PREDICTIVE MODELING PROJECT BUSINESS REPORT

Date- 25/10/2022

Table of contents

Contents

Problem 2

List of Tables

Problem 1

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:

1. Carat : Carat weight of the cubic zirconia in numbers.
2. Cut : Describe the cut quality of the cubic zirconia like- Fair, Good, Very Good, Premium, Ideal.

3. Color : Color of the cubic zirconia. With D being the best and J the worst.

4 Clarity : Clarity refers to the absence of the Inclusions and Blemishes. (In order from Best to Worst in terms of avg price) IF, VVS1, VVS2, VS1, VS2, Sl1, Sl2, l1.

5. Depth : The Height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter; in numbers

6. Table : The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter; in numbers

7. Price : The Price of the cubic zirconia; in numbers
8. X : Length of the cubic zirconia in mm.
9. Y : Width of the cubic zirconia in mm.
10. Z : Height of the cubic zirconia in mm.

Sample of 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

Exploratory Data Analysis:

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

Column	Dtype
Unnamed: 0	int64
carat	float64
cut	object
color	object
clarity	object
depth	float64
table	float64
x	float64
y	float64
z	float64
price	int64

we have multiple data types as 6 float data types, 2 integer data types and 3 object data types.

Checking for missing values in the dataset:

```
 #   Column        Non-Null Count
---  ------        --------------
 0   Unnamed: 0    26967 non-null
 1   carat         26967 non-null
 2   cut           26967 non-null
 3   color         26967 non-null
 4   clarity       26967 non-null
 5   depth         26270 non-null
 6   table         26967 non-null
 7   x             26967 non-null
 8   y             26967 non-null
 9   z             26967 non-null
 10  price         26967 non-null
```

From the above results we can see that there is missing value present in the dataset

Q1.1. 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 missing values present in the data but it's only in column 'depth' and we will impute this in question 1.2

Coulumn Unnamed: 0 contains serial number so we can remove it and here is the sample data without this column.

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

Duplicate Values

34 duplicate entries found in the data and will remove these values and after removing the Duplicate values data size is (26933, 10)

We are using describe function to get descriptive summary of the data and here is the sample.

	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

We can see that there are no anomalies found in data.

Data Visualization

Univariate Analysis

Non visual representation:

Using describe function to get descriptive analysis.

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
carat	26933.0	NaN	NaN	NaN	0.79801	0.477237	0.2	0.4	0.7	1.05	4.5
cut	26933	5	Ideal	10805	NaN	NaN	NaN	NaN	NaN	NaN	NaN
color	26933	7	G	5653	NaN	NaN	NaN	NaN	NaN	NaN	NaN
clarity	26933	8	SI1	6565	NaN	NaN	NaN	NaN	NaN	NaN	NaN
depth	26236.0	NaN	NaN	NaN	61.745285	1.412243	50.8	61.0	61.8	62.5	73.6
table	26933.0	NaN	NaN	NaN	57.45595	2.232156	49.0	56.0	57.0	59.0	79.0
x	26933.0	NaN	NaN	NaN	5.729346	1.127367	0.0	4.71	5.69	6.55	10.23
y	26933.0	NaN	NaN	NaN	5.733102	1.165037	0.0	4.71	5.7	6.54	58.9
z	26933.0	NaN	NaN	NaN	3.537769	0.719964	0.0	2.9	3.52	4.04	31.8
price	26933.0	NaN	NaN	NaN	3937.52612	4022.551862	326.0	945.0	2375.0	5356.0	18818.0

Insights:

1. Maximum price of zirconia is 18818.0 and minimum price is 326.0 so we can say that price spread in wide range.

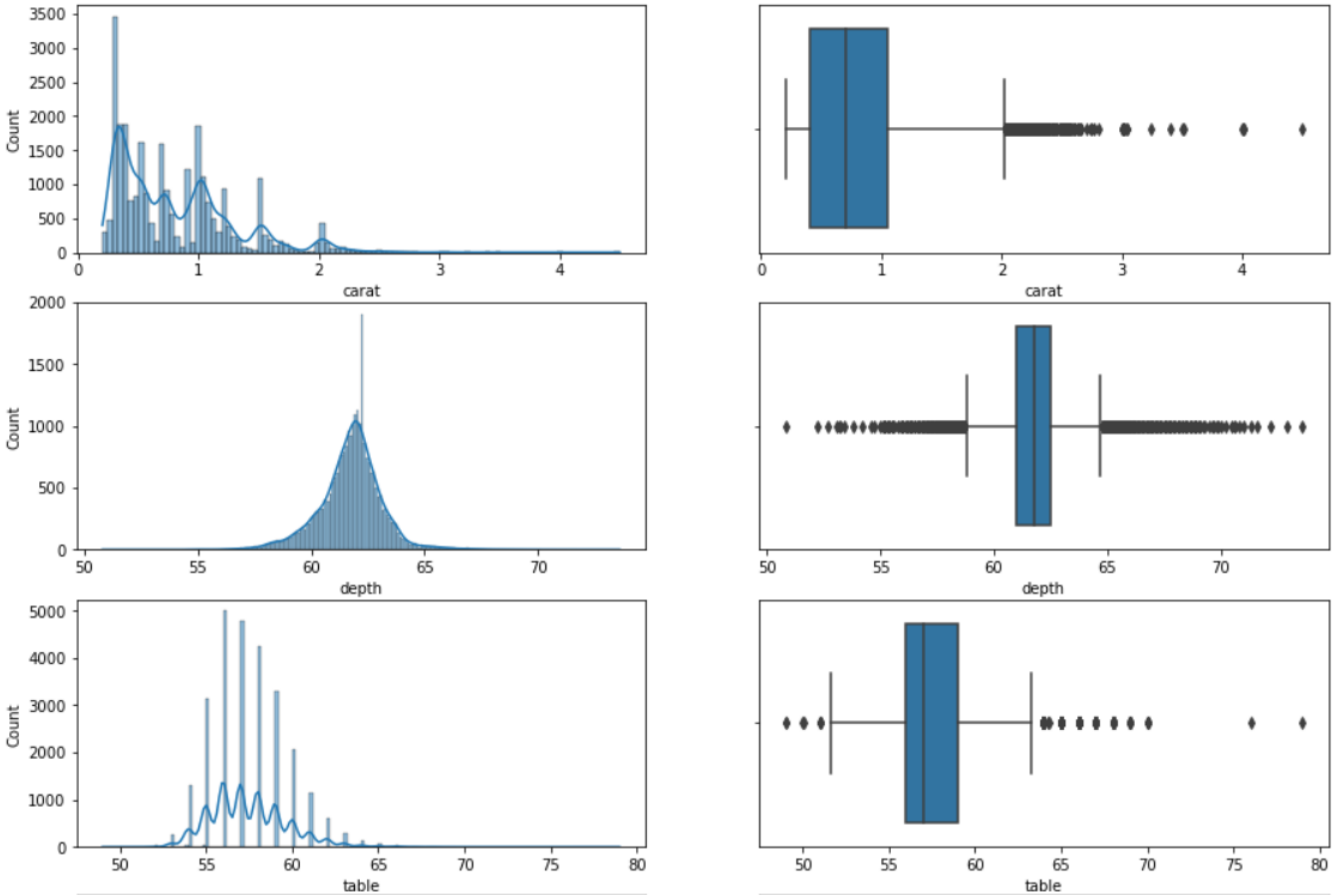
2. minimum Carat weight of zirconia is 0.2 and maximum is 4.5 so zirconia is available in different carat weight.

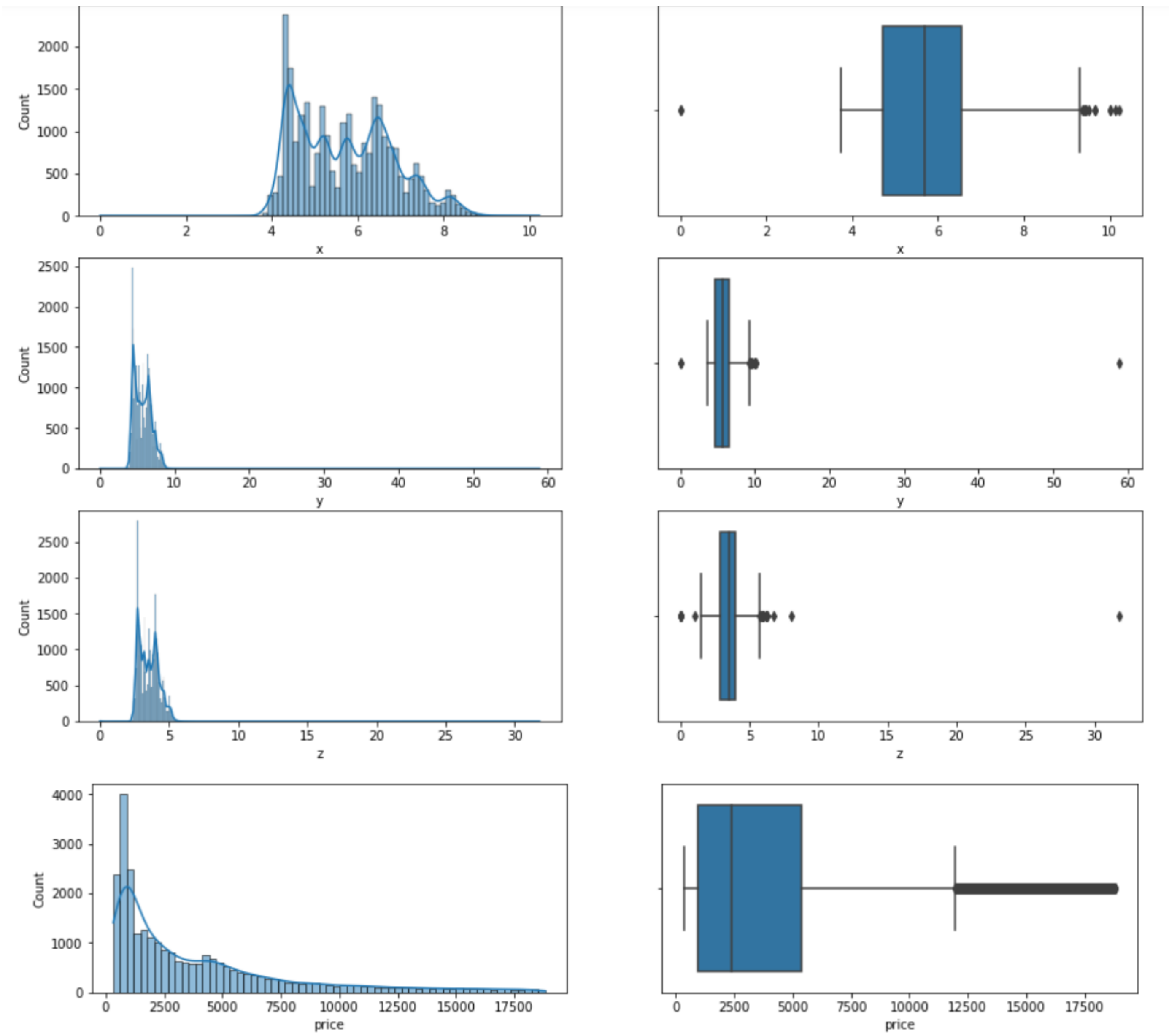
3. Cut quality of zirconia is mostly marked as Ideal which is the highest quality.

