

# PREDICTIVE MODELING (LINEAR, LOGISTIC & LDA)

Mohamed Rifaz Ali K S  
PGP-DSBA Online  
November' 21

## Contents

Problem 1: Linear Regression .....	7
Executive Summary.....	7
Introduction .....	7
Data Description .....	8
Sample of the Salary Data dataset.....	8
Let us check the types of variables in the data frame .....	9
Check for missing values in the dataset.....	9
Check for the duplicate records in the dataset.....	10
Summary of the Dataset .....	10
Categorical columns Inspection .....	11
Observations for Cubic Zirconia dataset.....	11
Boxplot to identify the Outliers for predictor variables.....	12
Histograms and Boxplots .....	13
Skewness.....	16
Observations from Histograms and Boxplots .....	16
Histogram for the dependent variable 'price' .....	17
Categorical variable graphs.....	17
Linear relationship graphs .....	22
Pair plot and Correlation Plot: .....	23
Heat Map .....	24
Null values check.....	25
Zero values check.....	25
Combining Sub levels .....	26
Considerations before training the model.....	27
Linear regression using sklearn.....	29
Linear regression using statsmodels library.....	31
Actual Price Vs Predicted Price .....	33

Checking the Linear Regression Assumptions .....	34
Assumption 1: No Multicollinearity .....	34
Summary with and without multicollinearity variable(depth) .....	36
Assumption 2: Mean of the residuals should be 0 .....	36
Assumption 3: No Heteroscedasticity.....	36
Assumption 4: Linearity of variables.....	37
Assumption 5: Normality of error terms .....	37
Summary .....	38
Recommendations and insights .....	38
Problem 2: Logistic Regression and LDA.....	40
Executive Summary.....	40
Introduction .....	40
Sample of the Salary Data dataset.....	41
Let us check the types of variables in the data frame. ....	41
Check for the duplicate and missing records in the dataset.....	42
Summary of the Dataset .....	42
Observations for Holiday Package Dataset.....	42
Histograms and Boxplots .....	43
Skewness.....	45
Observations from Histograms and Boxplots .....	45
Categorical Variables .....	45
Pair plot.....	50
Heat Map .....	51
Checking the Data types .....	52
Converting object data types to categorical codes.....	52
Target variable percentage for Train & Test.....	54
Building the Logistic Regression Model .....	54
Grid Search to find out the optimal hyper parameters for Logistic Model training set .....	54

Co-efficient for the Logistic Model .....	55
Linear Discriminant Analysis Model(LDA) .....	56
Building the Linear Discriminant Analysis Model.....	56
Logistic Regression – Check the performance on Training and Test dataset .....	57
Logistic Regression Classifier Model Predictions .....	57
LDA – Check the performance on Training and Test dataset .....	59
ROC Curve for Logistic and LDA models.....	61

## List of Figures

Figure1. Source: <a href="https://www.diamonds.pro/education/diamond-depth-and-table/">https://www.diamonds.pro/education/diamond-depth-and-table/</a> .....	7
Figure2. Boxplot for all the predictors .....	12
Figure3. Boxplot for all the predictors .....	16
Figure4. Histogram for price .....	17
Figure5. Countplot for cut variable in order .....	17
Figure6. Countplot for color variable in order .....	18
Figure7. Countplot for clarity variable in order .....	18
Figure8. Countplot for cut Vs clarity variables in order .....	19
Figure9. Stacked bar for color Vs clarity variables in order .....	20
Figure10. Average price for cut.....	20
Figure11. Average price for color .....	21
Figure12. Average price for clarity.....	21
Figure13. Linear relationship graphs between predictors and response variable .....	22
Figure 14. Pair plot for all the numerical variables.....	23
Figure 15. Heat Map for all the numerical variables .....	24
Figure 16. Standardized box plots to understand the outliers .....	28
Figure 17. Price and log price Histograms .....	28
Figure 18. Model coefficients .....	30
Figure19. Statsmodel(lm1) with all the X's .....	32

Figure 20. Actual Vs Predicted Price .....	33
Figure 21. Q-Q plot .....	34
Figure 23. Statsmodel without depth variable .....	35
Figure 24. Residual Vs Fitted plots .....	37
Figure 25. Normality of errors .....	37
Figure 26. Data information .....	42
Figure 27. Histogram and Box plot for numerical columns .....	44
Figure 28. Swarmplot for Salary Vs Holiday Package .....	45
Figure 29. Swarmplot for Age Vs Holiday Package .....	46
Figure 30. Barplot for Age Category Vs Holiday Package .....	48
Figure 31. Swarmplot for education/young/old children Vs Holiday Package .....	48
Figure 32. Swarmplot for young/old children Vs Age group Vs Holiday Package .....	49
Figure 36. Pair plot to see the correlation between the variables .....	50
Figure 37. Heat Map to see the correlation between the variables .....	51
Figure 38. Data Types .....	52
Figure 39. Verify data types after conversion .....	52
Figure 40. Coefficients by bar graph .....	56
Figure 41. ROC Curve for Logistic and LDA Models .....	61

## List of Tables

Table 1. Cubic Zirconia Data Description .....	8
Table 2. Cubic Zirconia Sample Dataset .....	8
Table 3. Cubic Zirconia Datatypes .....	9
Table 4. Cubic Zirconia Datatypes .....	9
Table 5. Summary of the Cubic Zirconia Dataset .....	10
Table 6. Stone counts based on the object values .....	11
Table 7. Outlier Proportions .....	12
Table 8. Null values check .....	25

Table9. Zero values check .....	25
Table10. Combining clarity sublevel table .....	26
Table11. Hard coding the ordinal values .....	26
Table12. Cubic Data types.....	27
Table13. Cubic Dataset summary to understand the outliers .....	27
Table14. Models score and RMSE value using sklearn .....	31
Table15. Models score and RMSE value using statsmodel.....	33
Table16. VIF values for the predictors .....	35
Table17. RMSE values .....	36
Table18. Holiday package Data Description .....	40
Table 19.Holiday Package Dataset Sample .....	41
Table 20.Holiday Package Data types .....	41
Table 21.Holiday Package Dataset Summary.....	42
Table 22.Holiday Package Outlier proportion.....	43
Table 23.Employees Vs Holiday Package proportion.....	47
Table 24.Grouping Employees by age range.....	47
Table 25.Sample Dataset after encoding .....	53
Table 26.Holiday Package Split percentage .....	54
Table 27.Attributes Coefficients .....	55
Table 28.Comparison of Logistic and LDA Models.....	61

## Problem 1: Linear Regression

### Executive Summary

A Gem Stones co Ltd, which is a cubic zirconia(which is an inexpensive diamond alternative with many of the same qualities as a diamond) manufacturer and the company is earning different profits on different prize slots. The company has provided us the approximately 27000 cubic zirconia details. Based on the stone details, they would like to distinguish the higher profitable and lower profitable stones. Also, they also wanted to understand which 5 attributes are most important for cubic zirconia to gain more profits.

### Introduction

The purpose of this problem is to 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. Using 'Linear Regression' technique, we will predict the price for the stone on the bases of the stone details.

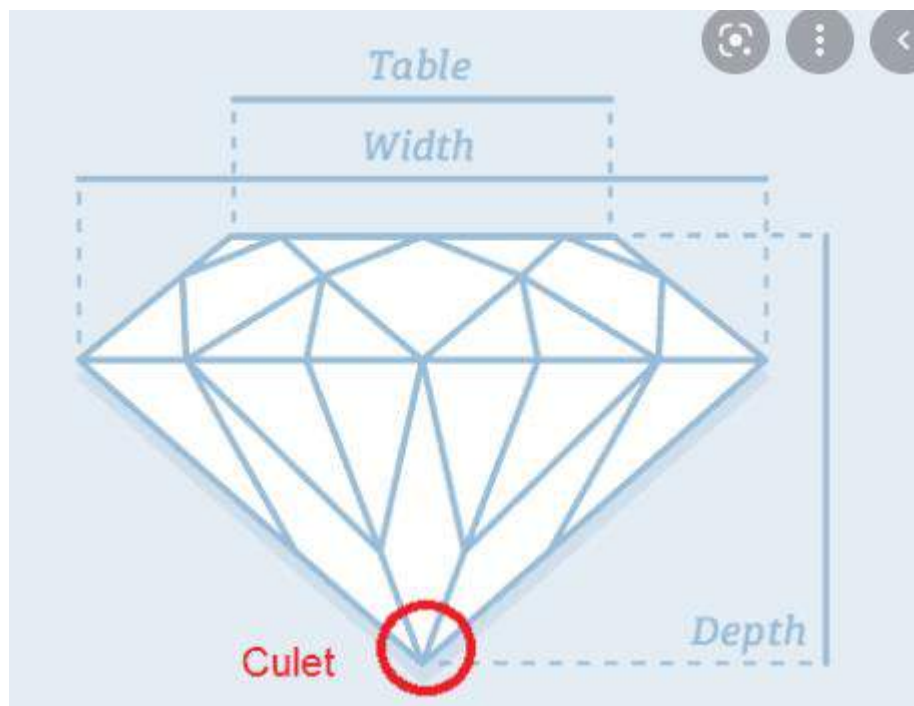


Figure1. Source: <https://www.diamonds.pro/education/diamond-depth-and-table/>

Linear Regression is a part of supervised learning. Linear regression is used for finding linear relationship between target and one or more predictors.

## 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.

There are 26967 records and 10 columns present in the cubic zirconia provided dataset.

### 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	Clarity refers to the absence of the Inclusions and Blemishes. (In order from Worst to Best in terms of avg price) 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.

Table 1. Cubic Zirconia Data Description

### Sample of the Salary Data 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 2. Cubic Zirconia Sample Dataset



Let us check the types of variables in the data frame

Variable Name	Data Types
Unnamed: 0	int64
Carat	float64
Cut	object
Color	object
Clarity	object
Depth	float64
Table	float64
X	float64
Y	float64
Z	float64
Price	int64

*Table 3. Cubic Zirconia Datatypes*

We can drop the column 'Unnamed: 0' as it's just a indexing to the records.

