2021

Predictive Analysis Project

Prajoth

Great learning

Table of Contents

List of Tables:

List of Figures:

Problem 1:Linear Regression:

Problem Statement:

You are hired by a company Gem Stones co ltd, which is a cubic zirconia manufacturer. You are provided with the dataset containing the prices and other attributes of almost 27,000 cubic zirconia (which is an inexpensive diamond alternative with many of the same qualities as a diamond). The company is earning different profits on different prize slots. You have to help the company in predicting the price for the stone on the bases of the details given in the dataset so it can distinguish between higher profitable stones and lower profitable stones so as to have better profit share. Also, provide them with the best 5 attributes that are most important.

Data Description:

Variable Name	Description
Carat	Carat weight of the cubic zirconia.
Cut	Describe the cut quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal.
Color	Colour of the cubic zirconia.With D being the worst and J the best.
Clarity	cubic zirconia Clarity refers to the absence of the Inclusions and Blemishes. (In order from Worst to Best) IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1
Depth	The Height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter.
Table	The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter.
Price	the Price of the cubic zirconia.
X	Length of the cubic zirconia in mm.
Y	Width of the cubic zirconia in mm.
Z	Height of the cubic zirconia in mm.

1.1 Read the data and do exploratory data analysis. Describe the data briefly. (Check the null values, Data types, shape, EDA, duplicate values). Perform Univariate and Bivariate Analysis

- Importing all necessary library packages to build a linear regression model. Read the dataset to build a model.

Reading the Dataset:

	Unnamed: 0	carat	cut	color	clarity	depth	table	x	y	z	price
0	1	0.30	Ideal	E	SI1	62.1	58.0	4.27	4.29	2.66	499
1	2	0.33	Premium	G	IF	60.8	58.0	4.42	4.46	2.70	984
2	3	0.90	Very Good	E	VVS2	62.2	60.0	6.04	6.12	3.78	6289
3	4	0.42	Ideal	F	VS1	61.6	56.0	4.82	4.80	2.96	1082
4	5	0.31	Ideal	F	VVS1	60.4	59.0	4.35	4.43	2.65	779

Table 1: Reading the Data

Inferences:

- Dataset has 26967 rows and 11 columns.
- Dataset has both categorical and continuous data. For categorical data we have cut, colour and clarity.
- For continuous data we have carat, depth, table, x. y, z and price. Price will be target variable.
- Quality is increasing order Fair, Good, Very Good, Premium, Ideal. Colour of the cubic zirconia. With D being the worst and J the best. In order from Worst to Best IF, VVS1, VVS2, VS1, VS2, Sl1, Sl2, l1.

Summary of Dataset:

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Unnamed: 0	26967	NaN	NaN	NaN	13484	7784.85	1	6742.5	13484	20225.5	26967
carat	26967	NaN	NaN	NaN	0.798375	0.477745	0.2	0.4	0.7	1.05	4.5
cut	26967	5	Ideal	10816	NaN	NaN	NaN	NaN	NaN	NaN	NaN
color	26967	7	G	5661	NaN	NaN	NaN	NaN	NaN	NaN	NaN
clarity	26967	8	SI1	6571	NaN	NaN	NaN	NaN	NaN	NaN	NaN
depth	26270	NaN	NaN	NaN	61.7451	1.41286	50.8	61	61.8	62.5	73.6
table	26967	NaN	NaN	NaN	57.4561	2.23207	49	56	57	59	79
x	26967	NaN	NaN	NaN	5.72985	1.12852	0	4.71	5.69	6.55	10.23
y	26967	NaN	NaN	NaN	5.73357	1.16606	0	4.71	5.71	6.54	58.9
z	26967	NaN	NaN	NaN	3.53806	0.720624	0	2.9	3.52	4.04	31.8
price	26967	NaN	NaN	NaN	3939.52	4024.86	326	945	2375	5360	18818

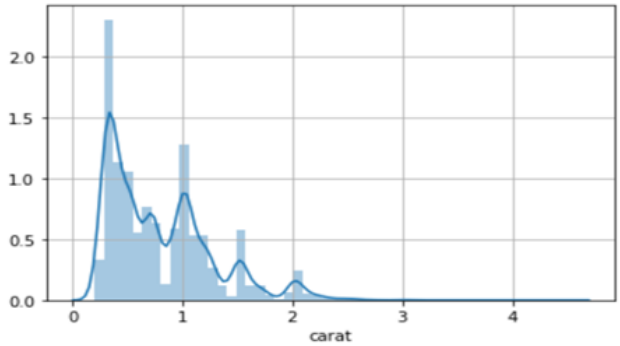
Table 2: Summary of Data

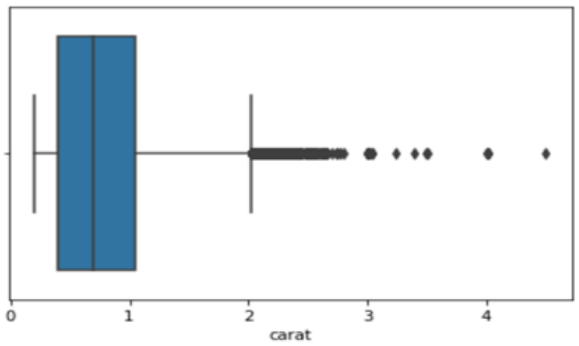
Inferences:

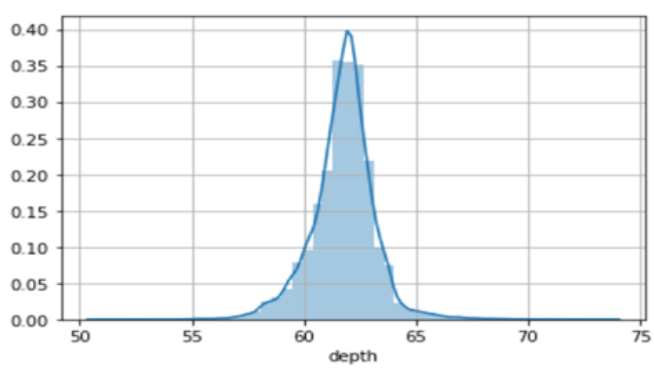
- Dataset has 26967 rows and 11 cloumns.
- Unnamed: 0 column has no information so we can remove this variable before modelling.
- Only depth variable as null values 697 which is 2.58% in data in depth.
- There is no duplicates in dataset.
- In cut variable, Ideal is more preferred which is 10816. In color variable, Ideal is more preferred which is 5658. In clarity variable, Ideal is more preferred which is 6570.

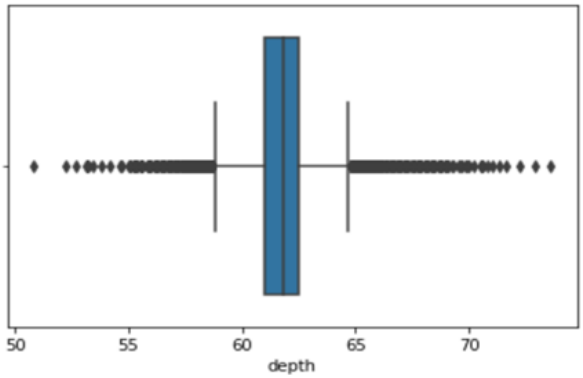
Univariate Analysis:

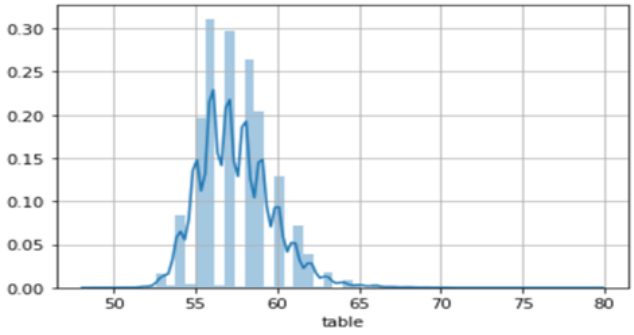
Continuous Variables:

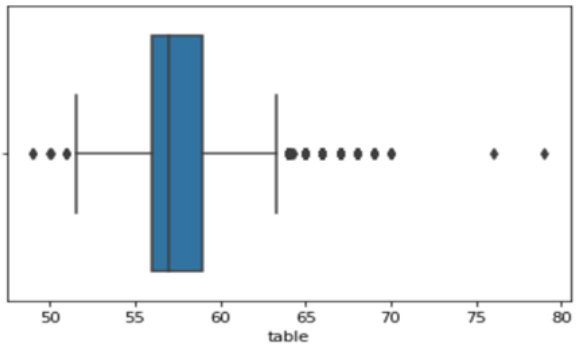


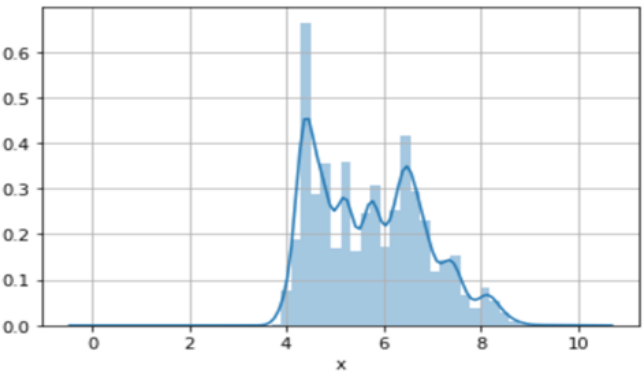


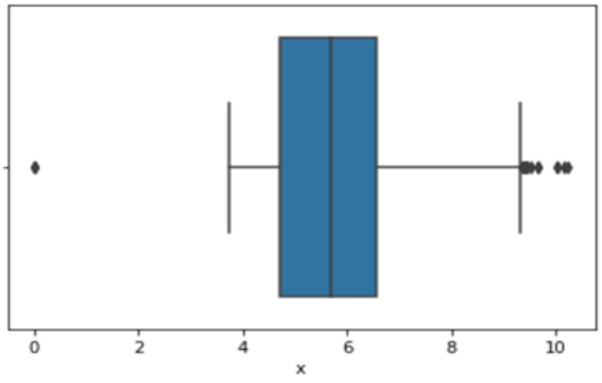

0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
50
55
60
65
70
75
depth

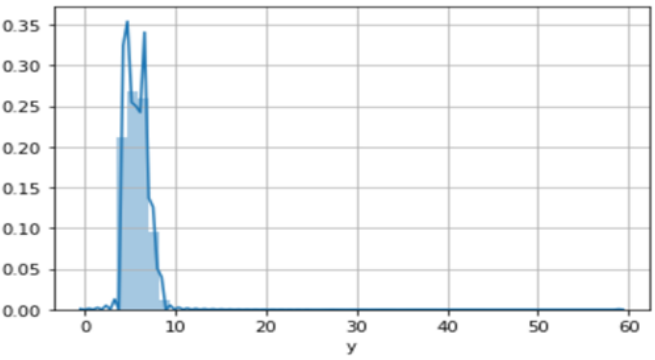

50
55
60
65
70
depth

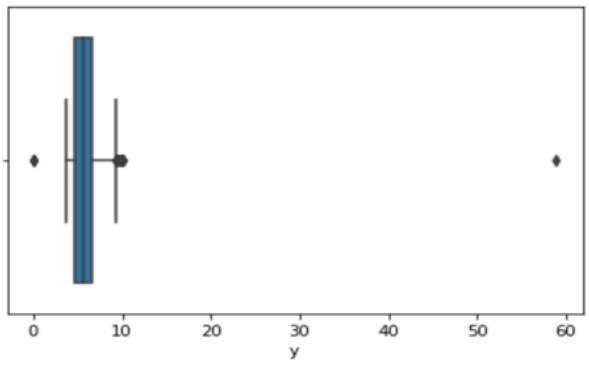

0.30
0.25
0.20
0.15
0.10
0.05
0.00
50
55
60
65
70
75
80
table


50
55
60
65
70
75
80
table


0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
2
4
6
8
10
x


0
2
4
6
8
10
x


0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
10
20
30
40
50
60
y


0
10
20
30
40
50
60
y

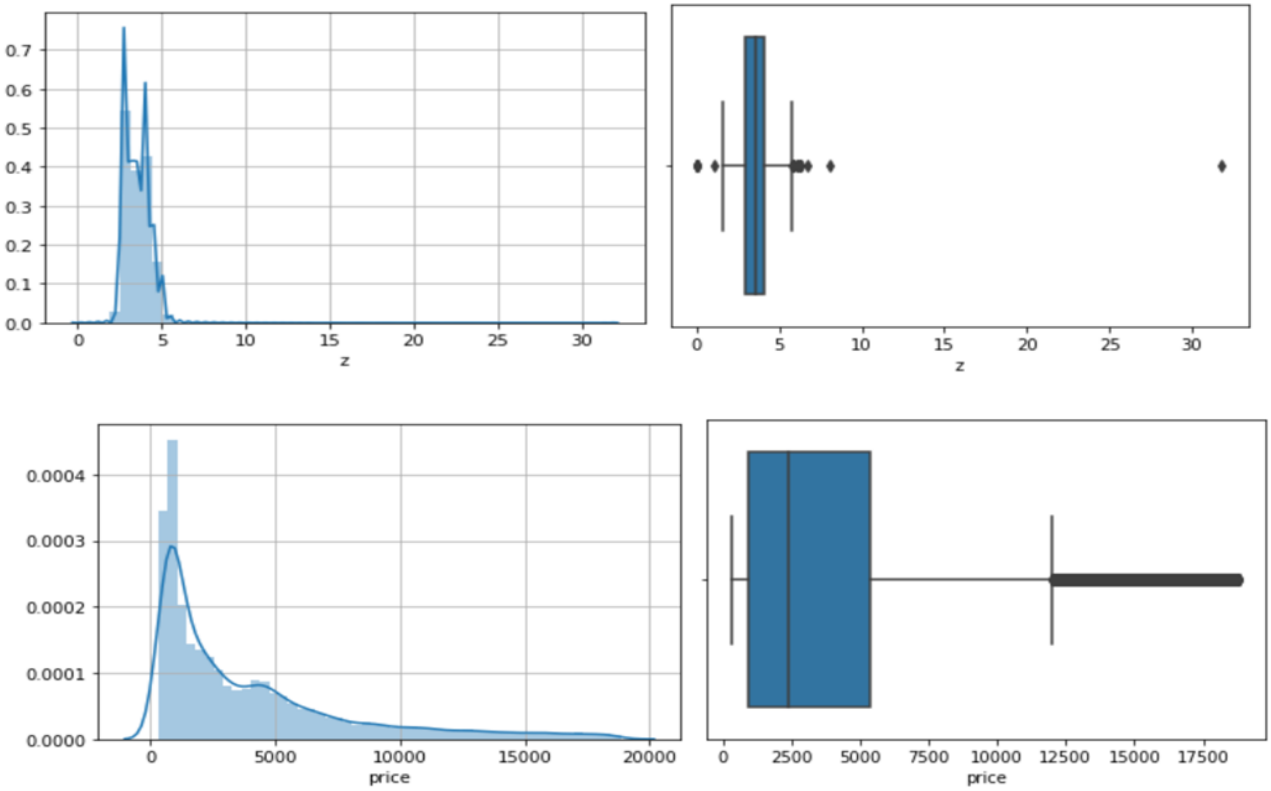


Figure 1:Univariate Analysis with Continuous Variable

Inference:

- Variable like carat,table,X(Length of the cubic zirconia in mm.) ,Y(Width of the cubic zirconia in mm),Z(Height of the cubic zirconia in mm) seems to be positively skewed.
- All variables as outliers.
- X,Y,Z has value 0 as outliers. So we need to remove zeros while data preprocessing. Majority of values of carat ranges from 0 to 1.
- The depth ranges from 55 to 65 which is normally distributed. Maximum distribution in table from 55 to 65.
- Length(X) ranges from 4 to 8,Width(Y) ranges fro 0 to 10,Height (Z) ranges from 0 to 5. The price distribution is from rs 100 to 8000.

