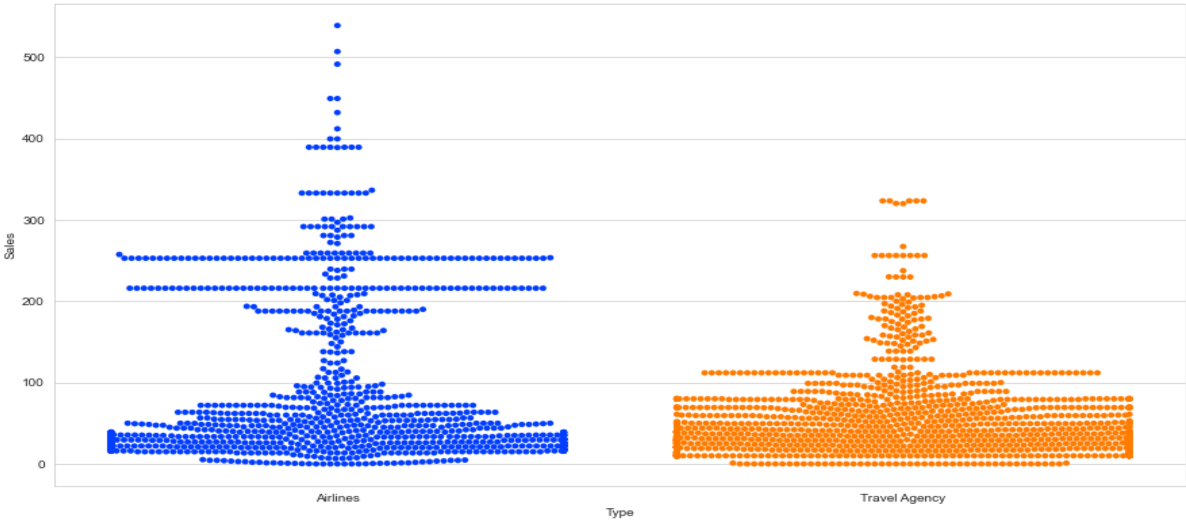


Figure no.53– Swarm plot of type in the categorical variable

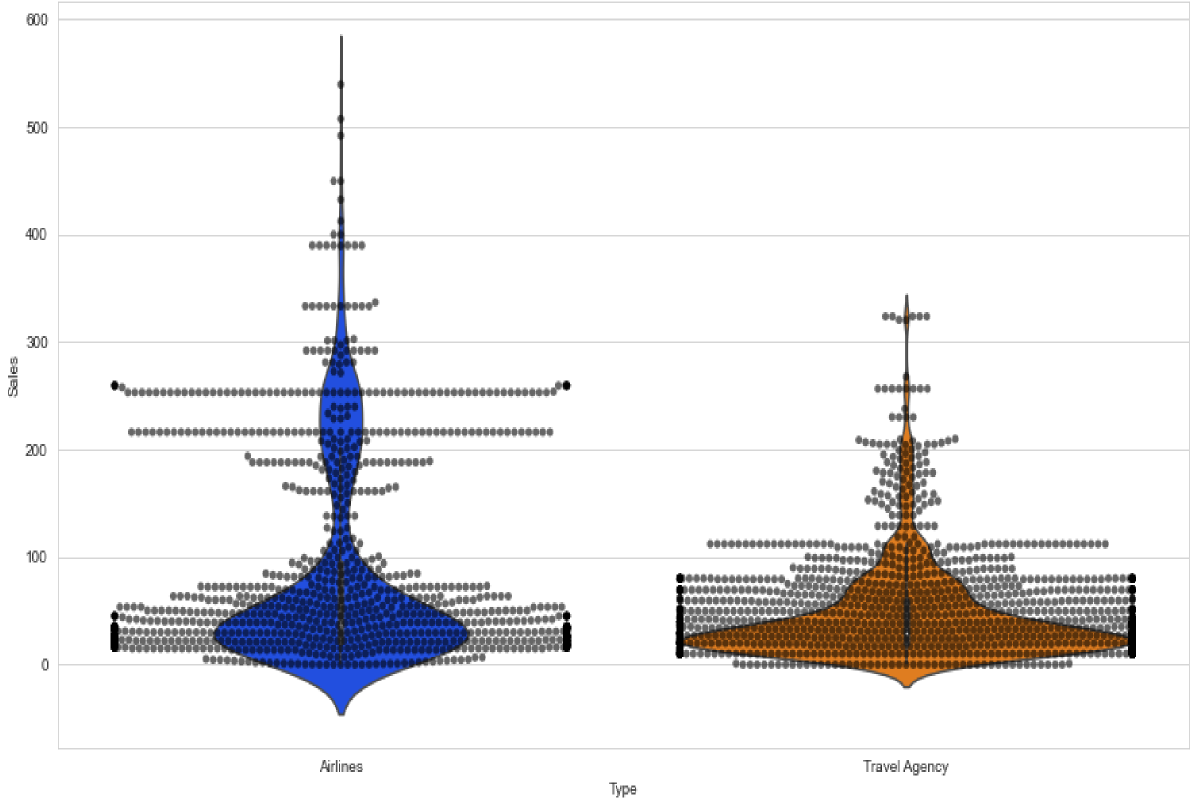


Figure no.54– Combine Violin plot and Swarmplot of type in the categorical variable