Check for missing values in the dataset

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 26933 entries, 0 to 26966
Data columns (total 10 columns):
#   Column    Non-Null Count  Dtype
---  ---
0   carat      26933 non-null  float64
1   cut        26933 non-null  object
2   color      26933 non-null  object
3   clarity    26933 non-null  object
4   depth      26236 non-null  float64
5   table      26933 non-null  float64
6   x          26933 non-null  float64
7   y          26933 non-null  float64
8   z          26933 non-null  float64
9   price      26933 non-null  int64
dtypes: float64(6), int64(1), object(3)
memory usage: 2.3+ MB
```

*Table 4. Cubic Zirconia Datatypes*

- From the above results we can see that the 'depth' column has quite some missing values.
- There are 697 missing values present in 'depth' column.
- Depth is nothing but the height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter. Thus, we will impute it with the median values (i.e., 61.8 with the missing values).

## Check for the duplicate records in the dataset

Among the 26967 records, there are 34 duplicate records present in the Dataset. i.e., 0.1%, we can simply drop them from our further analysis.

## Summary of the Dataset

	count	mean	std	min	25%	50%	75%	max
carat	26933.0	0.798010	0.477237	0.2	0.40	0.70	1.05	4.50
depth	26236.0	61.745285	1.412243	50.8	61.00	61.80	62.50	73.60
table	26933.0	57.455950	2.232156	49.0	56.00	57.00	59.00	79.00
x	26933.0	5.729346	1.127367	0.0	4.71	5.69	6.55	10.23
y	26933.0	5.733102	1.165037	0.0	4.71	5.70	6.54	58.90
z	26933.0	3.537769	0.719964	0.0	2.90	3.52	4.04	31.80
price	26933.0	3937.526120	4022.551862	326.0	945.00	2375.00	5356.00	18818.00

*Table5. Summary of the Cubic Zirconia Dataset*

- x,y and z are length, width and height which will never be '0' either it can be dropped or impute it with median/mode values. There are just 9 records of it, let's drop it from our data frame.
- Most of the maximum values deviated away from the mean values, so we can say, almost outliers are present in all of the columns. However, we will confirm it during the Univariate analysis.
- There is one record in the column 'y' and 'z' is totally outstanding from the crowd ('y' has 58.9 and 'z' has 31.8) which can be imputed using corresponding median values (5.7(y) and 3.52(z) respectively).

### Categorical columns Inspection

Column Name	Values (worst to best)	Counts
Cut	Fair	779
	Good	2434
	Very Good	6027
	Premium	6880
	Ideal	10805
Color	D	3341
	E	4916
	F	4722
	G	5650
	H	4091
	I	2765
	J	1440
Clarity	IF	891
	VVS1	1839
	VVS2	2530
	VS1	4086
	VS2	6092
	SI1	6564
	SI2	4561
	I1	362

*Table6. Stone counts based on the object values*

### Observations for Cubic Zirconia dataset

- There are 26967 rows and 11 columns are present in the dataset.
- Dropped 34 duplicate records from the dataset.
- Imputed 697 values with mode value (62.0) present in the 'Depth' column.
- There are 8 anomalies records where x, y and z are 0 values. Decided to drop from the dataset.
- Upon looking at the average and minimum/maximum values, there are outliers present in the dataset.
- The column 'y' and 'z' has one record outstanding from the crowd, will be imputed with median values.
- There are no anomalies present in the categorical columns.

- From the data description, categorical columns are ordinal in nature, so we will hardcode the label encoding for these columns.
- After inspecting the every column, now we have 26925 records present in the all the variables.

## Boxplot to identify the Outliers for predictor variables

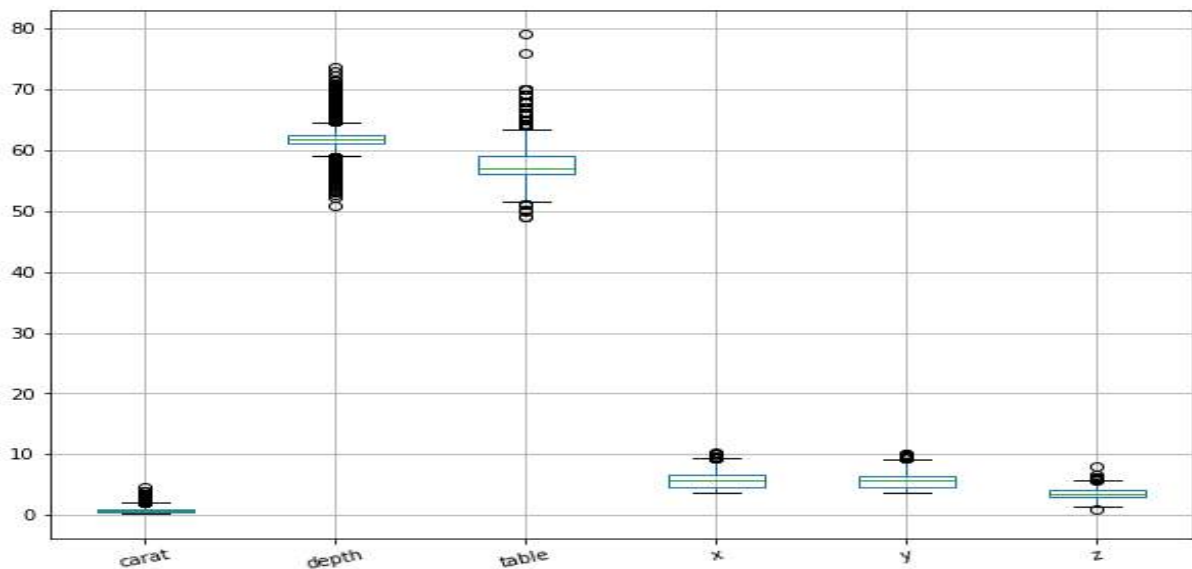


Figure2. Boxplot for all the predictors

As we presumed, all the variables have outliers.

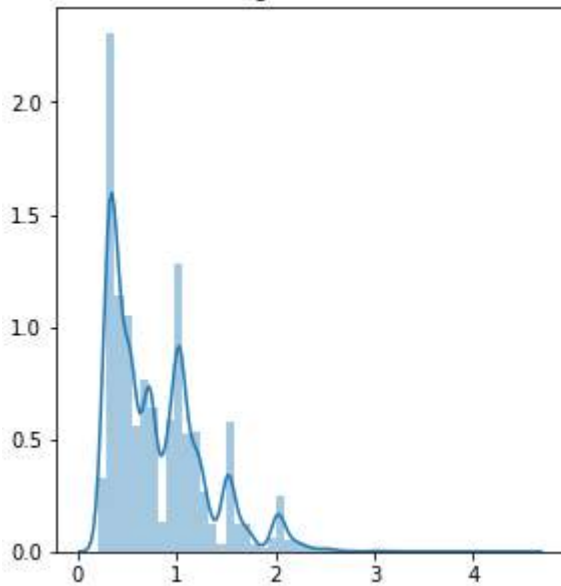
## Outliers Proportions(in %)

outlier proproction %	
carat	2.43
depth	5.24
table	1.18
x	0.04
y	0.04
z	0.05
price	6.60

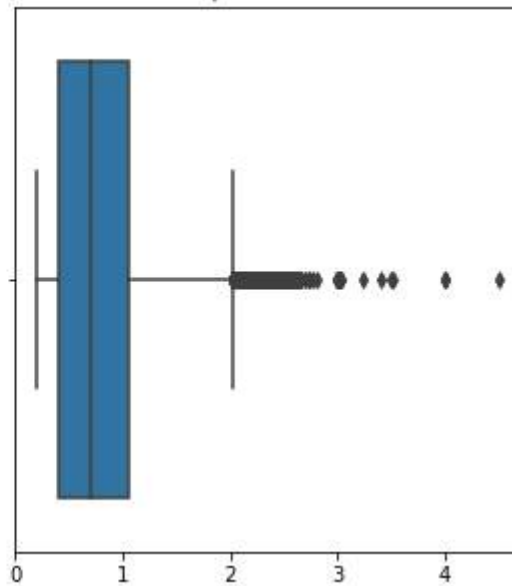
Table7. Outlier Proportions

## Histograms and Boxplots

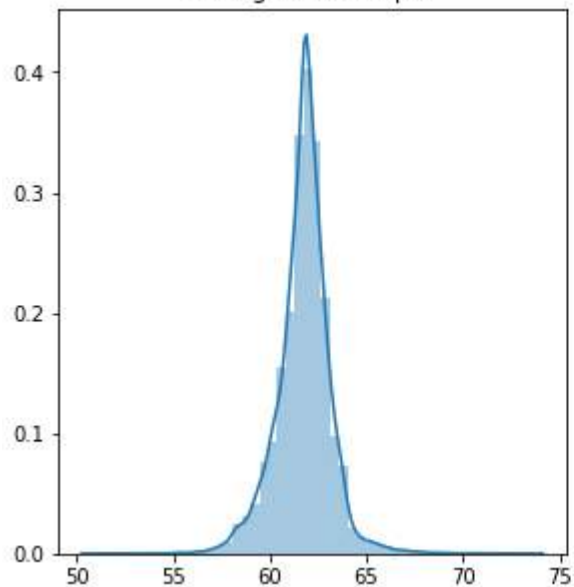
Histogram for carat



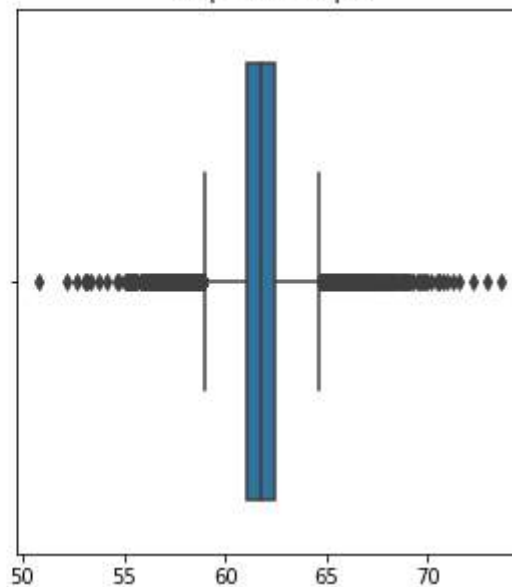
Boxplot for carat



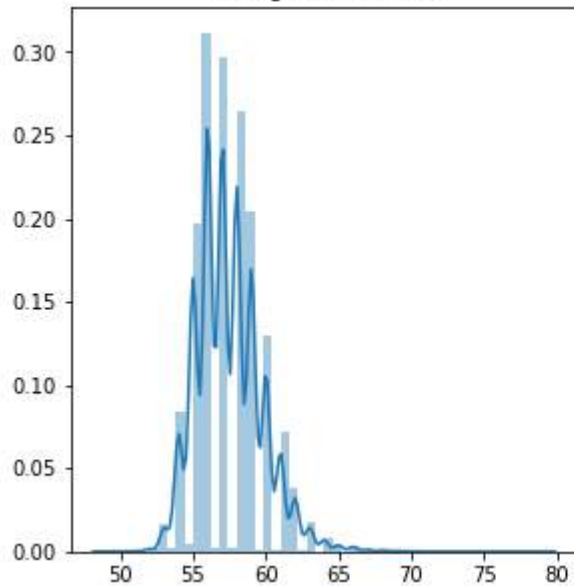
Histogram for depth



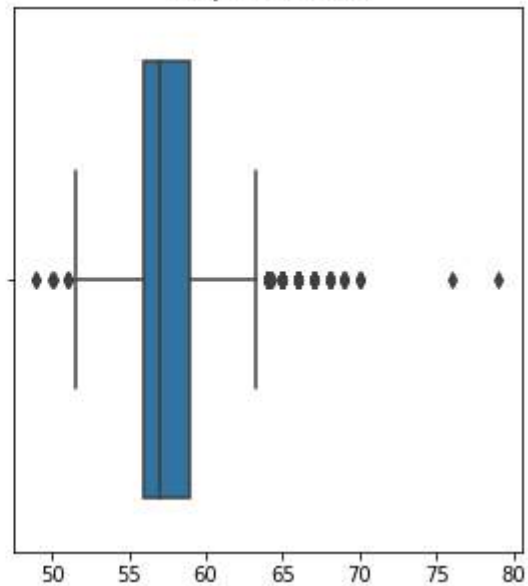
Boxplot for depth



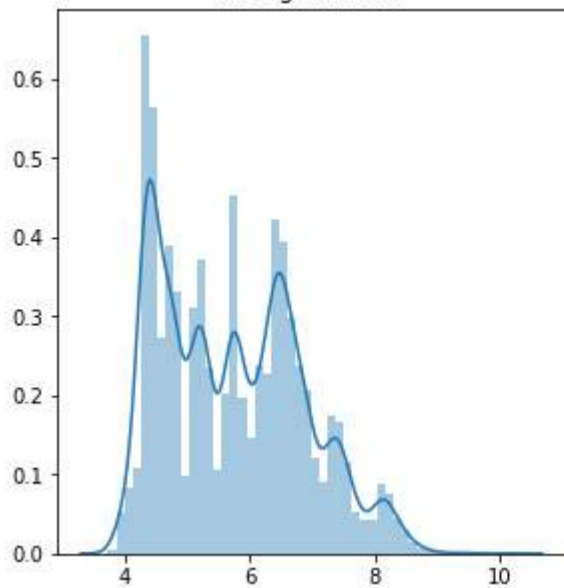
Histogram for table



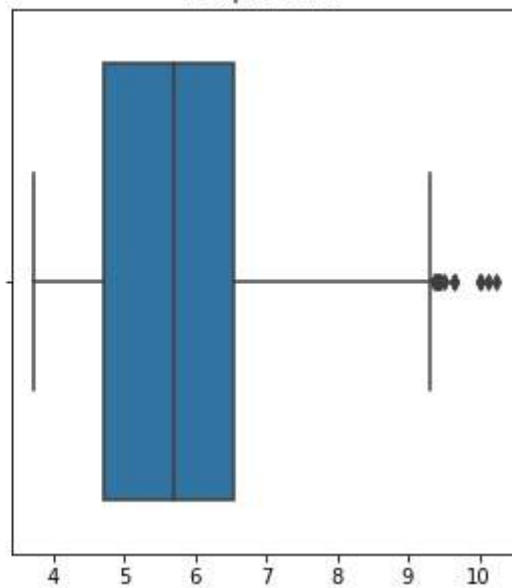
Boxplot for table



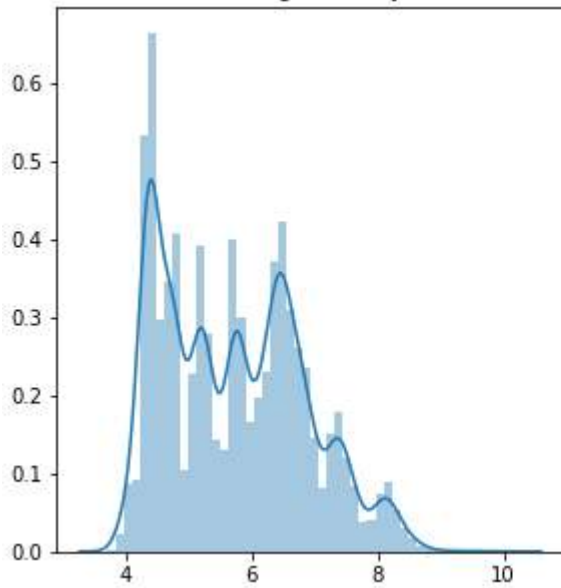
Histogram for x



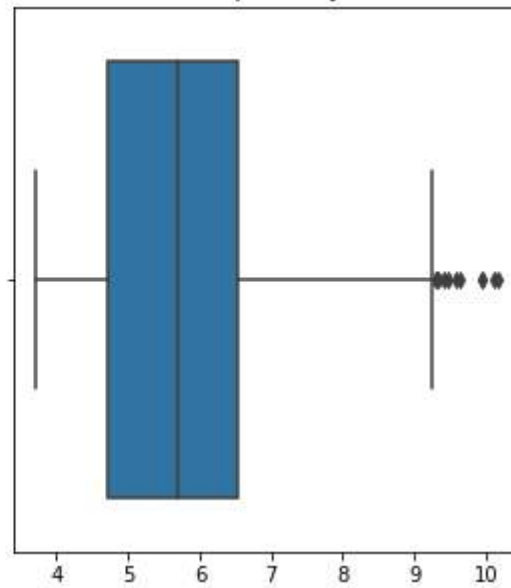
Boxplot for x



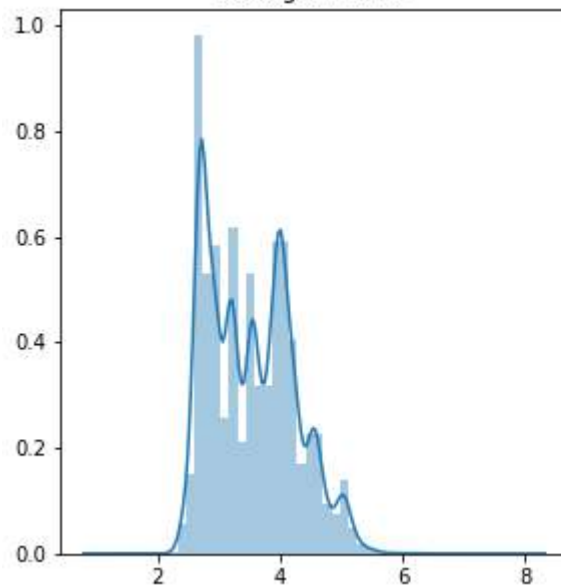
Histogram for y



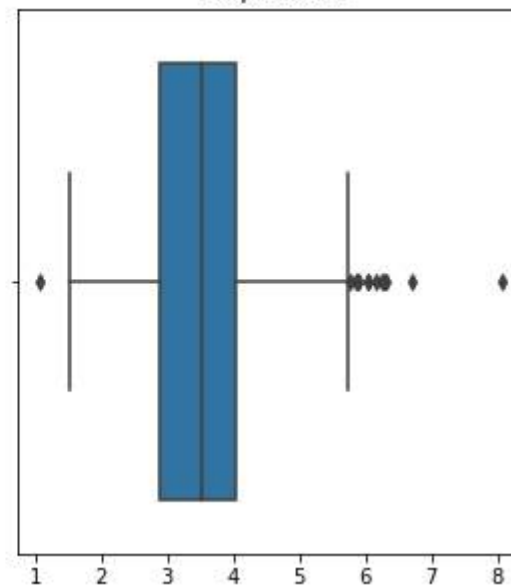
Boxplot for y



Histogram for z



Boxplot for z



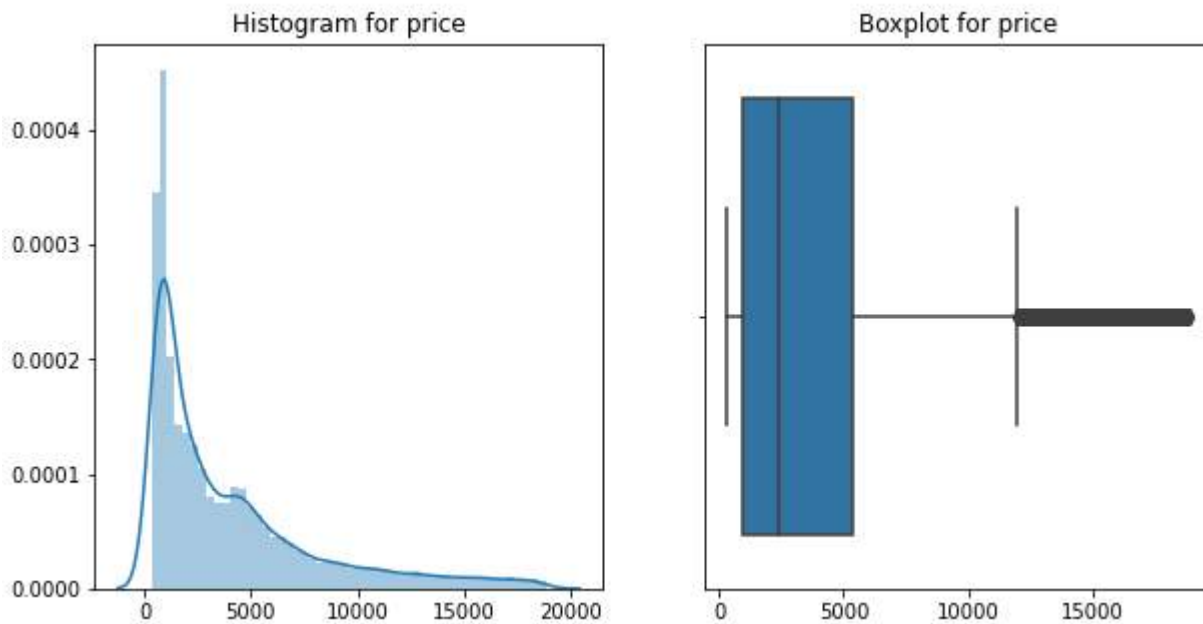


Figure3. Boxplot for all the predictors

## Skewness

Skewness for carat is: 1.11  
Skewness for depth is: -0.03  
Skewness for table is: 0.76  
Skewness for x is: 0.4  
Skewness for y is: 0.4  
Skewness for z is: 0.41  
Skewness for price is: 1.62

## Observations from Histograms and Boxplots

- The columns such as carat, table, x, y, z, price are positively skewed and their data's are clustered around the left tail of the distribution while the right tail of the distribution is longer.
- 'Depth' column is almost normally distributed.
- All the columns contain the (*valid*) outliers.
- Most of the cubic carats are falls between 0.2 and 1.5.



## Histogram for the dependent variable 'price'

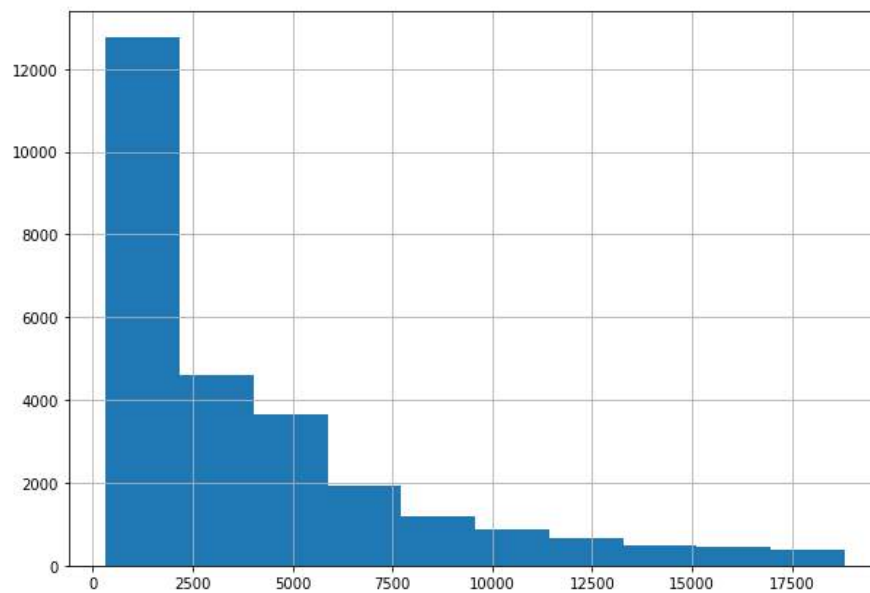


Figure4. Histogram for price

Approximately, 50% of the prices of the cubic falls less than 2500 range and 85% of the prices of the cubic falls less than 7500 range.

## Categorical variable graphs

### Cut

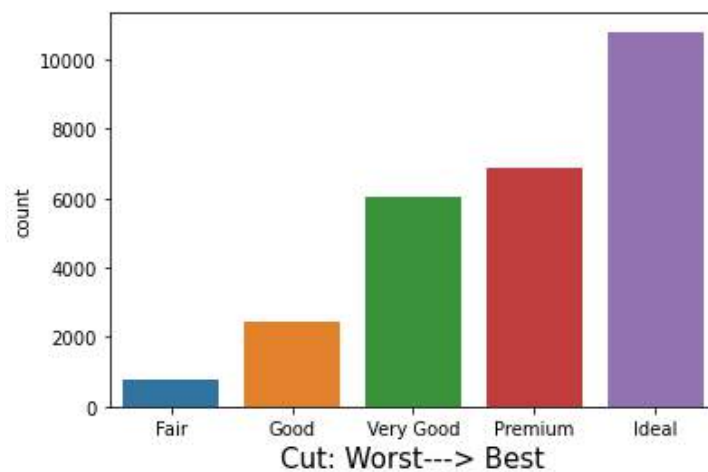
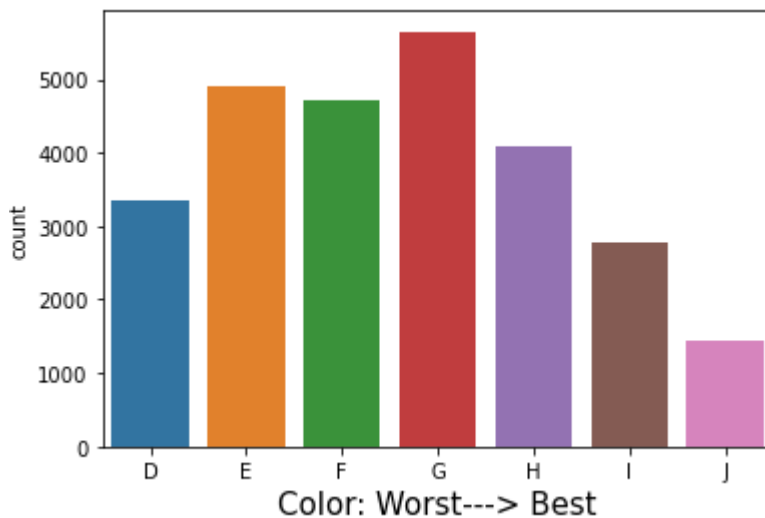


Figure5. Countplot for cut variable in order

Interesting pattern to be noted here is that there are number stones sold based on the cut quality order. So, where the manufacturer has the best cut quality sold the most number of stones.

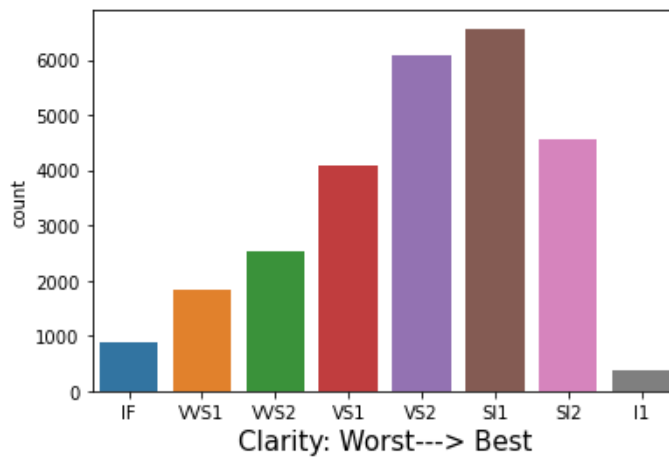
## Color



*Figure6. Countplot for color variable in order*

The manufacturer might thought the best colors would be sold more but it is not the case here that mid ranges of the colors of the stones sold more.

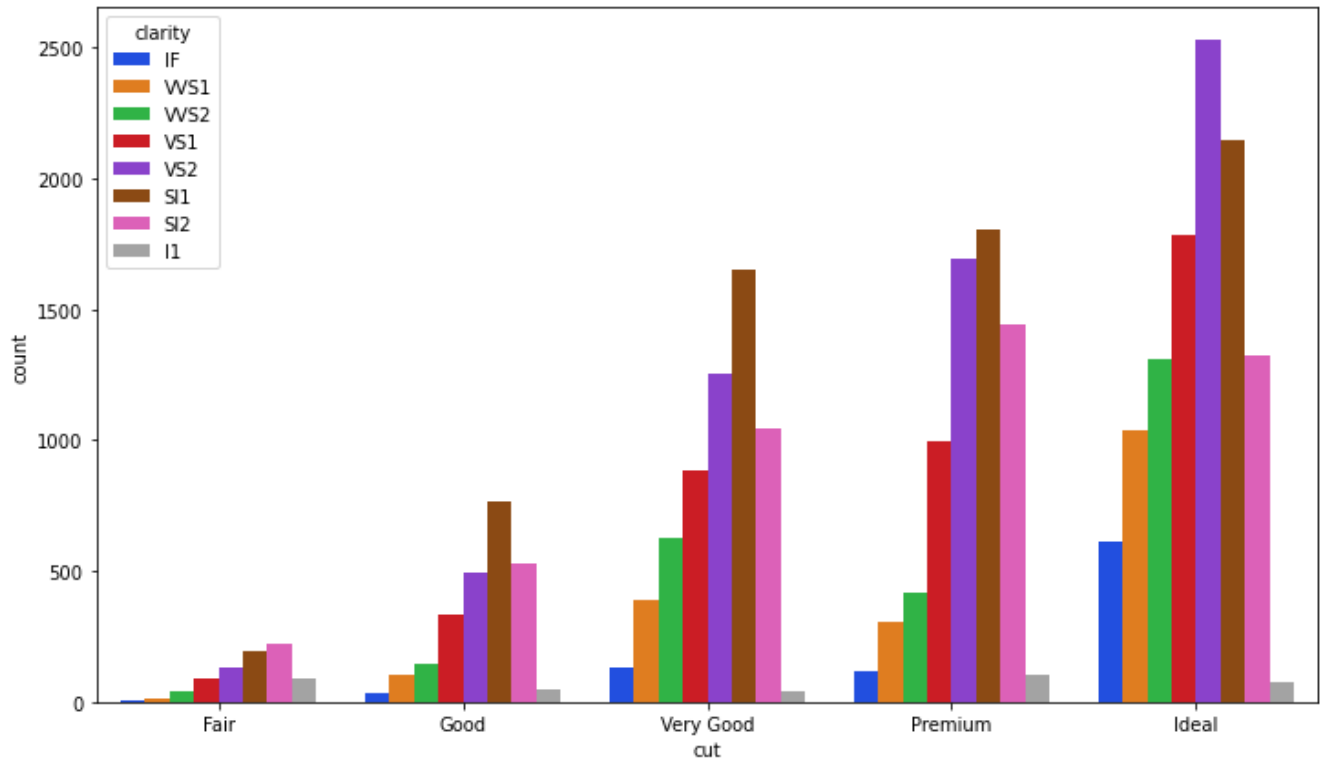
## Clarity



*Figure7. Countplot for clarity variable in order*

People like the very slightly and slightly included (inclusions/blemishes) clarity stones the most compared to the A1 carat and worst clarity stones.

## Cut and Clarity



*Figure8. Countplot for cut Vs clarity variables in order*

Very slightly included (VS1/2) and Slightly included (SI1/2) blemishes stones are the ones sold most across the cut quality.

## Color and Clarity

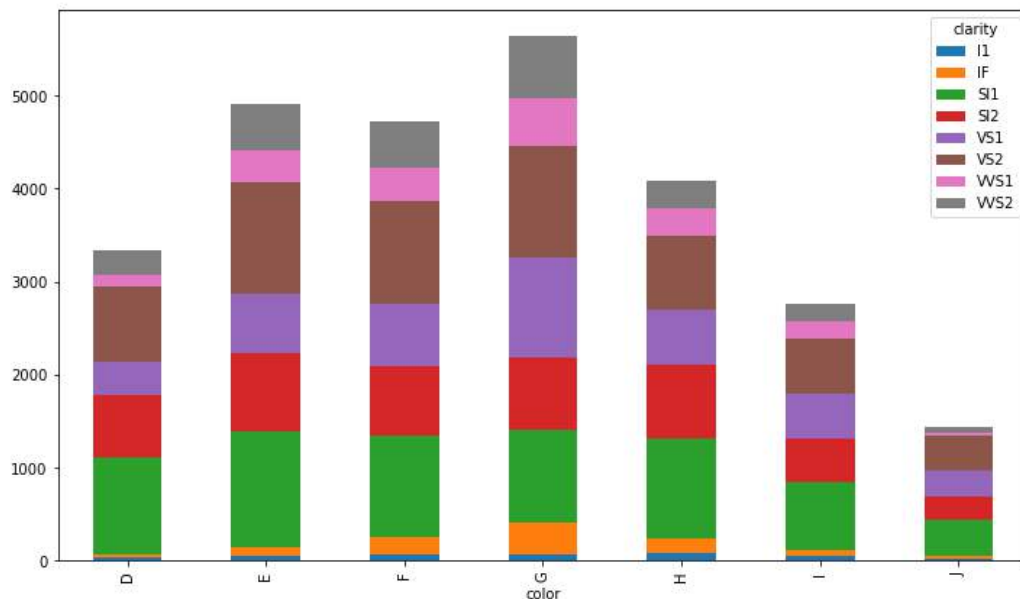


Figure9. Stacked bar for color Vs clarity variables in order

- People hardly buying the best A1 clarity (I1) and worst clarity (IF) stones.
- Also, the customers like the mid ranges clarity with mid ranges colors the most.

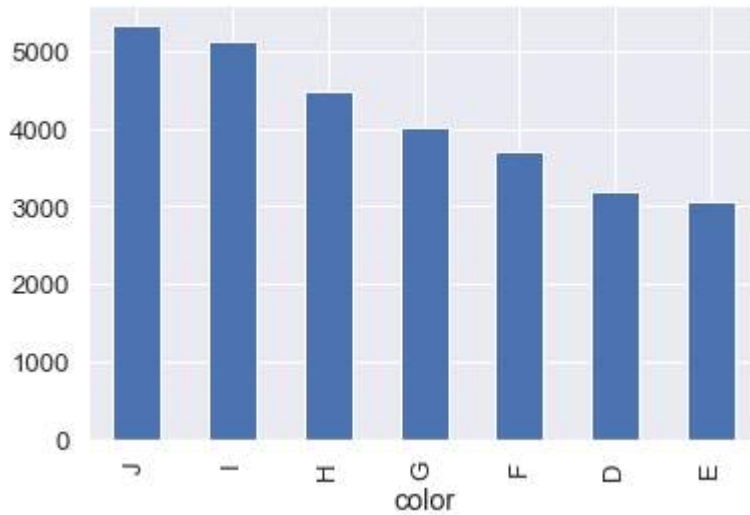
## Average price for cut



Figure10. Average price for cut

Average price of worst cut quality product (Fair) is high compared to the best cut quality as Ideal.

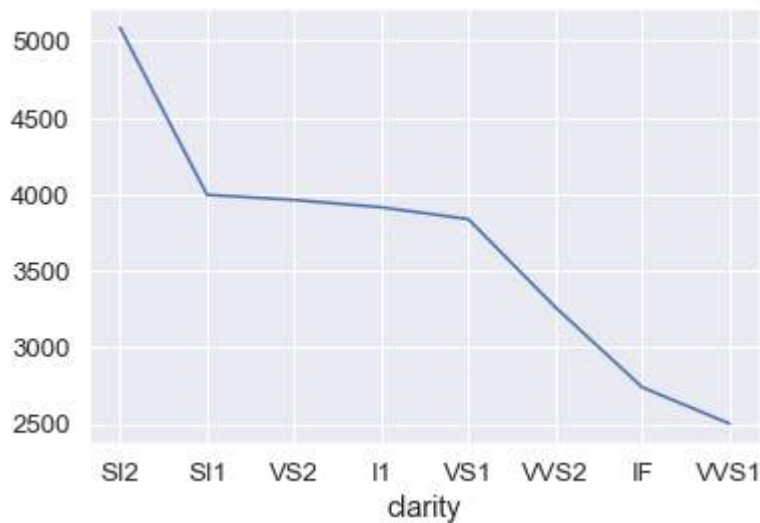
### Average price for color



*Figure11. Average price for color*

Average price of the best colors are holding the highest price.

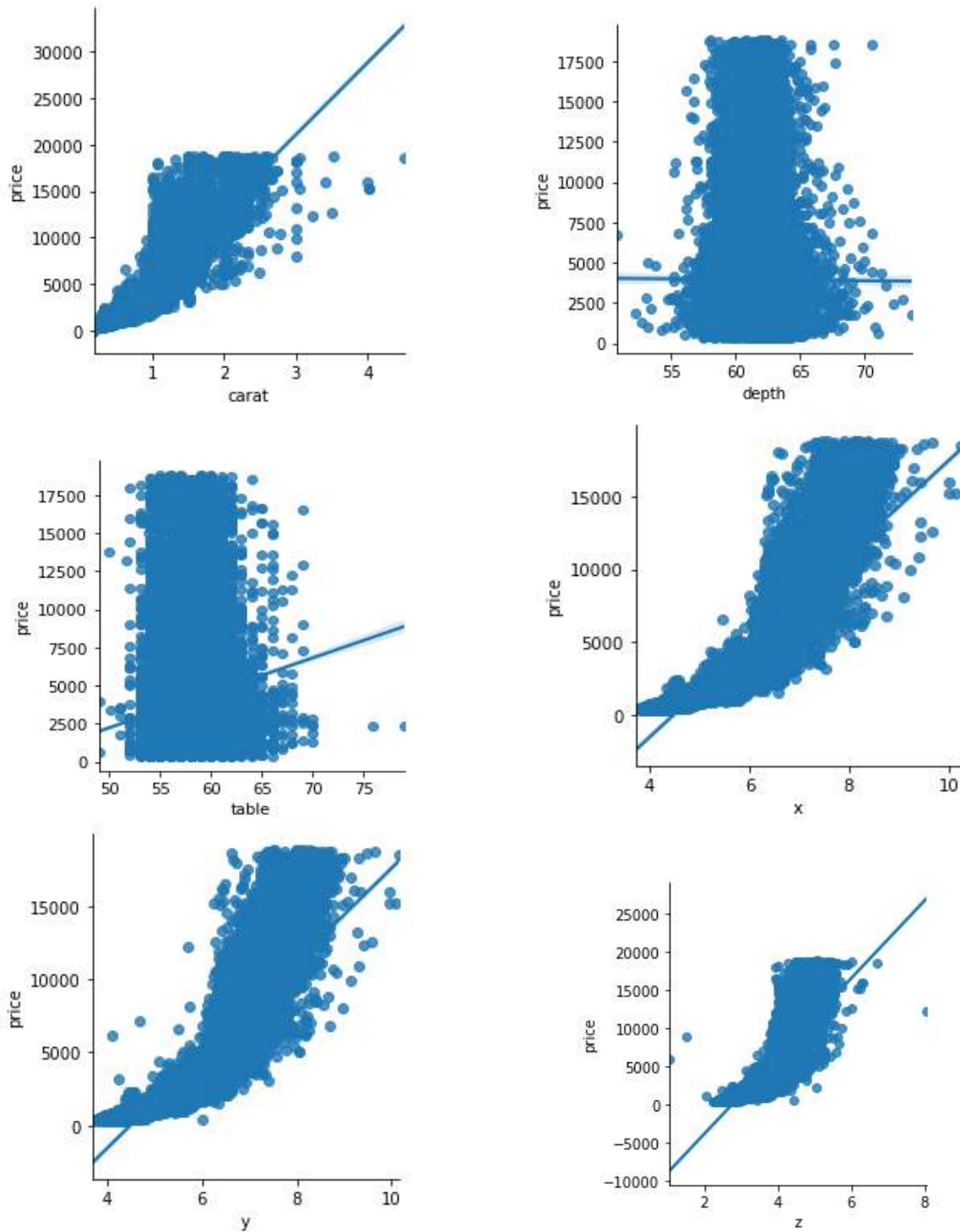
### Average price for clarity



*Figure12. Average price for clarity*

- Average price of SI2 is high compared to the other clarity values.
- SI1, VS2, I1 and VS1 are maintaining the almost similar average price.
- VVS1 average price is low compared to the worst clarity IF.

## Linear relationship graphs



*Figure13.Linear relationship graphs between predictors and response variable*

- From the LM plots, we could see that there is a positive linear relationship between the predictors such as carat, x, y, z and the dependent variable price.
- There is no linear relationship for table and depth variables.

## Pair plot and Correlation Plot:

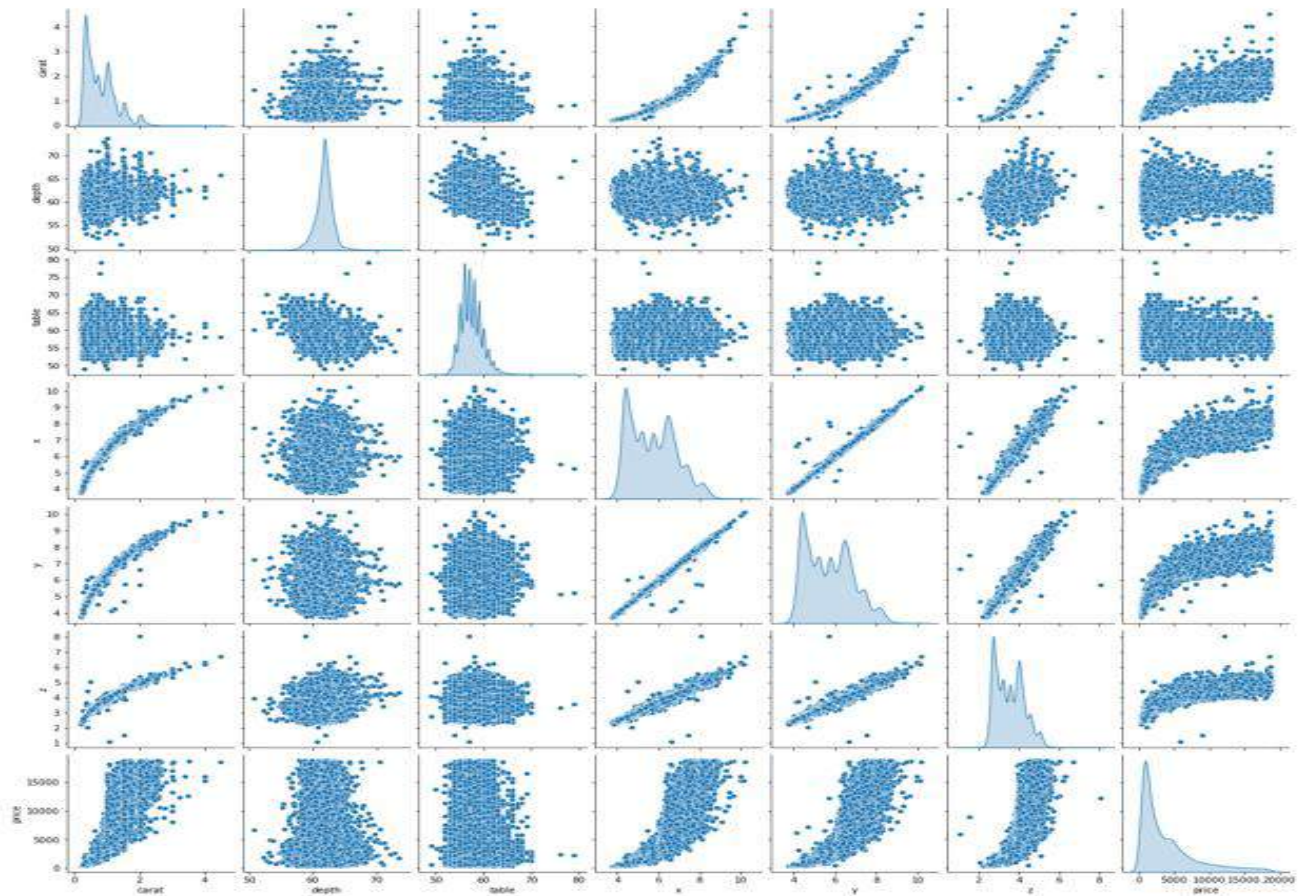


Figure 14. Pair plot for all the numerical variables

## Pair plot Inferences

- Depth and Table has no linear relationship between the other variables.
- Carat has positive relationship with x,y,z and price variables.
- Based on the plots, carat, x, y and z are good predictors for price.
- In general, length, weight and height must be having good linear relationship and we could see it in the scatter plots of x, y and z.







**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.**

## Null values check

- There are 697 null values present in the depth column, other columns does not have any null values. We will impute the depth missing values with median.
- The median value for depth is 61.8

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

*Table8. Null values check*

## Zero values check

	carat	cut	color	clarity	depth	table	x	y	z	price
5821	0.71	Good	F	SI2	64.1	60.0	0.00	0.00	0.0	2130
6034	2.02	Premium	H	VS2	62.7	53.0	8.02	7.95	0.0	18207
10827	2.20	Premium	H	SI1	61.2	59.0	8.42	8.37	0.0	17265
12498	2.18	Premium	H	SI2	59.4	61.0	8.49	8.45	0.0	12631
12689	1.10	Premium	G	SI2	63.0	59.0	6.50	6.47	0.0	3696
17506	1.14	Fair	G	VS1	57.5	67.0	0.00	0.00	0.0	6381
18194	1.01	Premium	H	I1	58.1	59.0	6.66	6.60	0.0	3167
23758	1.12	Premium	G	I1	60.4	59.0	6.71	6.67	0.0	2383

*Table9. Zero values check*

- There are just 9 records where the columns such as x or y or z are having 0 values. It does not make any sense where the stones are with no length or width or height.
- We can either drop or impute it with median values.
- There is no harm dropping these records from the dataset as it's very low with the total records. It is just approximately 0.02%.

## Combining Sub levels

- On the cut column, there is a possibility to combine 'good' and 'very good' cut quality into one value because there is not much difference between the 'good' quality cut and 'very good' quality cut. Additionally, we have the premium and ideal cut quality to describe the very high quality cut.
- After combining the color sublevels, we reduced the levels from 5 to 4.
- On the clarity column, we have IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1 clarities of the stones which describes the inclusion and blemishes present in the cubic zirconia.
- After combining the clarity sublevels, we reduced the levels from 8 to 5.

Clarity levels	Merging values
VVS1, VVS2	VVS(Very Very Slightly included)
VS1, VS2	VS (Very Slightly included)
SI1, SI2	SI (Slightly included)

*Table10. Combining clarity sublevel table*

**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.clusters.**

We use the label encoding for the categorical ordinal values because it is mentioned in the hierarchical order.

- Cut – Any gemstone have worst to best quality cut (ordinal).
- Color - Any gemstone will look dull or dirty over time if it is not maintained and cleaned (ordinal).
- Clarity - Any gemstone have worst to best clarity based on the inclusions/blemishes (ordinal).

Let's hard code the values as shown in the below table.

Cut		Color		Clarity	
Fair	1	D	1	IF	1
Good	2	E	2	VVS	2
Premium	3	F	3	VS	3
Ideal	4	G	4	SI	4
		H	5	I1	5
		I	6		
		J	7		

*Table11. Hard coding the ordinal values*

Now, all the objects are converted to integers which are required for building the model.