4. Most of zirconia marked with clarity as SI1 which is decent.

Visual representation:

We will use Boxplot and histogram to see distribution and pattern of continuous variables.



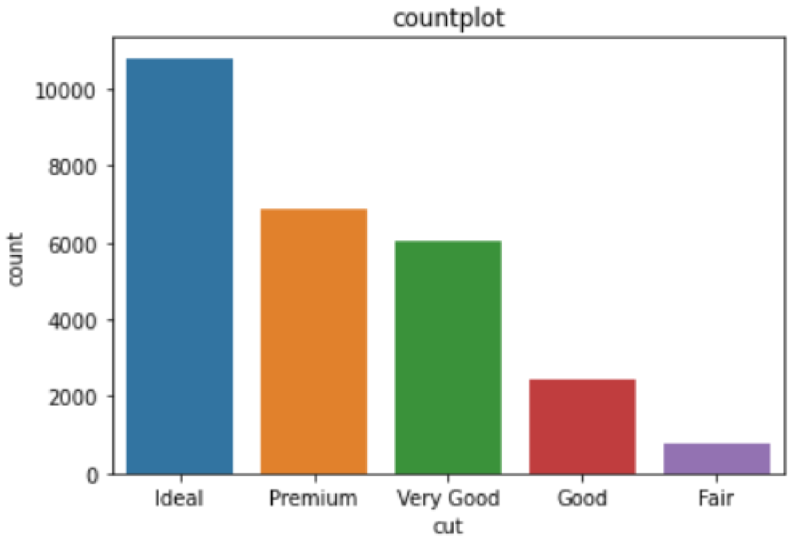


Insights:

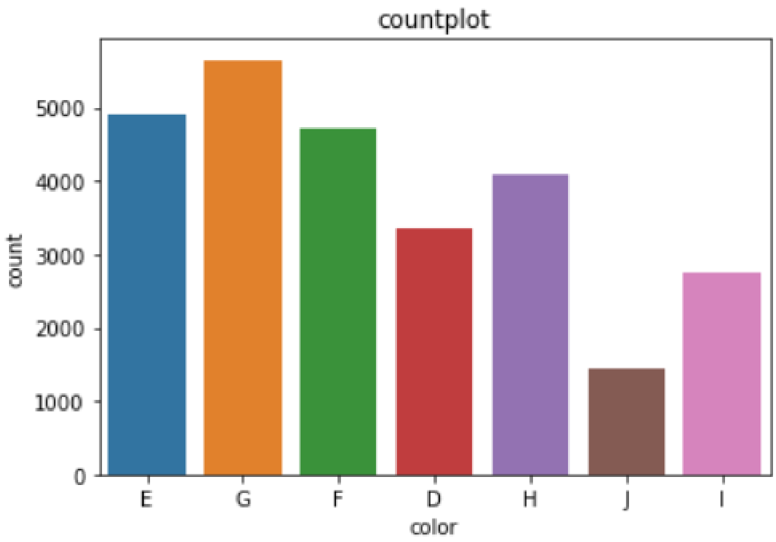
1. From the above box plots we can say that there are outliers present in the data.

2. For the variable 'depth' distribution is almost symmetric.

For Categorical variable we are using barplot

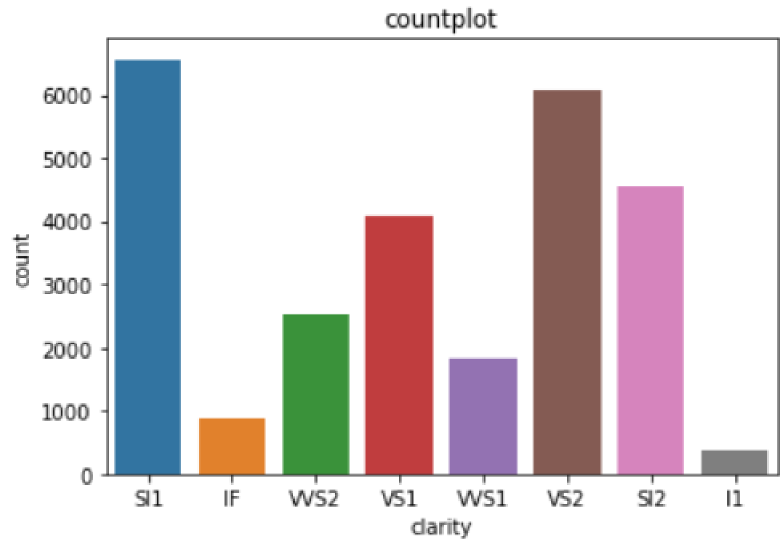


Best cut quality of zirconia is 'Ideal' and it holds maximum number of cubic other side 'Fair' is poor cut quality.



Most of the zirconia has color 'G' and color 'J' given to the least number of zirconia which is right because color 'J' is the worst color

Here count of zirconia having color 'D' requires improvement.

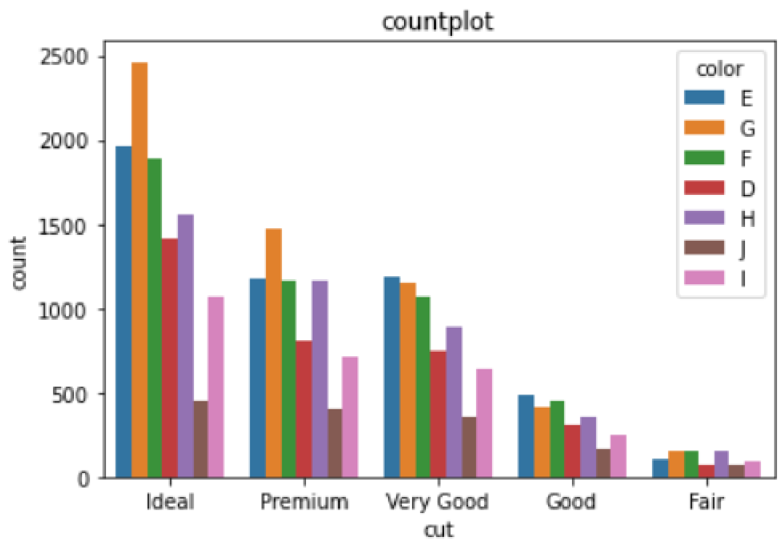


Clarity 'I1' has the minimum count and it also the worst clarity in the data where the other side Clarity 'SI1' has the maximum count and it's a decent clarity.

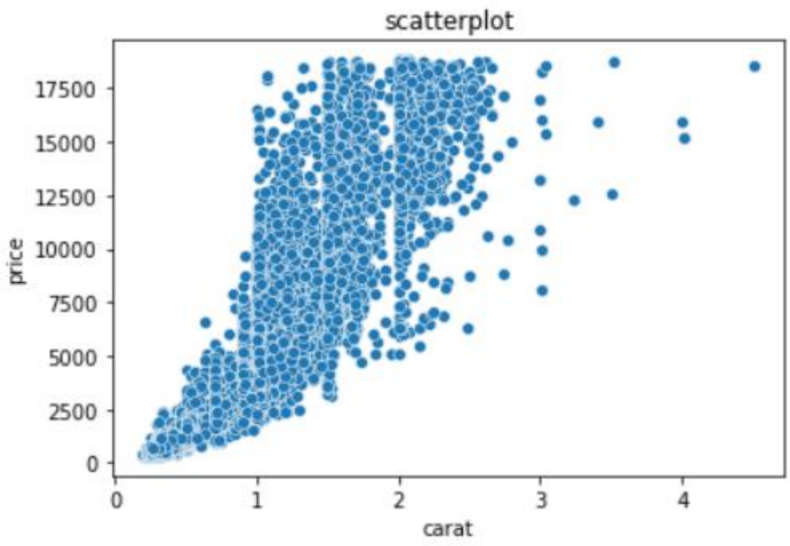
Best clarity is 'IF' but it has very less count whereas it must have maximum or close to maximum count to maintain good clarity of zirconia.

