Predictive Modelling Project

SUBMITTED BY: RISHAB SINGLA

Table of Contents

List Of tables

Table of figures

Problem 1: Linear Regression

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.

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	Color 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, Sl1, Sl2, l1
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.

Question 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.

Answer 1.1

The dataset consists of 26967 rows and 11 columns. 3 columns are of datatype object namely cut, clarity and color. All other columns are of continuous nature and datatype as int or float.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 26967 entries, 0 to 26966
Data columns (total 11 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  26967 non-null  int64
 1   carat       26967 non-null  float64
 2   cut         26967 non-null  object
 3   color       26967 non-null  object
 4   clarity     26967 non-null  object
 5   depth       26270 non-null  float64
 6   table       26967 non-null  float64
 7   x           26967 non-null  float64
 8   y           26967 non-null  float64
 9   z           26967 non-null  float64
 10  price       26967 non-null  int64
dtypes: float64(6), int64(2), object(3)
memory usage: 2.3+ MB
```

Figure 1: Dataset Information

The figure below shows the description of the data. The range of each variable is the min value to the max value. Here it is observed that min values for variables x,y,z is zero which seems to be incorrect as each diamond will have an x,y,z as they are dimensions of the diamond. There seems to be a large difference in 75 percentile and max values which suggests there are outliers in the data. Carat seems to be one of the most important factors affecting the price of the diamond but a lot more analysis is required for that. The price of diamonds varies from 326 to 18818.