```
carat      float64
cut         int64
color       int64
clarity     int64
depth       float64
table       float64
x           float64
y           float64
z           float64
price       int64
```

*Table12. Cubic Data types*

Considerations before training the model

	count	mean	std	min	25%	50%	75%	max
<b>carat</b>	26925.0	0.797821	0.477085	0.20	0.40	0.70	1.05	4.50
<b>cut</b>	26925.0	3.029192	0.911291	1.00	2.00	3.00	4.00	4.00
<b>color</b>	26925.0	3.604977	1.706043	1.00	2.00	4.00	5.00	7.00
<b>clarity</b>	26925.0	3.211625	0.846663	1.00	3.00	3.00	4.00	5.00
<b>depth</b>	26925.0	61.746982	1.393457	50.80	61.10	61.80	62.50	73.60
<b>table</b>	26925.0	57.455305	2.231327	49.00	56.00	57.00	59.00	79.00
<b>x</b>	26925.0	5.729385	1.126081	3.73	4.71	5.69	6.55	10.23
<b>y</b>	26925.0	5.731176	1.117804	3.71	4.71	5.70	6.54	10.16
<b>z</b>	26925.0	3.537770	0.696503	1.07	2.90	3.52	4.04	8.06
<b>price</b>	26925.0	3936.249991	4020.983187	326.00	945.00	2373.00	5353.00	18818.00

*Table13. Cubic Dataset summary to understand the outliers*

1. Outliers present in the current dataset is reasonable and acceptable.
2. Since the columns are in different scale, lets standardize the data only to understand the outliers presence using boxplots.

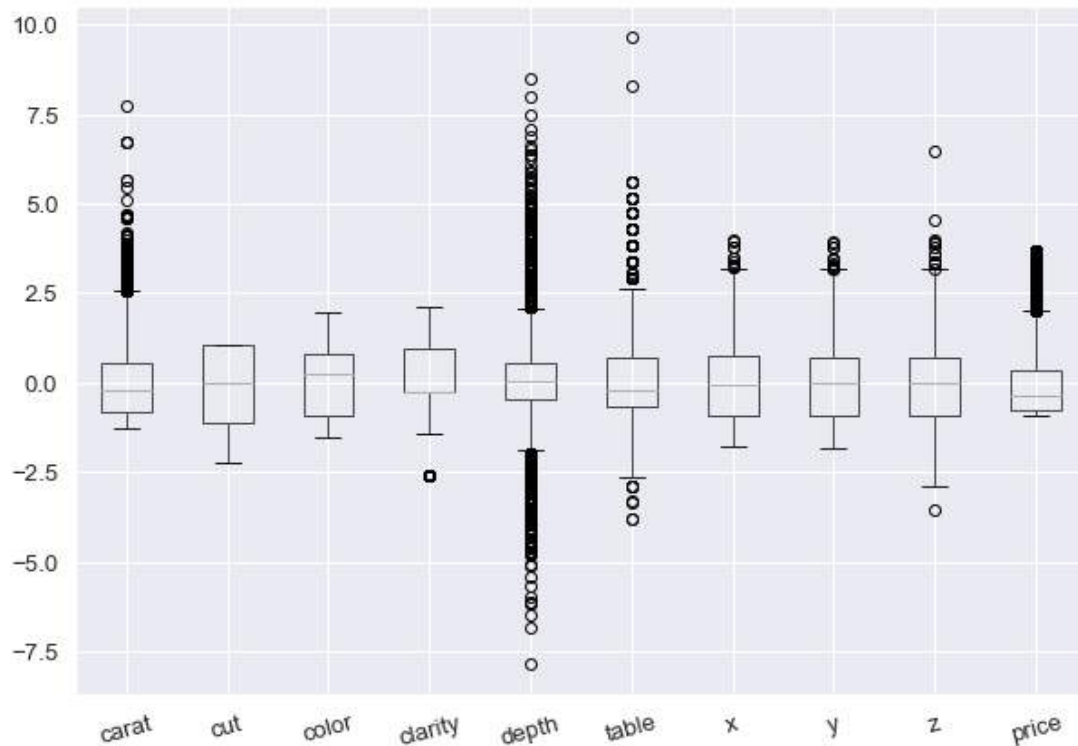


Figure 16. Standardized box plots to understand the outliers

3. Removing the outliers would increase our model's accuracy score, but that might not be best model when we deploy it in the production.
4. Based on our data analysis, let's train the model with the current reasonable outliers.
5. What we want to predict is the 'price' of the stone. We will use the normalized version of price for modelling.

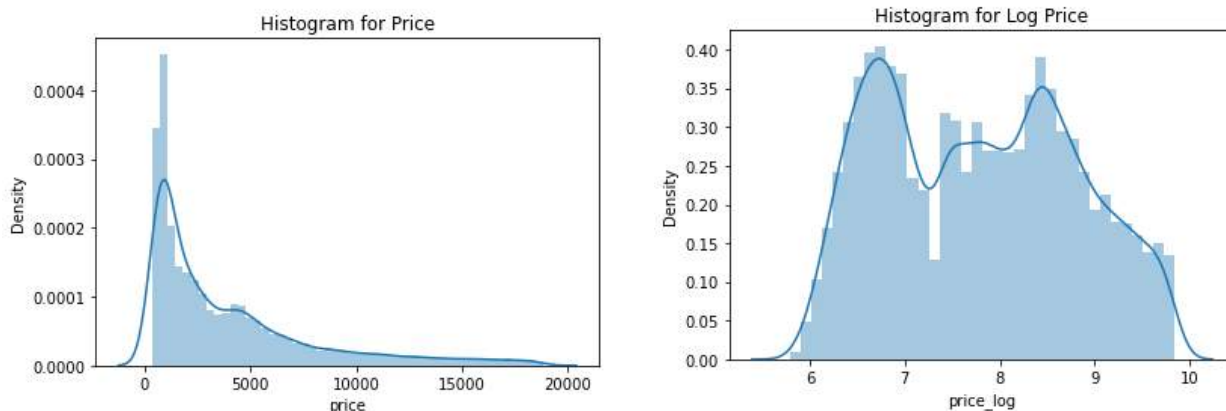


Figure 17. Price and log price Histograms

6. Though, standardizing the data would not impact the model's performance. We will use the original dataset to train the model for better interpretations.

Here, we assign independent columns to variable X and target (price) to Y.