Channel

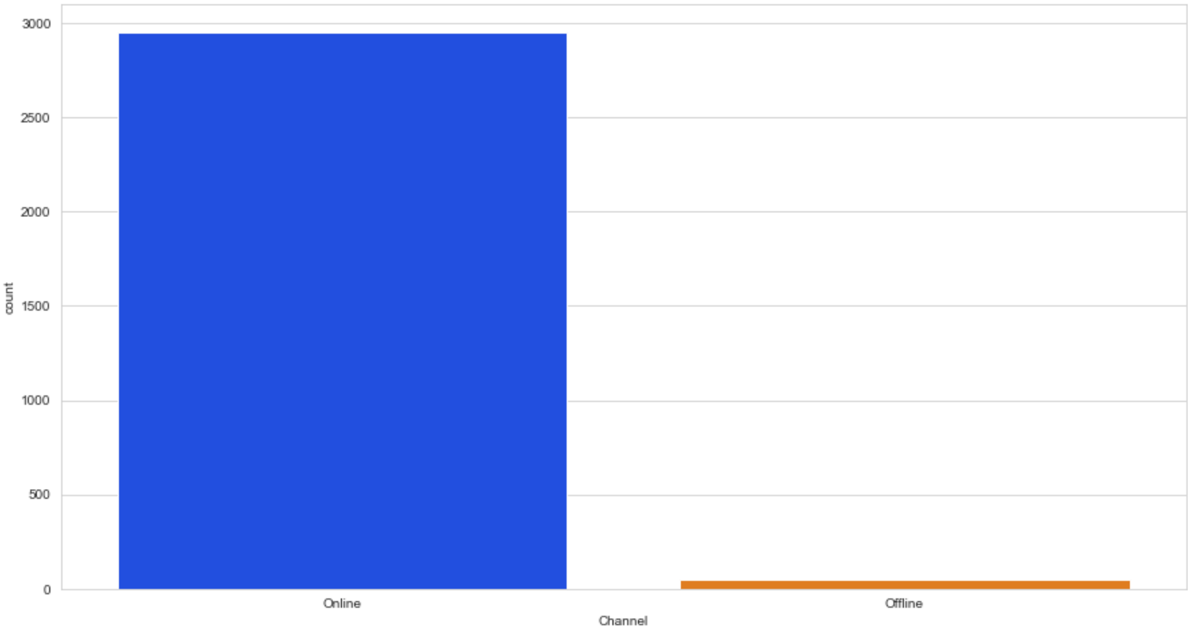


Figure no.55– Count plot of Channel In the categorical variable

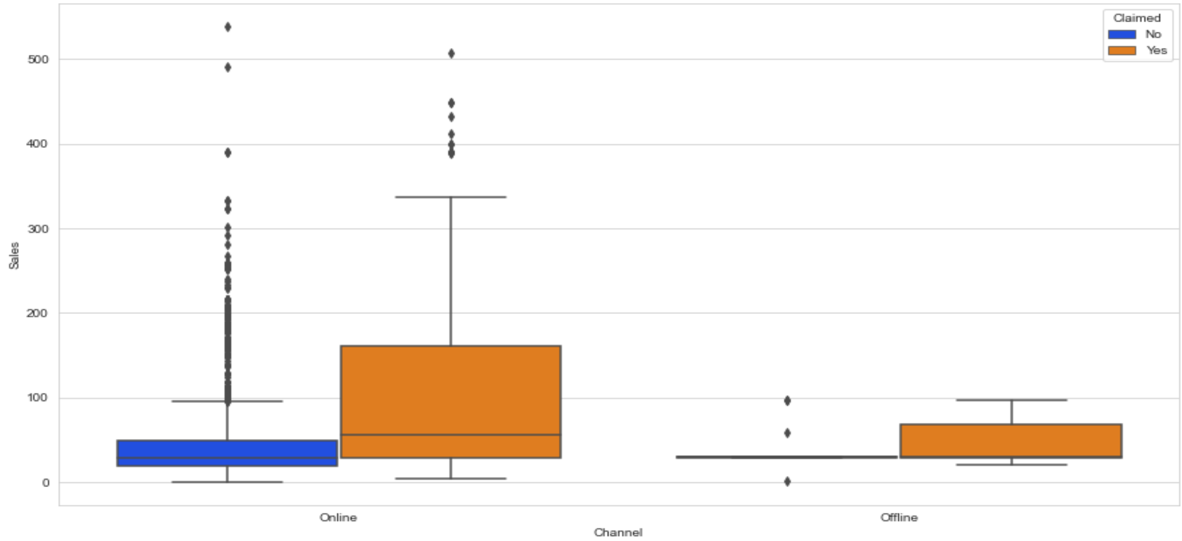


Figure no.56– Box plot of Channel In the categorical variable

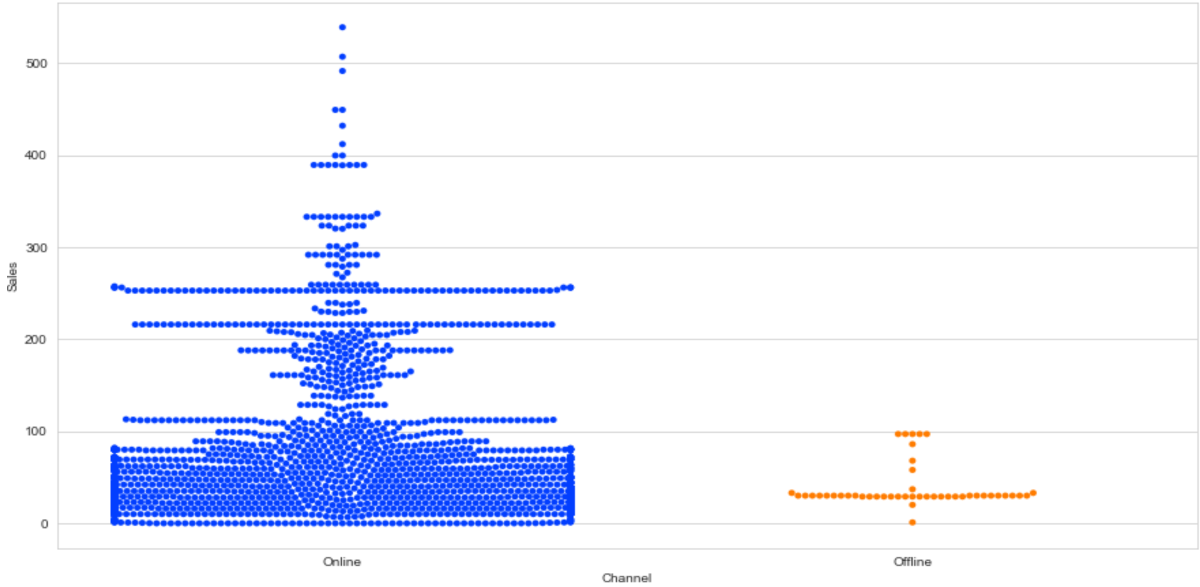


Figure no.57– Box plot of Channel In the categorical variable

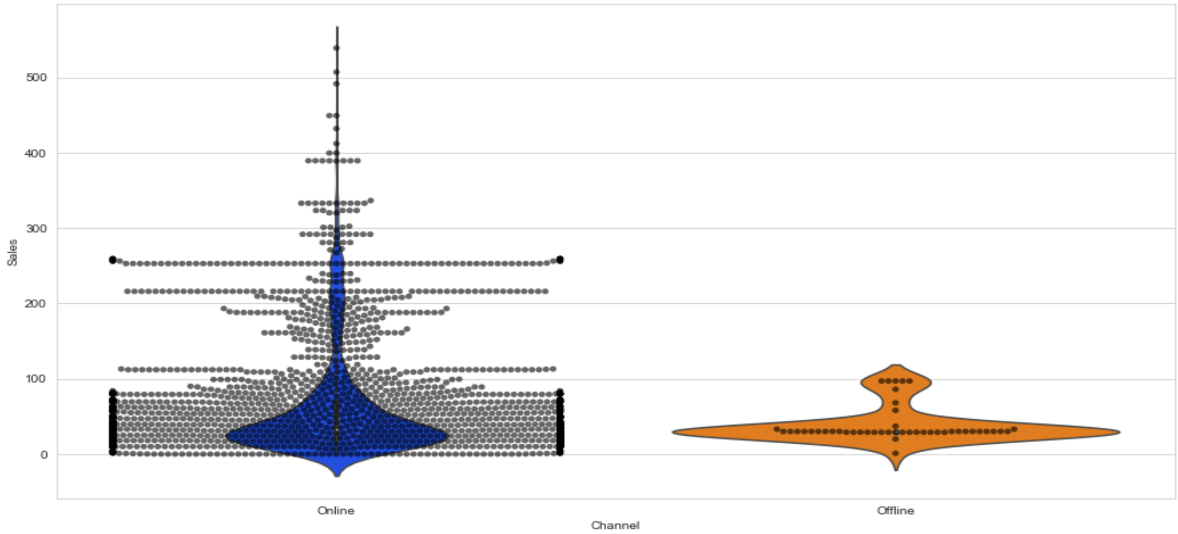


Figure no.58– Combine Violin plot and Swarmplot of Channel in the categorical variable