Categorical Variable:

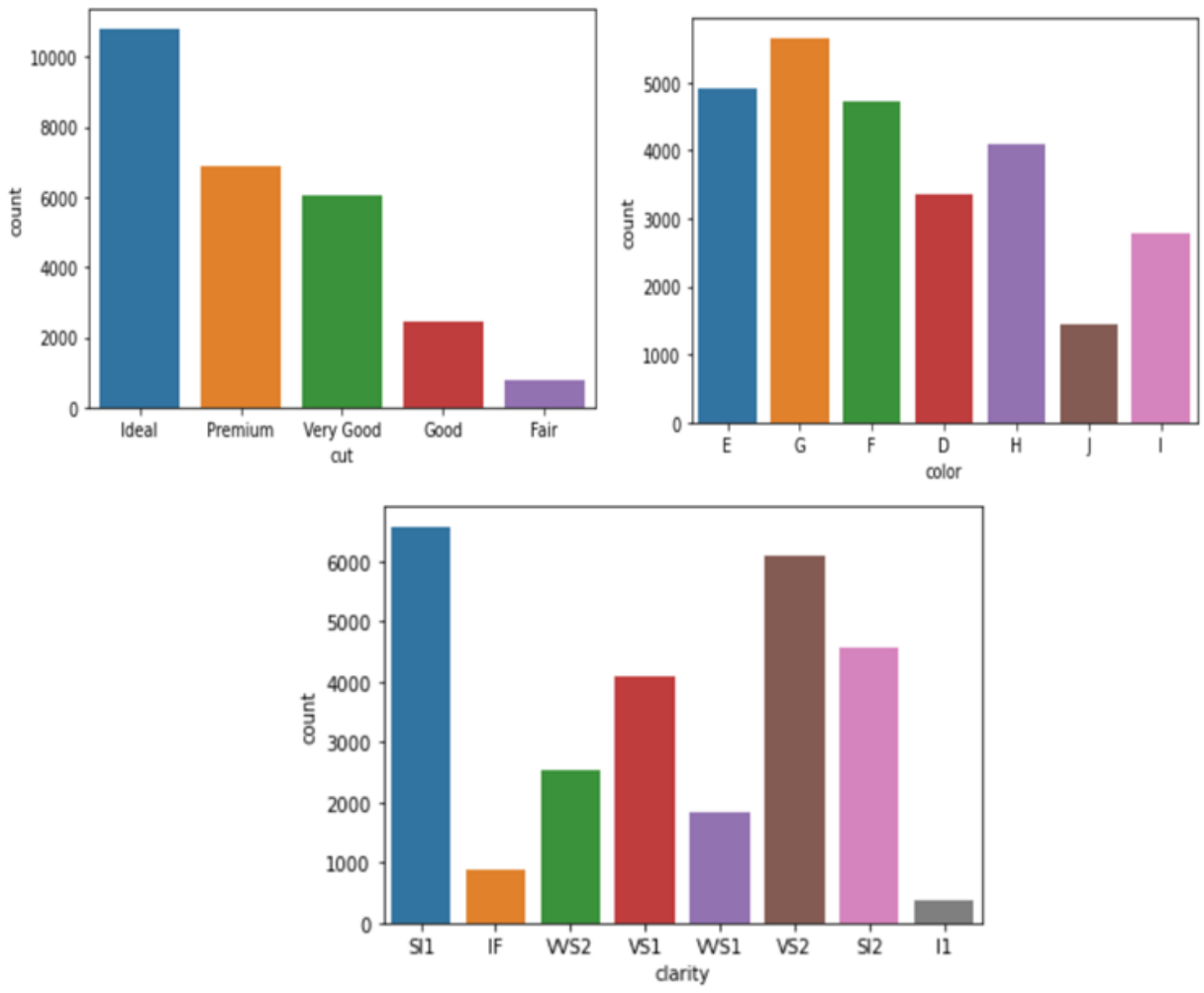


Figure 2: Univariate Analysis with Categorical Variabl

Inferences:

- Ideal cut is most preferred cut for diamond.

- We have 7 colours in the data, The G seems to be the preferred colour.J is low among other colours.
- Sl1 is more prefered clarity for diamond.L1 is less prefered clarity for diamond.

Multivariate Analysis:

Pair Plot with Continuous Variable:

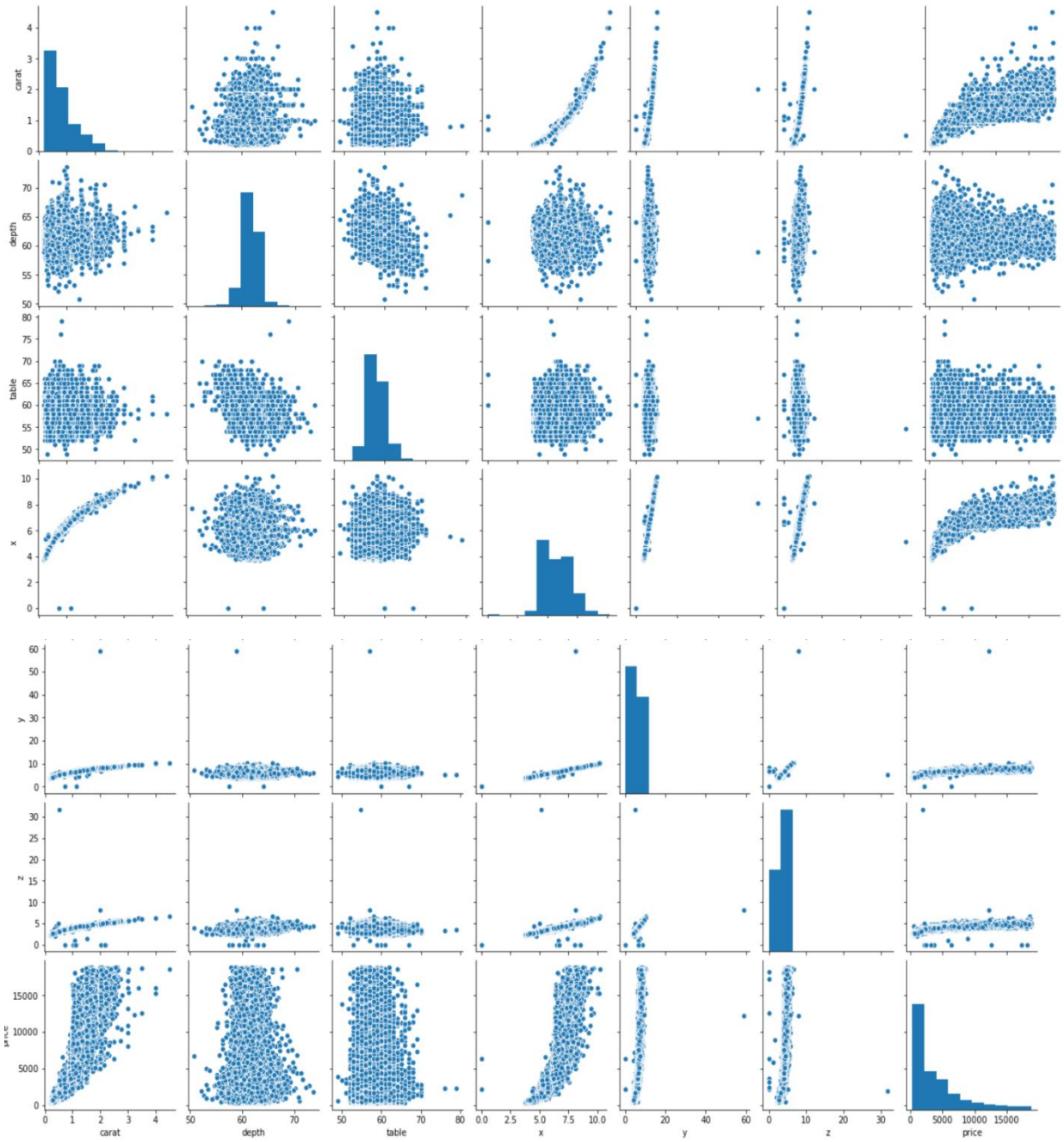


Figure 3: Pair Plot with Continuous Variable

Heat map:

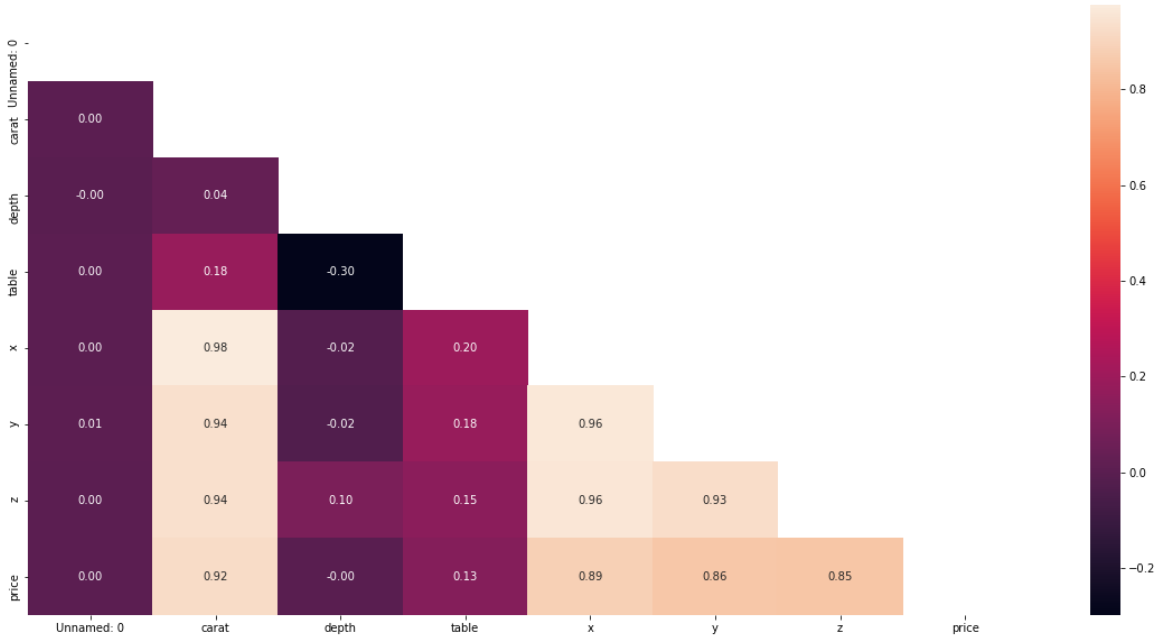


Figure 4: Correlation map with Continuous Variable

Inference:

- Correlation matrix clearly shows there is multicollinearity in the data.
- Carat has highly correlated with price(0.92),X(0.98),Y(0.94),Z(0.92) which is positive. X has good relation with Y,Z and price.
- Y has good correlation with Price and Z .Z has good relation with price.
- Depth has bad correlation with other variables. which is -0.0 with variable price and -0.02 with variable X & Y.

Bivariate Analysis:

Cut Vs Price:

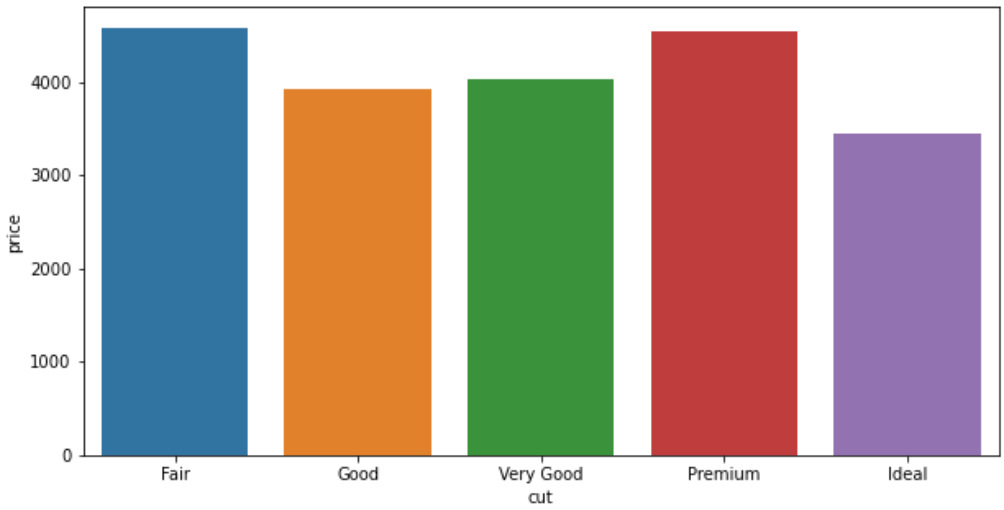


Figure 5: Cut Vs Price

Inference:

- Ideal is most preferred cut for diamond because price is low compared to other cuts. Price of fair cut is high so customer shows less interest to buy.
- Slight change in price in both Good and Very Good. Premium price is high but customer is preferred with this cut.