```
X = df.drop(['price', 'price_log'], axis=1)
y = df[['price', 'price_log']]
```

Using the sklearn package, we import train\_test\_split function. Split the dataset, one for training the model and another one for test the model (unseen data by the model).

Throughout this problem, we run the model with random state=1 to be consistent across the results.

### Linear regression using sklearn

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30
, random_state=1)
```

Assign 30% of the data's to the test set and 70% for training the model.

- Train set contains 18847 records and 9 columns.
- Test set contains 8078 records and 9 columns.

Invoke the LinearRegression function and find the bestfit model on training data

```
regression_model = LinearRegression()
regression_model.fit(X_train, y_train['price_log'])
```

## Co-efficient for the columns

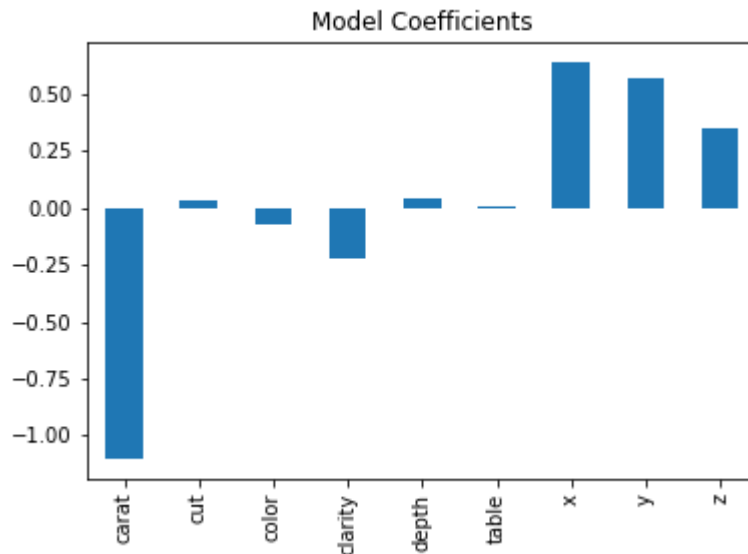


Figure 18. Model coefficients

The coefficient for carat is -1.1001  
The coefficient for cut is 0.0301  
The coefficient for color is -0.0753  
The coefficient for clarity is -0.2231  
The coefficient for depth is 0.0415  
The coefficient for table is 0.0083  
The coefficient for x is 0.6406  
The coefficient for y is 0.5669  
The coefficient for z is 0.3511

## Calculating $R^2$ score

```
regression_model.score(X_train, y_train['price_log'])  
regression_model.score(X_test, y_test['price_log'])
```

## Root Mean square

While calculating the RMSE error value, we must convert the normalized version of price column(log\_price) to back to original using exponential function(np.exp), which is the inverse of np.log()

```
rmfit=regression_model.fit(X_train, y_train['price_log'])
predicted_train=rmfit.predict(X_train)
orig_price_train=np.exp(predicted_train)
np.sqrt(metrics.mean_squared_error(y_train['price'],orig_price_train))
```

Data	Model's score	Root Mean Square
Train	0.97684	924.97477
Test	0.97680	940.95530

*Table14. Models score and RMSE value using sklearn*

## Linear regression using statsmodels library

Lets say, we collect more datas from the universe where the co-efficients between independent variables and dependent variable is mere 0 and coefficient achieved is likely by chance.

We make use of Hypothesis testing

**H0:** There is no relationship between Independent variables and Dependent variable.

**HA:** There is a relationship between Independent variables and Dependent variable.

At 95% confidence level (i.e.,  $p < 0.05$ ), we will reject the null Hypothesis.

$R^2$  is not a reliable metric as it always increases with addition of more attributes even if the attributes have no influence on the predicted variable. Instead we use adjusted  $R^2$  which removes the statistical chance that improves  $R^2$ . Scikit does not provide a facility for adjusted  $R^2$ . So, we use statsmodel, a library that gives results with more insights. This library expects the X and Y to be given in one single dataframe

```
data_train = pd.concat([X_train, y_train], axis=1)
```

```
expr='price_log ~ carat+cut+color+clarity+depth+table+x+y+z'
```

```
import statsmodels.formula.api as smf
lm1 = smf.ols(formula= expr, data = data_train).fit()
print(lm1.summary())
```

Coefficients of the predictor column remains same like we obtained using sklearn. In fact, using statsmodels library, we will get more insights for the data otherwise everything else remains the same.

```

=====
                        OLS Regression Results
=====
Dep. Variable:          price_log      R-squared:                0.977
Model:                  OLS           Adj. R-squared:           0.977
Method:                 Least Squares   F-statistic:             8.829e+04
Date:                  Tue, 23 Nov 2021   Prob (F-statistic):       0.00
Time:                  11:40:01         Log-Likelihood:          8439.1
No. Observations:      18847           AIC:                    -1.686e+04
Df Residuals:          18837           BIC:                    -1.678e+04
Df Model:              9
Covariance Type:       nonrobust
=====
                        coef      std err          t      P>|t|      [0.025      0.975]
-----
Intercept      -1.6440         0.121    -13.604     0.000     -1.881     -1.407
carat          -1.1001         0.012   -92.324     0.000     -1.123     -1.077
cut             0.0301         0.002    19.934     0.000      0.027      0.033
color          -0.0753         0.001   -107.767     0.000     -0.077     -0.074
clarity        -0.2231         0.001   -152.618     0.000     -0.226     -0.220
depth           0.0415         0.002    24.725     0.000      0.038      0.045
table           0.0083         0.001    12.862     0.000      0.007      0.010
x               0.6406         0.024    26.623     0.000      0.593      0.688
y               0.5669         0.021    26.925     0.000      0.526      0.608
z               0.3511         0.024    14.629     0.000      0.304      0.398
=====
Omnibus:                 660.173   Durbin-Watson:           2.010
Prob(Omnibus):           0.000   Jarque-Bera (JB):       1815.866
Skew:                   -0.104   Prob(JB):                0.00
Kurtosis:                4.506   Cond. No.               9.23e+03
=====

```

#### Notes:

- [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
- [2] The condition number is large, 9.23e+03. This might indicate that there are strong multicollinearity or other numerical problems.