	Unnamed: 0	carat	depth	table	x	y	z	price
count	26967.000000	26967.000000	26270.000000	26967.000000	26967.000000	26967.000000	26967.000000	26967.000000
mean	13484.000000	0.798375	61.745147	57.456080	5.729854	5.733569	3.538057	3939.518115
std	7784.846691	0.477745	1.412860	2.232068	1.128516	1.166058	0.720624	4024.864666
min	1.000000	0.200000	50.800000	49.000000	0.000000	0.000000	0.000000	326.000000
25%	6742.500000	0.400000	61.000000	56.000000	4.710000	4.710000	2.900000	945.000000
50%	13484.000000	0.700000	61.800000	57.000000	5.690000	5.710000	3.520000	2375.000000
75%	20225.500000	1.050000	62.500000	59.000000	6.550000	6.540000	4.040000	5360.000000
max	26967.000000	4.500000	73.600000	79.000000	10.230000	58.900000	31.800000	18818.000000

Figure 2: Dataset Description

There are 697 null values present in the depth column

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

Figure 3: Null Values

The Unnamed: 0 has been dropped as it is of no importance.

	carat	cut	color	clarity	depth	table	x	y	z	price
26	0.34	Ideal	D	SI1	NaN	57.0	4.50	4.44	2.74	803
86	0.74	Ideal	E	SI2	NaN	59.0	5.92	5.97	3.52	2501
117	1.00	Premium	F	SI1	NaN	59.0	6.40	6.36	4.00	5292
148	1.11	Premium	E	SI2	NaN	61.0	6.66	6.61	4.09	4177
163	1.00	Very Good	F	VS2	NaN	55.0	6.39	6.44	3.99	6340
...	...	...	...	...	...	...	...	...	...	...
26848	1.22	Very Good	H	VS1	NaN	59.0	6.91	6.85	4.29	7673
26854	1.29	Premium	I	VS2	NaN	58.0	7.12	7.03	4.27	6321
26879	0.51	Very Good	E	SI1	NaN	58.0	5.10	5.13	3.12	1343