Color Vs Price:

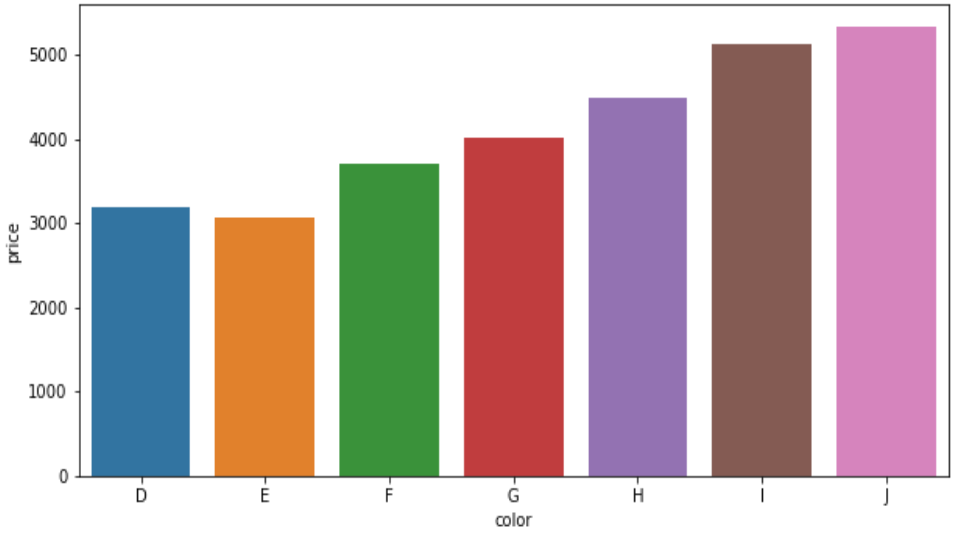


Figure 6: Color Vs Price

Inference:

- Price of colour J is high so customer is less prefered with this colour. Price of colour E is less than colour G but in sales G is good than E.

Clarity Vs Price:

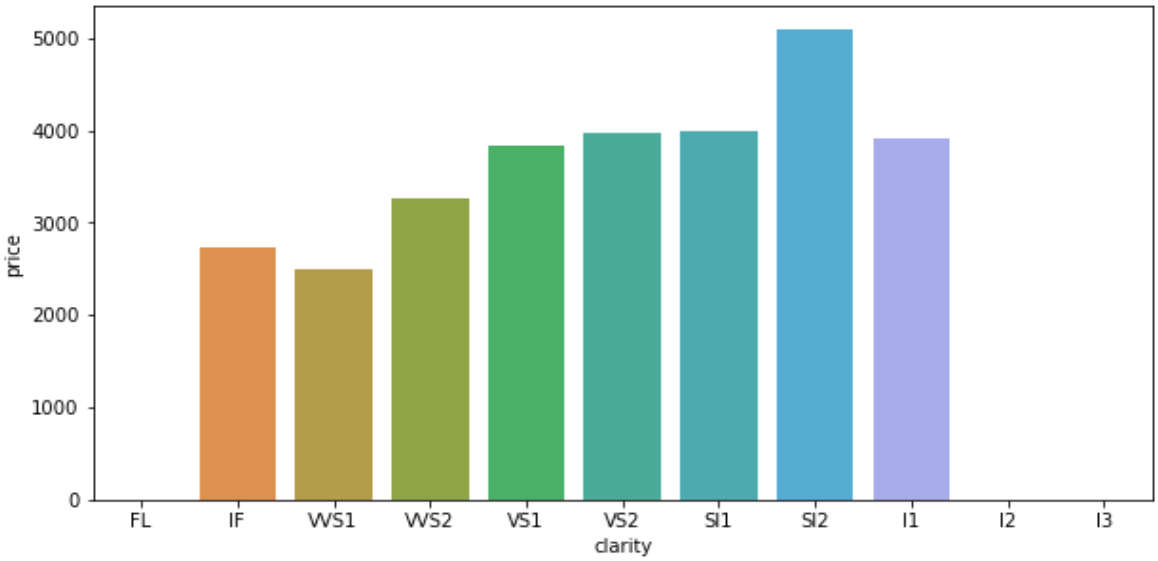


Figure 7: Clarity Vs Price

Inference:

- Sl1 and Vs2 is contributing good profit in diamond
- Sl2 price is high, So moderate customer preferred this clarity. Store has no FL,I2 and I3 diamonds.

Color Vs Price With hue Clarity:

clarity color	I1	IF	SI1	SI2	VS1	VS2	VVS1	VVS2	All
D	25	38	1040	671	369	804	121	276	3344
E	54	87	1249	849	625	1202	342	509	4917
F	67	183	1088	753	672	1107	360	499	4729
G	68	342	1001	779	1078	1205	507	681	5661
H	82	149	1082	796	595	804	288	306	4102
I	48	69	725	469	480	603	183	194	2771
J	21	26	386	258	274	374	38	66	1443
All	365	894	6571	4575	4093	6099	1839	2531	26967

Table 3: Cross tab with Color & Clarity

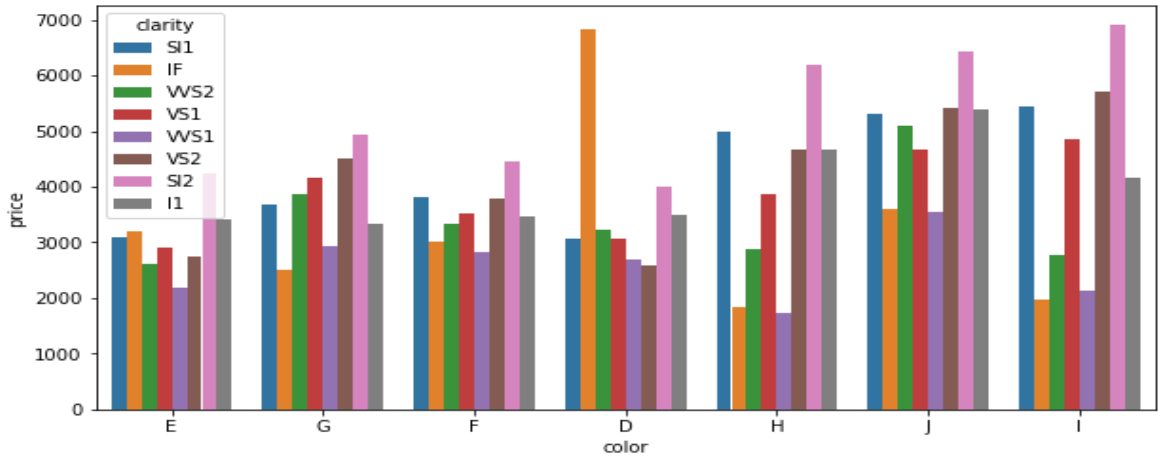


Figure 8:Color Vs Price With hue Clarity

Color Vs Price With hue Cut:

color cut	D	E	F	G	H	I	J	All
Fair	74	100	148	147	150	94	68	781
Good	311	491	454	419	352	253	161	2441
Ideal	1409	1966	1893	2470	1552	1073	453	10816
Premium	808	1174	1167	1471	1161	711	407	6899
Very Good	742	1186	1067	1154	887	640	354	6030
All	3344	4917	4729	5661	4102	2771	1443	26967

Table 4: Cross tab with Color & cut

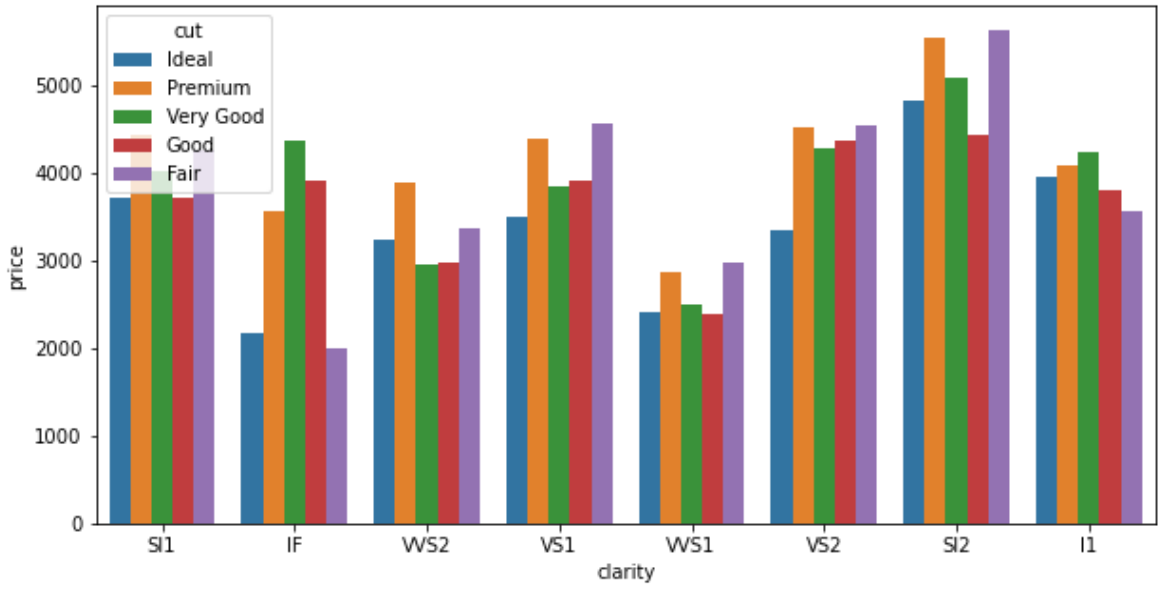


Figure 9:Color Vs Price With hue Cut

Clarity Vs Price With hue Color:

clarity cut	I1	IF	SI1	SI2	VS1	VS2	VVS1	VVS2	All
Fair	89	4	193	225	93	129	10	38	781
Good	51	30	765	530	331	491	100	143	2441
Ideal	74	613	2150	1324	1784	2528	1036	1307	10816
Premium	108	115	1809	1449	998	1697	307	416	6899
Very Good	43	132	1654	1047	887	1254	386	627	6030
All	365	894	6571	4575	4093	6099	1839	2531	26967

Table 5: Crosstab with Cut & Clarity

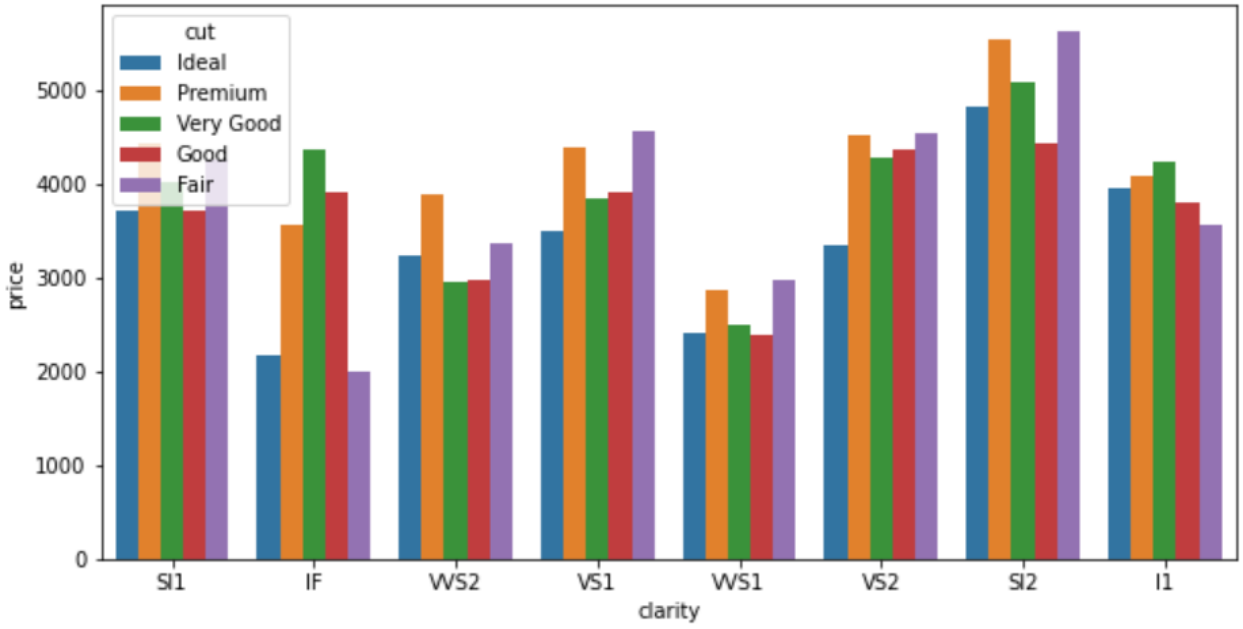


Figure 10: Clarity Vs Price With hue Color

1.2 Impute null values if present, also check for the values which are equal to zero. Do they have any meaning or do we need to change them or drop them? Check for the possibility of combining the sub levels of a ordinal variables and take actions accordingly. Explain why you are combining these sub levels with appropriate reasoning

Data Preprocessing:

```
Unnamed: 0      0
carat           0
cut             0
color           0
clarity         0
depth         697
table           0
x               0
y               0
z               0
price           0
dtype: int64
```

Inference:

- we have Null values in depth, since depth being continuous variable mean or median imputation can be done.
- The percentage of Null values is less than 5% (ie) 2.58%, we can also drop these if we want. After median imputation, we don't have any null values in the dataset.
- X,Y,Z are dimension of diamonds has zeors in data this is unacceptable because dimensions will not be zero's anymore. This will not help to build the model, So we can drop the Zero's which is less in data.

Data Scaling:

- Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual features do not more or less look like standard normally distributed data (e.g. Gaussian with 0 mean and unit variance).

Unnamed: 0	carat	depth	table	x	y	z	price
1	0.30	62.1	58.0	4.27	4.29	2.66	499
2	0.33	60.8	58.0	4.42	4.46	2.70	984
3	0.90	62.2	60.0	6.04	6.12	3.78	6289
4	0.42	61.6	56.0	4.82	4.80	2.96	1082
5	0.31	60.4	59.0	4.35	4.43	2.65	779

Table 6: Scaling the data

Outlier Teatment:

- Linear Regression models are sensitive to Outliers.
- Normalisation is used to transform all variables in the data to a same range. It doesn't solve the problem caused by outliers. So, as you can see the after normalisation also, the outliers remains outliers. Only the range is changed.
- Removal of outliers creates a normal distribution in some of my variables, and makes transformations for the other variables more effective. Therefore, it seems that removal of outliers before transformation is the better option.
- Compare the respective medians of each box plot. If the median line of a box plot lies outside of the box of a comparison box plot, then there is likely to be a difference between the two groups.