*Figure19. Statsmodel(lm1) with all the X's*

Based on the Durbin-Watson result, we get 2.010 which imply that there is no autocorrelation.

In this case,  $R^2$  is 97.7%, meaning, 97.7% of the variance in the price is explained by the 'carat', 'cut', 'color', 'clarity', 'depth', 'table', 'x', 'y', 'z' predictors. In other words, if we know the values of the predictors, we will have 97.7% information to make an accurate prediction about its prices.

This is a clear indication that we have been able to create a very good model that is able to explain variance in price of the cubic zirconia upto approximately 98%.

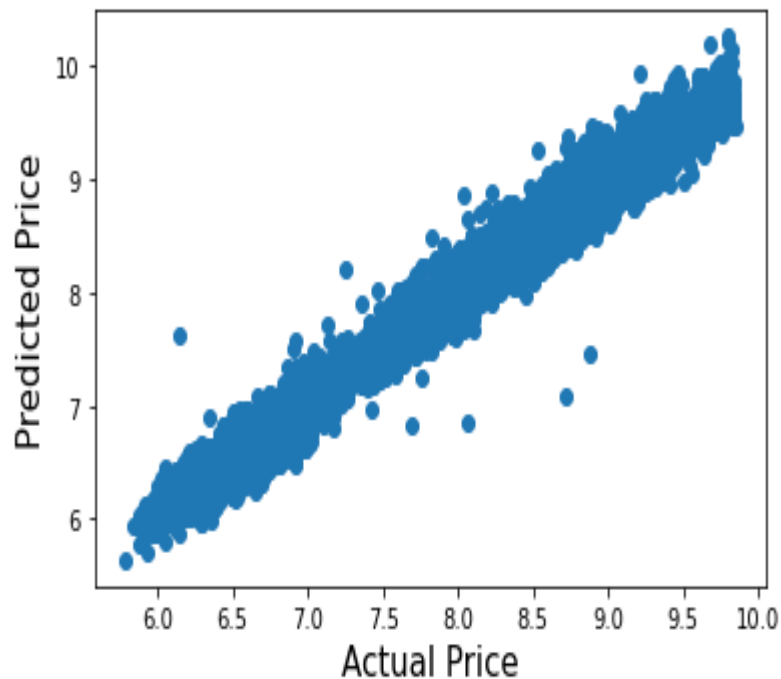


```
iv_data_test=data_test.drop(['price','price_log'],axis=1)
pred_log_price_test=lm1.predict(iv_data_test)
log_orig_price_test=np.exp(pred_log_price_test)
mse = np.mean((log_orig_price_test-data_test['price'])**2)
math.sqrt(mse)
```

Data	Model's score	Root Mean Square	MAE	MAPE
Train	0.977	924.97477	496.085777	12.199529
Test	0.97680	940.95530	500.595055	12.227167

*Table15. Models score and RMSE value using statsmodel*

## Actual Price Vs Predicted Price



*Figure 20. Actual Vs Predicted Price*

We see that there is a strong linear relationship between the actual and predicted price which indicates our model is pretty good model.

Q-Q or quantile-quantile is a scatter plot which helps us validate the assumption of normal distribution in a data set. Using this plot we can infer if the data comes from a normal distribution. If yes, the plot would show fairly straight line. Absence of normality in the errors can be seen with deviation in the straight line.

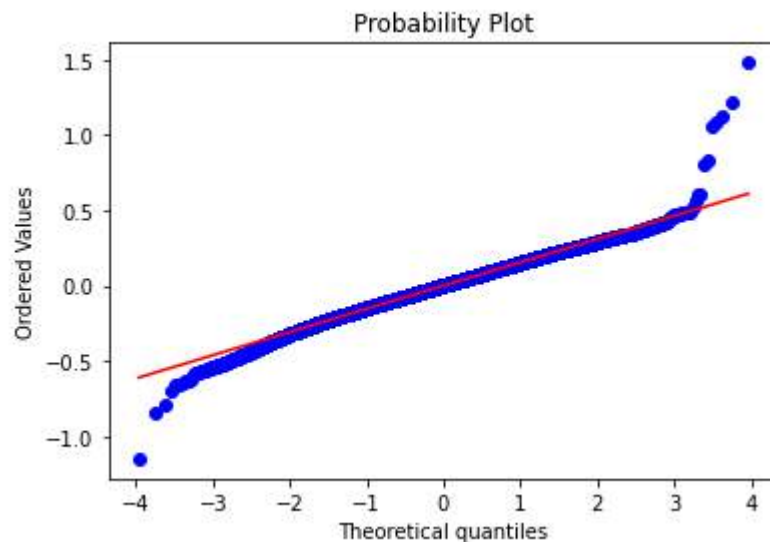


Figure 21.Q-Q plot

## Checking the Linear Regression Assumptions

1. No Multicollinearity
2. Mean of residuals should be 0
3. No Heteroscedasticity
4. Linearity of variables
5. Normality of error terms

### Assumption 1: No Multicollinearity

From the correlation, we observed the column depth and table has a very weak correlation with the price variable. However, we will use VIF, to check if there is multicollinearity in the data.

VIF value  $\leq 4$  suggests no multicollinearity whereas a value of  $\geq 10$  implies serious multicollinearity.

carat	25.166938
cut	1.484473
color	1.114907
clarity	1.208717
depth	4.784497
table	1.632172
x	445.923678
y	362.487239
z	250.463098

Table16. VIF values for the predictors

The columns carat, x, y, z are the very good predictors for the price variable. VIF for the depth column is greater than 4 and will rerun the model by dropping the 'depth' column.

OLS Regression Results						
Dep. Variable:	price_log		R-squared:	0.976		
Model:	OLS		Adj. R-squared:	0.976		
Method:	Least Squares		F-statistic:	9.613e+04		
Date:	Tue, 23 Nov 2021		Prob (F-statistic):	0.00		
Time:	18:30:16		Log-Likelihood:	8138.1		
No. Observations:	18847		AIC:	-1.626e+04		
Df Residuals:	18838		BIC:	-1.619e+04		
Df Model:	8					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	1.1431	0.044	25.819	0.000	1.056	1.230
carat	-1.0767	0.012	-89.215	0.000	-1.100	-1.053
cut	0.0241	0.002	15.895	0.000	0.021	0.027
color	-0.0749	0.001	-105.509	0.000	-0.076	-0.073
clarity	-0.2208	0.001	-148.937	0.000	-0.224	-0.218
table	0.0052	0.001	8.027	0.000	0.004	0.006
x	0.4265	0.023	18.697	0.000	0.382	0.471
y	0.4684	0.021	22.300	0.000	0.427	0.510
z	0.8419	0.014	61.384	0.000	0.815	0.869
Omnibus:	3431.663	Durbin-Watson:	2.019			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	67914.225			
Skew:	0.310	Prob(JB):	0.00			
Kurtosis:	12.279	Cond. No.	2.34e+03			

#### Notes:

- [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
- [2] The condition number is large, 2.34e+03. This might indicate that there are strong multicollinearity or other numerical problems.

Figure23. Statsmodel without depth variable

Data	Root Mean Square	MAE	MAPE
Train	987.38038	497.671896	12.333218
Test	940.95530	501.572188	12.398176

Table17. RMSE values

## Summary with and without multicollinearity variable(depth)

- Model R-squared and Adjusted R squared obtained for **olsmodel1** is same as the previous model – **lm1**. Removal of multicollinear variable has not causes any information loss in the model.
- The RMSE of the model on train data has increased after dropping the depth variable. But it is still small on the test data which is a welcome sign.
- Mean Absolute Percentage Error is ~12% on the test data.

## Assumption 2: Mean of the residuals should be 0

```
residuals = lm1.resid
np.mean(residuals)
```

Result is  $-1.1661207245637731e-13$  which is very close to 0. Assumption 2 is satisfied.

## Assumption 3: No Heteroscedasticity

**Homoscedacity** - If the residuals are symmetrically distributed across the regression line, then the data is said to homoscedastic.

**Heteroscedasticity**- If the residuals are not symmetrically distributed across the regression line, then the data is said to be heteroscedastic. In this case the residuals can form a funnel shape or any other non symmetrical shape.

We will use 'Goldfeldquandt' Test to test the following hypothesis ( $\alpha = 0.05$ )

**H0:** Residuals are homoscedastic

**HA:** Residuals have hetroscedasticity

```
test = sms.het_goldfeldquandt(residuals, x_train)
```

```
p-value 0.7654062186137036
```

Since p-value > 0.05 we cannot reject the Null Hypothesis that the residuals are homoscedastic.

Assumption 3 is also satisfied by our lm1.

## Assumption 4: Linearity of variables

Predictor variables must have a linear relation with the dependent variable.

To test the assumption, we will plot residuals and fitted values on a plot and ensure that residuals do not form a strong pattern. They should be randomly and uniformly scattered on the x axis.

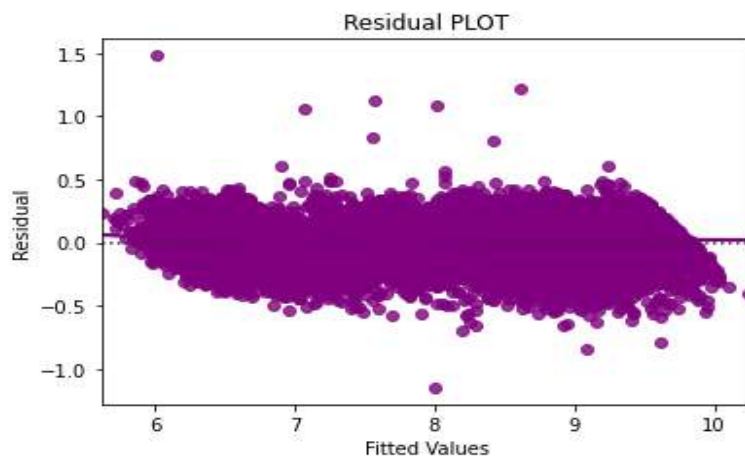


Figure 24. Residual Vs Fitted plots

Assumption 4 is also satisfied.

## Assumption 5: Normality of error terms

The residuals should be normally distributed.

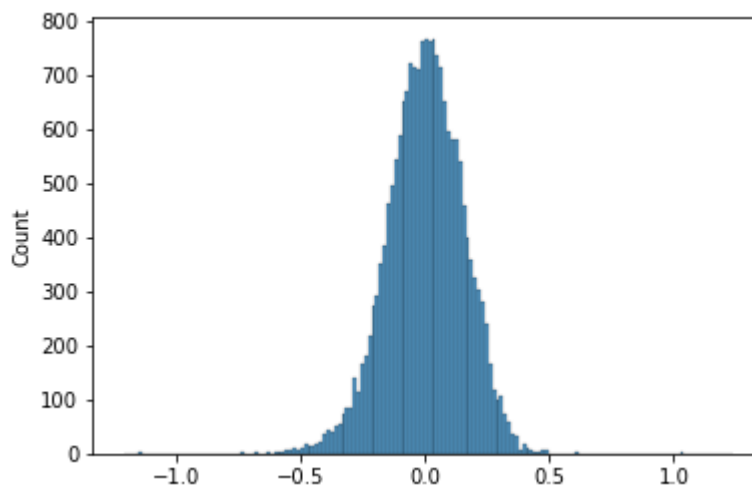


Figure 25. Normality of errors

Assumption 5 is also satisfied.

- Run the model with/without outliers and removing the Depth (*assuming the coefficient is likely by chance*), merging/without merging the sublevels of categorical, but these steps does not make great impact to the model results.
- Removing the 'depth' variable increasing the mean square error and bring down the Adjusted R<sup>2</sup> value by 0.1%.
- From our original statsmodel model lm1 results, we can conclude that there is some correlation on all the predictors with price. Depth and Table variable are the very weak predictors and rests of the variables are very strong predictors for deciding the price. With all the variables, we have been able to achieve 97.7%, Adjusted R<sup>2</sup> value, i.e., the co-efficient of determinant.
- Our samples are obtained from the normal distribution, confirmed with Q-Q plot.
- All of the linear assumptions are satisfied with our original model 'lm1'.

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

Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present.

#### Summary

- We had a problem to predict the price of the cubic zirconia based on the provided dataset using the Linear regression model. We have combined the sublevels of clarity/cut quality of the stone to some extent and also normalized the price before fitting the data into the regression model.
- We have used the sklearn and statsmodel package to figure out the best coefficient of determinant. With all the provided predictors, we have achieved approximately 97% of explained variance of the dataset. After dropping the multicollinearity variables depth and table, there is no change in the adjusted R<sup>2</sup>. So, for our final model, we can fit it without depth and table variables.
- Most of our linear assumptions are satisfied by the model.
- There is a linear relationship between the actual and predicted price.

#### Manual prediction

```
(-1.64) * Intercept + (-1.1) * carat + (0.03) * cut + (-0.08) * color +  
(-0.22) * clarity + (0.04) * depth + (0.01) * table + (0.64) * x +  
(0.57) * y + (0.35) * z
```

#### Recommendations and insights

- Customers are interested to buy the stones where we had the best cut. i.e., 'Ideal' cut in our case. Also, stones sold is based on the hierarchical cut quality from Fair<Good<Very Good<Premium<Ideal. Recommend to concentrate on the cut quality as the first step to bring more profits. Additionally, the average price of ideal cut stones are less compared to the Fair cut, price of the ideal cut stones can be increased to gain more profits.

- When it comes to the color of the stone, manufactures identified as the best color sold less compared to the mid range colors. Also, the average prices of the best color stones are high. This may be one of the reasons that the customers not buying the stones identified as the best color. As a matter of fact, mid range colors bring more profits to the company. It is advisable to produce more stones in the mid range colors. At some point, it is good to revisit the prices of best colored stones.
- Customers preferred the slight inclusions in the stones. There were very less profits for the A1 carat stones and no profits to the highly included blemishes on the stones. Based on this fact, inclusions in the stones to be maintained moderately (not very high and not little). Slightly included inclusions in the stones bring profits across all the cut quality.
- The clarity of the stone is one of the most important attribute to choose the stone. Because, the more clear the stones, it attracts the customers more. The average price of SI2 is high compared to other clarities. SI2 stones have more noticeable inclusions than SI1 stones. The choice between SI1 and SI2 clarity comes down to the tradeoff between price and quality. All else being equal, SI2 cubic zirconias are cheaper, but SI1 stones are cleaner, on average. Also, the average prices of SI1, VS2, I1 and VS1 are almost same. The benchmark of SI2 stone average is high which is good for the company profits. Additionally, we can increase the prices of SI1, VS2, I1, VS1 stones to differentiate from SI2 stones. With this, we will make the customers to think of the quality of cubic rather than just being choosing cheap stones.
- Ask for feedback from the customers on a regular basis. More importantly, you need to pay attention to what's being said – good or bad. When you encourage your customers to speak up and show them that their opinions really matter, you endear them to your brand. As a bonus, you will also be better prepared to correct problems and make improvements in a more proactive way.

The five best attributes of Cubic Zirconia are:

1. Carat weight of the cubic zirconia.
2. Clarity (visibility) of the cubic zirconia.
3. Length of the cubic zirconia.
4. Width of the cubic zirconia.
5. Height of the cubic zirconia.

## Problem 2: Logistic Regression and LDA

### Executive Summary

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

### Introduction

This is a supervised classification problem where the employees would opt for the holiday package or not. Here, we will be using the Logistic regression and Linear Discriminant Analysis to predict the classification. A tour and travel agency provided the 872 employee details. There are 7 columns (*excluding index column*) and 'Holiday Package' column is the target variable, which tells us whether the holiday package is opted or not. Once the models are trained, we will run the model using the test set and compare each model's precision/recall value to finalize the model for our business problem.

**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.**

There are 872 records and 8 columns present in the Holiday Package dataset.

### Data Description

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

*Table18. Holiday package Data Description*



## Sample of the Salary Data 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 19. Holiday Package Dataset Sample*

Let us check the types of variables in the data frame.

Variable Name	Data Types
Unnamed: 0	int64
Holiday_Package	object
Salary	int64
age	int64
edu	int64
no_young_children	int64
no_older_children	int64
foreign	object

*Table 20. Holiday Package Data types*

We can drop the column 'Unnamed: 0' as it's just a indexing to the records.

There are only 5 continuous columns such as salary, education, age, number of young children and number of older children. Other variables such as Foreign and Holiday package (*target variable*) are the object data types which we will be encoding shortly for our model building.

Check for the duplicate and missing records in the dataset

There are no missing and duplicate records present in the dataset.

```
#      Column      Non-Null Count  Dtype
---  -
0      Unnamed: 0      872 non-null    int64
1      Holliday_Package  872 non-null    object
2      Salary           872 non-null    int64
3      age              872 non-null    int64
4      educ             872 non-null    int64
5      no_young_children  872 non-null    int64
6      no_older_children  872 non-null    int64
7      foreign          872 non-null    object
dtypes: int64(6), object(2)
```

Figure 26.Data information

Let's rename the columns from 'Holliday package' to 'Holiday Package'.