26923	0.51	Ideal	D	VS2	NaN	57.0	5.12	5.09	3.18	1882
26960	1.10	Very Good	D	SI2	NaN	63.0	6.76	6.69	3.94	4361

697 rows × 10 columns

Figure 4: Records with Null Values

There are zeroes present in the dataset in variables ‘x’,’y’,’z’ which makes no sense as each diamond will have certain length, breadth and width so these records are not correct and the corresponding mean values of these variables have been imputed to the zeroes.

	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

Figure 5: Zero Value Records

There are 34 duplicate records in the dataset. The below figure shows the 34 duplicated records.

	carat	cut	color	clarity	depth	table	x	y	z	price
4756	0.35	Premium	J	VS1	62.4	58.0	5.67	5.64	3.53	949
6215	0.71	Good	F	SI2	64.1	60.0	0.00	0.00	0.00	2130
8144	0.33	Ideal	G	VS1	62.1	55.0	4.46	4.43	2.76	854
8919	1.52	Good	E	I1	57.3	58.0	7.53	7.42	4.28	3105
9818	0.35	Ideal	F	VS2	61.4	54.0	4.58	4.54	2.80	906
10473	0.79	Ideal	G	SI1	62.3	57.0	5.90	5.85	3.66	2898
10500	1.00	Premium	F	VVS2	60.6	54.0	6.56	6.52	3.96	8924
12894	1.21	Premium	D	SI2	62.5	57.0	6.79	6.71	4.22	6505
13547	0.43	Ideal	G	VS1	61.9	55.0	4.84	4.86	3.00	943
13783	0.79	Ideal	G	SI1	62.3	57.0	5.90	5.85	3.66	2898
14389	0.60	Premium	D	SI2	62.0	57.0	5.43	5.35	3.34	1196
14410	1.00	Very Good	D	SI1	63.1	56.0	6.34	6.30	3.99	5645
15798	0.90	Very Good	I	VS2	58.4	62.0	6.29	6.35	3.69	3334
16852	0.79	Ideal	G	SI1	62.3	57.0	5.90	5.85	3.66	2898
17263	1.04	Premium	I	SI2	62.0	57.0	6.53	6.47	4.03	3774
18025	1.51	Good	I	SI1	63.8	57.0	7.21	7.18	4.59	6046
18777	0.32	Premium	H	VS2	60.6	58.0	4.47	4.44	2.70	648
18837	1.01	Premium	H	VS1	61.2	61.0	6.44	6.41	3.93	5294
19731	0.30	Good	J	VS1	63.4	57.0	4.23	4.26	2.69	394