Before Treatment:

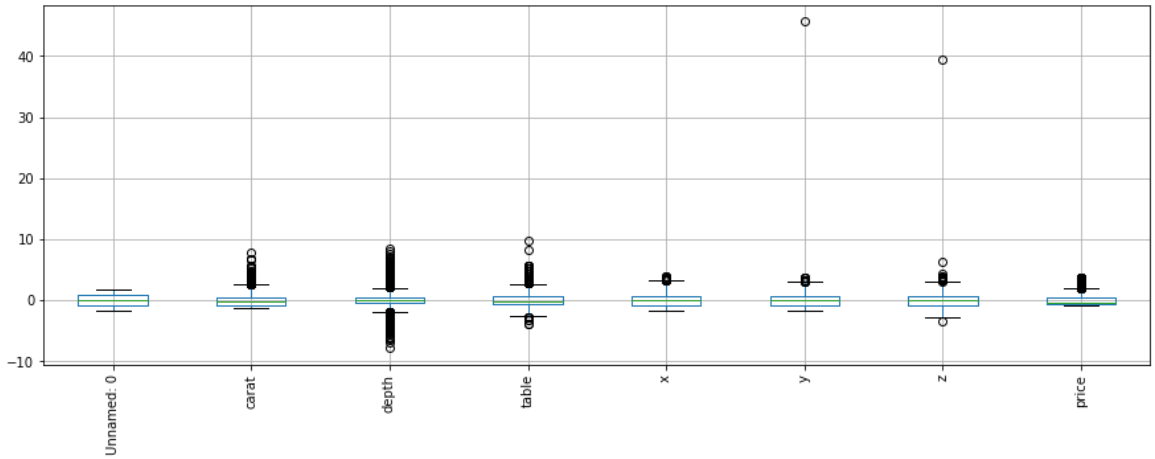


Figure 11: Outliers before Treatment

After Treatment:

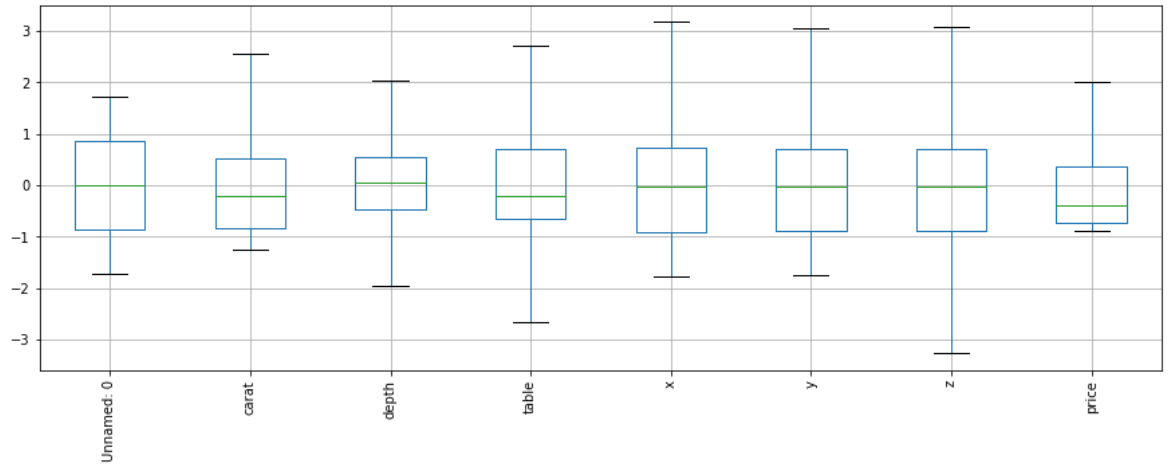


Figure 12: Outliers after Treatment

1.3 Encode the data (having string values) for Modelling. Split the data into train and test (70:30). Apply Linear regression using scikit learn.

- Perform checks for significant variables using appropriate method from statsmodel. Create multiple models and check the performance of Predictions on Train and Test sets using Rsquare, RMSE & Adj Rsquare. Compare these models and select the best one with appropriate reasoning.

Creating get dummies for categorical Variable:

carat	depth	table	x	y	z	price	cut_Good	cut_Ideal	...	color_H	color_I	color_J	clarity_IF	clarity_SI1	clarity_SI2	clarity_VS1	clarity_VS2	clarity_VVS1	clarity_VVS2
-1.043125	0.253399	0.244112	-1.295920	-1.240065	-1.224865	-0.854851	0	1	...	0	0	0	0	1	0	0	0	0	0
-0.980310	-0.679158	0.244112	-1.162787	-1.094057	-1.169142	-0.734303	0	0	...	0	0	0	1	0	0	0	0	0	0
0.213173	0.325134	1.140496	0.275049	0.331668	0.335404	0.584271	0	0	...	0	0	0	0	0	0	0	0	0	1
-0.791865	-0.105277	-0.652273	-0.807766	-0.802041	-0.806936	-0.709945	0	1	...	0	0	0	0	0	0	1	0	0	0
-1.022187	-0.966099	0.692304	-1.224916	-1.119823	-1.238796	-0.785257	0	1	...	0	0	0	0	0	0	0	0	1	0

Table 7: Dummies for categorical Variable

Train and Test Split:

```
# Split X and y into training and test set in 70:30 ratio

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3 , random_state=1)
```

Building Linear Regression Model:

- Regression analysis is one of the most commonly used toolstofind a relationship (linear or non- linear) between a responseand one or more predictorsand exploit that relationship in predicting the expected value of the response for certain values of the predictor(s) with maximum accuracy possible.

```
# invoke the LinearRegression function and find the bestfit model on training data

regression_model = LinearRegression()
regression_model.fit(X_train, y_train)
```

Let us explore the coefficients for each of the independent attributes:

1) The coefficient for carat is 1.1009417847804501.
2) The coefficient for depth is 0.005605143445570377 .
3) The coefficient for table is -0.013319500386804035.
4) The coefficient for x is -0.30504349819633475.
5) The coefficient for y is 0.30391448957926553 .
6) The coefficient for z is -0.13916571567987943.
7) The coefficient for cut_Good is 0.09403402912977911.
8) The coefficient for cut_Ideal is 0.1523107462056746.
9) The coefficient for cut_Premium is 0.14852774839849378.
10) The coefficient for cut_Very Good is 0.12583881878452705 .
11) The coefficient for color_E is -0.04705442233369822.
12) The coefficient for color_F is -0.06268437439142825.
13) The coefficient for color_G is -0.10072161838356786 .
14) The coefficient for color_H is -0.20767313311661612 .
15) The coefficient for color_I is -0.3239541927462737.
16) The coefficient for color_J is -0.46858930275015803.
17) The coefficient for clarity_IF is 0.9997691394634902.
18) The coefficient for clarity_SI1 is 0.6389785818271332.
19) The coefficient for clarity_SI2 is 0.42959662348315514.
20) The coefficient for clarity_VS1 is 0.8380875826737564 .
21) The coefficient for clarity_VS2 is 0.7660244466083613 .
22) The coefficient for clarity_VVS1 is 0.9420769630114072 .
23) The coefficient for clarity_VVS2 is 0.9313670288415696.

Let us check the intercept for the model:

```
intercept = regression_model.intercept_

print("The intercept for our model is {}".format(intercept))
```

```
The intercept for our model is -0.7567627863049391
```

```
# R square on training data
regression_model.score(X_train, y_train)
```

```
0.9419557931252712
```

```
# R square on testing data
regression_model.score(X_test, y_test)
```

```
0.9381643998102491
```

```
#RMSE on Training data
predicted_train=regression_model.fit(X_train, y_train).predict(X_train)
np.sqrt(metrics.mean_squared_error(y_train,predicted_train))
```

```
0.20690072466418796
```

```
#RMSE on Testing data
predicted_test=regression_model.fit(X_train, y_train).predict(X_test)
np.sqrt(metrics.mean_squared_error(y_test,predicted_test))
```

```
0.21647817772382869
```

Inferential statistics:

The simple linear regression model involves unknown parameters 0 and 1,which need to be estimated from data. There are several different methods of estimating the parameters. The simplest and the most widely used method is known as the Ordinary Least Squares method (OLS).

Intercept	-0.756763
carat	1.100942
depth	0.005605
table	-0.013320

x	-0.305043
y	0.303914
z	-0.139166
cut_Good	0.094034
cut_Ideal	0.152311
cut_Premium	0.148528
cut_Very_Good	0.125839
color_E	-0.047054
color_F	-0.062684
color_G	-0.100722
color_H	-0.207673
color_I	-0.323954
color_J	-0.468589
clarity_IF	0.999769
clarity_SI1	0.638979
clarity_SI2	0.429597
clarity_VS1	0.838088
clarity_VS2	0.766024
clarity_VVS1	0.942077
clarity_VVS2	0.931367

Table 8:Coefficient with all Variables

OLS Regression Results

Dep. Variable:	price	R-squared:	0.942
Model:	OLS	Adj. R-squared:	0.942
Method:	Least Squares	F-statistic:	1.330e+04
Date:	Mon, 04 Oct 2021	Prob (F-statistic):	0.00
Time:	13:43:42	Log-Likelihood:	2954.6
No. Observations:	18870	AIC:	-5861.
Df Residuals:	18846	BIC:	-5673.
Df Model:	23		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-0.7568	0.016	-46.999	0.000	-0.788	-0.725
carat	1.1009	0.009	121.892	0.000	1.083	1.119
depth	0.0056	0.004	1.525	0.127	-0.002	0.013
table	-0.0133	0.002	-6.356	0.000	-0.017	-0.009
x	-0.3050	0.032	-9.531	0.000	-0.368	-0.242
y	0.3039	0.034	8.934	0.000	0.237	0.371
z	-0.1392	0.024	-5.742	0.000	-0.187	-0.092
cut_Good	0.0940	0.011	8.755	0.000	0.073	0.115
cut_Ideal	0.1523	0.010	14.581	0.000	0.132	0.173
cut_Premium	0.1485	0.010	14.785	0.000	0.129	0.168
cut_Very_Good	0.1258	0.010	12.269	0.000	0.106	0.146
color_E	-0.0471	0.006	-8.429	0.000	-0.058	-0.036
color_F	-0.0627	0.006	-11.075	0.000	-0.074	-0.052
color_G	-0.1007	0.006	-18.258	0.000	-0.112	-0.090
color_H	-0.2077	0.006	-35.323	0.000	-0.219	-0.196
color_I	-0.3240	0.007	-49.521	0.000	-0.337	-0.311
color_J	-0.4686	0.008	-58.186	0.000	-0.484	-0.453
clarity_IF	0.9998	0.016	62.524	0.000	0.968	1.031
clarity_SI1	0.6390	0.014	46.643	0.000	0.612	0.666
clarity_SI2	0.4296	0.014	31.177	0.000	0.403	0.457
clarity_VS1	0.8381	0.014	59.986	0.000	0.811	0.865
clarity_VS2	0.7660	0.014	55.618	0.000	0.739	0.793
clarity_VVS1	0.9421	0.015	63.630	0.000	0.913	0.971
clarity_VVS2	0.9314	0.014	64.730	0.000	0.903	0.960

Omnibus:	4696.785	Durbin-Watson:	1.994
Prob(Omnibus):	0.000	Jarque-Bera (JB):	17654.853
Skew:	1.208	Prob(JB):	0.00
Kurtosis:	7.076	Cond. No.	57.0

Inference:

- Depth is a continuous predictor is not significant, i.e. if the p-value in the regression table is less than a pre-fixed level α=0.05, we simply eliminate the variable from the regression equation.
- However, the numerical value of R square is non-decreasing even if non-significant predictors are included in the model. Addition of non-significant predictors adversely affects the predictive quality of the model. Therefore, there is a need of another measure of model adequacy.
- Square measures the proportion of the total variation in$Y$that is explained by the regression model. It ranges from 0≤R^2≤1.The higher the value of R2,the more powerful is the predictor to predict the response. Regression model with high R2value indicates that the model fits the data well. In that case a high proportion of variance in the response is explained by the dependence of the response on the predictor.
- To ideally bring down the values to lower levels we can drop depth variable that is highly correlated. Dropping depth variables would bring down the multi co linearity level down.

```
# Let us check the sum of squared errors by predicting value of y for test cases and
# subtracting from the actual y for the test cases

mse = np.mean((regression_model.predict(X_test)-y_test)**2)
```

```
# underroot of mean_sq_error is standard deviation i.e. avg variance between predicted and actual

import math

math.sqrt(mse)
```

```
0.21647817772382863
```

```
# Model score - R2 or coeff of determinant
# R^2=1-RSS / TSS

regression_model.score(X_test, y_test)
```

```
0.9381643998102491
```

Intercept	-0.756657
carat	1.101954
table	-0.013928
x	-0.315617
y	0.283420
z	-0.108789
cut_Good	0.095123
cut_Ideal	0.151173
cut_Premium	0.147355
cut_Very_Good	0.125514
color_E	-0.047114
color_F	-0.062727
color_G	-0.100657
color_H	-0.207568
color_I	-0.323689
color_J	-0.468428
clarity_IF	1.000046
clarity_SI1	0.639804
clarity_SI2	0.430195
clarity_VS1	0.838626
clarity_VS2	0.766683
clarity_VVS1	0.942390
clarity_VVS2	0.931898

Table 9: Coefficient of all variables without depth

OLS Regression Results

Dep. Variable:	price	R-squared:	0.942
Model:	OLS	Adj. R-squared:	0.942
Method:	Least Squares	F-statistic:	1.390e+04
Date:	Mon, 04 Oct 2021	Prob (F-statistic):	0.00
Time:	13:17:14	Log-Likelihood:	2953.5
No. Observations:	18870	AIC:	-5861.
Df Residuals:	18847	BIC:	-5680.
Df Model:	22		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-0.7567	0.016	-46.991	0.000	-0.788	-0.725
carat	1.1020	0.009	122.331	0.000	1.084	1.120
table	-0.0139	0.002	-6.770	0.000	-0.018	-0.010
x	-0.3156	0.031	-10.101	0.000	-0.377	-0.254
y	0.2834	0.031	9.069	0.000	0.222	0.345
z	-0.1088	0.014	-7.883	0.000	-0.136	-0.082
cut_Good	0.0951	0.011	8.876	0.000	0.074	0.116
cut_Ideal	0.1512	0.010	14.508	0.000	0.131	0.172
cut_Premium	0.1474	0.010	14.711	0.000	0.128	0.167
cut_Very_Good	0.1255	0.010	12.239	0.000	0.105	0.146
color_E	-0.0471	0.006	-8.439	0.000	-0.058	-0.036
color_F	-0.0627	0.006	-11.082	0.000	-0.074	-0.052
color_G	-0.1007	0.006	-18.246	0.000	-0.111	-0.090
color_H	-0.2076	0.006	-35.306	0.000	-0.219	-0.196
color_I	-0.3237	0.007	-49.497	0.000	-0.337	-0.311
color_J	-0.4684	0.008	-58.169	0.000	-0.484	-0.453
clarity_IF	1.0000	0.016	62.544	0.000	0.969	1.031
clarity_SI1	0.6398	0.014	46.738	0.000	0.613	0.667
clarity_SI2	0.4302	0.014	31.232	0.000	0.403	0.457
clarity_VS1	0.8386	0.014	60.042	0.000	0.811	0.866
clarity_VS2	0.7667	0.014	55.691	0.000	0.740	0.794
clarity_VVS1	0.9424	0.015	63.655	0.000	0.913	0.971
clarity_VVS2	0.9319	0.014	64.784	0.000	0.904	0.960

Omnibus:	4699.504	Durbin-Watson:	1.994
Prob(Omnibus):	0.000	Jarque-Bera (JB):	17704.272
Skew:	1.208	Prob(JB):	0.00
Kurtosis:	7.084	Cond. No.	56.5

Inference:

Notice that now, all the predictors are statistically significant with very low p-values. The value of R square has not decreased much and adjusted R squares almost same as R square(around 94%).