Product Name

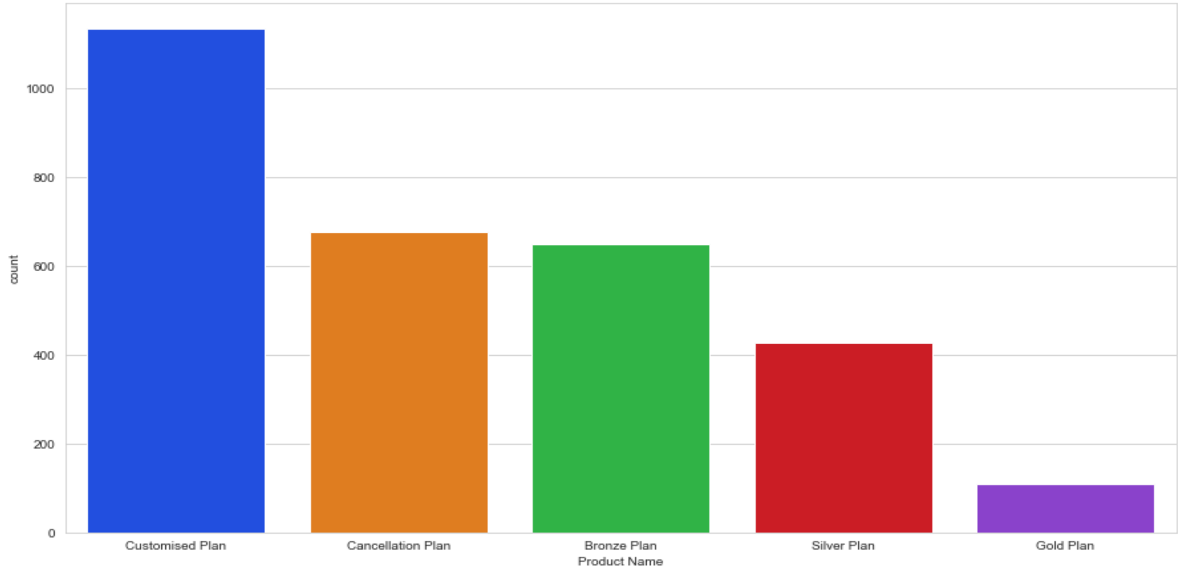


Figure no.59– Count plot of Product Name In the categorical variable

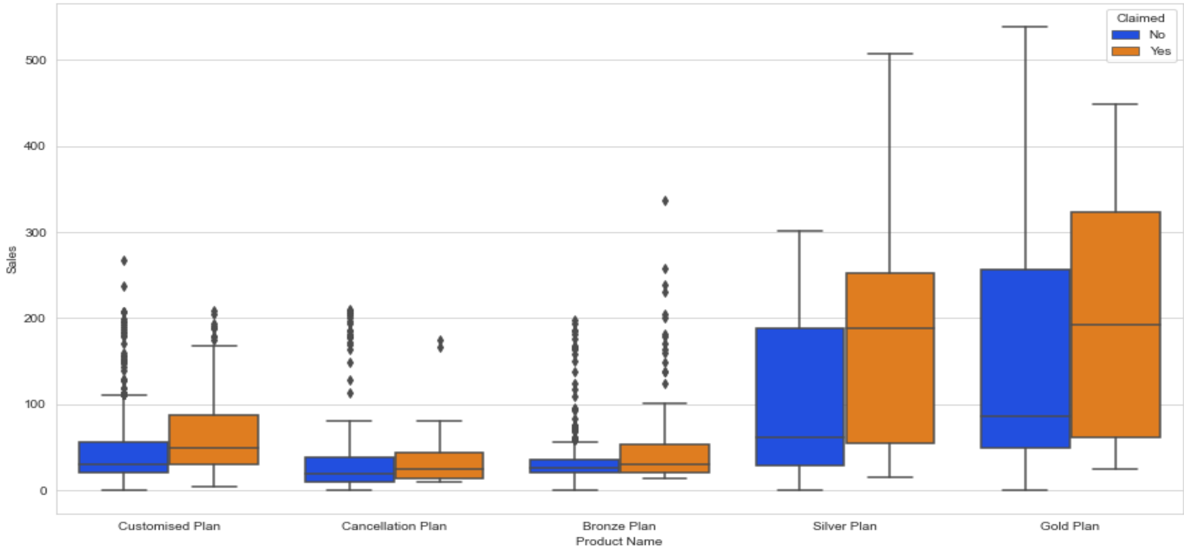


Figure no.60– **Box** plot of Product Name In the categorical variable

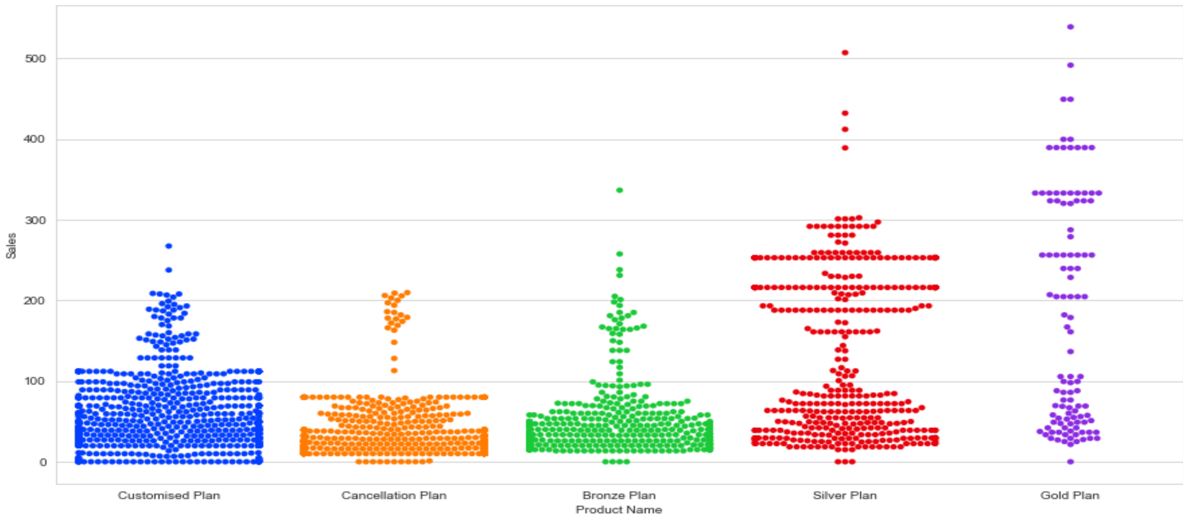


Figure no.57– **Swarm** plot of Product Name In the categorical variable

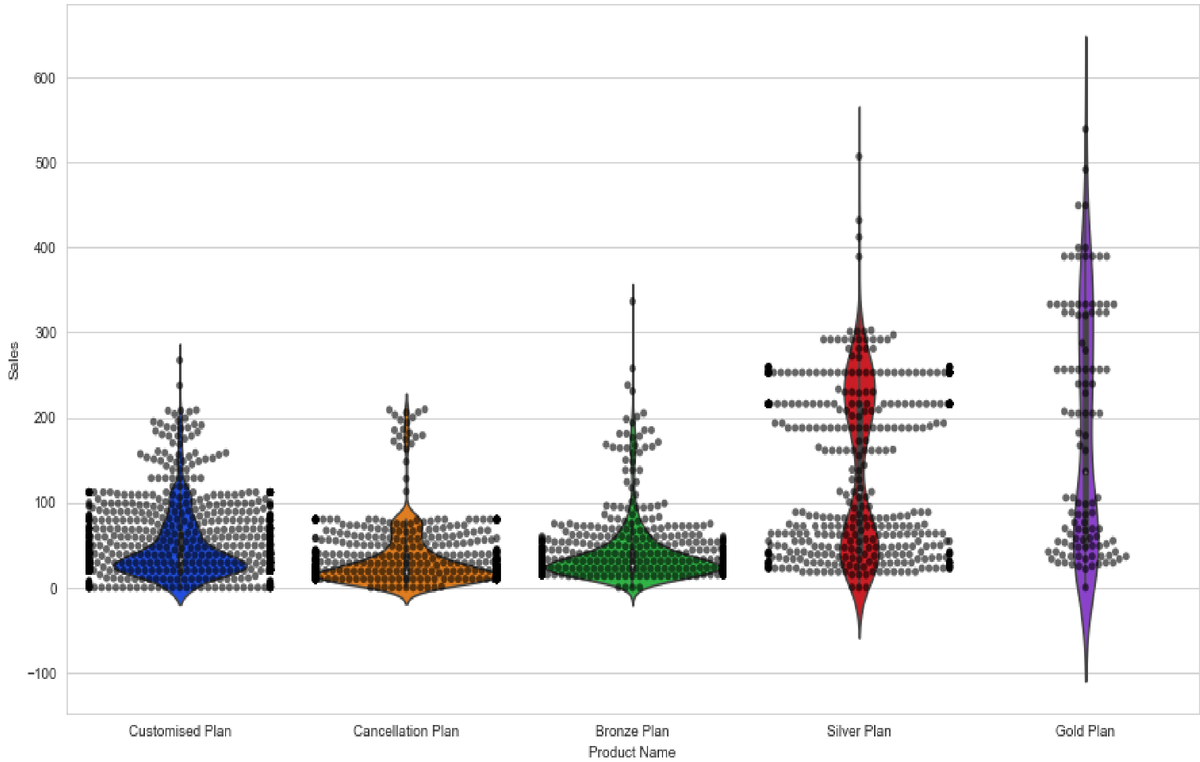


Figure no.61– Combine Violin plot and Swarmplot of Product Name in the categorical variable