19877	2.01	Premium	I	VS2	60.3	62.0	8.13	8.08	4.89	15939
20301	0.30	Ideal	H	SI1	62.2	57.0	4.26	4.29	2.66	450
20760	1.80	Ideal	H	VS1	62.3	56.0	7.79	7.76	4.84	15105
22322	2.05	Premium	I	SI2	62.0	58.0	8.13	8.08	5.02	9850
22488	2.42	Premium	J	VS2	61.3	59.0	8.61	8.58	5.27	17168
22583	0.33	Ideal	F	IF	61.2	56.0	4.47	4.49	2.74	1240
23458	2.66	Good	H	SI2	63.8	57.0	8.71	8.65	5.54	16239
23564	1.50	Premium	F	SI2	58.5	60.0	7.52	7.48	4.39	7644
24351	2.50	Fair	H	SI2	64.9	58.0	8.46	8.43	5.48	13278
24816	1.50	Good	G	SI2	57.5	63.0	7.53	7.49	4.32	6006
25268	1.20	Premium	I	VS2	62.6	58.0	6.77	6.72	4.22	5699
25759	0.30	Ideal	G	IF	62.1	55.0	4.32	4.35	2.69	863
25941	0.51	Premium	F	SI2	58.1	59.0	5.26	5.24	3.05	1052
26191	2.54	Very Good	H	SI2	63.5	56.0	8.68	8.65	5.50	16353
26530	0.41	Ideal	G	IF	61.7	56.0	4.77	4.80	2.95	1367

Figure 6: Duplicated Records

Univariate Analysis:

Depth

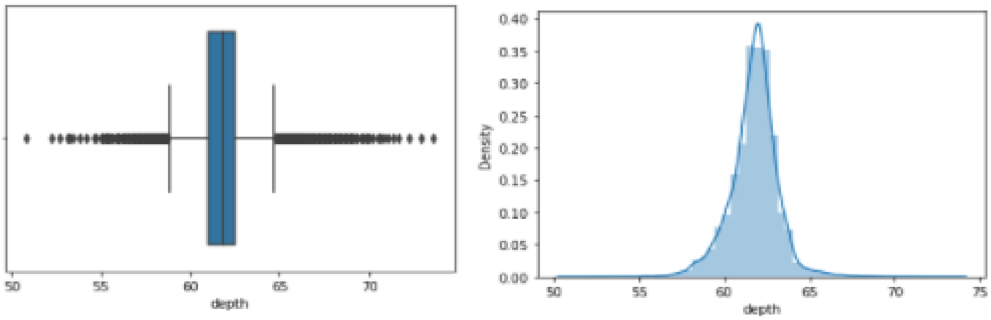

Figure 7: Variable 'depth'

Observation: The boxplot of depth variable shows there are outliers on both sides. The distribution of the depth variable is almost normally distributed.

Carat

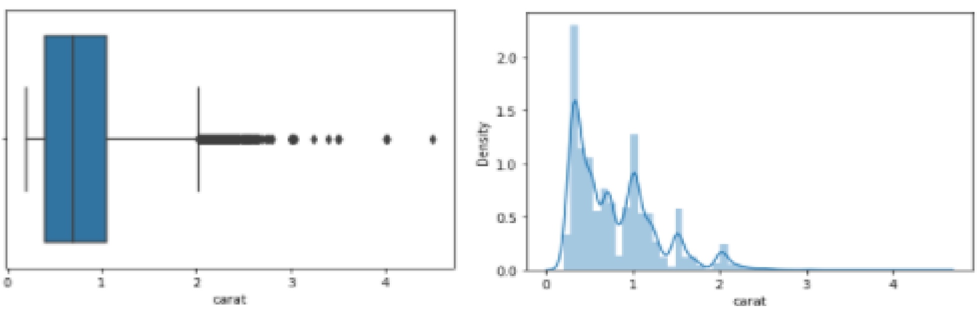

Figure 8: Variable 'carat'

Observation: The boxplot of the carat variable shows a lot of outliers. The distribution plot shows that the distribution is right skewed.

Table

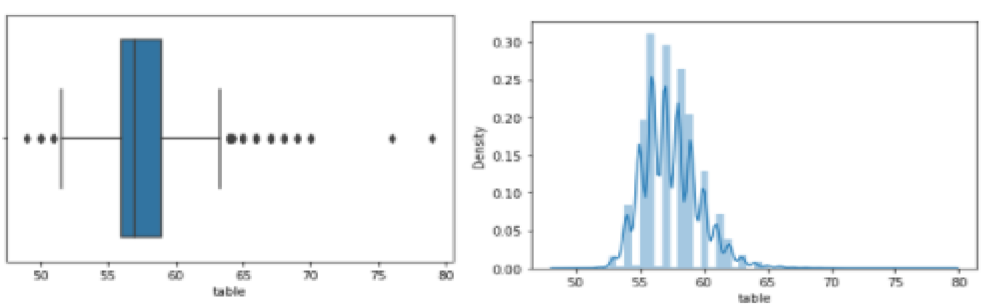

Figure 9: Variable 'table'

Observation: The boxplot of the variable table shows that there are outliers on both ends. The distribution shows that table is right skewed.

X

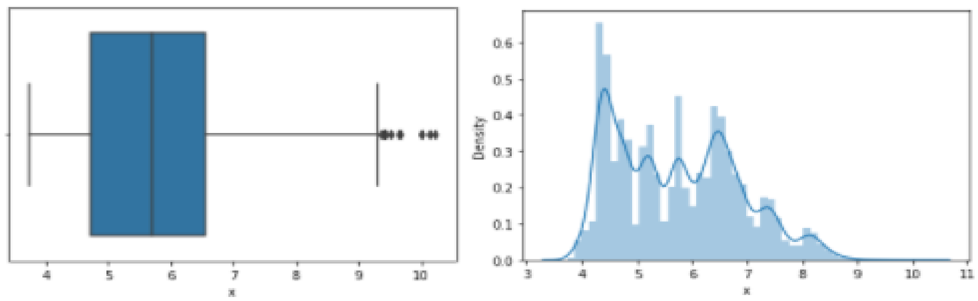

Figure 10: Variable 'x'

Observation: The boxplot of the x variable shows that there are some outliers. The distribution plot shows that the distribution is right skewed.