Summary of the Dataset

	count	mean	std	min	25%	50%	75%	max
Salary	872.0	47729.172018	23418.668531	1322.0	35324.0	41903.5	53469.5	236961.0
age	872.0	39.955275	10.551675	20.0	32.0	39.0	48.0	62.0
educ	872.0	9.307339	3.036259	1.0	8.0	9.0	12.0	21.0
no_young_children	872.0	0.311927	0.612870	0.0	0.0	0.0	0.0	3.0
no_older_children	872.0	0.982798	1.086786	0.0	0.0	1.0	2.0	6.0

Table 21.Holiday Package Dataset Summary

Observations for Holiday Package Dataset

Salary: The minimum salary is 1322/- rupees and the maximum salary is 236961/- rupees. Upon investigating the minimum salary, the age of the person is 57 and it must be a bad record entry, let's impute it with mean salary.

Age: Minimum age is 20 and maximum age is 62, which looks good.

Education: Formal years of education seem to be valid.

No\_young\_children: Looks good.

No\_old\_children: Looks good.

## Outliers Proportion (in %)

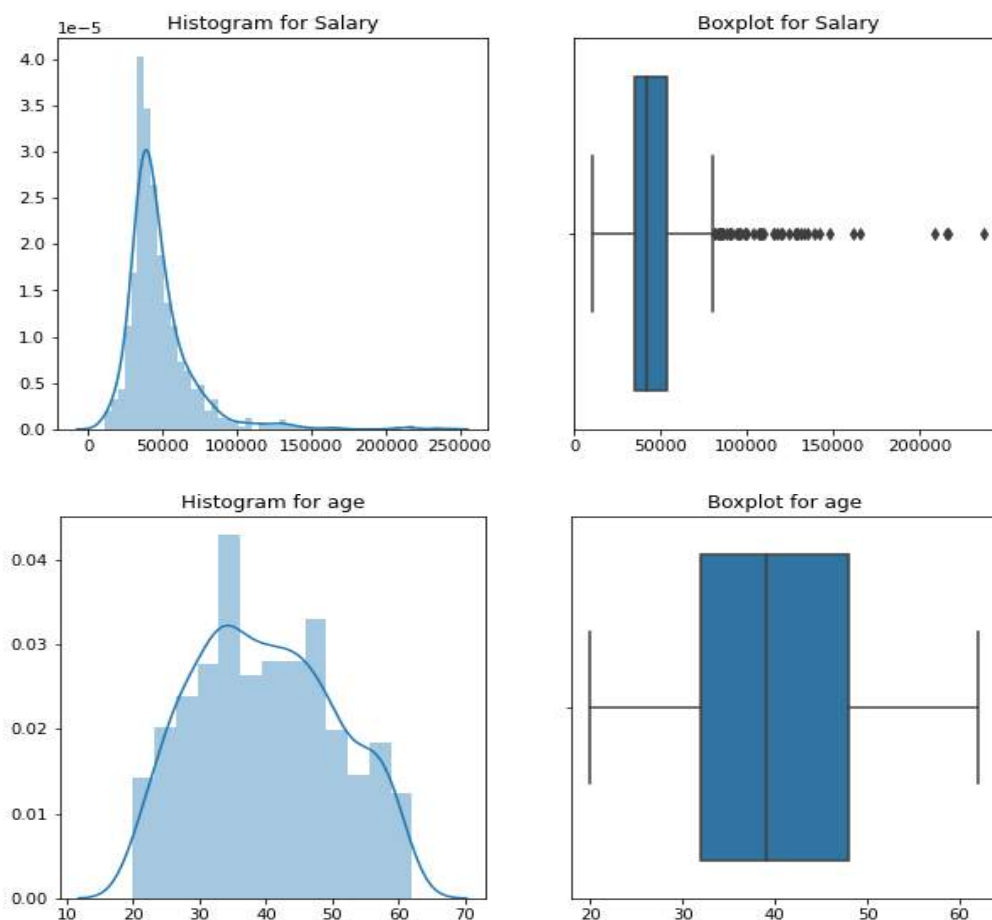
	outlier proportion %
Salary	6.42
age	0.00
educ	0.46
no_young_children	23.74
no_older_children	0.23

Table 22. Holiday Package Outlier proportion

Outlier for Salary and number of young children seems to be valid.

## Histograms and Boxplots

Below graphs helps us identifying the data distribution for continuous columns.



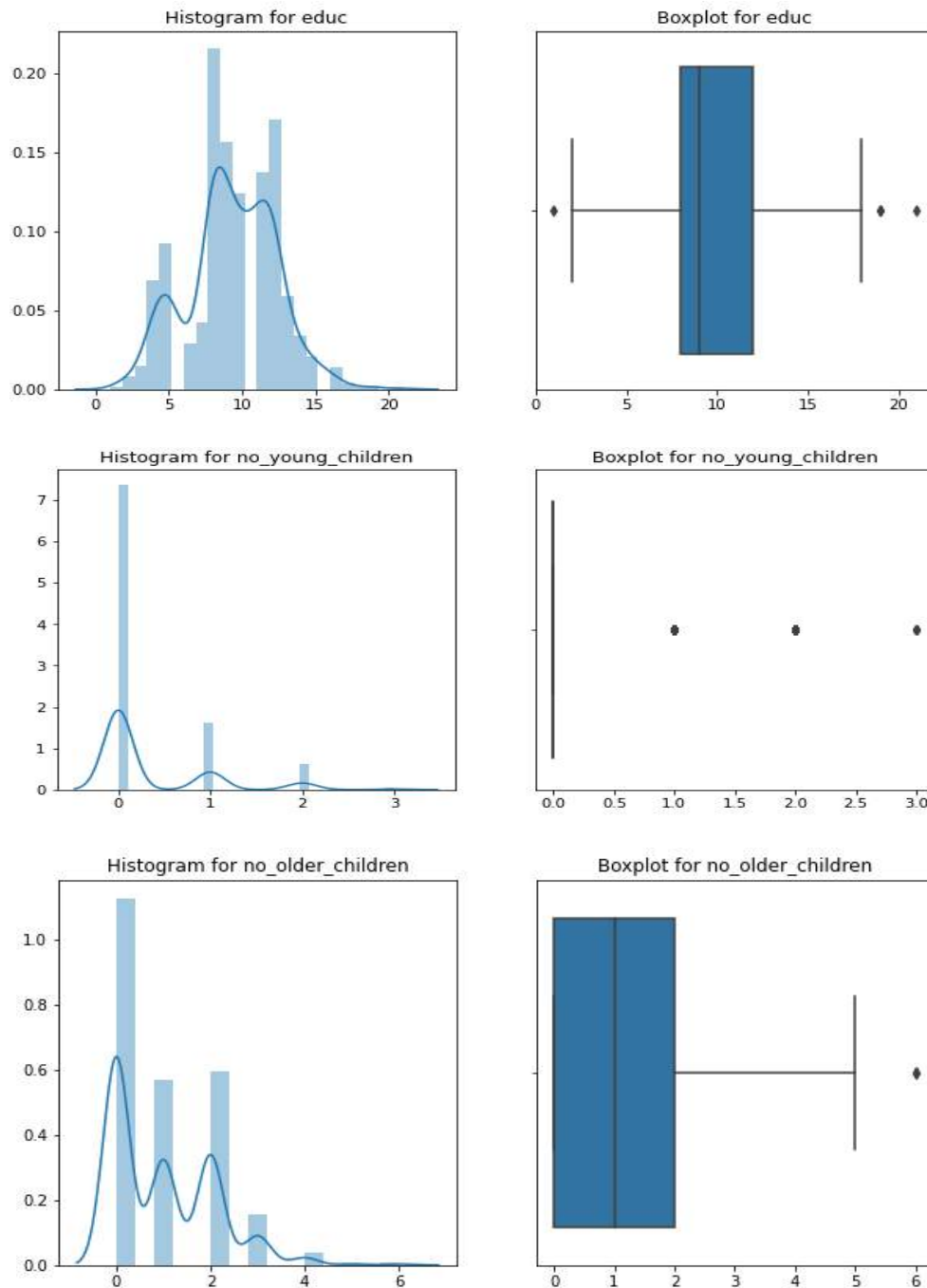


Figure 27. Histogram and Box plot for numerical columns

- Histogram shows the data's distributed from 0 to ~350.
- The 'Sales' variable has outliers. Few records are out from the crowd.
- 'Sales' is positively skewed with the value of 2.3446426921667585
- Most of the data's are clustered around the left tail of the distribution while the right tail of the distribution is longer.

## Skewness

Skewness for Salary is: 3.13

Skewness for age is: 0.15

Skewness for educ is: -0.05

Skewness for no\_young\_children is: 1.95

Skewness for no\_older\_children is: 0.95

## Observations from Histograms and Boxplots

- The columns such as Salary, no\_young\_children, no\_older\_children are positively skewed and their data's are clustered around the left tail of the distribution while the right tail of the distribution is longer.
- 'Education' and 'age' columns are almost normally distributed.
- Outliers present in the variables are valid.

## Categorical Variables

### Holiday Package Vs Salary

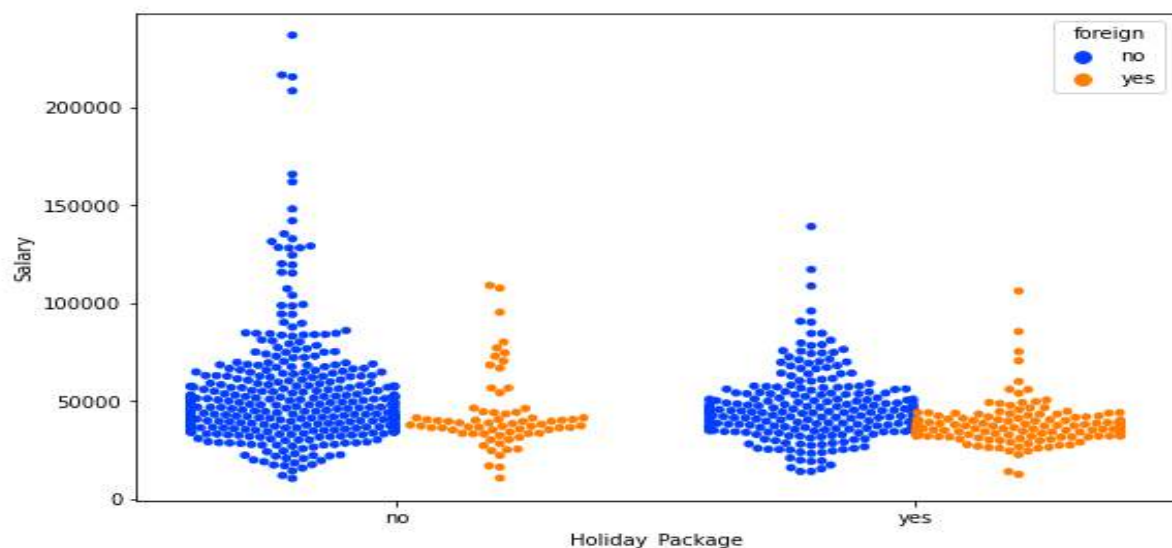


Figure 28. Swarmplot for Salary Vs Holiday Package

- When salary of the employees increases, they are rarely opting for the Holiday package. In fact, when it goes above 1L, opting for the package is null.
- Employees earning between 10K and 80K is opting for the Holiday package, even 50% within them are foreigners.

## Holiday Package Vs Age



Figure 29. Swarmplot for Age Vs Holiday Package

Irrespective of foreigners or local, between the employees age 30 and 50 is mostly opted for the Holiday package.

### Foreigner Vs Local Vs Holiday Package

Holiday_Package	no	yes	All
foreign			
no	0.461009	0.291284	0.752294
yes	0.079128	0.168578	0.247706
All	0.540138	0.459862	1.000000

*Table 23. Employees Vs Holiday Package proportion*

- Among 872 employees, 75% of them are local and 25% of them are foreigner.
- The percentage of local availed the package is 29% and foreigner is 17%. However, if we calculate the percentage of employees based out of local from total local is 33% whereas the foreigner is 58%. These shows up the foreigners are more interested in opting for the Holiday packages.
- In whole, 46% of the employees availed the Holiday packages.

### Grouping the employees by Age

Age range	Category
20-25	Young
25-35	Adults
35-50	Mid-age
50-60	Seniors

*Table 24. Grouping Employees by age range*

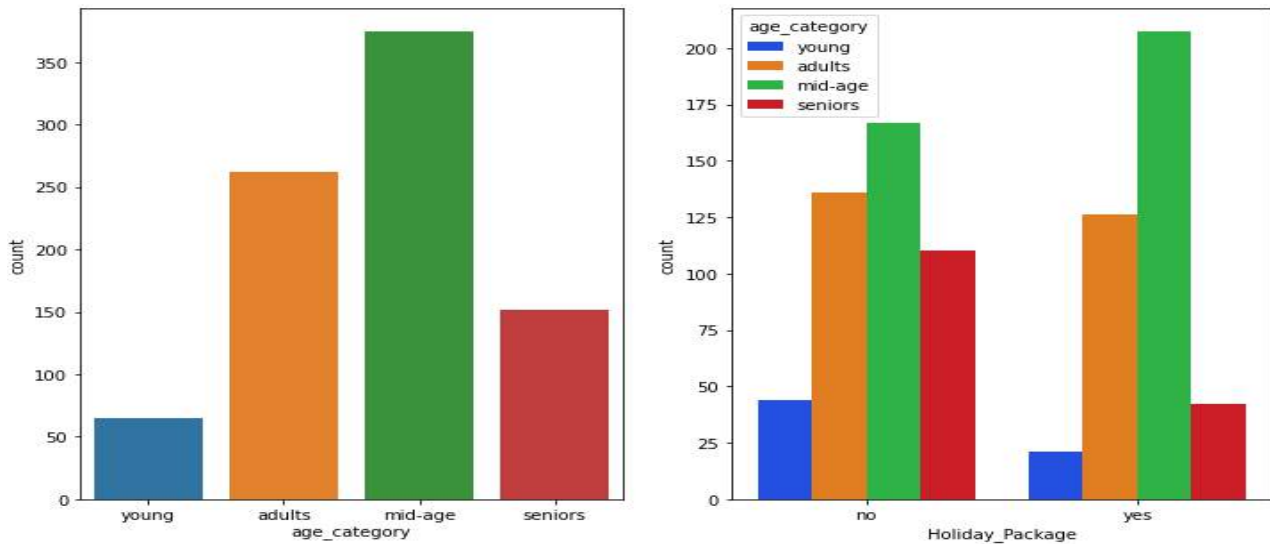


Figure 30.Barplot for Age Category Vs Holiday Package

Approximately, 45% of the employees falls under mid-age category and 30% of them are Adults category. Among the mid-age and adults availed the package mostly. Seniors and young employees are showing less interest in availing the packages.