Thus we select this as our final regression model.

```
for i,j in np.array(lm2.params.reset_index()):
print('({}) * {} +'.format(round(j,2),i),end=' '
```

(-0.76) * Intercept + (1.1) * carat + (-0.01) * table + (-0.32) * x + (0.2 8) * y + (-0.11) * z + (0.1) * cut_Good + (0.15) * cut_Ideal + (0.15) * cut

_Premium + (0.13) * cut_Very_Good + (-0.05) * color_E + (-0.06) * color_F + (-0.1) * color_G + (-0.21) * color_H + (-0.32) * color_I + (-0.47) * color_ J + (1.0) *

clarity_IF + (0.64) * clarity_SI1 + (0.43) * clarity_SI2 + (0.8 4) * clarity_VS1 + (0.77) * clarity_VS2 + (0.94) * clarity_VVS1 + (0.93) * clarity_VVS

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

- We had a business problem to predict the price of the stone and provide insights for the company on the profits on different prize slots
- we can clearly observe that Carat has the highest coefficient . However Carat value increase the price also increase significantly when compared to any other feature in the data.
- Ideal cut is most preferred cut for diamond.
- We have 7 colors in the data, The G seems to be the preferred color.J is low among other colors.
- The predictions were able to capture 95% variations in the price and it is explained by the predictors in the training set.
- Using stats model if we could run the model again we can have P values and coefficients which will give us better understanding of the relationship, so that values more 0.05 we can drop those variables and re run the model again for better results.
- For better accuracy dropping depth column in iteration for better results.

Equation that coefficient varies with respect to price

(-0.76) * Intercept + (1.1) * carat + (-0.01) * table + (-0.32) * x + (0.2 8) * y + (-0.11) * z + (0.1) * cut_Good + (0.15) * cut_Ideal + (0.15) * cut

_Premium + (0.13) * cut_Very_Good + (-0.05) * color_E + (-0.06) * color_F + (-0.1) * color_G + (-0.21) * color_H + (-0.32) * color_I + (-0.47) * color_ J + (1.0) * clarity_IF + (0.64) * clarity_SI1 + (0.43) * clarity_SI2 + (0.8 4) * clarity_VS1 + (0.77) * clarity_VS2 + (0.94) * clarity_VVS1 + (0.93) * clarity_VVS

Problem 2: Logistic Regression and LDA

Problem Statement:

You are hired by a tour and travel agency which deals in selling holiday packages. You are provided details of 872 employees of a company. Among these employees, some opted for the package and some didn't. You have to help the company in predicting whether an employee will opt for the package or not based on the information given in the data set. Also, find out the important factors based on which the company will focus on particular employees to sell their packages

Data Dictionary:

Variable Name-Description

Holiday_Package-Opted for Holiday Package yes/no?

Salary- Employee salary

age -Age in years

edu- Years of formal education

no_young_children -The number of young children (younger than 7 years)

no_older_children- Number of older children

foreign- foreigner Yes/No

2.1 Data Ingestion:Read the dataset. Do the descriptive statistics and do null value condition check, write an inference on it. Perform Univariate and Bivariate Analysis. Do exploratory data analysis.

Reading the Dataset:

	Unnamed: 0	Holliday_Package	Salary	age	educ	no_young_children	no_older_children	foreign
0	1	no	48412	30	8	1	1	no
1	2	yes	37207	45	8	0	1	no
2	3	no	58022	46	9	0	0	no
3	4	no	66503	31	11	2	0	no
4	5	no	66734	44	12	0	2	no

Table 10: Reading the data

Inference:

- Dataset has 872 rows and 8 columns.
- Dataset has both numerical and categorical variables.
- Unnamed: 0 column has no information so we can remove this variable before modeling
- Numerical variables are Salary,educ,no young_ children and no_older_children.
- Categorical variable has Holliday Package and foreign.

- Holliday Package is target variable in dataset.

Summary of Dataset:

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Unnamed: 0	872	NaN	NaN	NaN	436.5	251.869	1	218.75	436.5	654.25	872
Holliday_Package	872	2	no	471	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Salary	872	NaN	NaN	NaN	47729.2	23418.7	1322	35324	41903.5	53469.5	236961
age	872	NaN	NaN	NaN	39.9553	10.5517	20	32	39	48	62
educ	872	NaN	NaN	NaN	9.30734	3.03626	1	8	9	12	21
no_young_children	872	NaN	NaN	NaN	0.311927	0.61287	0	0	0	0	3
no_older_children	872	NaN	NaN	NaN	0.982798	1.08679	0	0	1	2	6
foreign	872	2	no	656	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Table 11: Summary of Data

Inference:

- Dataset has 872 rows and 8 columns.
- Unnamed: 0 column has no information so we can remove this variable before modelling.
- Most of the employees preferred no for Holliday pacages.
- Salary packages for employee from 1322 to max 236961.
- Maximum age for an employee in organization is 62.
- Mean is higher than median so we can expect positive skewness

Target Variable:

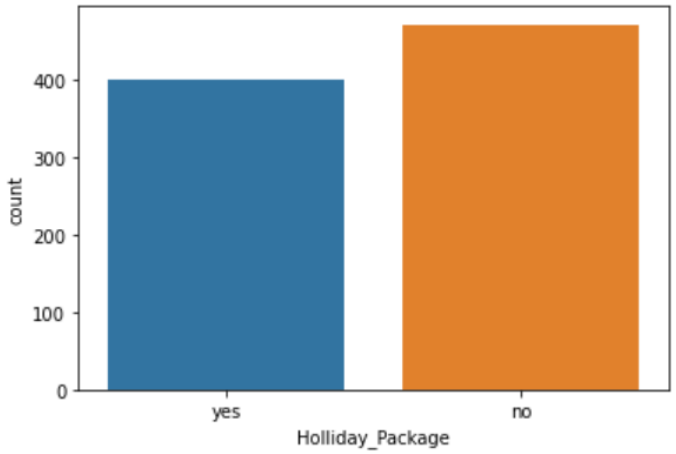


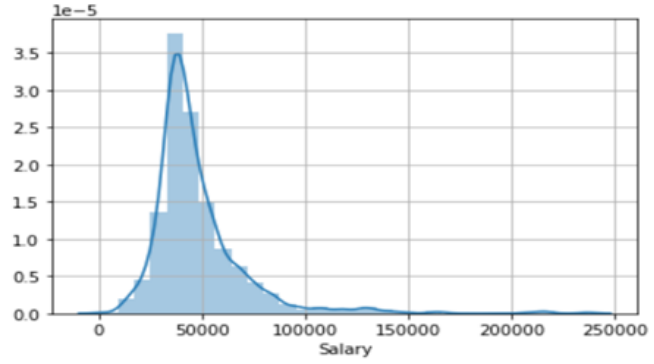
Figure 12:Target Variable

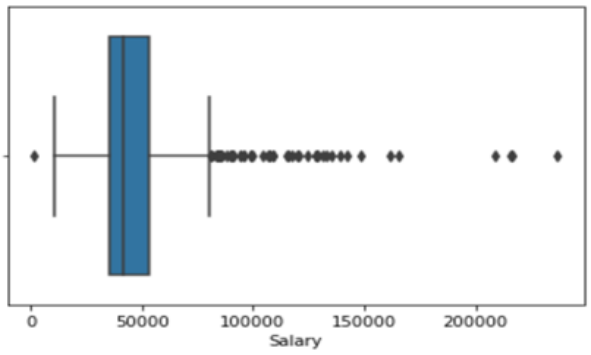
Inference:

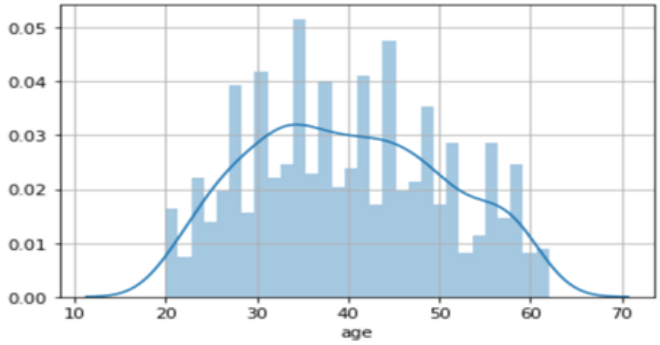
- 46%(401 out of 872) of employees in organization opted for holliday packages and 54%(471 out of 872) are not opted for Holliday packages.

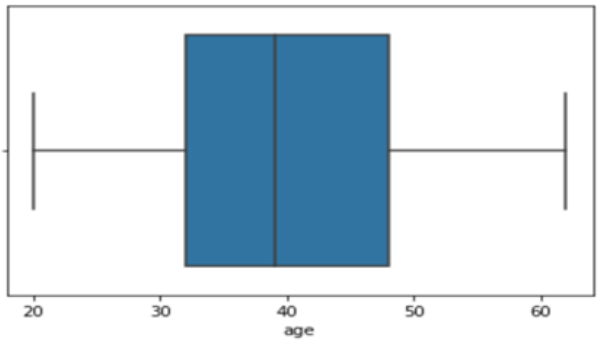
univariate Analysis:

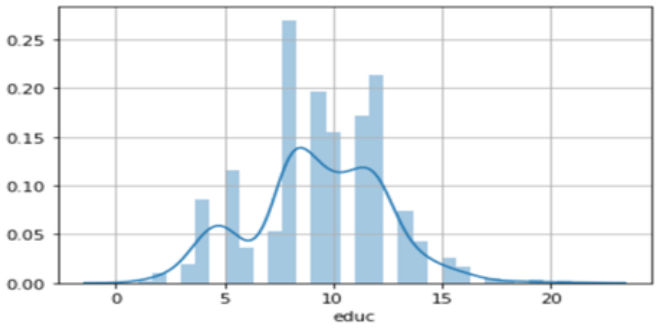
Continuous Variables:

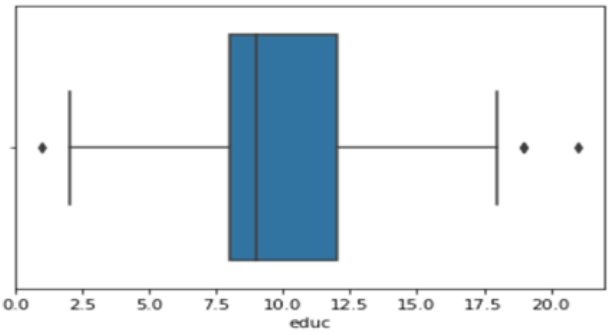


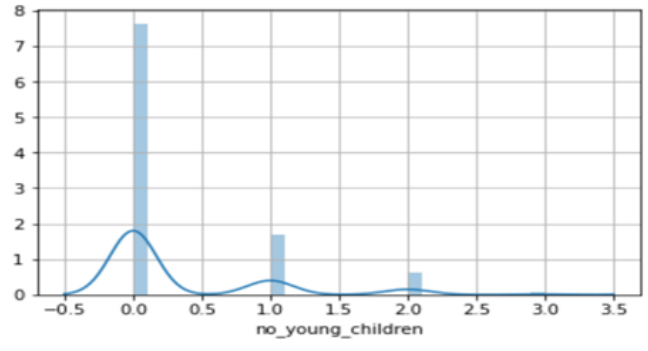


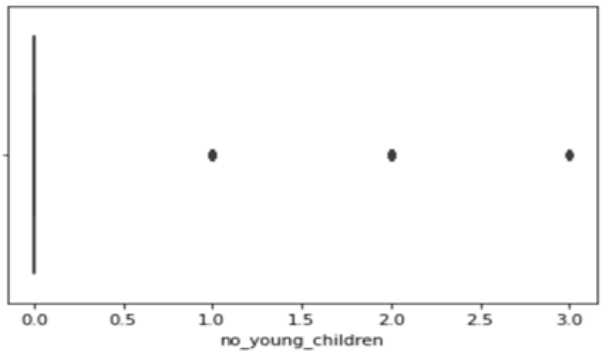












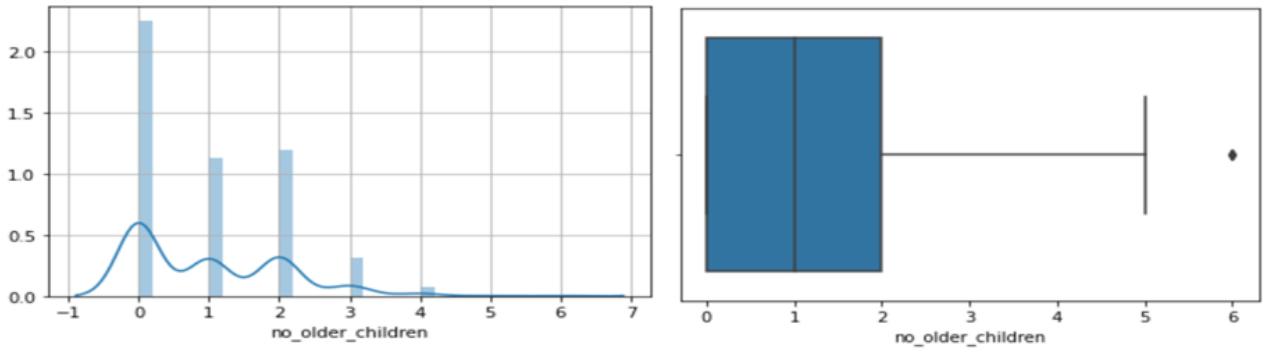


Figure 13: Univariate with Continuous Variable

Inference:

- salary has positively skewed and has more number of outliers in data.
- Age is widely spread with min 20 and max 62. There is no ouliers in data.
- No of younger children is 665 zeros's which seem to be valid and has no outliers.
- Most of the data in no of younger children lies between 0 to 2 and has outliers.

Skewness:

Salary	3.103216
age	0.146412
educ	0.045501
no_young_children	1.946515
no_older_children	0.953951

Table 12: Skewness for Continuous Variable

Categorical Variable:

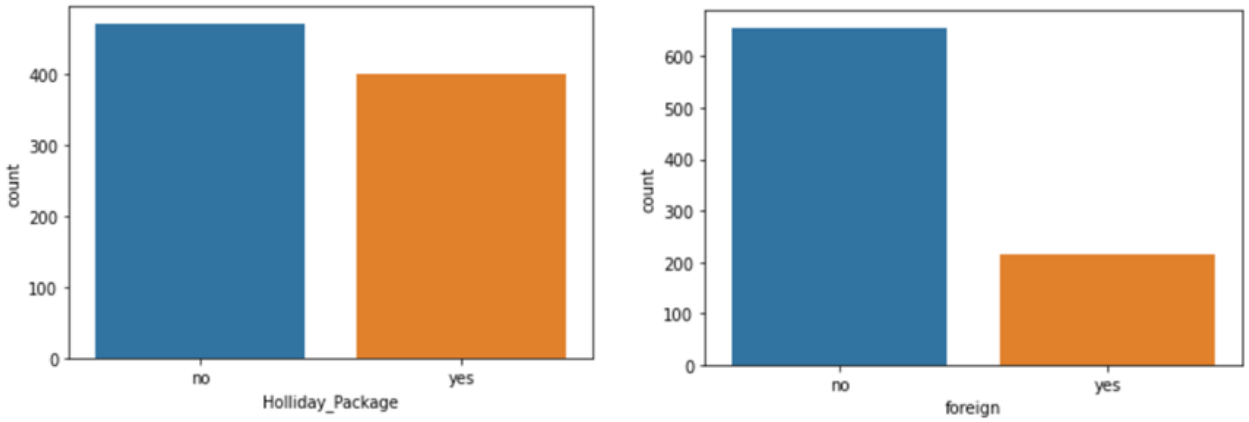


Figure 14: Univariate with Categorical Variable

Inference:

46%(401 out of 872) of employees in organization opted for holliday packages and 54%(471 out of 872) are not opted for holliday packages.