Destination

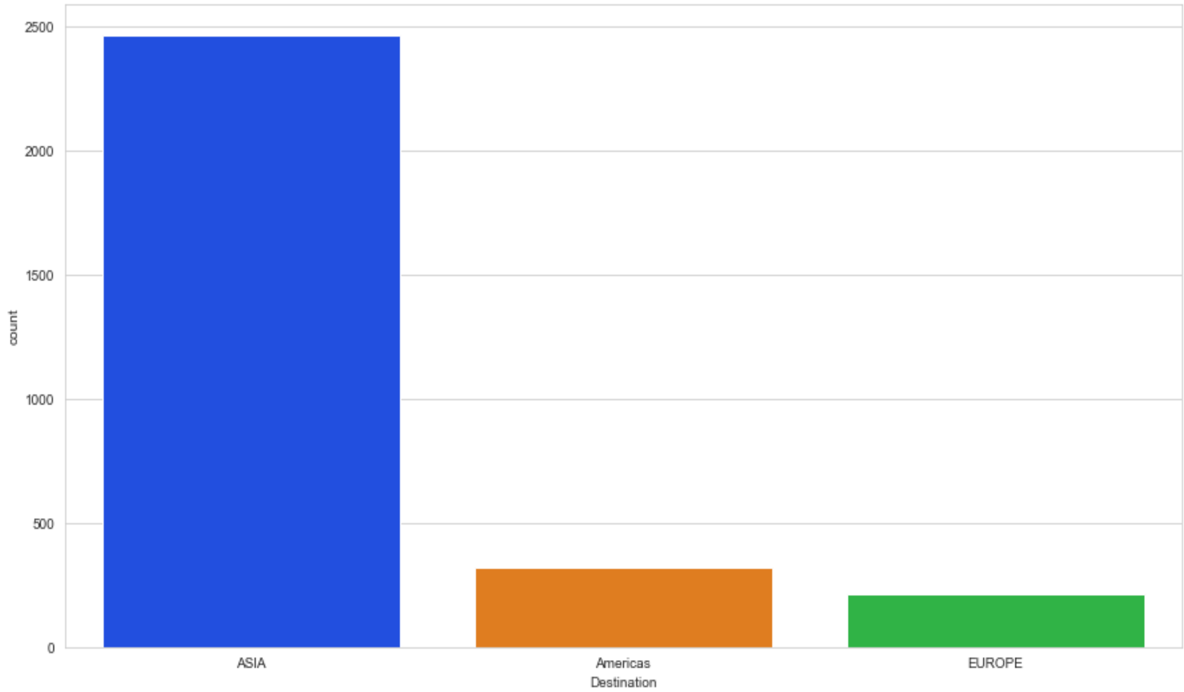


Figure no.62– **Count** plot of Destination In the categorical variable

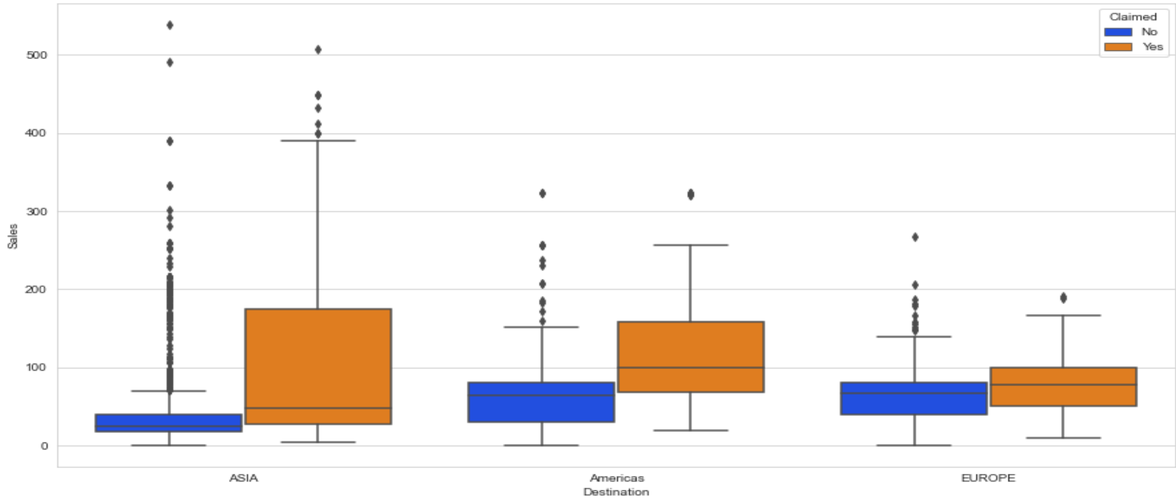


Figure no.63– **Box** plot of Destination In the categorical variable

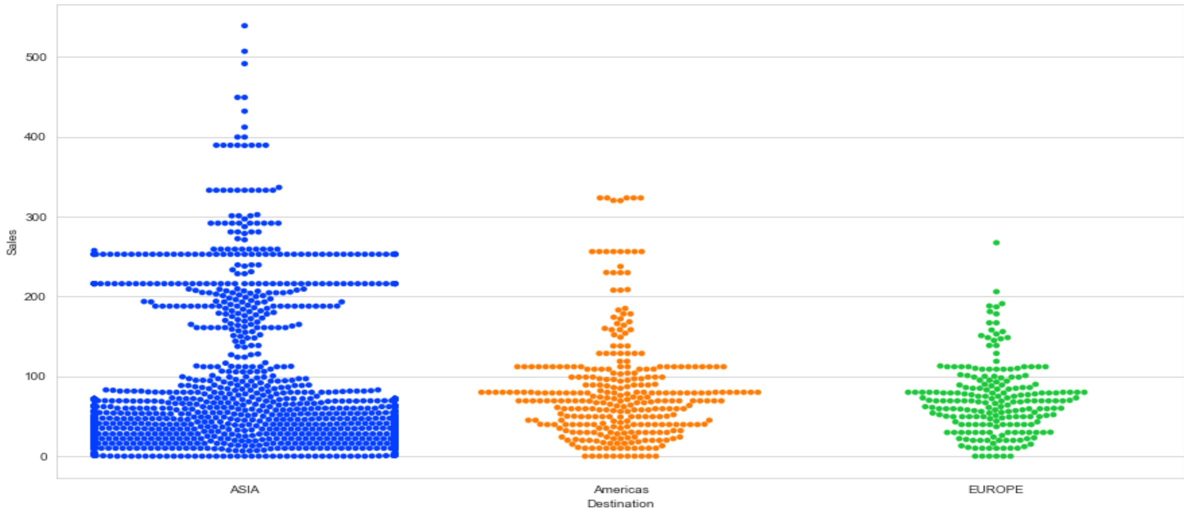


Figure no.64– **Swarm** plot of Destination In the categorical variable

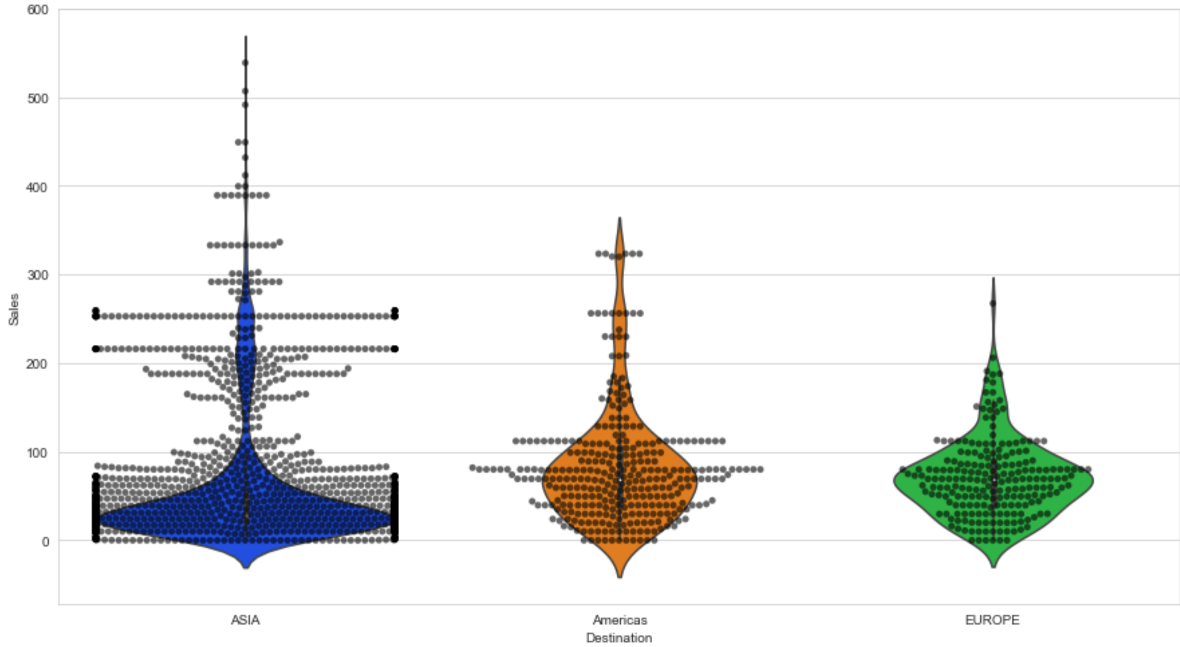


Figure no.65– Combine Violin plot and Swarmplot of Destination in the categorical variable