Bivariate Analysis

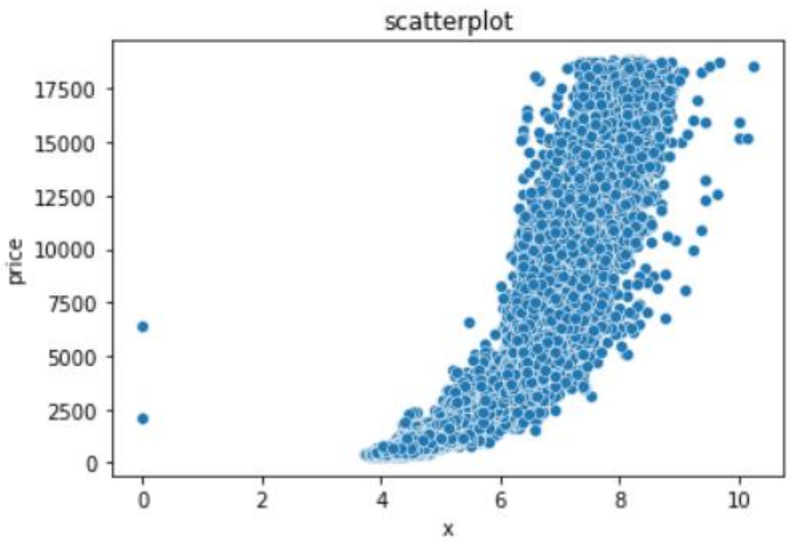
We will use countplot, boxplot and scatterplot to compare 2 variables.



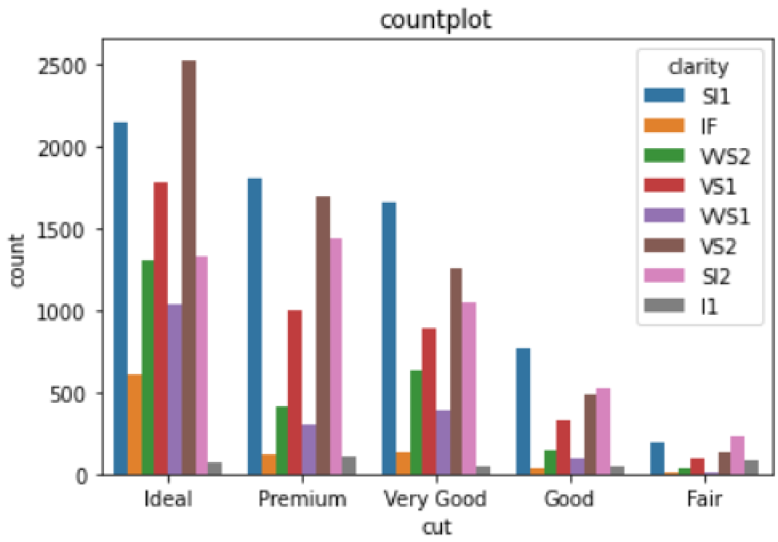
Mostly color G given into the all cut quality of zirconia.



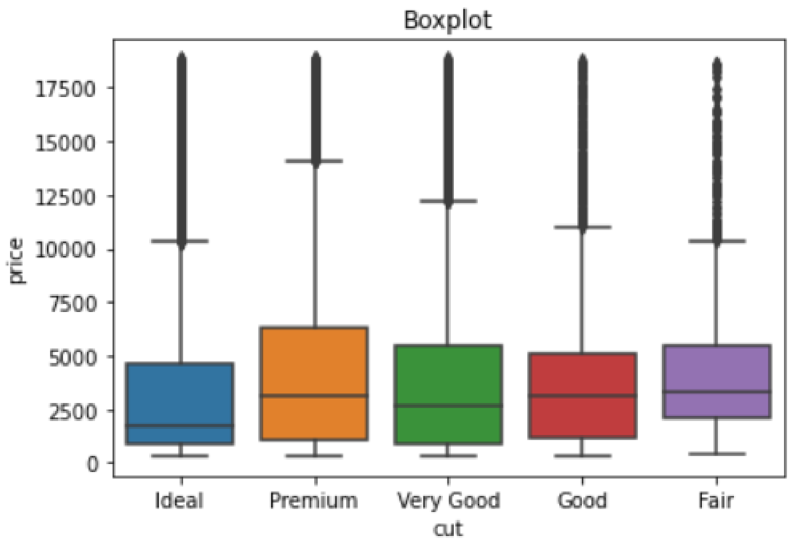
we can see that 'Carat' is positively correlated with 'price'.



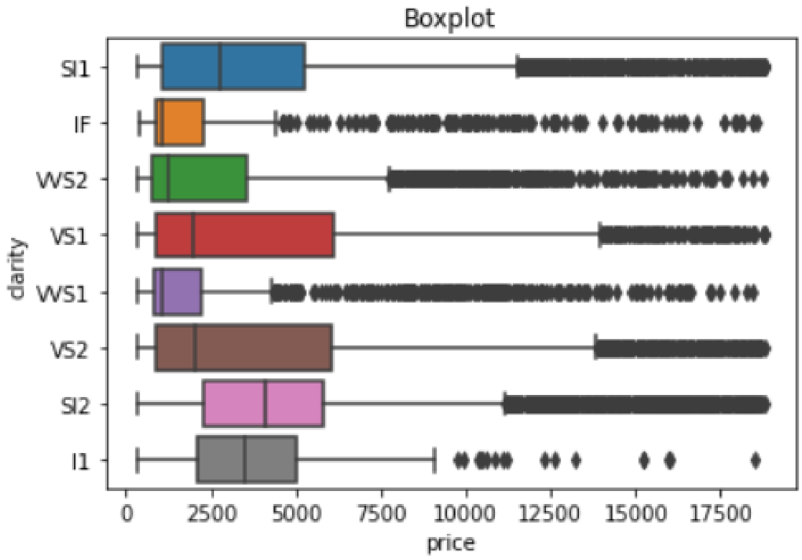
we can see that x (Length of the cubic zirconia) is positively correlated with price.



In Cut quality 'Fair' very few zirconia found with best Clarity 'IF' so this quality lacks top Clarity zirconia.



The zirconia diamonds with 'Premium' Cut are the most Expensive.



Most expensive diamond belongs to 'Clarity' S1.

Q1.2 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?

As we seen above that there is missing values in the column 'depth' and outliers also present in this variable so will replace null values with median.

Null Value check after imputation:

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

Unique values for categorical variables:

```
CUT :  5
Fair            780
Good           2435
Very Good      6027
Premium        6886
Ideal         10805
Name: cut, dtype: int64


COLOR :  7
J    1440
I    2765
D    3341
H    4095
F    4723
E    4916
G    5653
Name: color, dtype: int64


CLARITY :  8
I1       364
IF       891
VVS1    1839
VVS2    2530
VS1     4087
SI2     4564
VS2     6093
SI1     6565
Name: clarity, dtype: int64
```

Unique values looks fine as there is no repetition, sort abbreviation, ? .

Checking Values equal to 0 :

From descriptive summary We have seen that value equal to zero is only in columns 'x', 'y', 'z' so what these variables are? these variables indicate the dimension of a diamond and any of the parameters (Length, Width, Height) which shows the dimension cannot be zero and number of observations containing value=0 is very less so we will remove these.

After removing 0 values rows reduce to 26925.

combining ordinal variable:

All 3 categorical variables are ordinal as these follows an order like worst to best OR best to worst.

Combining ordinal variables in a way which gives us minimum number of groups

In variable 'CUT' we have 5 groups and we can separate these groups in 3 groups like average, good, best where Fair will be the part of group 'average' and good, Very Good in group 'good' and rest in group 'best' .

In variable 'color' D is the best color and J is the worst color and other color falls between these 2 which we can say in a alphabetic order best to worst like- D,E,F,G,H,I,J.

we can separate these in 4 group where J will be in group 'poor' & H,I in group 'average' .
F, G in group 'good' and rest in group 'best'.

In variable 'CLARITY' IF is the best and l1 is the worst so we can combine IF, VVS1 in group 'best' and VS2, Sl1, Sl2 in group 'average' and VVS2, VS1 in group 'Good' rest in group 'Worst'.

And Value counts of ordinal variables after combining groups:

```
Best       17685
Good        8461
Average      779
Name: cut, dtype: int64

Good       10372
Best        8257
Average     6856
Poor        1440
Name: color, dtype: int64

Average    17217
Good        6616
Best        2730
Worst        362
Name: clarity, dtype: int64
```

We have combined the group of ordinal variables and these were in an order so we ranked them with new variable in ordered form.

Q1.3 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?

Encoding the categorical variable.

We will use label encoding as our variable is in ordinal form.

Data types after encoding the categorical variables.

```
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   carat    26925 non-null  float64
 1   cut      26925 non-null  int32  
 2   color    26925 non-null  int32  
 3   clarity  26925 non-null  int32  
 4   depth    26925 non-null  float64
 5   table    26925 non-null  float64
 6   x        26925 non-null  float64
 7   y        26925 non-null  float64
 8   z        26925 non-null  float64
 9   price    26925 non-null  int64  
dtypes: float64(6), int32(3), int64(1)
```

Data types changed to integer after encoding.

Train Test Split:

Copy all the predictor variables into X Dataframe

Copy target into the y Dataframe.

We are using 70:30 ratio which means 70% data assign for training set and 30% for testing set.