Out of 876 employes in organization 216 has foreigners and remaining 656 has no foreigner.

Bivariate Analysis:

Holliday_Package vs Salary:

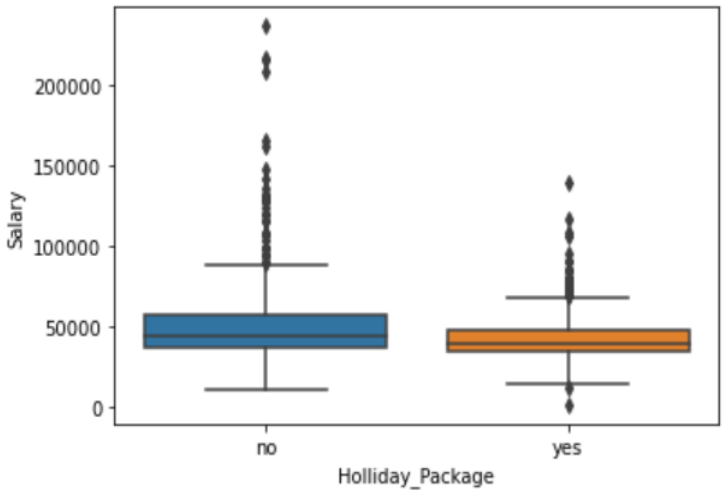


Figure 15: Holliday Package VS Salary

Inference:

Salary more than 150000 not opted for holiday packages.

Most of the employes less than 50000 salary is opted for holiday packages

Holliday Package vs Salary vs Age:

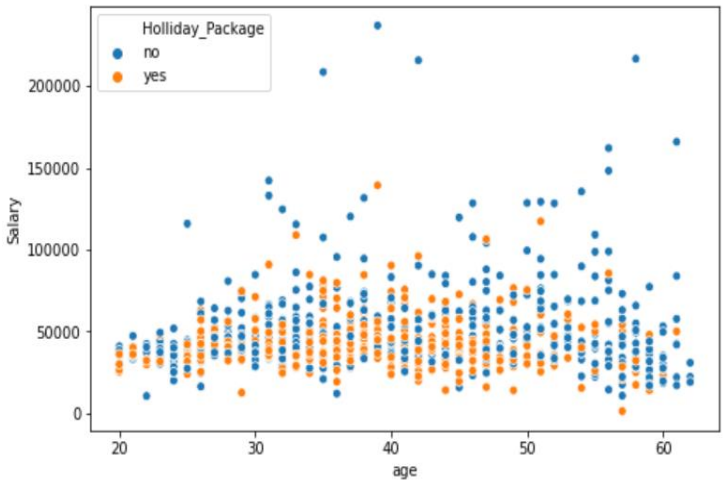


Figure 16:Holliday Package vs Salary vs Age

Holliday Package vs Salary vs educ:

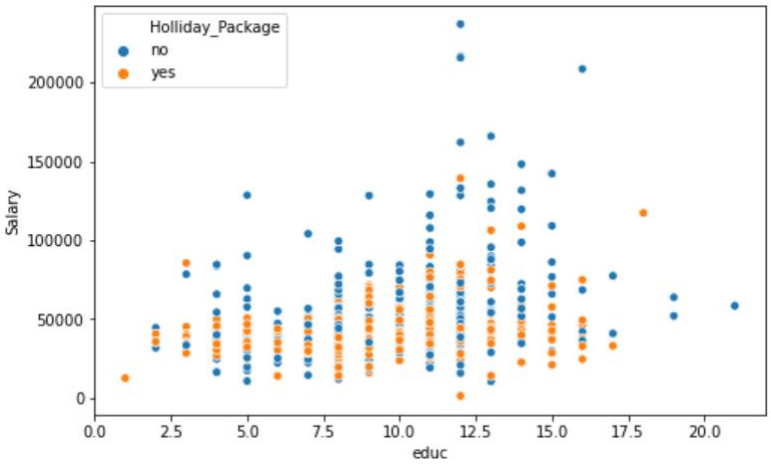


Figure 17:Holliday Package vs Salary vs Age

Holliday Package VS no young children VS Salary :

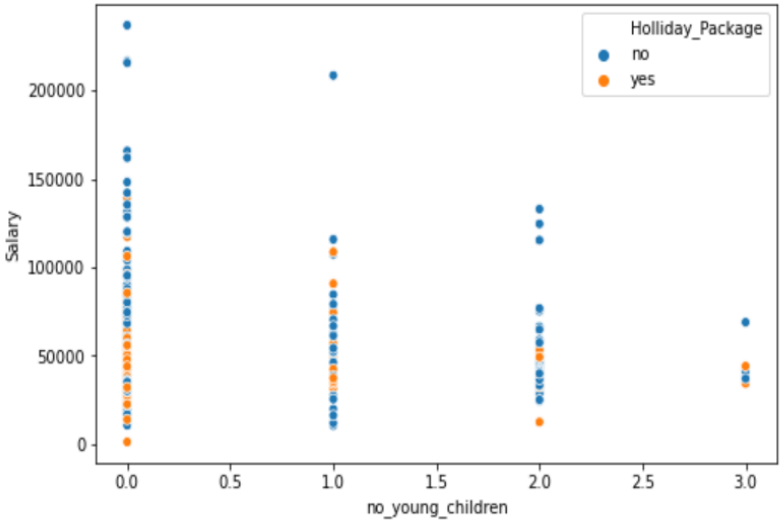


Figure 18:Holliday Package VS no young children VS Salary

Holliday Package vs no older children vs Salary:

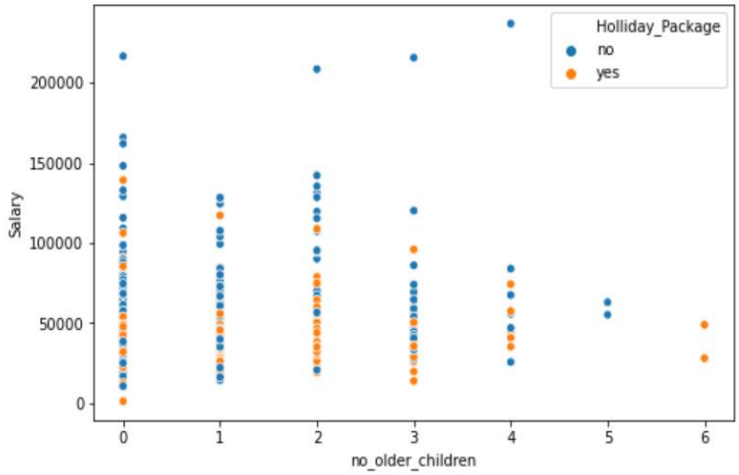


Figure 19: Holliday Package vs no older children vs Salary

Multivariate Analysis:

Pair Plot with Contineous Variable:

Figure 20: Pair Plot with Continuous Variable

Heatmap:

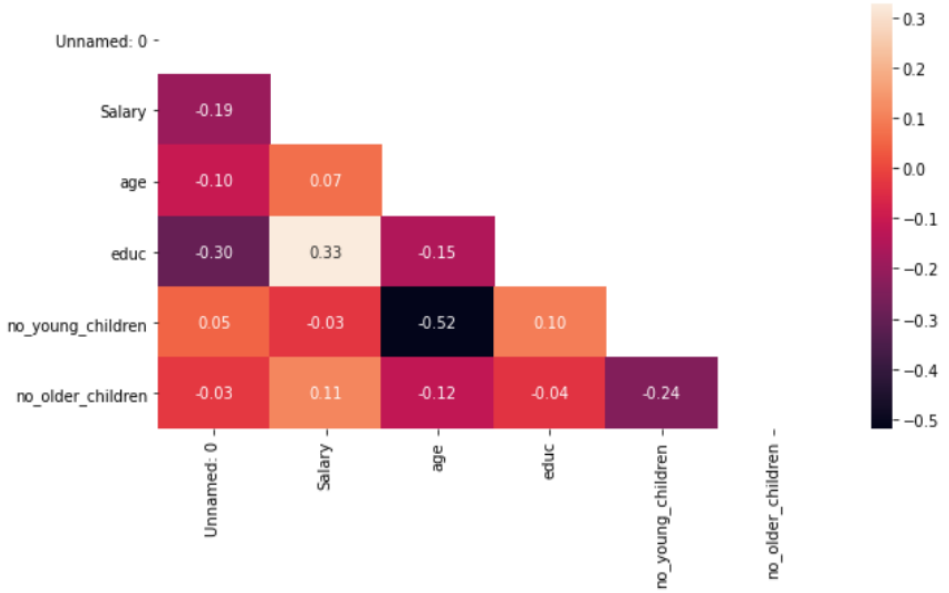


Figure 21: Correlation Map

Inference:

- There is no multicollinearity in data.
- No of young children and age is negatively correlated which is high.

Outlier Teatment:

- Logistic Regression and LDA models are sensitive to Outliers.
- Removal of outliers creates a normal distribution in some of my variables, and makes transformations for the other variables more effective. Therefore, it seems that removal of outliers before transformation is the better option.
- Compare the respective medians of each box plot. If the median line of a box plot lies outside of the box of a comparison box plot, then there is likely to be a difference between the two groups.

Before Treatment:

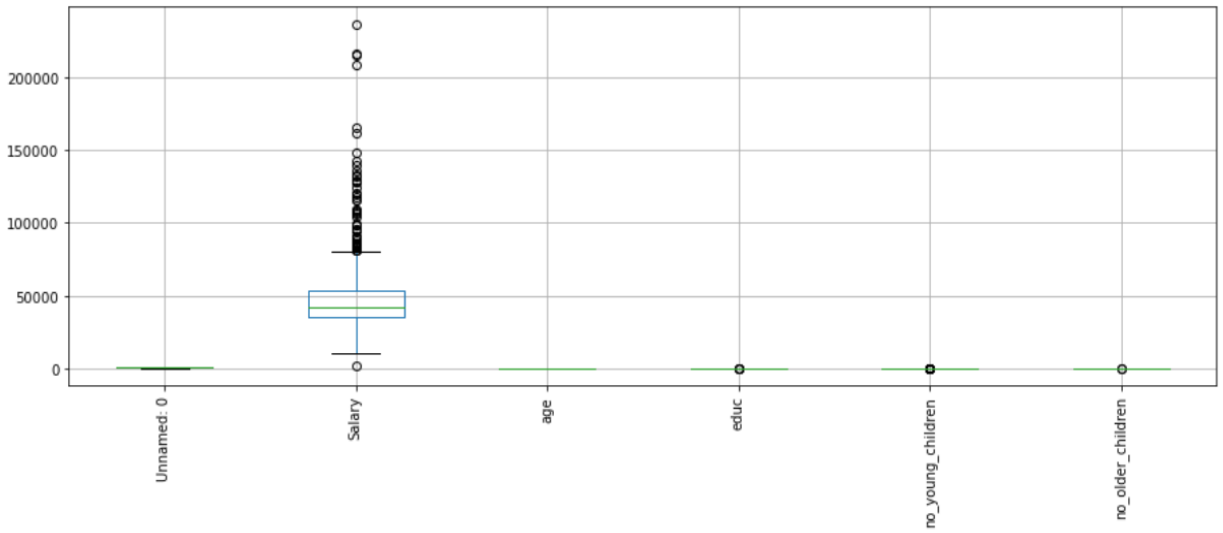


Figure 22: Outliers Before Treatment

After Treatment:

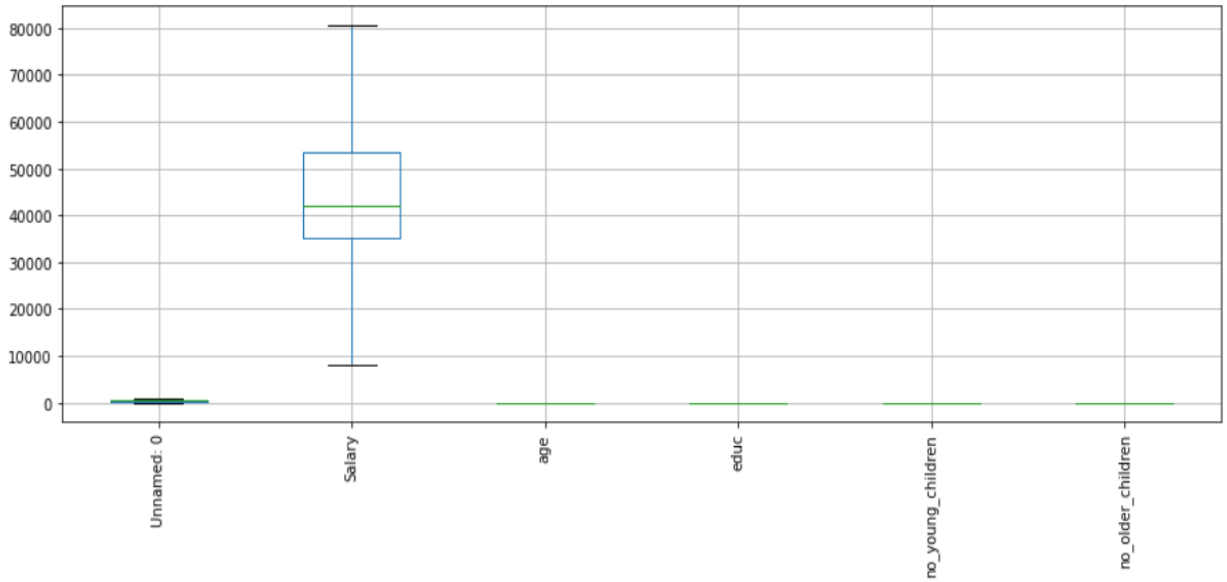


Figure 23: Outliers after Treatment

2.2 Do not scale the data. Encode the data (having string values) for Modelling. Data Split: Split the data into train and test (70:30). Apply Logistic Regression and LDA (linear discriminant analysis).