Checking pairwise distribution of the continuous variables

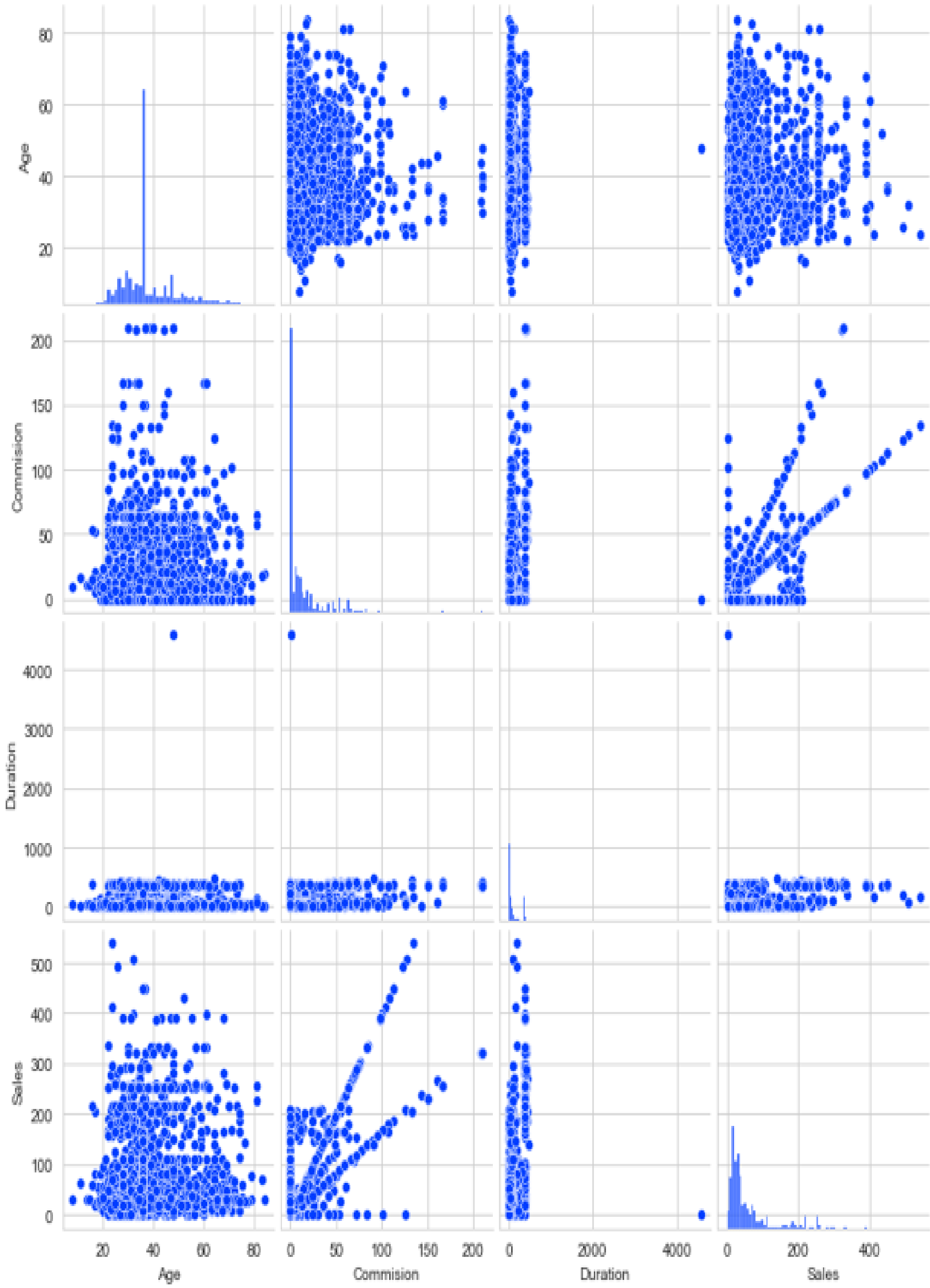


Figure no.66– pairwise distribution of the continuous variables

Checking for Correlations

Constructed heatmap with only continuous variables

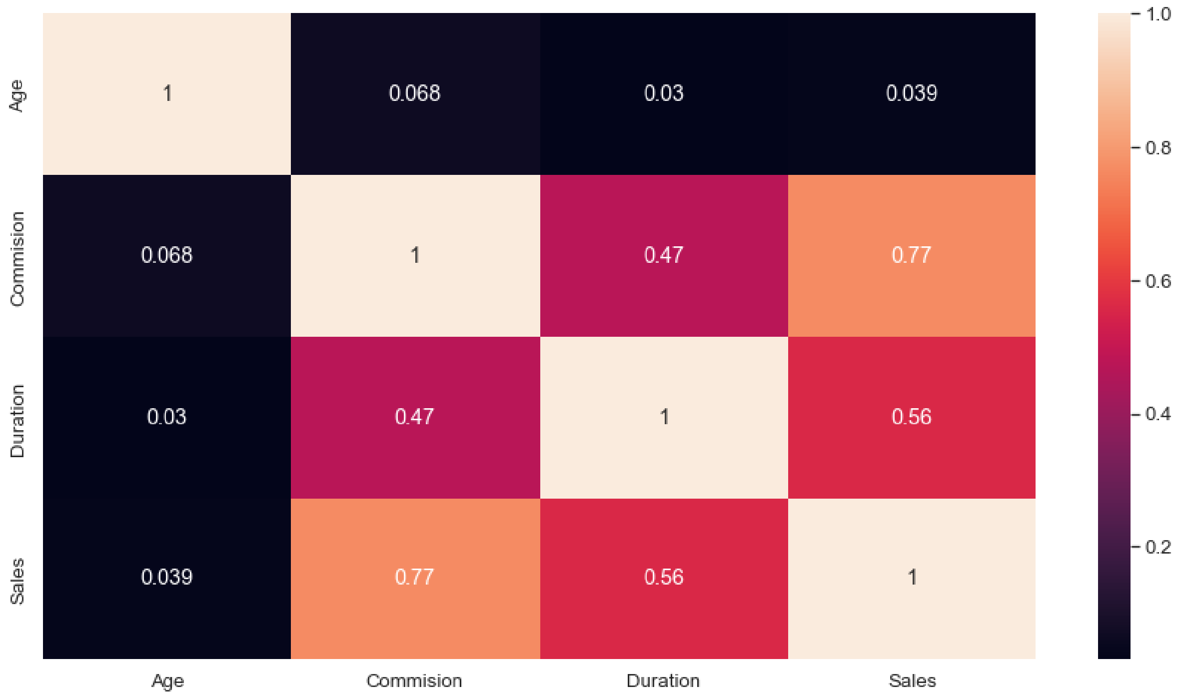


Figure no.67– correlation heatmap with only continuous variables

Converting all objects to categorical codes

```
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', 'Gol
d Plan']
Categories (5, object): ['Bronze Plan', 'Cancellation Plan', 'Customised Pl
an', 'Gold Plan', 'Silver Plan']
[2 1 0 4 3]


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

Checking the info of the converted data set

```
<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
```

Figure no.68– Info of numerical converted data set

	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

Figure no.69– Head of converted data set

Proportion of 1s and 0s