After building Linear Regression model with default parameters, we got Coefficient of each Variables.

```
The coefficient for carat is 11005.921547059015
The coefficient for cut is 58.1048980122618
The coefficient for color is 95.98651587888155
The coefficient for clarity is 289.1842209725049
The coefficient for depth is -197.39424122917015
The coefficient for table is -101.25030342033497
The coefficient for x is -1297.1955148649827
The coefficient for y is 7.3566647440026
The coefficient for z is -31.348666735059165

The intercept for our model is 20283.187895261588

R square on training data: 0.8646606102865697

R square on testing data: 0.8677285505530588
```

86% of the variation in the variable y is explained by the predictors in the model for test set.

In this regression model we can see the R-square value on Training and Test data is close to each other

Linear regression Performance using sklearn model:

intercept for the model= 20283.187895261588
R square on training data= 0.8646606102865697
R square on testing data= 0.8677285505530588
RMSE on Training data= 1474.507368277916
RMSE on Testing data= 1473.1909463247393

We can see that train and test score is close to each other so no overfitting found in the model.

Linear Regression using statsmodels (OLS) :

OLS Regression Results

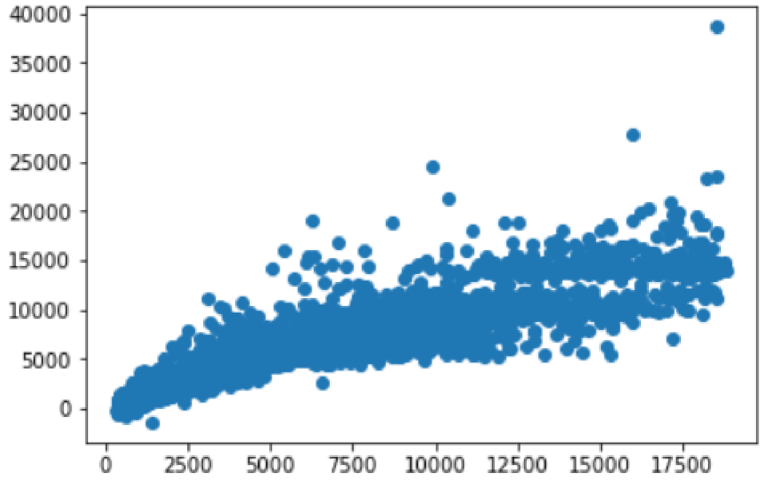
Dep. Variable:	price	**R-squared:**	0.865
Model:	OLS	**Adj. R-squared:**	0.865
Method:	Least Squares	**F-statistic:**	1.337e+04
Date:	Sun, 30 Oct 2022	**Prob (F-statistic):**	0.00
Time:	15:38:22	**Log-Likelihood:**	-1.6425e+05
No. Observations:	18847	**AIC:**	3.285e+05
Df Residuals:	18837	**BIC:**	3.286e+05
Df Model:	9		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.028e+04	741.488	27.355	0.000	1.88e+04	2.17e+04
carat	1.101e+04	112.389	97.927	0.000	1.08e+04	1.12e+04
cut	58.1049	21.379	2.718	0.007	16.201	100.009
color	95.9865	12.071	7.952	0.000	72.326	119.647
clarity	289.1842	12.253	23.601	0.000	265.167	313.201
depth	-197.3942	8.846	-22.314	0.000	-214.733	-180.055
table	-101.2503	5.223	-19.387	0.000	-111.487	-91.014
x	-1297.1955	61.417	-21.121	0.000	-1417.579	-1176.812
y	7.3567	29.003	0.254	0.800	-49.492	64.205
z	-31.3487	50.621	-0.619	0.536	-130.571	67.874

Omnibus:	4376.971	**Durbin-Watson:**	2.002
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	159274.110
Skew:	0.385	**Prob(JB):**	0
Kurtosis:	17.221	**Cond. No.**	5.87e-

The coefficients tell us how one unit change in X can affect y.

The sign of the coefficient indicates if the relationship is positive or negative.



We can see that in both model and model1(OLS based model) performance score is similar.

check for Multicollinearity

Multicollinearity is the presence of a strong correlation between the independent variables and We can check Multicollinearity with the VIF (Variance Inflation factor) score.

If VIF is 1 then no collinearities exist among the predictors and if VIF exceeds 5, we say there is moderate VIF, and if it is 10 or exceeding 10, it is signs of high multi-collinearity.

let's check the VIF of the predictors:

```
VIF values:

const       4763.503431
carat         24.687104
cut            1.034921
color          1.006263
clarity        1.045658
depth          1.322249
table          1.179995
x             41.124845
y             10.133346
z             11.626518
```

The VIF values indicate that the features carat, x, y, and z are correlated with one or more independent features.

To treat multicollinearity, we will have to drop one or more of the correlated features (carat, x, y, and z).

We will drop the variable that has the least impact on the adjusted R-squared of the model.

Let's remove/drop multicollinear columns one by one and observe the effect on our predictive model.

On dropping 'carat', adj. R-squared decreased by 0.069

This is a sharp decline indicates that 'carat' is an important predictor and shouldn't be removed.

On dropping 'x', adj. R-squared decreased by 0.004

On dropping 'y', 'z' adj. R-squared remains the same.

<u>Since there is no major effect on adj. R-squared after dropping the 'z', 'y', 'x' column, we can remove it from the training set</u>

VIF values After dropping variable 'Z'

```
VIF values:
const      4504.413857
carat        24.684807
cut           1.034003
color         1.006263
clarity       1.045434
depth         1.176757
table         1.179393
x            33.008558
y             9.993276
```

We know that 'carat' is an important predictor, so let's see the effect of dropping 'x' and 'y' now.

```
const      3704.195228
carat         1.071709
cut           1.031114
color         1.005016
clarity       1.034301
depth         1.109705
table         1.177175
```

<u>OLS summary after dropping x, y, z.</u>

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.859
Model:                            OLS   Adj. R-squared:                  0.859
Method:                 Least Squares   F-statistic:                 1.917e+04
Date:                Sun, 30 Oct 2022   Prob (F-statistic):               0.00
Time:                        15:38:23   Log-Likelihood:            -1.6462e+05
No. Observations:               18847   AIC:                         3.293e+05
Df Residuals:                   18840   BIC:                         3.293e+05
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       1.211e+04    666.771     18.169      0.000    1.08e+04    1.34e+04
carat       7989.5606     23.879    334.585      0.000    7942.756    8036.366
cut           69.8805     21.761      3.211      0.001      27.228     112.533
color         84.5029     12.302      6.869      0.000      60.391     108.615
clarity      323.7820     12.427     26.055      0.000     299.424     348.140
depth       -145.3873      8.264    -17.593      0.000    -161.585    -129.189
table       -104.0721      5.319    -19.565      0.000    -114.498     -93.646
==============================================================================
Omnibus:                     4305.974   Durbin-Watson:                   2.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            62836.287
Skew:                           0.696   Prob(JB):                         0.00
Kurtosis:                      11.836   Cond. No.                     5.14e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.14e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

After dropping the features causing strong multicollinearity and the statistically insignificant ones, our model performance hasn't dropped sharply (adj. R-squared has dropped from 0.865 to 0.859). This shows that these variables did not have much predictive power.

After treating *multicollinearity* Linear regression Performance using OLS model.

RMSE for training set:1503.7314672115162

RMSE for testing set:1502.3943677899188

R-squared: 0.859

Adj. R-squared: 0.859

<u>We can see that RMSE on the train and test sets are comparable. So, our model is not suffering from overfitting and OLS giving us list of important variables.</u>

The final Linear Regression equation is:
Price = (12114.40748719299) * Intercept + (7989.560637532106) * carat + (69.88053100665627) * cut + (84.50288930241051) * color + (323.7819678919295) * clarity + (-145.38733694822466) * depth + (-104.07209352527796) * table

Q1.4 Inference: Basis on these predictions, what are the business in sights and recommendations?

Insights:

We tried multiple models with different variable and at the end we got 6 independent variable which are important for our prediction.

1. When carat increases by 1 unit, diamond price increases by 7989.56 units, keeping all other predictors constant.

2. When cut increases by 1 unit, diamond price increases by 69.88 units, keeping all other predictors constant.

3. When color increases by 1 unit, diamond price increases by 84.50 units, keeping all other predictors constant.

4. When clarity increases by 1 unit, diamond price increases by 436.44 units, keeping all other predictors constant.

As per model these 4 variables are most important variable 'Carat', 'Cut', 'color', 'clarity' for predicting the diamond price.

We also have negative co-efficient values -145.38 for 'depth' & -104.07 for 'table' shows that these variables are inversely proportion to diamond price.

<u>We prepare our model with in different steps which we listed below:</u>

1. First we split the data in 30:70 where 70 % is for train and 30% is for test.

2. Then we build Linear regression model by using Sklearn library and calculate Rmse, R squared values.

3. we observed that Rsquare, RMSE was almost close for train and test data.

4. we then build regression model from statsmodels and check multicollinearity in the data and found multicollinearity in the data by using VIF.

5. we reduce the multicollinearity from the data by dropping variables and kept an eye on Adj Rsquare.

We can see R-squared: and Adj. R-squared: 0.859 are same. The overall P value is less than alpha.

Recommendations:

1. The Gem Stones company should focus more on the features 'Carat', 'Cut', 'color', 'clarity' as these are most important for predicting the price.

2. The zirconia diamonds with 'Premium' Cut are the most Expensive so need to check on sale of this diamond if it's good then it great as it's expensive one.

3. 'Depth' has negative impact on diamond price so this needs to be low as much as possible.

4. 'table' which shows Width of diamond needs to be minimum as it will reduce the price of diamond.

5. 'carat' is the most important feature out of all features as it is highly related to price and higher the carat weight of a diamond will have higher price.