Creating get-dumies for categorical Variable:

	Salary	age	educ	no_young_children	no_older_children	Holliday_Package_yes	foreign_yes
0	48412.0	30.0	8.0	0.0	1.0	0	0
1	37207.0	45.0	8.0	0.0	1.0	1	0
2	58022.0	46.0	9.0	0.0	0.0	0	0
3	66503.0	31.0	11.0	0.0	0.0	0	0
4	66734.0	44.0	12.0	0.0	2.0	0	0

Table 13: dummies for categorical Variable

Train & Test split:

- Splitting the data into train and test (70:30) to apply Logistic Regression and LDA.

Logistic Regression Model:

- Linear regression is a basic and commonly used type of predictive analysis.
- The overall idea of regression is to examine two things: (1) does a set of predictor variables do a good job in predicting an outcome (dependent) variable? (2) Which variables in particular are significant predictors of the outcome variable, and in what way do they–indicated by the magnitude and sign of the beta estimates–impact the outcome variable? These regression

estimates are used to explain the relationship between one dependent variable and one or more independent variables.

- The simplest form of the regression equation with one dependent and one independent variable is defined by the formula y = c + b*x, where y = estimated dependent variable score, c = constant, b = regression coefficient, and x = score on the independent variable.

Grid Search:

- Grid-search is used to find the optimal hyperparameters of a model which results in the most 'accurate' predictions

```
grid={'penalty':['l1','l2','none'],
      'solver':['lbfgs', 'liblinear'],
      'tol':[0.0001,0.000001]}
```

- After performing grid search it shows the best parameters,

```
{'penalty': 'l2', 'solver': 'liblinear', 'tol': 1e-06}

LogisticRegression(max_iter=100000, n_jobs=2, solver='liblinear', tol=1e-06)
```

- Liblinear most probably used in small dataset.

LDA Model:

- Linear Discriminant Analysis or LDA is a dimensionality reduction technique. It is used as a pre-processing step in Machine Learning and applications of pattern classification.
- The goal of LDA is to project the features in higher dimensional space onto a lower-dimensional space in order to avoid the curse of dimensionality and also reduce resources and dimensional costs.
- The original technique was developed in the year 1936 by Ronald A. Fisher and was named Linear Discriminant or Fisher's Discriminant Analysis. T
- he original Linear Discriminant was described as a two-class technique. The multi-class version was later generalized by C.R Rao as Multiple Discriminant Analysis. They are all simply referred to as the Linear Discriminant Analysis.
- LDA is a supervised classification technique that is considered a part of crafting competitive machine learning models.
- This category of dimensionality reduction is used in areas like image recognition and predictive analysis in marketing.

	Holliday_Package	Salary	age	educ	no_young_children	no_older_children	foreign
0	no	48412.0	30.0	8.0	0.0	1.0	no
1	yes	37207.0	45.0	8.0	0.0	1.0	no
2	no	58022.0	46.0	9.0	0.0	0.0	no
3	no	66503.0	31.0	11.0	0.0	0.0	no
4	no	66734.0	44.0	12.0	0.0	2.0	no

Table 14:Reading data to perform LDA

```
#Build LDA Model
clf = LinearDiscriminantAnalysis()
model=clf.fit(X_train,Y_train)
```

2.3 Performance Metrics:Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model Final Model:Compare Both the models and write inference which model is best/optimized.

Performance Metrics for Logistic Regression:

Accuracy for Training Data:

```
# Accuracy - Training Data

lr_train_acc = best_model.score(X_train, y_train)
lr_train_acc
```

```
0.6344262295081967
```

Classification Report for Training Data:

```
## Confusion matrix on the training data

plot_confusion_matrix(best_model,X_train,y_train)
print(classification_report(y_train, ytrain_predict),'\n');
```

```
              precision    recall  f1-score   support

           0       0.63      0.79      0.70       329
           1       0.65      0.45      0.53       281

    accuracy                           0.63       610
   macro avg       0.64      0.62      0.62       610
weighted avg       0.64      0.63      0.62       610
```

Confusion Matrix for Training Data:

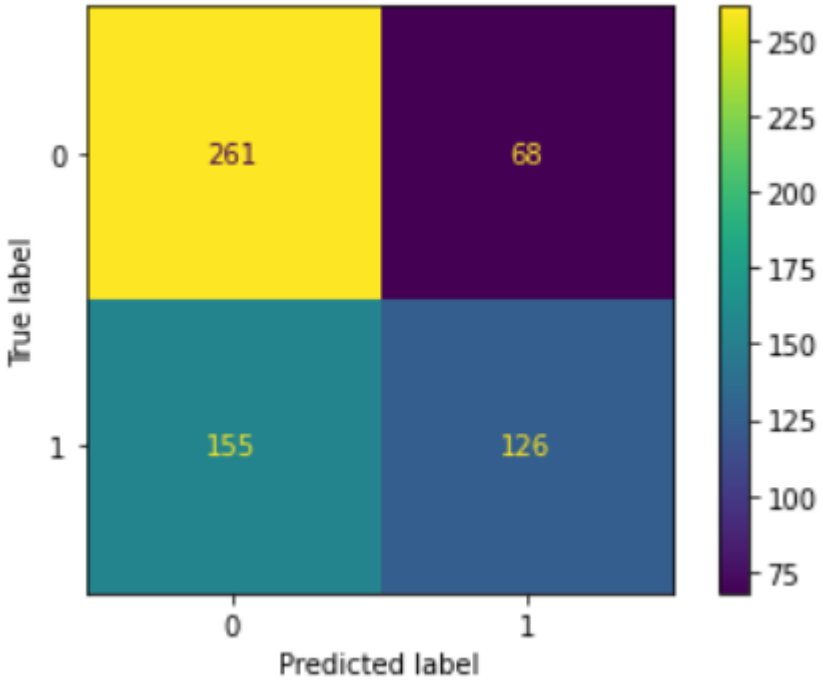


Figure 24:Confusion Matrix for Training Data(LR)

AUC and ROC for the training data:

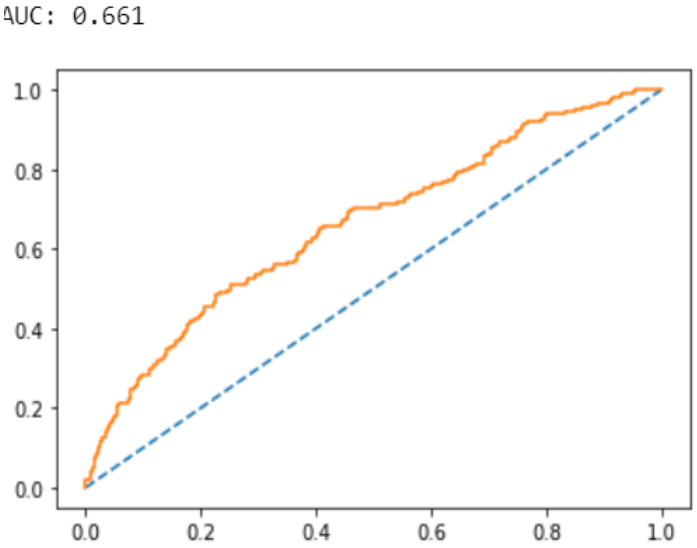


Figure 25:AUC and ROC for the training data(LR)

Accuracy for Test Data:

```
# Accuracy - Training Data

lr_train_acc = best_model.score(X_train, y_train)
lr_train_acc
```

```
0.6344262295081967
```

Classification Report for Test Data:

```
## Confusion matrix on the test data

plot_confusion_matrix(best_model,X_test,y_test)
print(classification_report(y_test, ytest_predict),'\n');
```

```
              precision    recall  f1-score   support

           0       0.64      0.83      0.72       142
           1       0.69      0.45      0.55       120

    accuracy                           0.66       262
   macro avg       0.67      0.64      0.63       262
weighted avg       0.66      0.66      0.64       262
```

Confusion matrix for Test Data:

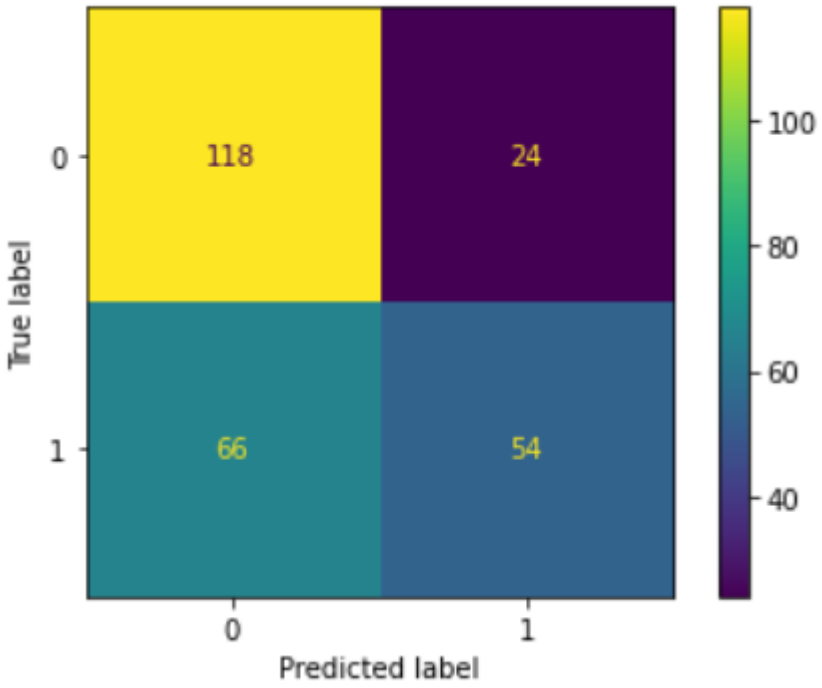


Figure 26:Confusion matrix for Test Data(LR)

AUC and ROC for the test data:

```
AUC: 0.675
```

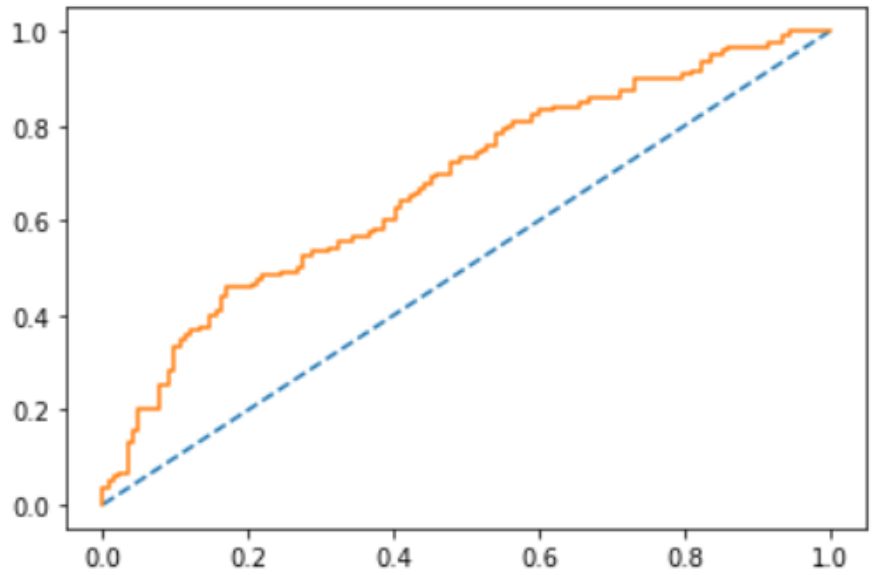


Figure 27: AUC and ROC for the test data(LR)

Performance Metrics for LDA:

Accuracy for Training Data:

```
lda_train_acc = model.score(X_train,Y_train)
lda_train_acc
```

```
0.6327868852459017
```

Classification Report for Training Data:

	precision	recall	f1-score	support
0	0.62	0.80	0.70	329
1	0.65	0.44	0.52	281
accuracy			0.63	610
macro avg	0.64	0.62	0.61	610
weighted avg	0.64	0.63	0.62	610

Confusion Matrix for Training Data:

```
confusion_matrix(Y_train, pred_class_train)
```

```
array([[263,  66],
       [158, 123]], dtype=int64)
```

Accuracy for Test Data:

```
lda_test_acc = model.score(X_test,Y_test)
lda_test_acc
```

```
0.6564885496183206
```

Confusion Matrix for Test Data:

```
confusion_matrix(Y_test, pred_class_test)
```

```
array([[118,  24],
       [ 66,  54]], dtype=int64)
```

Classification Report for Test Data:

	precision	recall	f1-score	support
0	0.64	0.83	0.72	142
1	0.69	0.45	0.55	120
accuracy			0.66	262
macro avg	0.67	0.64	0.63	262
weighted avg	0.66	0.66	0.64	262

- **Training and test data in this model is performing almost similar way, So we change cut-off value to get better accuracy in the model.**

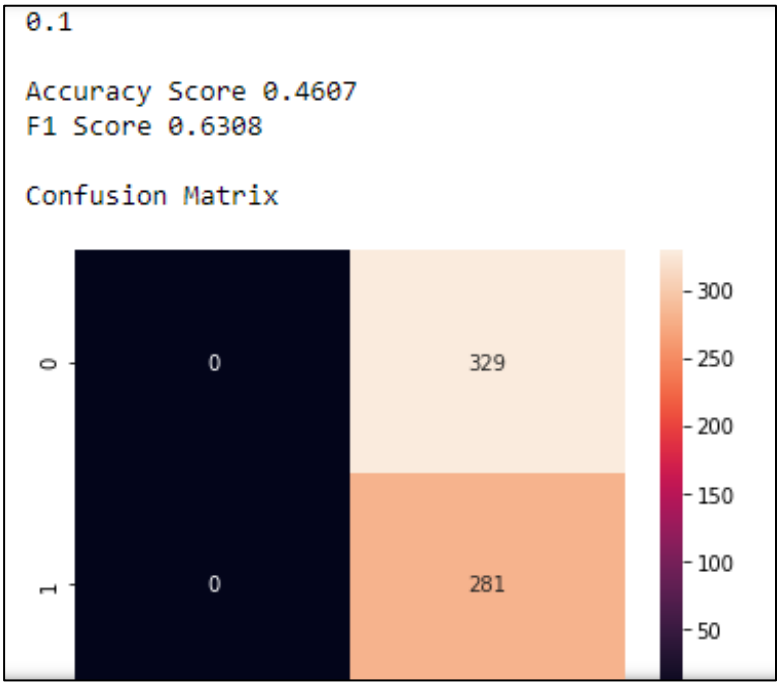


Figure 28: Confusion Matrix for 0.1 Cutoff.