```
0    0.692
1    0.308
Name: Claimed, dtype: float64
```

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

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

Splitting the into X and y

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

Figure no.70– head of split X data

After performed the scaling on the data, we see the below changes as shown in the visual

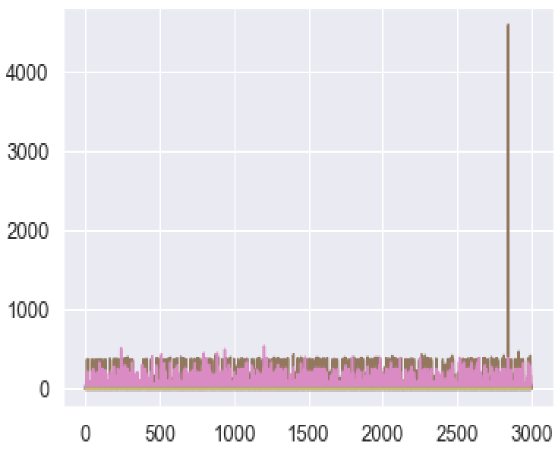


Figure no.71– Prior to scaling After Scaling

The dimensions of the training and test data

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

Building a Decision Tree Classifier

After checking the many param grid values, we see the below best grid values that going for used further.

Variable Importance - DTCL

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

The Predicted Classes and Probs

	0	1
0	0.697947	0.302053
1	0.979452	0.020548
2	0.921171	0.078829
3	0.510417	0.489583
4	0.921171	0.078829

Figure no.72– Predicted classes and probes values of DTCL

Inference of DTCL:

- Random state value used as 1
- Criterion is used as 'gini’
- By doing the minimal and maximum changes in the ‘max_depth’, 'min_samples_leaf' and 'max_samples_split', we are getting the optimum values one certain stage as we finalized as mentioned in the above
- Used DecisionTreeClassifier function for the above,

Building a Random Forest Classifier

After checking the many param grid values, we see the below best grid values that going for used further.

The Predicted Classes and Probs

	0	1
0	0.778010	0.221990
1	0.971910	0.028090
2	0.904401	0.095599
3	0.651398	0.348602
4	0.868406	0.131594

Figure no.73– Predicted classes and probes values of RFCL

Variable Importance via RF

```
                    Imp
Agency_Code    0.276015
Product Name   0.235583
Sales          0.152733
Commision      0.135997
Duration       0.077475
Type           0.071019
Age            0.039503
Destination    0.008971
Channel        0.002705
```

Inference of RFCL:

- Random state value used as 1
- Criterion is used as 'gini’
- By doing the minimal and maximum changes in the ‘max_depth’,’max_features’, 'min_samples_leaf' and 'max_samples_split', we are getting the optimum values one certain stage as we finalized as mentioned in the above
- Used RandomForestClassifier function for the above,

Building a Neural Network Classifier

After checking the many param grid values, we see the below best grid values that going for used further.

Predicted Classes and Probs

	0	1
0	0.822676	0.177324
1	0.933407	0.066593
2	0.918772	0.081228
3	0.688933	0.311067
4	0.913425	0.086575

Figure no.74– Predicted classes and probes values of NNCL

Inference of NNCL:

- Random state value used as 1
- Solver is used as 'adam'
- By doing the minimal and maximum changes in the 'hidden_layer_sizes','max_iter', and 'tol', we are getting the optimum values one certain stage as we finalized as mentioned in the above
- Used MLPClassifier function for the above,

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 - AUC and ROC for the training data

```
AUC: 0.823
[<matplotlib.lines.Line2D at 0x7f7d16dadee0>]
```

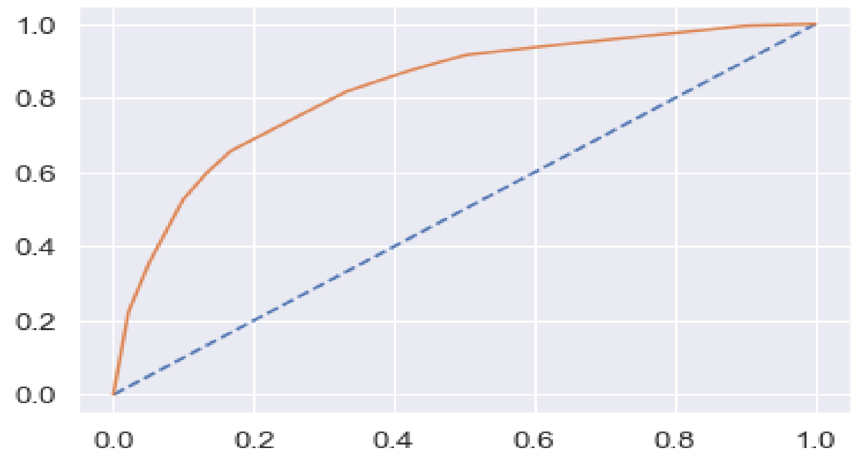


Figure no.75– CART - AUC and ROC for the training data

CART - AUC and ROC for the testing data