6. The Diamond's with clarity 'VS1' &'VS2' are the most Expensive So these two categories are very important also most expensive diamond is from clarity SI1 and this is the 3 most expensive diamond clarity after 'VS1' &'VS2'.

Problem 2

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 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.

Data Description:

1. Holiday_Package : Opted for Holiday Package yes/no?
2. Salary : Employee salary; in numbers
3. age : Age in years
4. edu : Years of formal education; in numbers
5. no_young_children: The number of young children (younger than 7 years)
6. no_older_children: Number of older children
7. foreign : foreigner Yes/No

Sample of 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

Exploratory Data Analysis:

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

Column	Dtype
Holliday_Package	object
Salary	int64
age	int64
educ	int64
no_young_children	int64
no_older_children	int64
foreign	object

we have 2 data types as 5 integer data types and 2 object data types.

Checking for missing values in the dataset:

```
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Holliday_Package   872 non-null    object
 1   Salary             872 non-null    int64
 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    object
dtypes: int64(5), object(2)
```

No missing value found.

Q2.1 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?

As we have seen above that there are no missing values present in the data

Column Unnamed: 0 contains serial number so we can remove it and here is the sample data without this column.

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

Number of rows 872 and number of columns 7.

Duplicate Values

No Duplicate value found

Data Types looks good as per data dictionary.

We are using describe function to get descriptive summary of the data and here is the sample.

	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

Salary spread to the wide range.

maximum age is 62 and minimum is 20

We can see that there are no anomalies found in data.

Unique counts of categorical variable

```
no     471
yes    401
Name: Holliday_Package, dtype: int64
```

```
no     656
yes    216
Name: foreign, dtype: int64
```

Unique values looks fine as there is no repetition, sort abbreviation, ? .

Data Visualization

Univariate Analysis

Non visual representation:

Using describe function to get descriptive analysis.

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Holliday_Package	872	2	no	471	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Salary	872.0	NaN	NaN	NaN	47729.172018	23418.668531	1322.0	35324.0	41903.5	53469.5	236961.0
age	872.0	NaN	NaN	NaN	39.955275	10.551675	20.0	32.0	39.0	48.0	62.0
educ	872.0	NaN	NaN	NaN	9.307339	3.036259	1.0	8.0	9.0	12.0	21.0
no_young_children	872.0	NaN	NaN	NaN	0.311927	0.61287	0.0	0.0	0.0	0.0	3.0
no_older_children	872.0	NaN	NaN	NaN	0.982798	1.086786	0.0	0.0	1.0	2.0	6.0
foreign	872	2	no	656	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Insights:

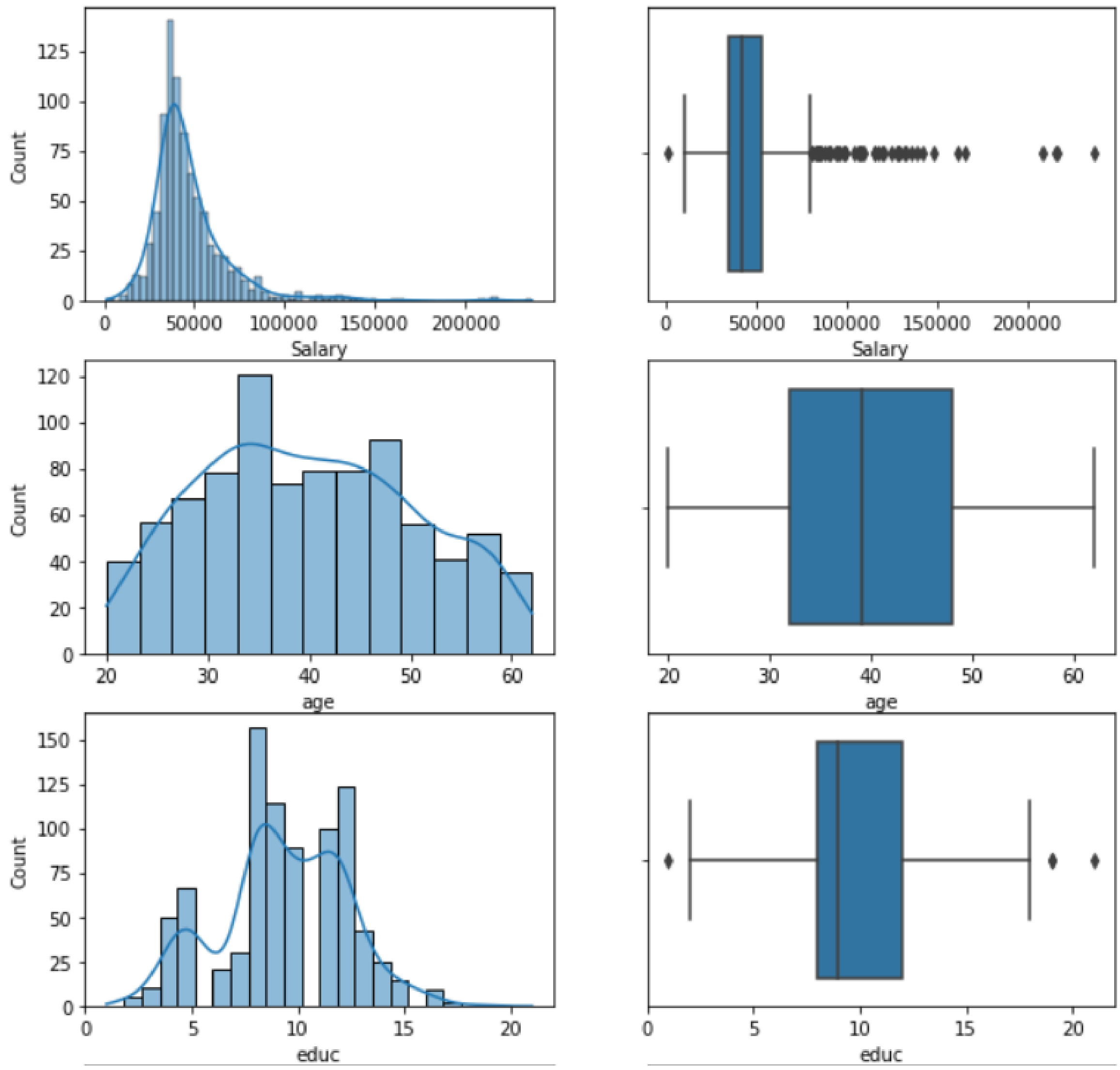
1. Maximum Salary of an employee is 236961.0 and minimum is 1322.0 so we can say that Salary spread in wide range.

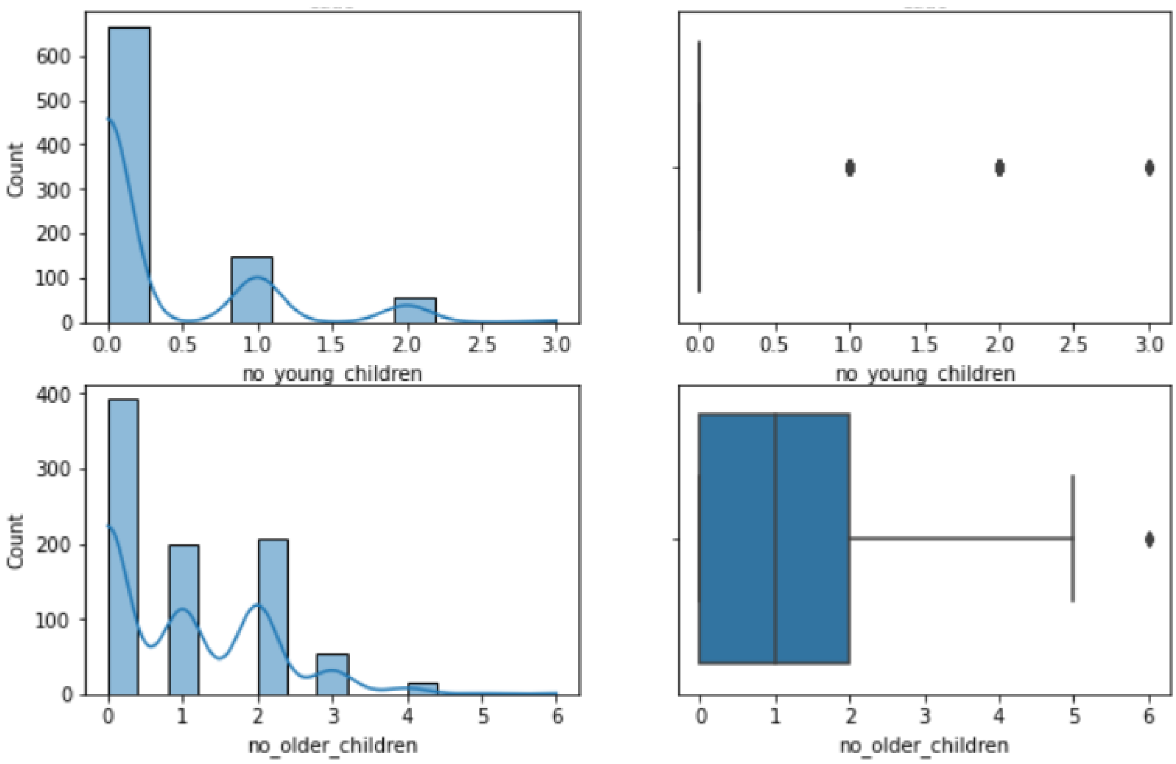
2. Mostly employee did not opt for Holliday_Package which is a concern.

3. 75% of employee's age is less than or equal to 48 years.