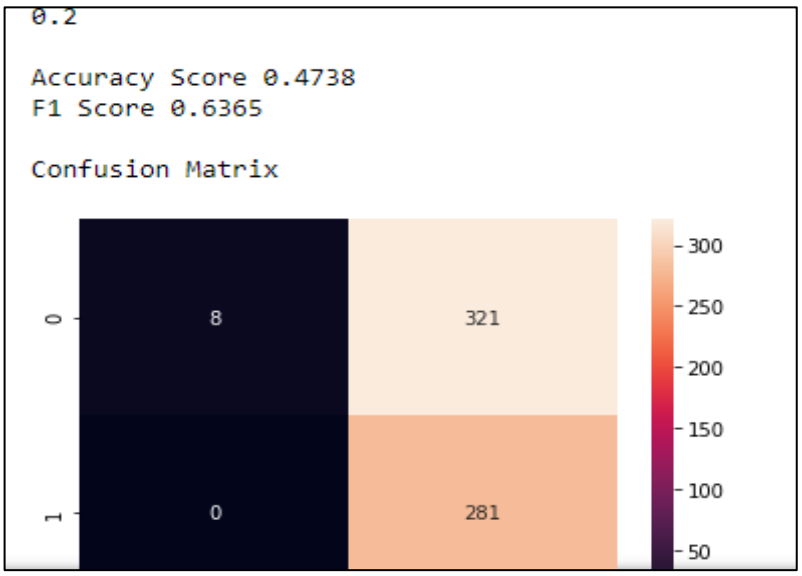


Figure 29: Confusion Matrix for 0.2 Cutoff.

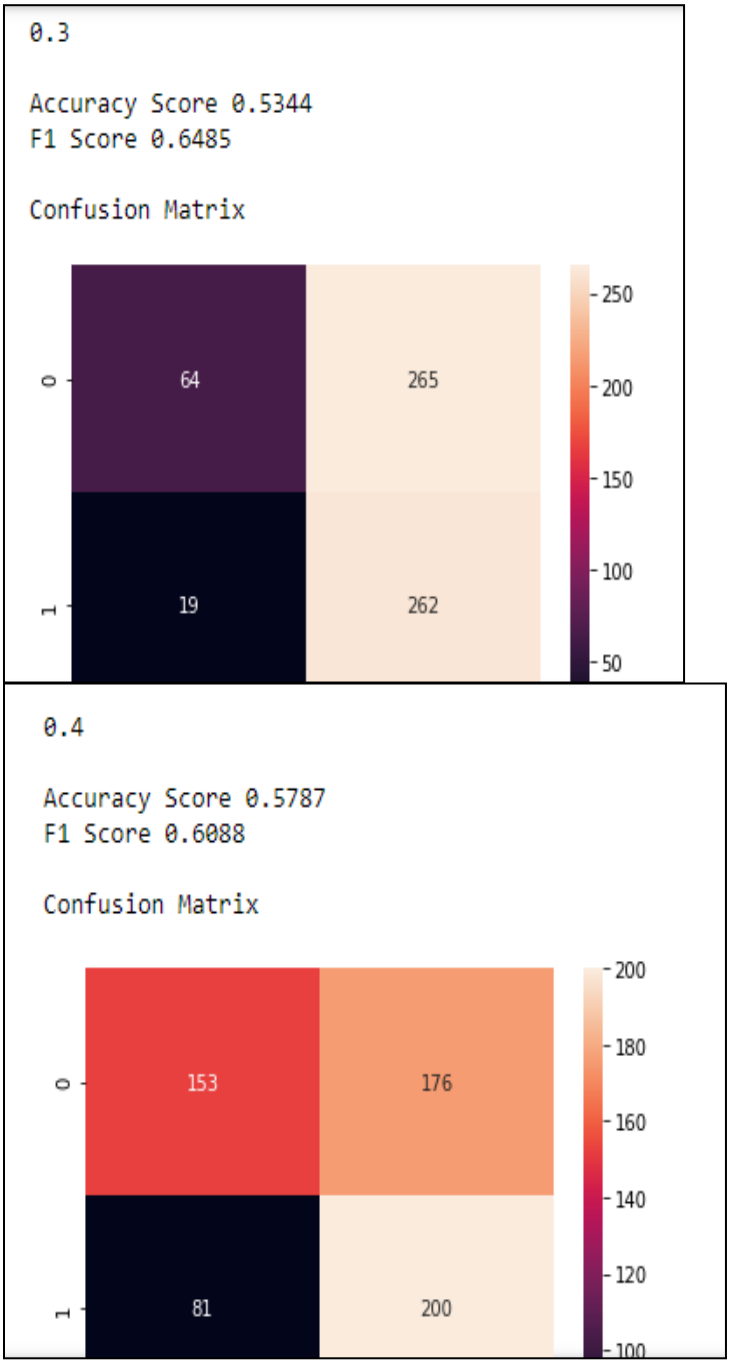


Figure 30:Confusion Matrix for 0.3 & 0.4 Cutoff.

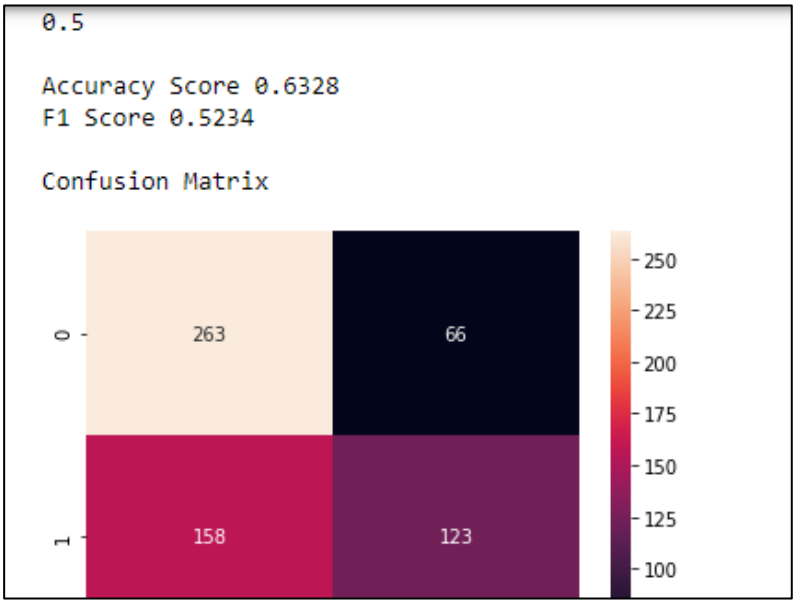


Figure 31: Confusion Matrix for 0.5 Cutoff.

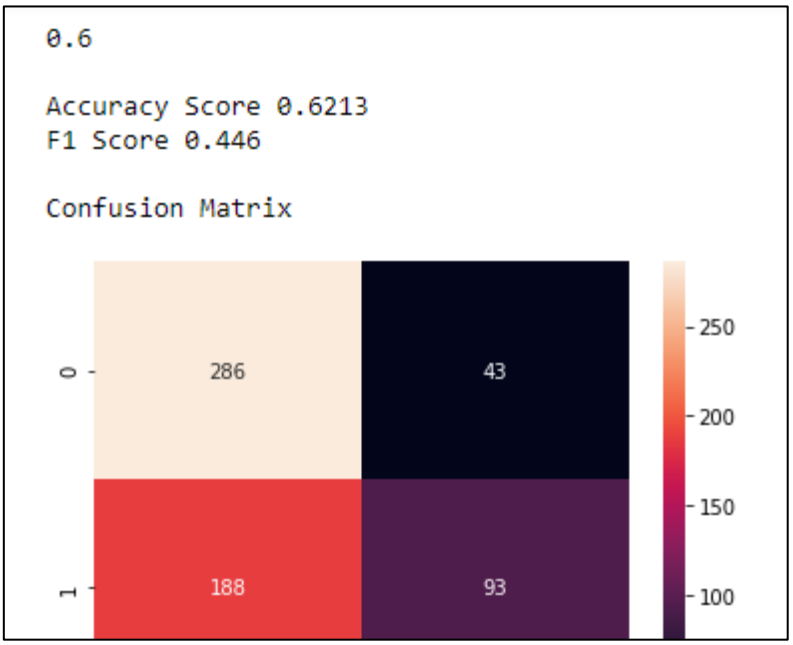
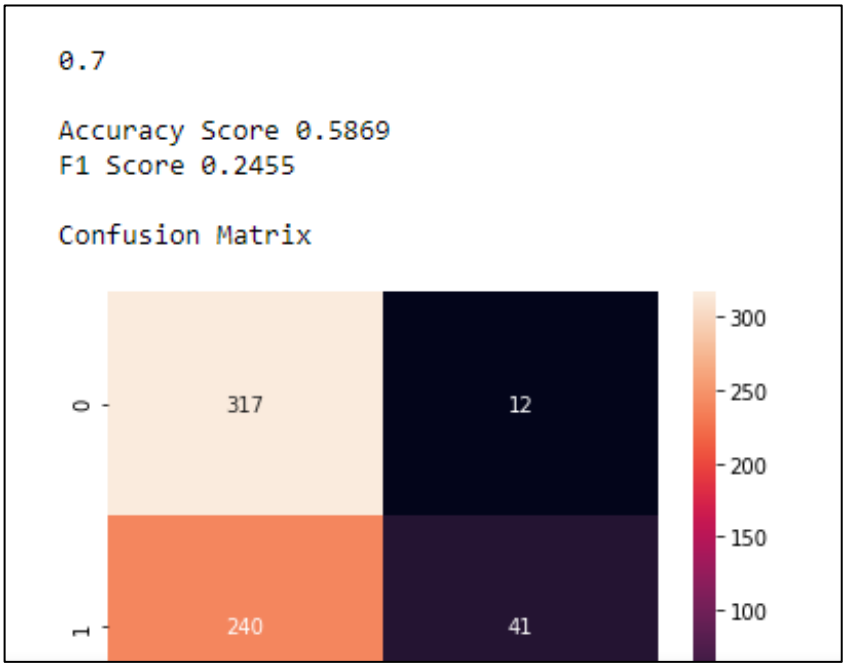


Figure 32: Confusion Matrix for 0.6 Cutoff.



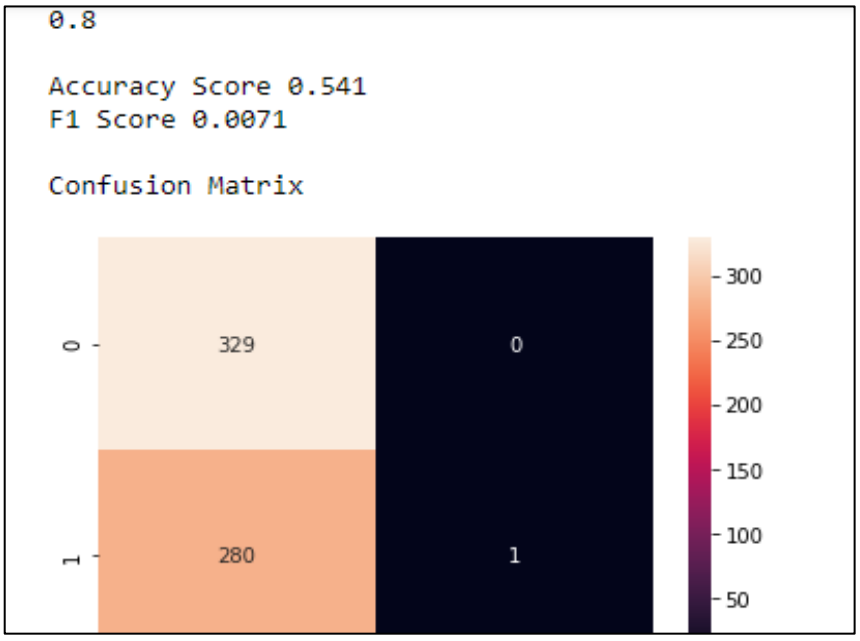


Figure 33: Confusion Matrix for 0.7 & 0.8 Cutoff.

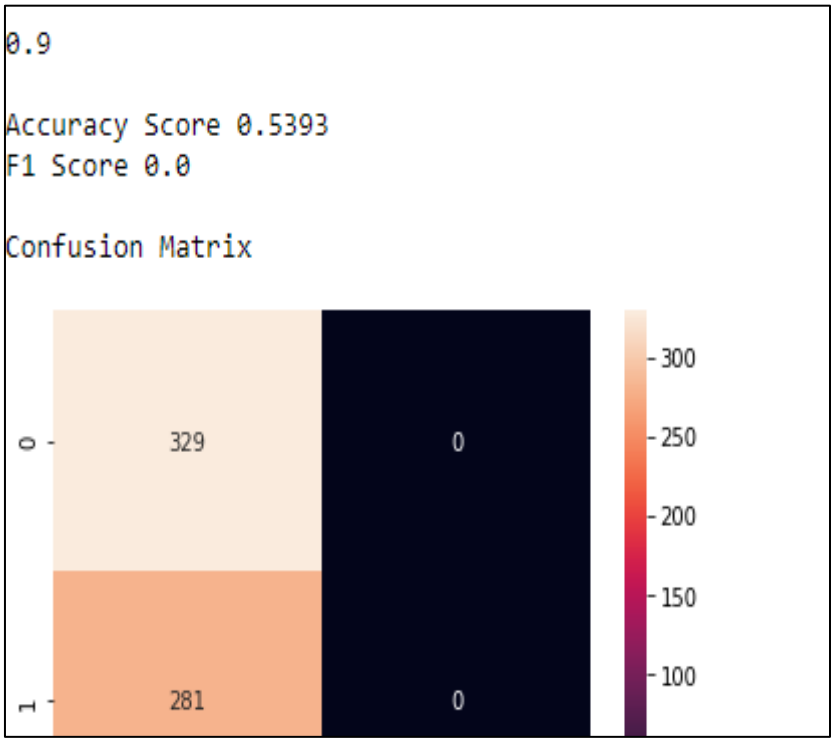


Figure 34: Confusion Matrix for 0.9 Cutoff.

AUC and ROC score for training and Test data:

Figure 35: AUC and ROC score for training and Test data(LDA)

Performance Matrices of both model:

	LR Train	LR Test	LDA Train	LDA Test
Accuracy	0.63	0.66	0.63	0.66
AUC	0.66	0.68	0.66	0.68
Recall	0.45	0.45	0.44	0.45
Precision	0.65	0.69	0.65	0.69
F1 Score	0.53	0.55	0.52	0.55

Table 15: Performance Matrices

Inference:

- **When we compare the both model, performance metrics reacts in similar way.**

2.4 Basis on these predictions, what are the insights and recommendations.Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present.

- We performed both Logestic regression and LDA model, all performace metrices reacts in similar way. We need additional data to predict better.
- In EDA it clearly shows age 30 to 50 opted for Holiday packages. But in age 50 to 60 employes are not opted for holiday packages.
- People of a lower age even from a lower salary bracket choose holiday more than people from a higher age group.
- Salary higher than 150000 are not opted for holiday packages. so we need to deep dive to get better solution. Basically aged person not opted for holiday packages so we need to offer some destination place. We need discuss with those employee to get better idea.