```
AUC: 0.801
[<matplotlib.lines.Line2D at 0x7f7d16da0970>]
```

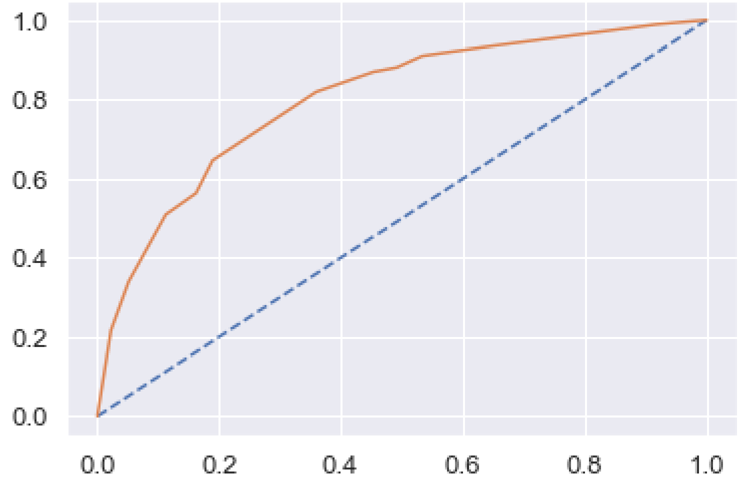


Figure no.76– CART - AUC and ROC for the testing data

CART Confusion Matrix and Classification Report for the training data

	precision	recall	f1-score	support
0	0.81	0.90	0.85	1453
1	0.70	0.53	0.60	647
accuracy			0.79	2100
macro avg	0.76	0.71	0.73	2100
weighted avg	0.78	0.79	0.78	2100

Figure no.77– Confusion matrix and Classification report of training data

CART Confusion Matrix and Classification Report for the testing data

	precision	recall	f1-score	support
0	0.80	0.89	0.84	623
1	0.67	0.51	0.58	277
accuracy			0.77	900
macro avg	0.74	0.70	0.71	900
weighted avg	0.76	0.77	0.76	900

Figure no.78– Confusion matrix and Classification report of testing data

CART Conclusion

Train Data:

- AUC: 82%
- Accuracy: 79%
- Precision: 70%
- f1-Score: 60%

Test Data:

- AUC: 80%
- Accuracy: 77%
- Precision: 80%
- f1-Score: 84%

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

Change is the most important variable for predicting diabetes

RF Model Performance Evaluation on Training data

	precision	recall	f1-score	support
0	0.84	0.89	0.86	1453
1	0.72	0.61	0.66	647
accuracy			0.80	2100
macro avg	0.78	0.75	0.76	2100
weighted avg	0.80	0.80	0.80	2100

Figure no.79– DF Model Classification report of training data

Area under Curve is 0.8563713512840778

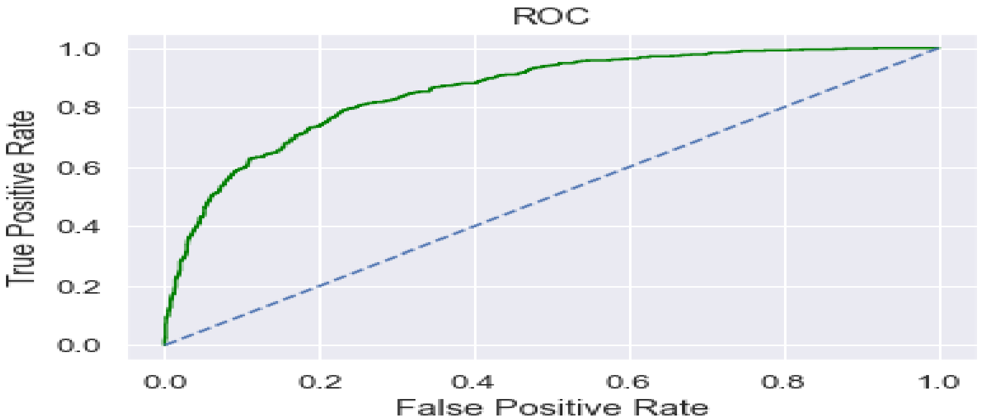


Figure no.80– DF Model ROC visual of training data

RF Model Performance Evaluation on Test data

	precision	recall	f1-score	support
0	0.82	0.88	0.85	623
1	0.68	0.56	0.62	277
accuracy			0.78	900
macro avg	0.75	0.72	0.73	900
weighted avg	0.78	0.78	0.78	900

Figure no.81– DF Model Classification report of testing data

Area under Curve is 0.8181994657271499

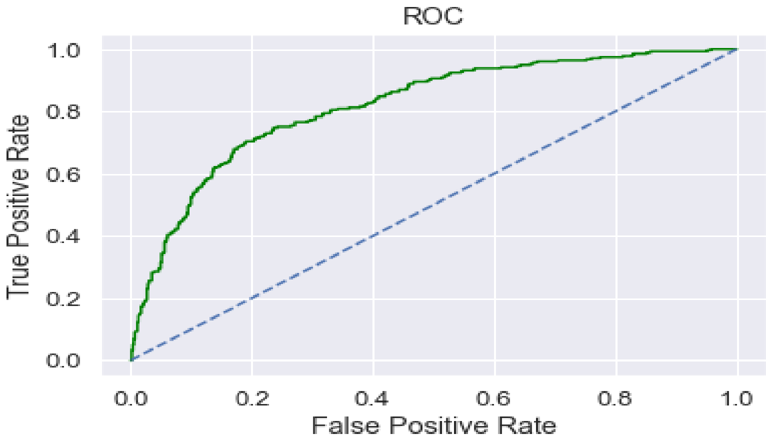


Figure no.82– DF Model ROC visual of testing data

Random Forest Conclusion

Train Data:

- AUC: 86%
- Accuracy: 80%
- Precision: 72%
- f1-Score: 66%

Test Data:

- AUC: 82%
- Accuracy: 78%
- Precision: 68%
- f1-Score: 62

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

Change is again the most important variable for predicting diabetes

NN Model Performance Evaluation on Training data

	precision	recall	f1-score	support
0	0.80	0.89	0.85	1453
1	0.68	0.51	0.59	647
accuracy			0.78	2100
macro avg	0.74	0.70	0.72	2100
weighted avg	0.77	0.78	0.77	2100

Figure no.83– NN Model Classification report of training data

Area under Curve is 0.8166831721609928

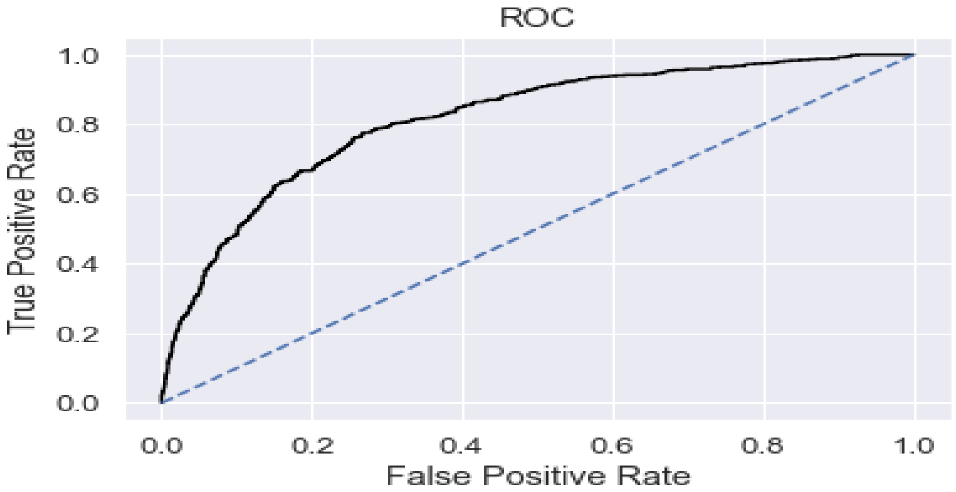


Figure no.84–NN Model ROC visual of training data

NN Model Performance Evaluation on Test data

	precision	recall	f1-score	support
0	0.80	0.89	0.84	623
1	0.67	0.50	0.57	277
accuracy			0.77	900
macro avg	0.73	0.69	0.71	900
weighted avg	0.76	0.77	0.76	900

Figure no.85– NN Model Classification report of testing data

Area under Curve is 0.8044225275393896

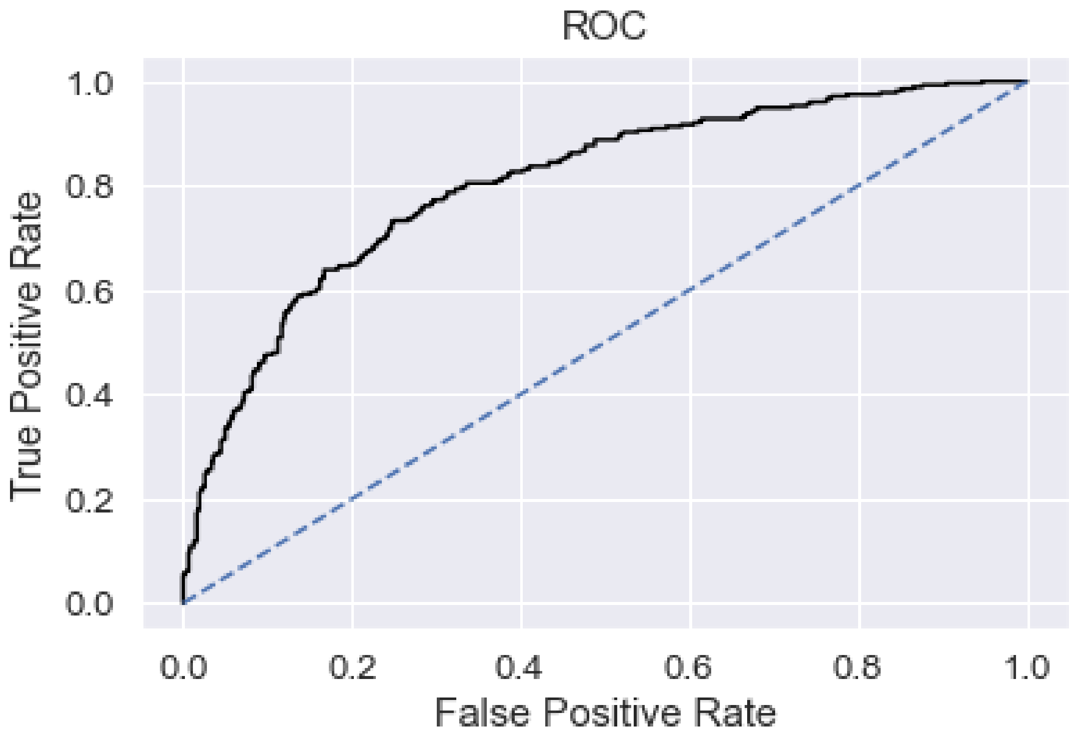


Figure no.86–NN Model ROC visual of testing data

Neural Network Conclusion

Train Data:

- AUC: 82%
- Accuracy: 78%
- Precision: 68%
- f1-Score: 59