Visual representation:

We will use Boxplot and histogram to see distribution and pattern of continuous variables.


Count
Salary
Count
age
Count
educ
Salary
age
educ

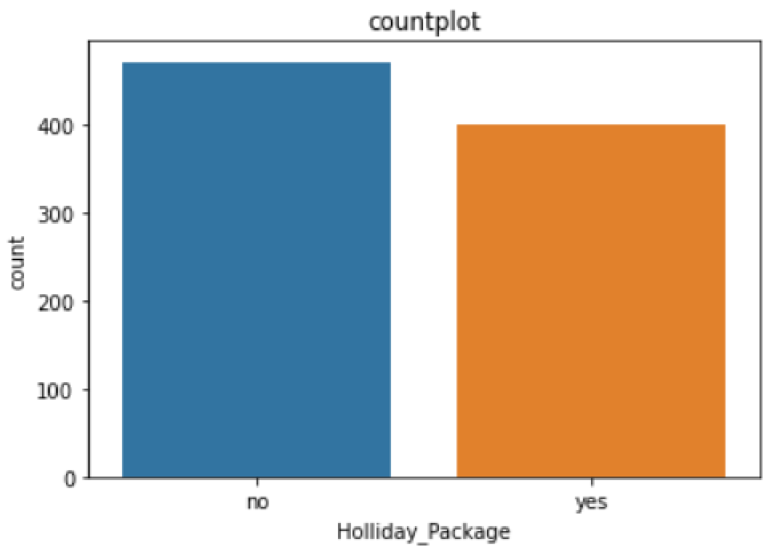


Insights:

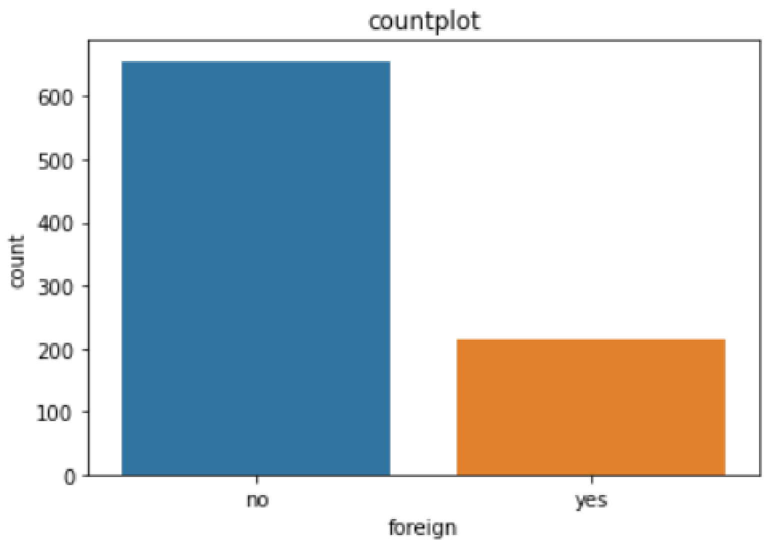
1. From the above box plots we can say that there are outliers present in the data.

2. For the variable 'age' distribution is almost symmetric.

For Categorical variable we are using barplot.



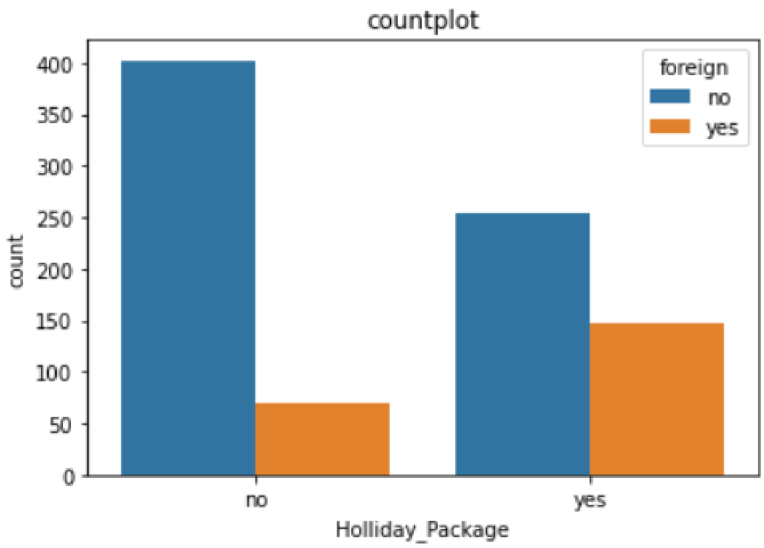
As we observed earlier number of Opted for 'Holliday_Package' is less than number of not-Opted.



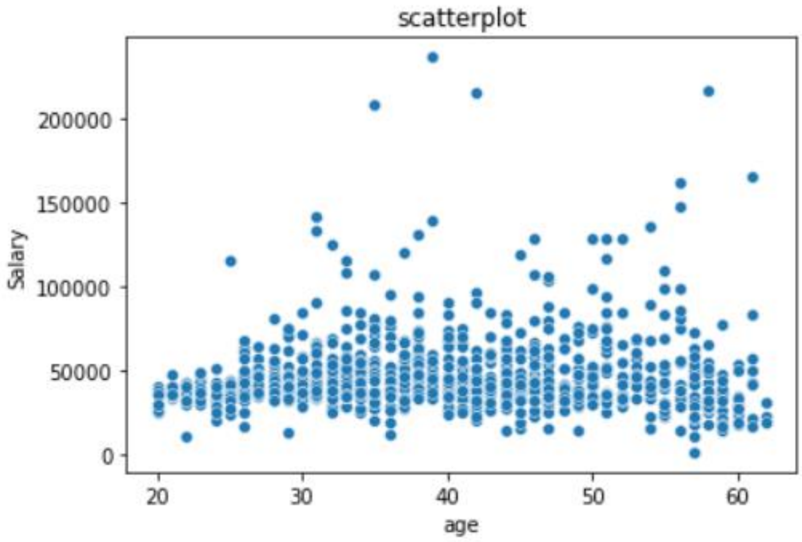
Mostly non foreigner present in the data.

Bivariate Analysis:

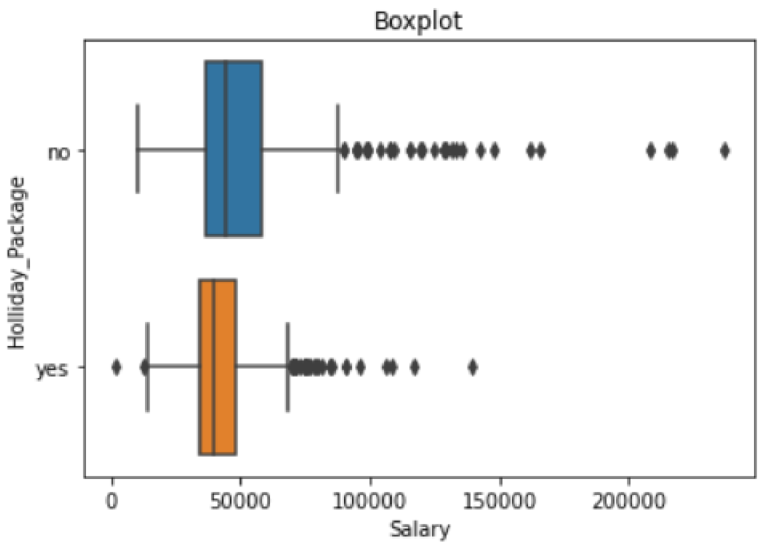
We will use countplot, boxplot and scatterplot to compare 2 variables.



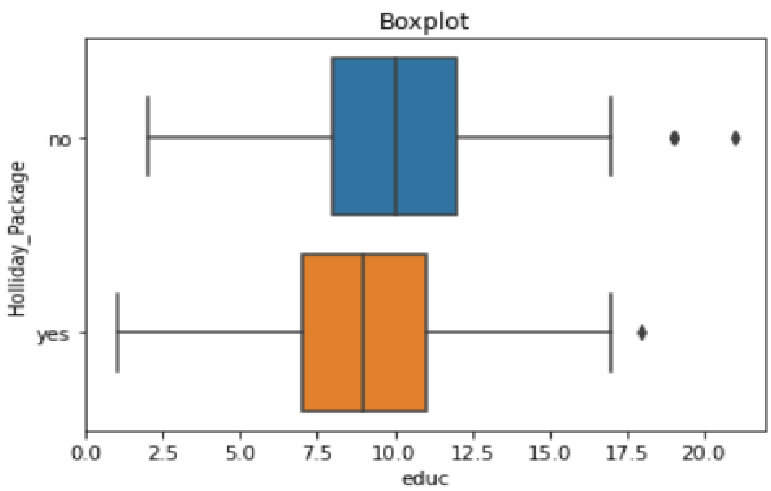
We can say that number of foreigners who stand for yes highly opted for Holliday_Package.



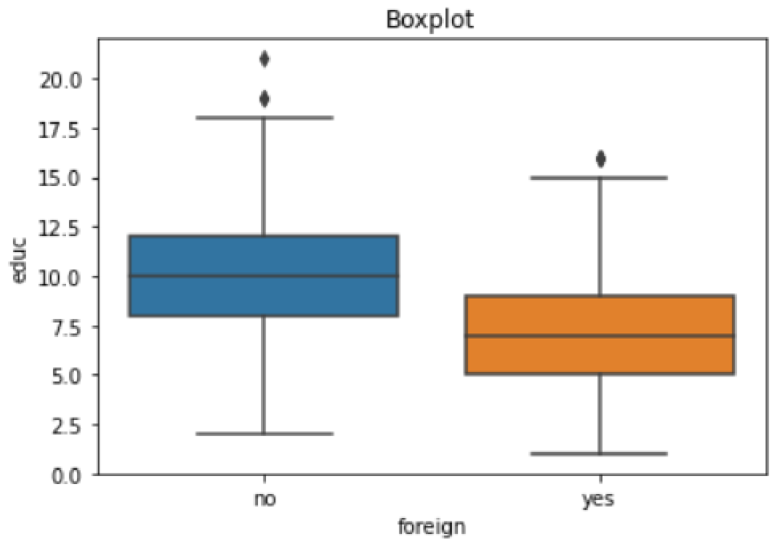
We found that age is not related to salary which is not common because generally salary and age are correlated.