#### Holiday Package Vs Education/Number of Young & Old Children

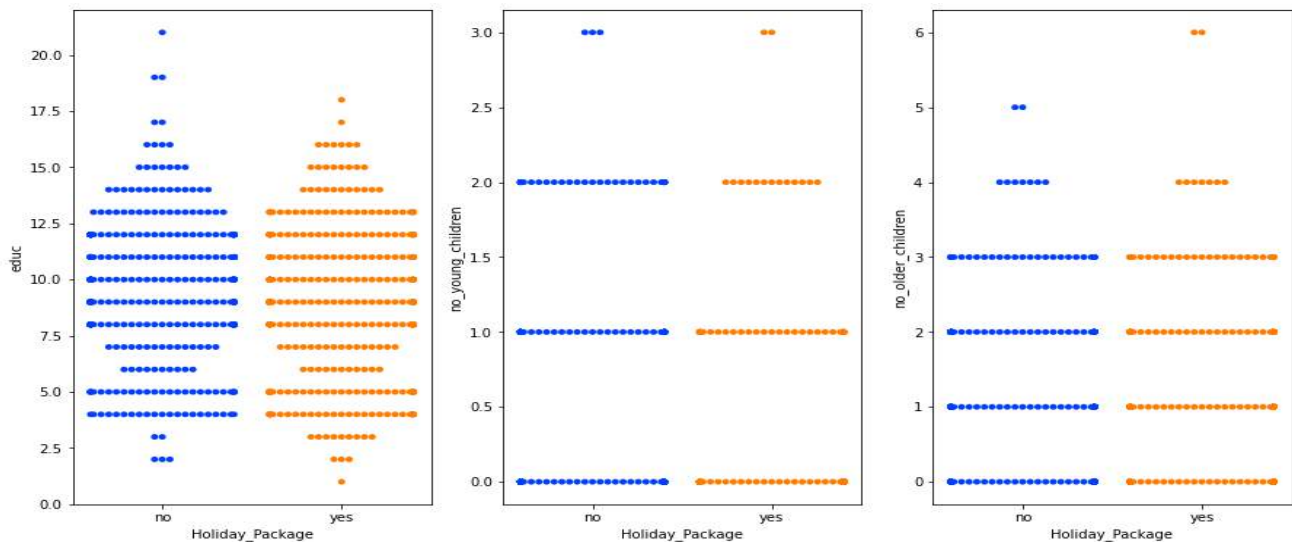
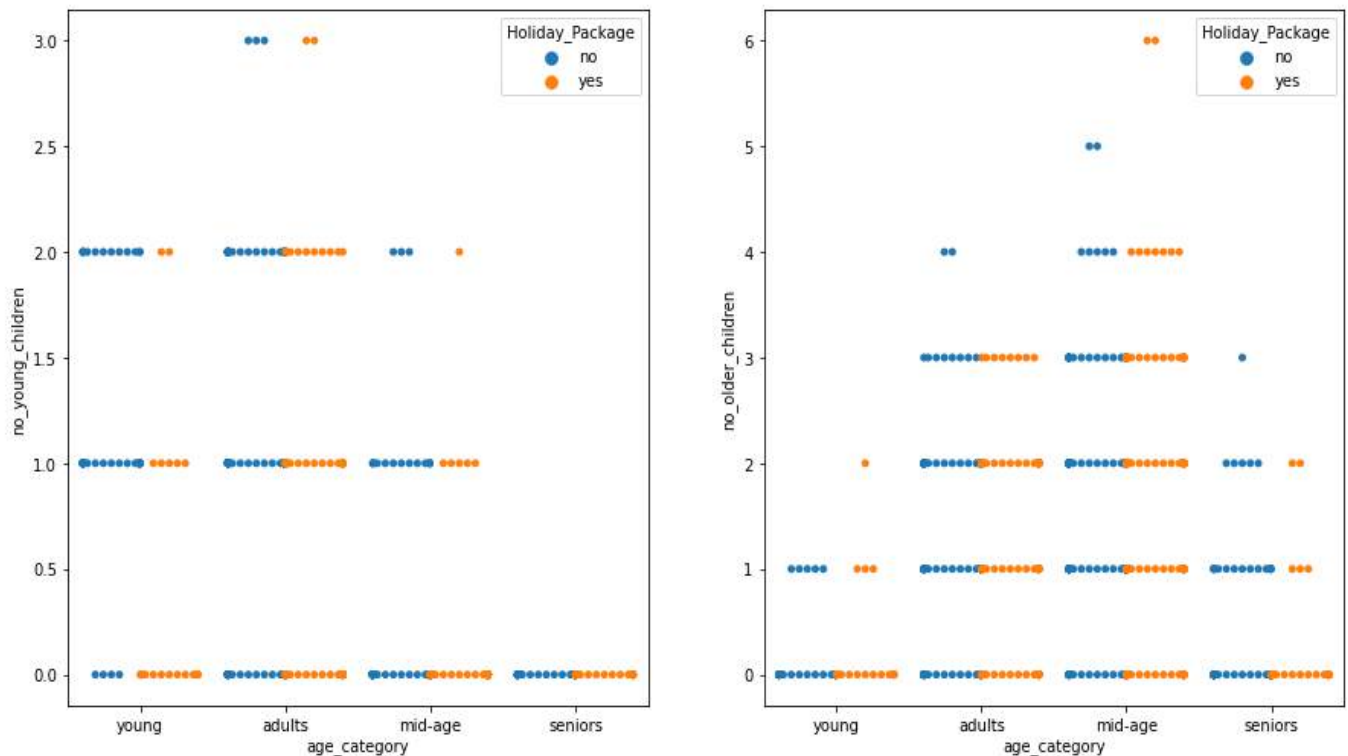


Figure 31.Swarmplot for education/young/old children Vs Holiday Package

Availing Holiday package is irrelevant to their education and children upto some levels. Observed, symmetric proportion between opted and not opted the packages.





*Figure 32. Swarmplot for young/old children Vs Age group Vs Holiday Package*

- Employees with no young or older children opted for the Holiday package.
- There is a consistency observed in the 'adults' opting for the package nevertheless of 0, 1 or 2 young children.
- When the 'young' (age 20-25) and 'mid-age' (age 35-50) employees started having young children, opting for the holiday package is gradually decreasing.
- Again, among 'adults' and 'mid-age' employees, there is not much change in opting for the package nevertheless of 0, 1, 2 or 3 older children.
- When 'seniors' started to have old children, opting for the holiday package is reducing significantly.

## Pair plot

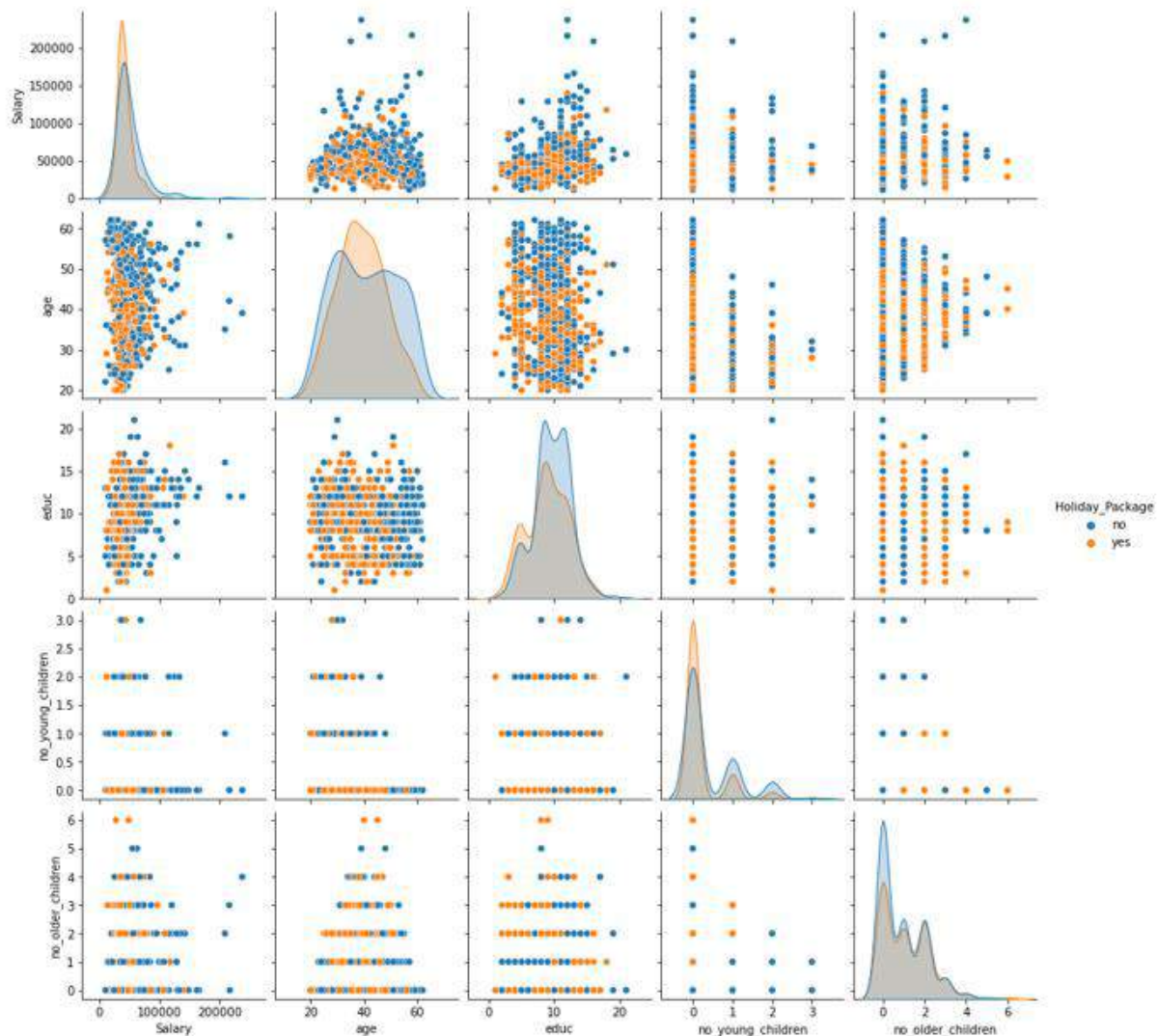


Figure 36. Pair plot to see the correlation between the variables

- Orange color indicates the employees opted the package and the blue implies not opted.
- Going through the diagonals (KDE), most of the attributes are significantly overlapping with slight difference which tells us the attributes are unable to differentiate between the employees opted the package and those not. These attributes are weak or poor predictors from classification point of view.
- Salary and children attributes have a sharp peak which indicates the outliers.
- Age attributes indicates the slightly differentiating between the classes.

## Heat Map

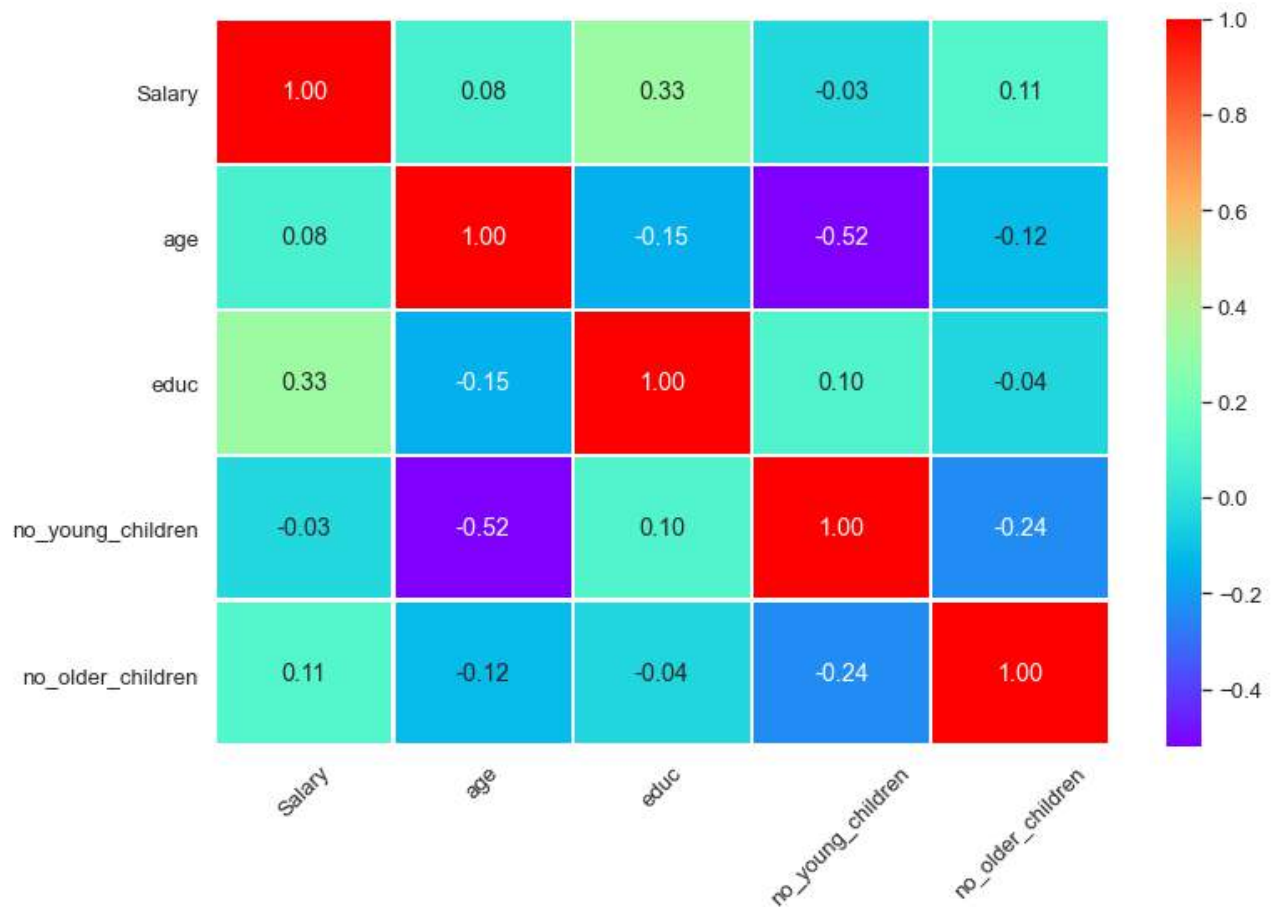


Figure 37. Heat Map to see the correlation between the variables

- There is no correlation between the variables, no multi-collinearity.
- The number of young children is slightly correlated with the age and it indicates the negative correlation between these attributes.

**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).**

### Checking the Data types

```
Holiday_Package    object
Salary             float64
age                int64
educ               int64
no_young_children  int64
no_older_children  int64
foreign            object
dtype: object
```

*Figure38. Data Types*

### Converting object data types to categorical codes

Using the Label encoder method, let's convert the Holiday package variable.

```
from sklearn.preprocessing import LabelEncoder

LE = LabelEncoder()

hdf['Holiday_Package'] = LE.fit_transform(hdf['Holiday_Package'])
```

Using the pd.Categorical method, let's convert the foreign variable.

```
hdf['foreign'] = pd.Categorical(hdf['foreign']).codes
```

```
RangeIndex: 872 entries, 0 to 871
Data columns (total 7 columns):
#   Column              Non-Null Count  Dtype
---  ---
0   Holiday_Package      872 non-null    int32
1   Salary               872 non-null    float64
2   age                  872 non-null    int64
3   educ                 872 non-null    int64
4   no_young_children    872 non-null    int64
5   no_older_children    872 non-null    int64
6   foreign              872 non-null    int8
dtypes: float64(1), int32(1), int64(4), int8(1)
memory usage: 38.4 KB
```

*Figure39. Verify data types after conversion*

Now, all the objects are converted to integers which are required for building the model.

	Holiday_Package	Salary	age	educ	no_young_children	no_older_children	foreign
0	0	48412.0	30	8	1	1	0
1	1	37207.0	45	8	0	1	0
2	0	58022.0	46	9	0	0	0
3	0	66503.0	31	11	2	0	0
4	0	66734.0	44	12	0	2	0

Table 25. Sample Dataset after encoding

Here, we assign independent columns to variable X and target to Y.

```
X = hdf.drop(['Holiday_Package'], axis=1)
y = hdf[['Holiday_Package']]
```

Using the sklearn package, we import the train\_test\_split function. Split the dataset, one for training the model and another one for test the model (*unseen data by the model*).

Throughout this problem, we run the model with random state=1 to be consistent across the results.

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30
, random_state=1, stratify=hdf['Holiday_Package'])
```

Assign 30% of the data's to the test set and 70% for training the model. Using the stratify option; will split equally among the train and test records according to the target classification variable.

- Train set contains 610 records.
- Test set contains 262 records.

Target variable percentage for Train & Test

	Train Labels		Test Labels	
Ratio	0	0.539344	0	0.541985
	1	0.460656	1	0.458015

*Table 26.Holiday Package Split percentage*

Ratio of split between train dataset and test dataset is nominal.

Building the Logistic Regression Model

```
lgt_model=LogisticRegression(max_iter=10000,n_jobs=2,random_state=1)
```

Grid Search to find out the optimal hyper parameters for Logistic Model training set

```
param_grid = {
    'solver': ['lbfgs','newton-cg','liblinear'],
    'penalty' : ['l1', 'l2', 'none'],
    'tol': [0.0001,0.000001]
}
```

```
lgtcl = LogisticRegression(random_state=1)
```

```
grid_search = GridSearchCV(estimator = lgt_model, param_grid =
param_grid,scoring='f1', cv = 3)
```

```
grid_search.fit(X_train, y_train)
```

This is the point, logistic regression model calculates the weights of each classification, and whichever the features discriminate the classes well gets more weightage and finds the best sigmoid curve using the logloss (or) cross entropy function internally.

$$H_p(q) = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i))$$

Binary Cross-Entropy — the usual formula

**Best Parameters are**

```
{'penalty': 'l1', 'solver': 'liblinear', 'tol': 0.0001}
```

Co-efficient for the Logistic Model

	0	Coefficient
0	Salary	-0.000016
1	age	-0.049031
2	educ	0.068478
3	no_young_children	-1.218315
4	no_older_children	-0.010368
5	foreign	1.252361

*Table 27.Attributes Coefficients*

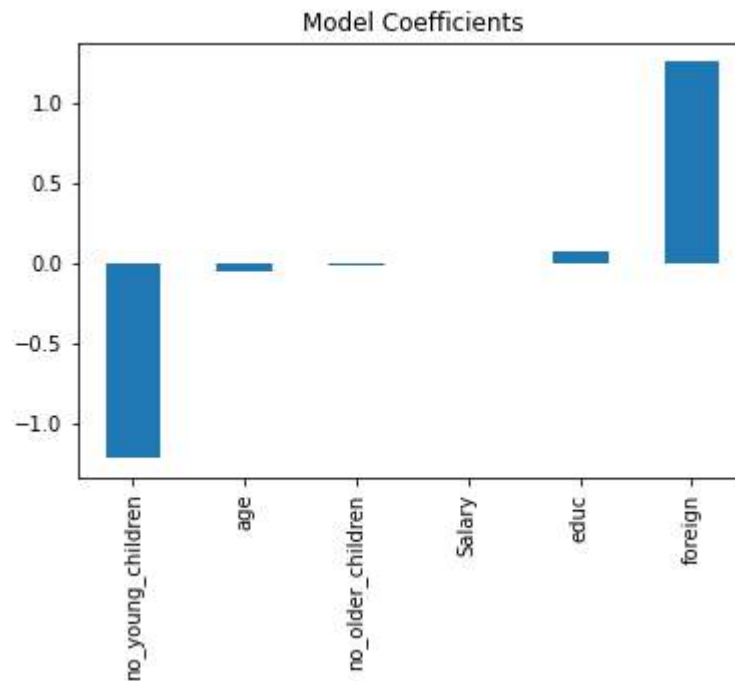


Figure40.Coefficients by bar graph

## Linear Discriminant Analysis Model(LDA)

LDA is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling differences in groups i.e. separating two or more classes.

### Building the Linear Discriminant Analysis Model