Test Data:

- AUC: 80%
- Accuracy: 77%
- Precision: 67%
- f1-Score: 57%

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

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

Comparison of the performance metrics from the 3 models

	CART Train	CART Test	Random Forest Train	Random Forest Test	Neural Network Train	Neural Network Test
Accuracy	0.79	0.77	0.80	0.78	0.78	0.77
AUC	0.82	0.80	0.86	0.82	0.82	0.80
Recall	0.53	0.51	0.61	0.56	0.51	0.50
Precision	0.70	0.67	0.72	0.68	0.68	0.67
F1 Score	0.60	0.58	0.66	0.62	0.59	0.57

Figure no.87– Comparison chart of the performance metrics from the 3 models

ROC Curve for the 3 models on the Training data

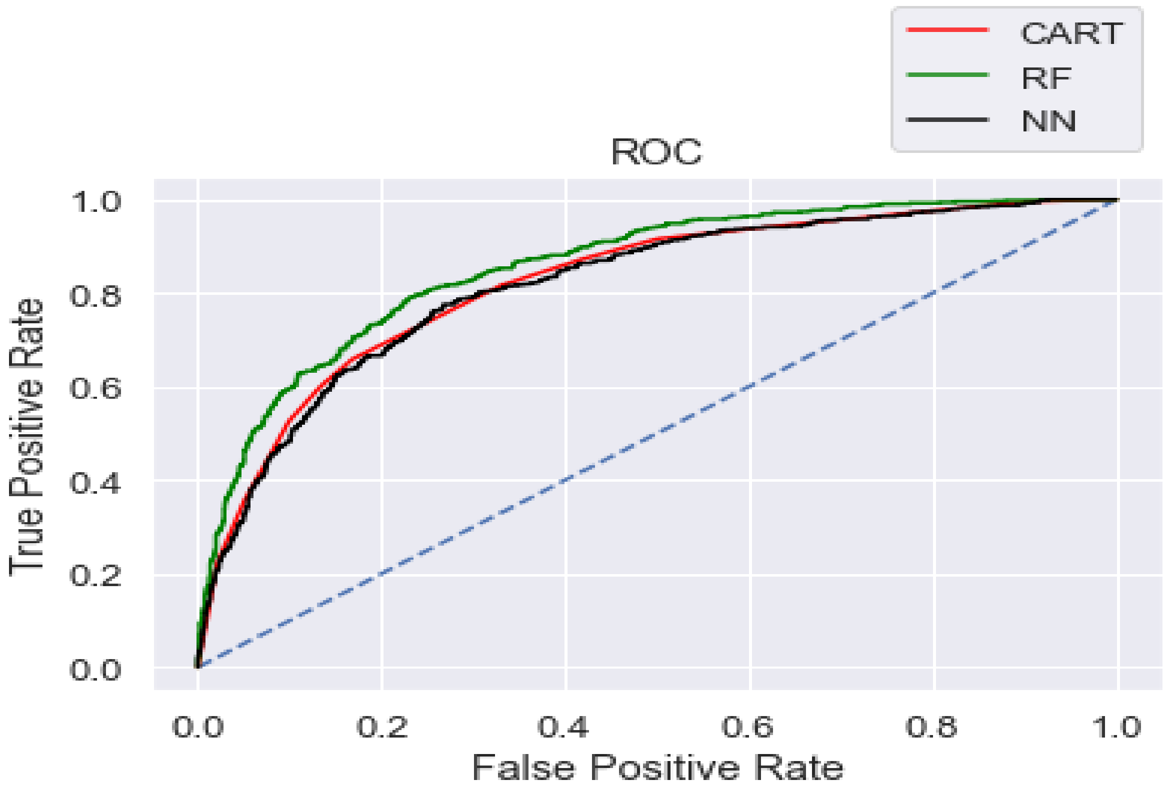


Figure no.88– ROC Curve visuals for the 3 models on the Training data

ROC Curve for the 3 models on the Test data

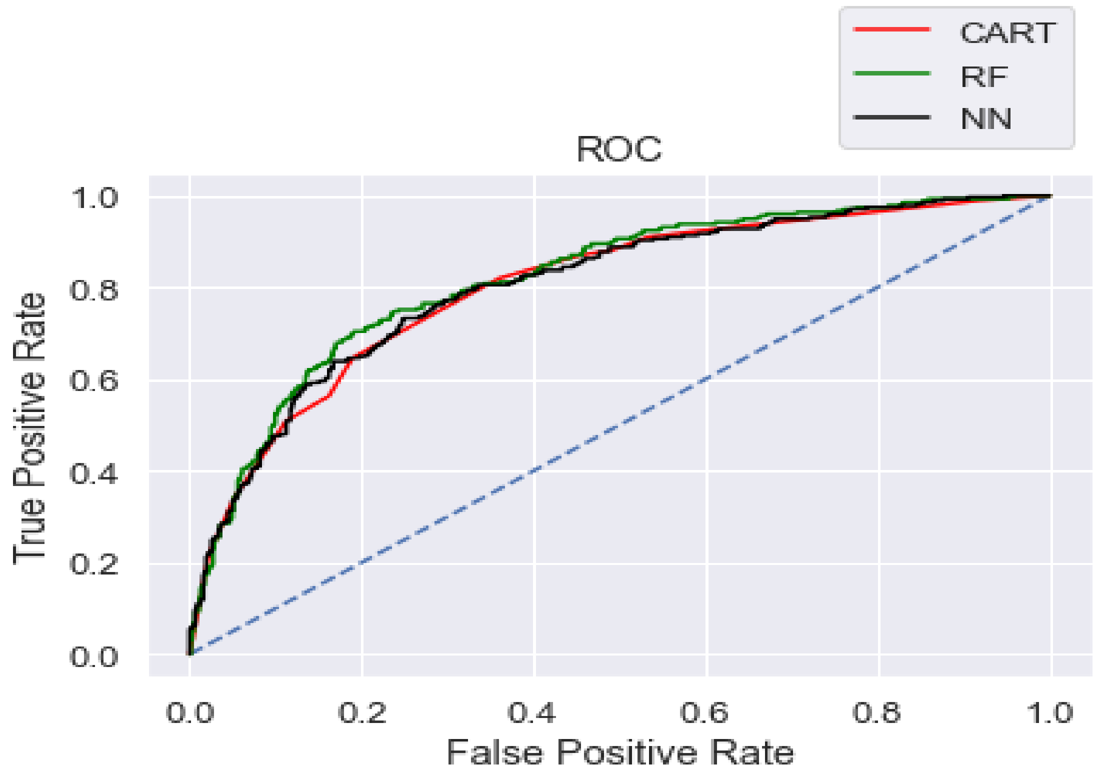


Figure no.89– ROC Curve visuals for the 3 models on the Test data

CONCLUSION:

I am selecting the 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

I strongly recommended we collect more real time unstructured data and past data if possible.

This is understood by looking at the insurance data by drawing relations between different variables such as day of the incident, time, age group, and associating it with other external information such as location, behaviour patterns, weather information, airline/vehicle types, etc.

• Streamlining online experiences benefitted customers, leading to an increase in conversions, which subsequently raised profits.

• As per the data 90% of insurance is done by online channel.

• Other interesting fact, is almost all the offline business has a claimed associated, need to find why?

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

• Also based on the model we are getting 80%accuracy, so we need customer books airline tickets or plans, cross sell the insurance based on the claim data pattern.

• Other interesting fact is more sales happen via Agency than Airlines and the trend shows the claim are processed more at Airline. So we may need to deep dive into the process to understand the workflow and why?

Key performance indicators (KPI) The KPI's of insurance claims are:

• Reduce claims cycle time

• Increase customer satisfaction

• Combat fraud

• Optimize claims recovery

• Reduce claim handling costs Insights gained from data and AI-powered analytics could expand the boundaries of insurability, extend existing products, and give rise to new risk transfer solutions in areas like a non-damage business interruption and reputational damage.

*****END OF PROBLEM2****

Used Library details for this project:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set_style('whitegrid')
sns.set_palette('bright')
from warnings import filterwarnings
filterwarnings('ignore')
from scipy.stats import norm
from sklearn.preprocessing import StandardScaler
from scipy.cluster.hierarchy import dendrogram, linkage
from scipy.cluster.hierarchy import fcluster
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_samples, silhouette_score
```