We can see that those who have higher salary did not opted for Holliday_Package.



We can see that those who have high Years of formal education did not opted for Holliday_Package.



Mostly high Years of formal education is not foreigner.

Q2.2 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)?

Encoding the data:

We are using one hot encoding and here is the sample after encoding.

	Salary	age	educ	no_young_children	no_older_children	Holliday_Package_yes	foreign_yes
0	48412	30	8	1	1	0	0
1	37207	45	8	0	1	1	0
2	58022	46	9	0	0	0	0
3	66503	31	11	2	0	0	0
4	66734	44	12	0	2	0	0

0 In column (Holliday_Package_yes) means holiday package not opted and 1 means holiday package opted similarly with column (foreign_yes) 0 means not foreigner and 1 means foreigner.

Train Test Split:

Copy all the predictor variables into X Dataframe

Copy target into the y Dataframe.

We are using 70:30 ratio which means 70% data assign for training set and 30% for testing set.

Building Logistic Regression model

We have built multiple models with different set of parameters so with first model named model1 we got following values.

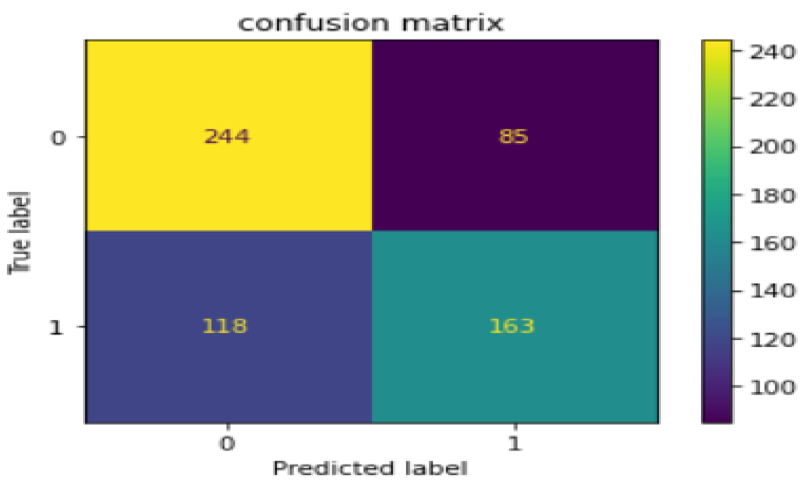
Getting the Predicted Classes and Probs:

	0	1
0	0.685349	0.314651
1	0.539469	0.460531
2	0.697042	0.302958
3	0.496348	0.503652
4	0.557723	0.442277

It's same for both train and test data.

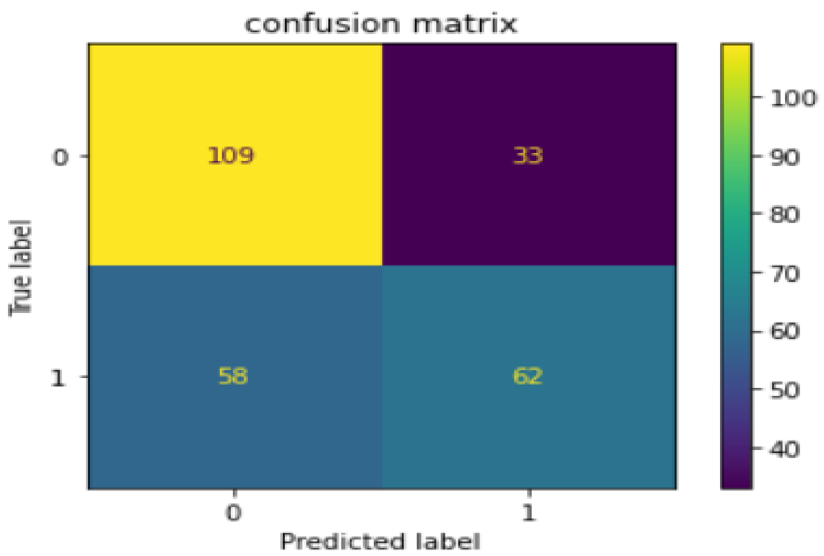
Model1 Evaluation:

Confusion Matrix & classification report for the training data



	precision	recall	f1-score	support
0	0.67	0.74	0.71	329
1	0.66	0.58	0.62	281
accuracy			0.67	610
macro avg	0.67	0.66	0.66	610
weighted avg	0.67	0.67	0.66	610

Confusion Matrix & classification report for the test data



	precision	recall	f1-score	support
0	0.65	0.77	0.71	142
1	0.65	0.52	0.58	120
accuracy			0.65	262
macro avg	0.65	0.64	0.64	262
weighted avg	0.65	0.65	0.65	262

We used GridsearchCV to try out impact of different parameters on our model and finally we are going with one model which gives us slightly better result and we called this model a final model named as model_F

We got following co-efficient from our model_F:

```
array([[-1.60772099e-05, -4.91773354e-02,  6.92519913e-02,
        -1.21965216e+00, -7.97142993e-03,  1.25278311e+00]])
```

Building linear discriminant (LDA) model

First, we have built LDA with default parameter and got following scores.

Classification Report of the training data:

	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 of the test data:

	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

We built our model with default parameters and got Accuracy more than 65 now will use different combination of parameters and will see if our model score improves.

We got following classification matrix from one of the models named as- model7

```
Classification Report of the training data:

              precision    recall  f1-score   support

           0       0.68      0.69      0.68       329
           1       0.63      0.63      0.63       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 of the test data:

              precision    recall  f1-score   support

           0       0.69      0.71      0.70       142
           1       0.65      0.62      0.64       120

    accuracy                           0.67       262
   macro avg       0.67      0.67      0.67       262
weighted avg       0.67      0.67      0.67       262
```

We are going with this model because train test score is not much different and score is improving from train to test.

<u>We got following co-efficient from our model:</u>

array([[-1.44748476e-05, -5.73218187e-02, 6.09200685e-02,
    -1.28700142e+00, -3.23170095e-02, 1.29994632e+00]])

Q2.3 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?

<u>For Logistic Regression Confusion Matrix, Classification Report, AUC and ROC for the training data:</u>

<u>Confusion Matrix:</u>

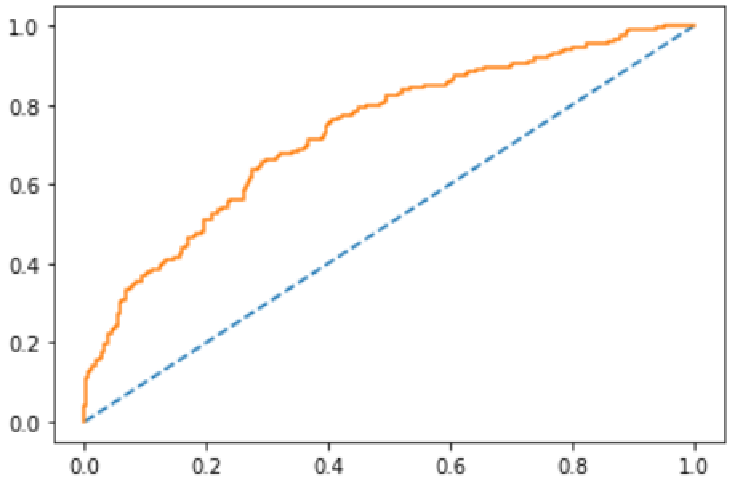
```
array([[247,  82],
       [122, 159]], dtype=int64)
```

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
```

ROC & AUC:

```
AUC: 0.715
[<matplotlib.lines.Line2D at 0x2a9d5e78070>]
```



For Logistic Regression Confusion Matrix, Classification Report, AUC and ROC for the testing data:

Confusion Matrix:

```
array([[110,  32],
       [ 57,  63]], dtype=int64)
```

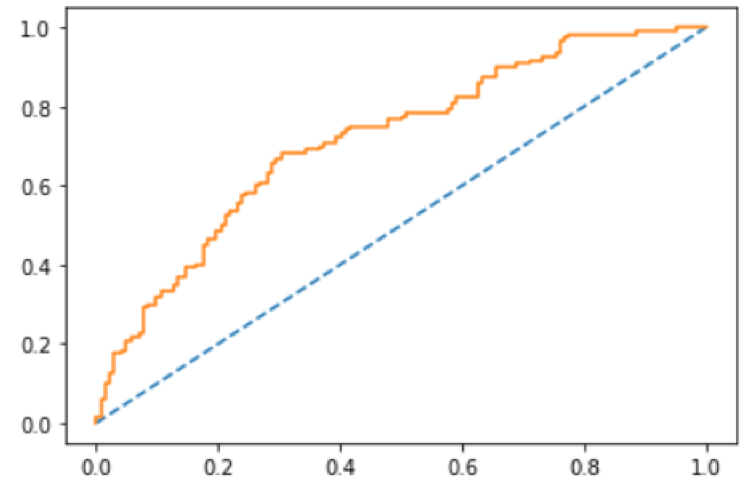
Classification Report:

```
              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
```

<u>ROC & AUC:</u>

```
AUC: 0.718
```

```
[<matplotlib.lines.Line2D at 0x2a9db763a90>]
```



LR Conclusion

Train Data:

AUC- 72%

Accuracy- 67%

Precision- 65%

f1-Score- 60%

test Data:

AUC- 72%

Accuracy- 66%

Precision- 68%

f1-Score- 59%

For LDA Confusion Matrix, Classification Report, AUC and ROC for the training data:

Confusion Matrix:

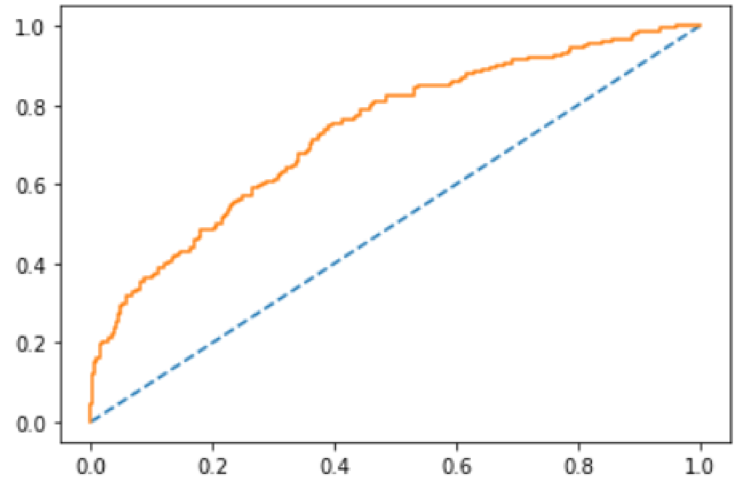
```
array([[226, 103],
       [105, 176]], dtype=int64)
```

Classification Report:

```
Classification Report of the training data:

              precision    recall  f1-score   support

           0       0.68      0.69      0.68       329
           1       0.63      0.63      0.63       281

    accuracy                           0.66       610
   macro avg       0.66      0.66      0.66       610
weighted avg       0.66      0.66      0.66       610
```

ROC & AUC:

```
AUC: 0.733

[<matplotlib.lines.Line2D at 0x2a9da28fee0>]
```



For LDA Confusion Matrix, Classification Report, AUC and ROC for the testing data:

Confusion Matrix:

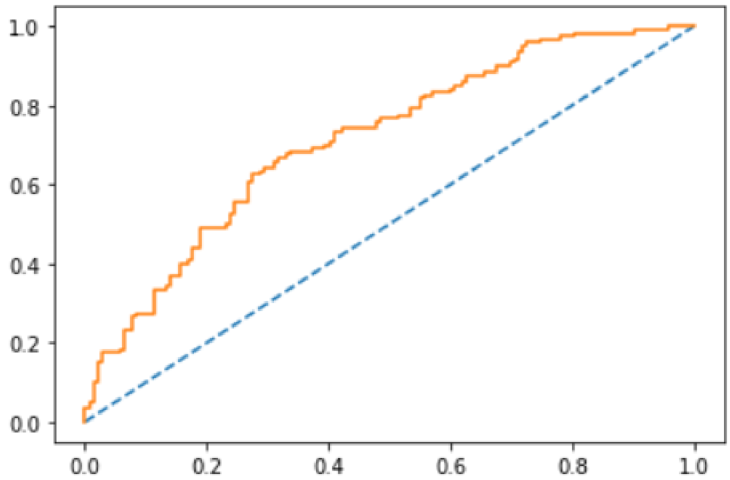
```
array([[101,  41],
       [ 45,  75]], dtype=int64)
```

Classification Report:

```
Classification Report of the training data:

              precision    recall  f1-score   support

           0       0.69      0.71      0.70       142
           1       0.65      0.62      0.64       120

    accuracy                           0.67       262
   macro avg       0.67      0.67      0.67       262
weighted avg       0.67      0.67      0.67       262
```

ROC & AUC:

```
AUC: 0.714
[<matplotlib.lines.Line2D at 0x2a9d8382700>]
```



LDA Conclusion

Train Data:

AUC- 73%

Accuracy- 66%

Precision- 63%

f1-Score- 63%

test Data:

AUC- 71%

Accuracy- 67%

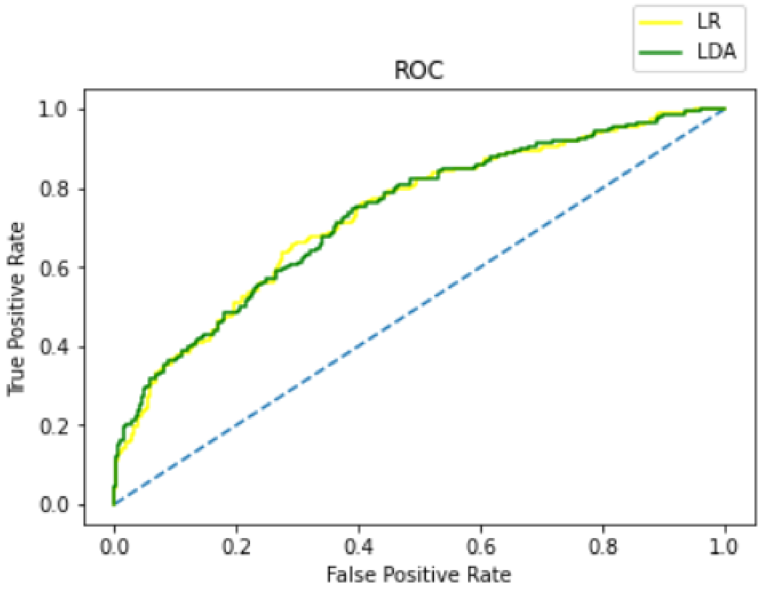
Precision- 65%

f1-Score- 64%

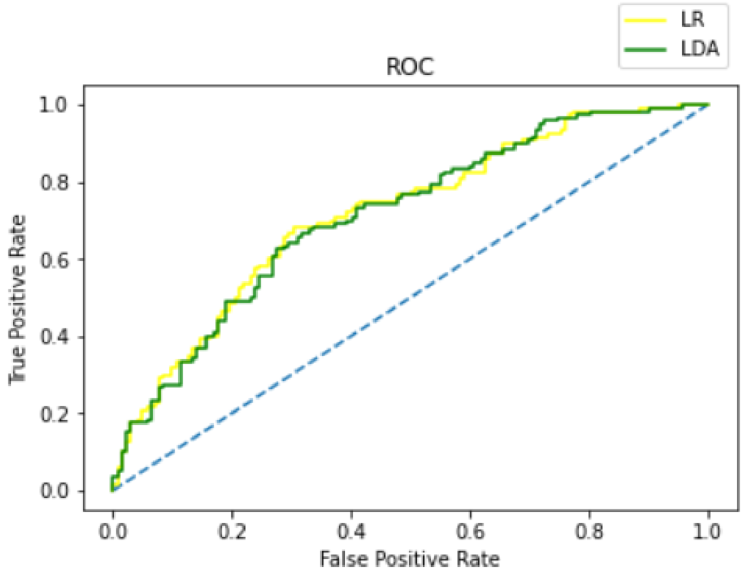
Comparison of the performance metrics from the 2 models:

	LR Train	LR Test	LDA Train	LDA Test
Accuracy	0.67	0.66	0.66	0.67
AUC	0.72	0.72	0.73	0.71
Recall	0.57	0.53	0.63	0.62
Precision	0.66	0.66	0.63	0.65
F1 Score	0.61	0.59	0.63	0.64

ROC Curve for the 2 models on the training data:



ROC Curve for the 2 models on the testing data:



Out of the 2 models, LDA has slightly better performance than the Logistic Regression (LR) in terms of overall accuracy and difference between train and test score is very minimum in LDA.

Co-efficient from LR model.

```
array([[-1.60772099e-05, -4.91773354e-02,  6.92519913e-02,
        -1.21965216e+00, -7.97142993e-03,  1.25278311e+00]])
```

Co-efficient from LDA model.

```
array([[-1.44748476e-05, -5.73218187e-02,  6.09200685e-02,
        -1.28700142e+00, -3.23170095e-02,  1.29994632e+00]])
```

As per our model Education and Foreigner is very important features for prediction and other features also related to dependent variable y but in negative direction.

Q2.4 Inference: Basis on these predictions, what are the insights and Recommendations?

Insights:

1. We have seen that LDA is better model with 0.65 precision in test data.
2. we want to reduce FP as we don't want that we predict employee will opt but in actual he didn't.
3. We need to go for higher number of educations as this is important variable.
4. 65% of employee who did not opted for Holiday Package are correctly predicted.
5. Out of all employees who actually did not opted , 62% of employees who didn't opted have been predicted correctly.
6. Accuracy, AUC, Precision and Recall for test data is almost in line with training data. This proves no overfitting or underfitting has happened, and overall, the model is a decent model for classification

Recommendations:

1. Mostly who opted for Holliday_Package is between salary range 30k to 60k so need to focus on other salary range that why they are not opting for packages.
2. non foreigner less opted for package so need to provide additional benefit, resources to them.
3. The number of opted for Holliday_Package decreases when number of children increases so company should provide attractive packages which has good benefits for children.
4. Company should focus on age more than 40 and need to check how they can attract them for Holliday_Package
5. Those we have higher salaries not opting for Holliday_Package so need to add some benefits, offers and tour location to attract higher salary employees.

6. Packages provided by company not seems to be in expensive range they can add some luxury/premium packages for higher salary employees.

THE END