```
lda = LinearDiscriminantAnalysis()
```

```
ldamodel=lda.fit(X_train,y_train)
```

LDA divided the dataset into 2 groups(0 and 1), once it is separated out, now looks at the X variables and uses the method of unsupervised learning(PCA) to try and find the structure of X's. LDA model maximizes the between class variances (*like ANOVA technique*) and minimizes the within class variance (*like PCA*). It uses the Bayes' theorem to estimate the probabilities for the every new input, the class which has the highest probability is considered as the output class.



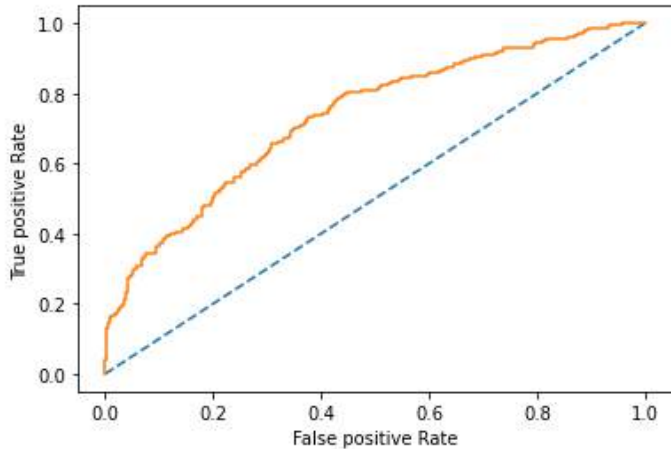
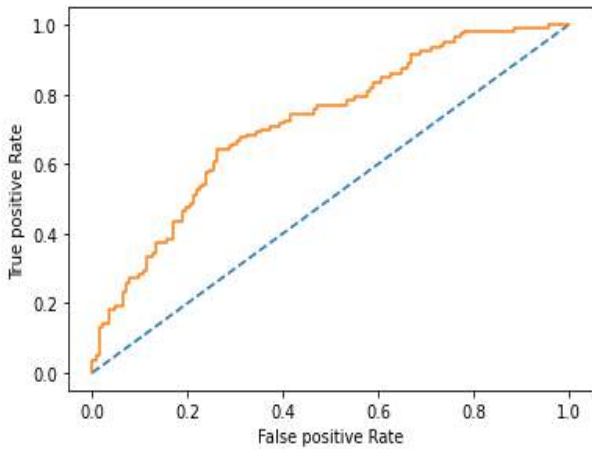
**2.3 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC\_AUC score for each model Final Model: Compare Both the models and write inference which model is best/optimized.**

Logistic Regression – Check the performance on Training and Test dataset

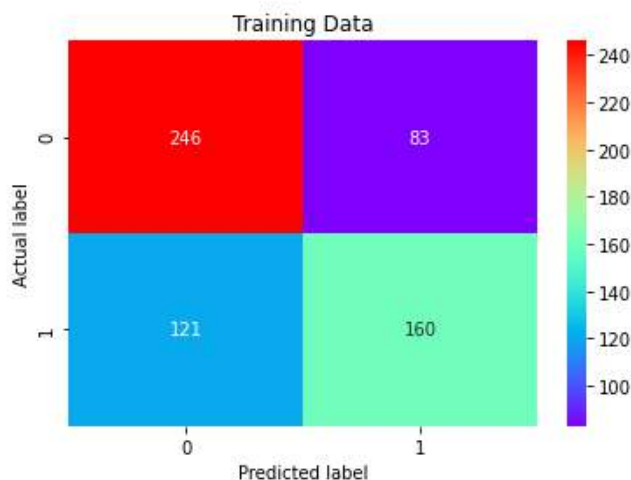
```
ytrain_predict = model.predict(X_train)
```

```
ytest_predict = model.predict(X_test)
```

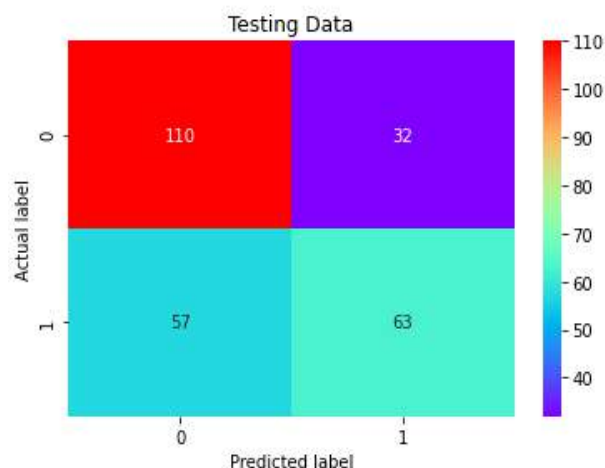
Logistic Regression Classifier Model Predictions

Logistic Regression Train Set	Logistic Regression Test Set
<b>Predictions</b> <pre>ytrain_predict = model.predict(X_train)</pre>	<b>Predictions</b> <pre>ytest_predict = model.predict(X_test)</pre>
<b>Data shape</b> There are 610 records	<b>Data shape(30% of the data)</b> There are 262 records
<b>ROC</b> 	<b>ROC</b> 

## Confusion Matrix



## Confusion Matrix



## Classification Report

	precision	recall	f1-score	support
0	0.67	0.75	0.71	329
1	0.66	0.57	0.61	281
accuracy			0.67	610
macro avg	0.66	0.66	0.66	610
weighted avg	0.66	0.67	0.66	610

## Classification Report

	precision	recall	f1-score	support
0	0.66	0.77	0.71	142
1	0.66	0.53	0.59	120
accuracy			0.66	262
macro avg	0.66	0.65	0.65	262
weighted avg	0.66	0.66	0.65	262

## Summary

Not opted the package	Opted the package
<b>Train Data:</b> Precision: 67 % Recall: 75% <b>f1-Score: 71%</b>	<b>Train Data:</b> AUC: 73.3% Accuracy: 67% Precision: 66 % Recall: 57% <b>f1-Score: 61%</b>

## Summary

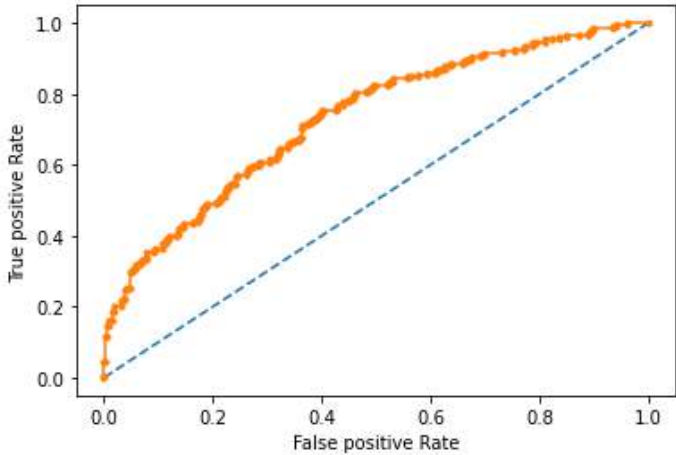
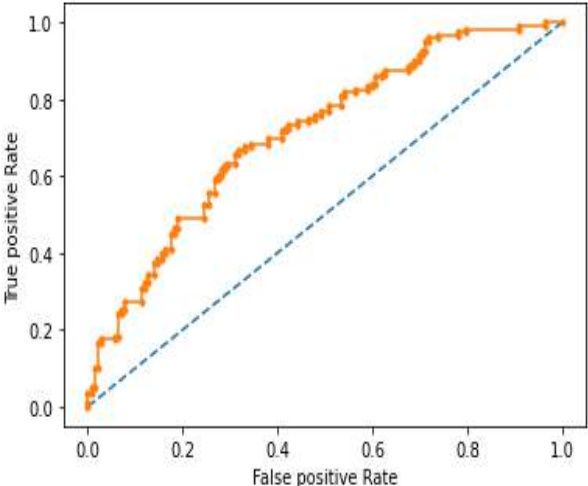
Not opted the package	Opted the package
<b>Test Data:</b> Precision: 66% Recall: 77% <b>f1-Score: 71%</b>	<b>Test Data:</b> AUC: 73.3% Accuracy: 66.03% Precision: 66% Recall: 53% <b>f1-Score: 59%</b>

By default, threshold value is 0.5 for the predictions. We have achieved 59% of F1 score for the test set. We run through the loop with the threshold values ranging from 0.1 to 1. For the threshold value 0.4, we have achieved the maximum F1 score of 68% for the training set and 65% for the test set.

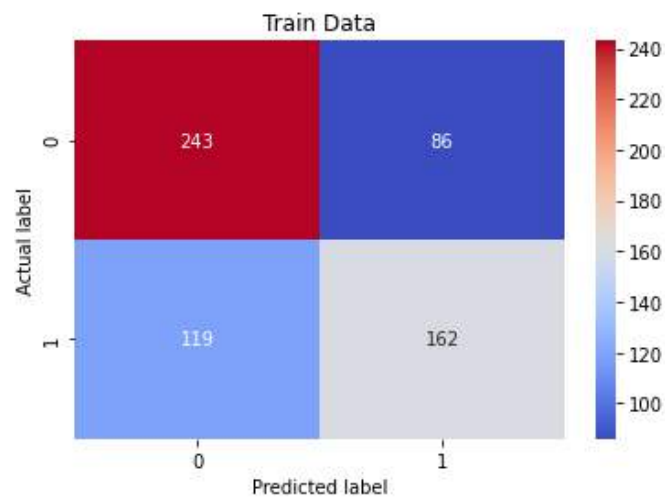
LDA – Check the performance on Training and Test dataset

```
pred_class_train = ldamodel.predict(X_train)
```

```
pred_class_test = ldamodel.predict(X_test)
```

Linear Discriminant Analysis Train Set	Linear Discriminant Analysis Test Set
<b>Predictions</b> <code>pred_class_train = ldamodel.predict(X_train)</code>	<b>Predictions</b> <code>pred_class_test = ldamodel.predict(X_test)</code>
<b>Data shape</b> There are 610 records	<b>Data shape(30% of the data)</b> There are 262 records
<b>ROC Train Data</b> 	<b>ROC Test Data</b> 

**Confusion Matrix**



**Confusion Matrix**



**Classification Report**

	precision	recall	f1-score	support
0	0.67	0.74	0.70	329
1	0.65	0.58	0.61	281
accuracy			0.66	610
macro avg	0.66	0.66	0.66	610
weighted avg	0.66	0.66	0.66	610

**Classification Report**

	precision	recall	f1-score	support
0	0.64	0.77	0.70	142
1	0.64	0.49	0.56	120
accuracy			0.64	262
macro avg	0.64	0.63	0.63	262
weighted avg	0.64	0.64	0.63	262

**Summary**

Not opted the package	Opted the package
<b>Train Data:</b> Precision:67 % Recall: 74% <b>f1-Score: 70%</b>	<b>Train Data:</b> AUC: 73.2% Accuracy: 66% Precision:65 % Recall: 58% <b>f1-Score: 61%</b>

**Summary**

Not opted the package	Opted the package
<b>Test Data:</b> Precision: 64% Recall: 77% <b>f1-Score: 70%</b>	<b>Test Data:</b> AUC: 71.4% Accuracy: 64% Precision: 64% Recall: 49% <b>f1-Score: 56%</b>

**Overfitting and Underfitting:** In both the models, we could see our models are neither overfitted nor underfitted. Because, our training set and testing set results are almost equal.

**F1 score:** Understanding Accuracy made us realize, we need a tradeoff between precision and recall. F1 Score is the weighted average of Precision and Recall.

$$\text{F1 Score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

## ROC Curve for Logistic and LDA models

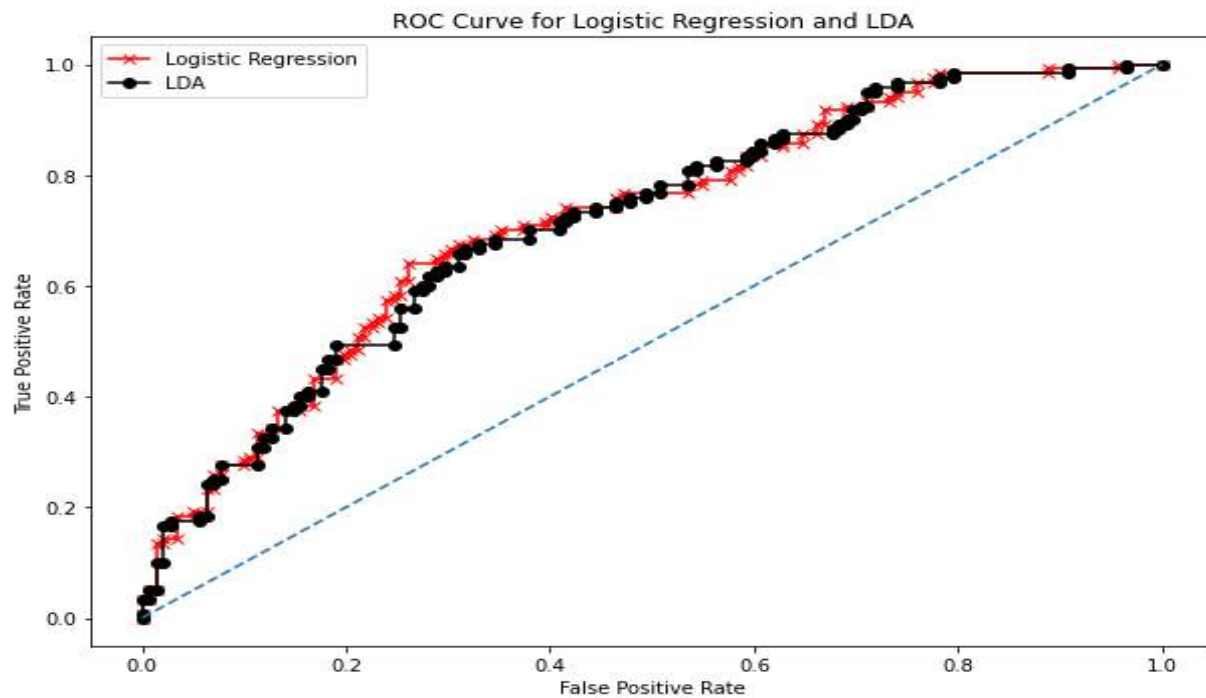


Figure 41. ROC Curve for Logistic and LDA Models

	Logistic Train	Logistic Test	Logistic Test with Threshold 0.4	LDA Train	LDA Test	LDA Test with Threshold 0.4
Accuracy	0.67	0.66	0.77	0.66	0.64	0.79
AUC	0.73	0.72	0.65	0.73	0.71	0.65
Recall	0.57	0.52	0.74	0.58	0.49	0.72
Precision	0.66	0.66	0.58	0.65	0.64	0.59
F1 Score	0.61	0.59	0.65	0.61	0.56	0.65

Table 28. Comparison of Logistic and LDA Models

Logistic Regression Model has the high F1 score for test set compared to the Linear Discriminant Analysis models. Otherwise, both the model results are same. From the above table, even when we run the model with 0.4 as the threshold, f1 score results the same and the ROC curve is superimposed over one another. However, LDA works generally well when the assumptions are violated.

## **2.4 Inference: 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.**

### Summary

We are provided with the records of 872 employees of a company. Among them, some opted for the Holiday package and some did not. Since this is a classification problem, we have used the Logistic Regression and LDA models to predict whether an employee will opt for the package or not, on the basis of the information given in the data set. We have performed the EDA to discover the problems listed below

### Recommendations and Insights

- In general, between the age 30 and 50 is opted for the Holiday package mostly. As an experience provider, request them to share the pictures and comment about the experience they had in social media platform. One of the most influential things can do to attract more employees.
- Seniors are rarely opted for the Holiday package. Provide them the information's such as local events, religious packages, familiar places (*reminiscence of the old days*), restaurants nearby, public transports nearby, toilets, and traditional map handy. Basically, ease to access details, form a package where you get all in one place.
- Foreigner's shows up more interest than local employee which is good sign that they like to explore the local culture and traditional places. Create packages to explore adventure and sometimes local haunted places, list of the foodie highlights including a succinct list of local markets and local events.
- Employees earning between 10K and 80K is opted for the Holiday package. Additionally, whilst the salary is above 50K, opting for the package is gradually decreasing and approaches to nil. Offer them the isolated calm resorts, places with private beach.
- When number of young children is more than 1, employees opted for the package is reducing. Suggest them the one day trip, places nearby like amusement parks.

- Employees with no children opted for the Holiday package. Offer them the package which has islands, mountains, glacial lakes, hill stations, inside the forest trips.
- When 'young' (age 20-25) and 'seniors' (50-60) started to have old children, opting for the holiday package is reducing significantly. May be, young employees married early and could not be able manage alone with the children when they go for trips. Seniors might have grown up children. So, offer them the discounts on combine family trips or group travels.
- Sending an email asking for feedback after their trip and also thanking them for joining your experience is essential. Sooner the better for the business.

**THE END!**