Y

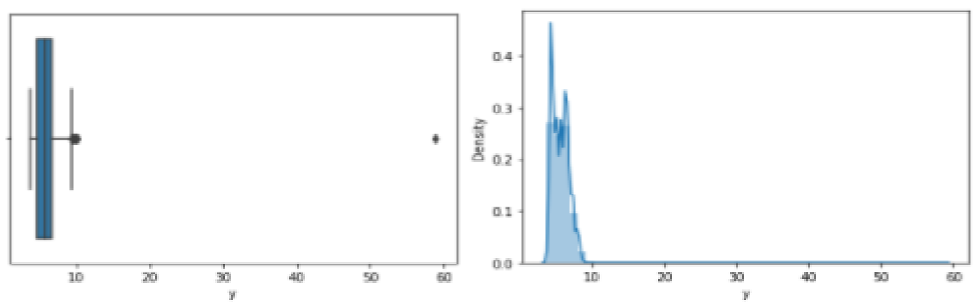

Figure 11: Variable 'y'

Observation: The boxplot of the y variable shows that there are very few outliers. The distribution plot shows that the distribution is right skewed.

Z

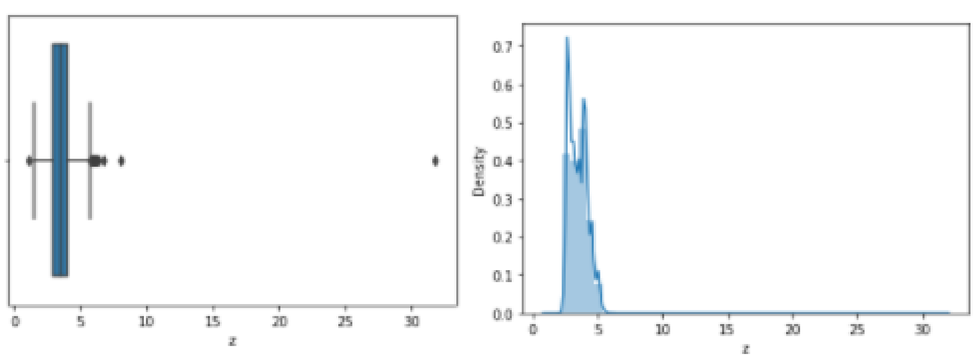

Figure 12: Variable 'Z'

Observation: The boxplot of the y variable shows that there are very few outliers. The distribution plot shows that the distribution is right skewed.

Price

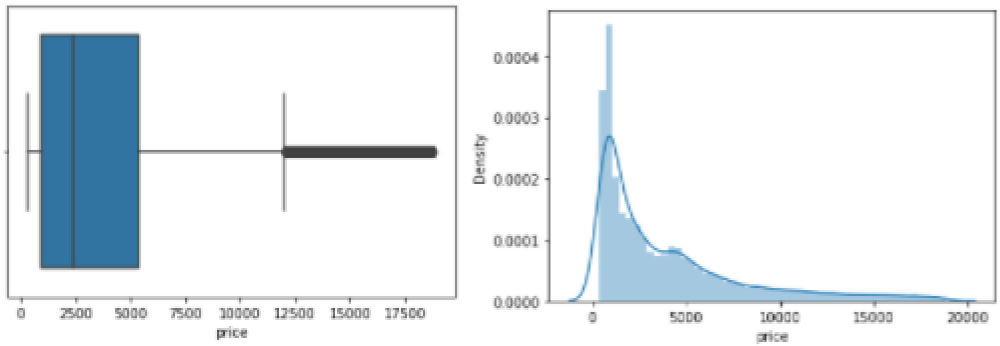

Figure 13: Variable 'Price'

Observation: The boxplot of the price variable shows that there are some outliers. The distribution plot shows that the distribution is right skewed.

Bivariate Analysis

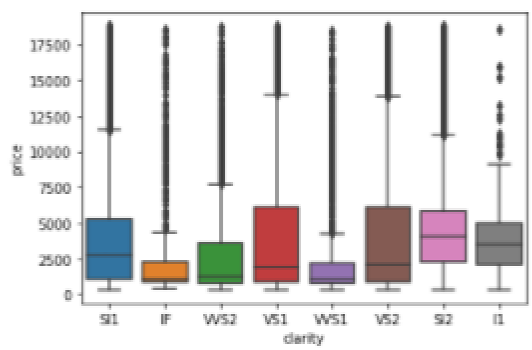

Figure 14: Boxplot

Boxplot between clarity and price shows that median for SI1, SI2, I1 is higher than the rest of them.

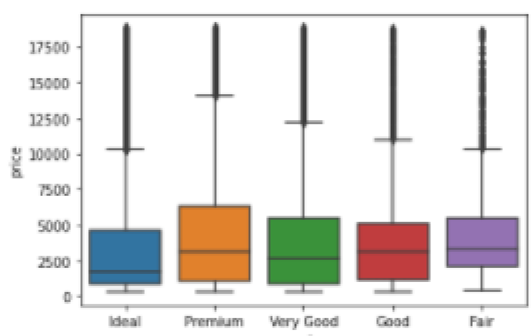

Figure 15: Boxplot

Boxplot between cut and price show that ideal cut has the least median value which is expected and premium has highest median which suggests premium cut will have high prices and ideal cut will low prices but the price ranges are seen spread out because there are different factors also affecting price.

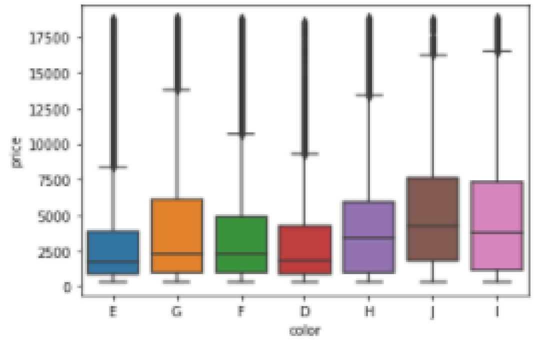

Figure 16: Boxplot

Boxplot between color and price shows that H, I, J have higher median price than other colors.

Multivariate Analysis

Heatmap

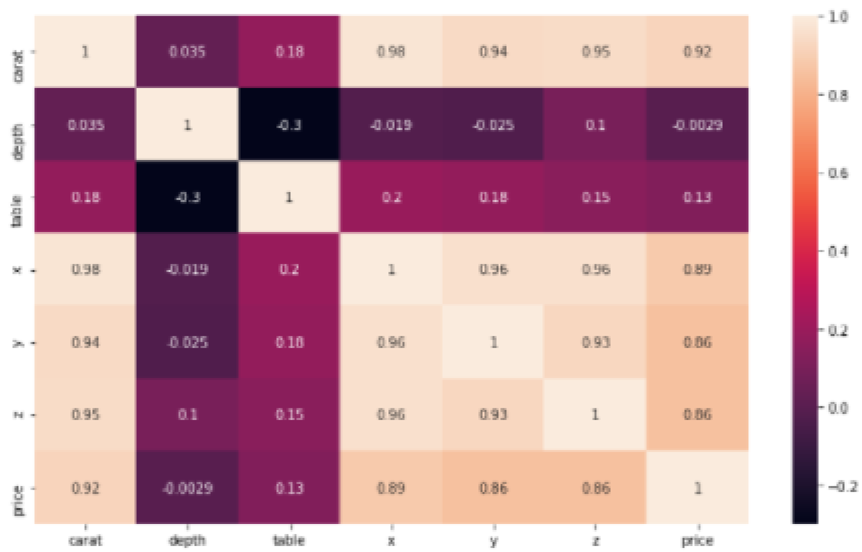

Figure 17: Heatmap

The heatmap shows that there is a strong correlation between these variables:

- Carat and x
- Carat and y
- Carat and z

- Price and carat
- X and y
- Y and z
- X and z

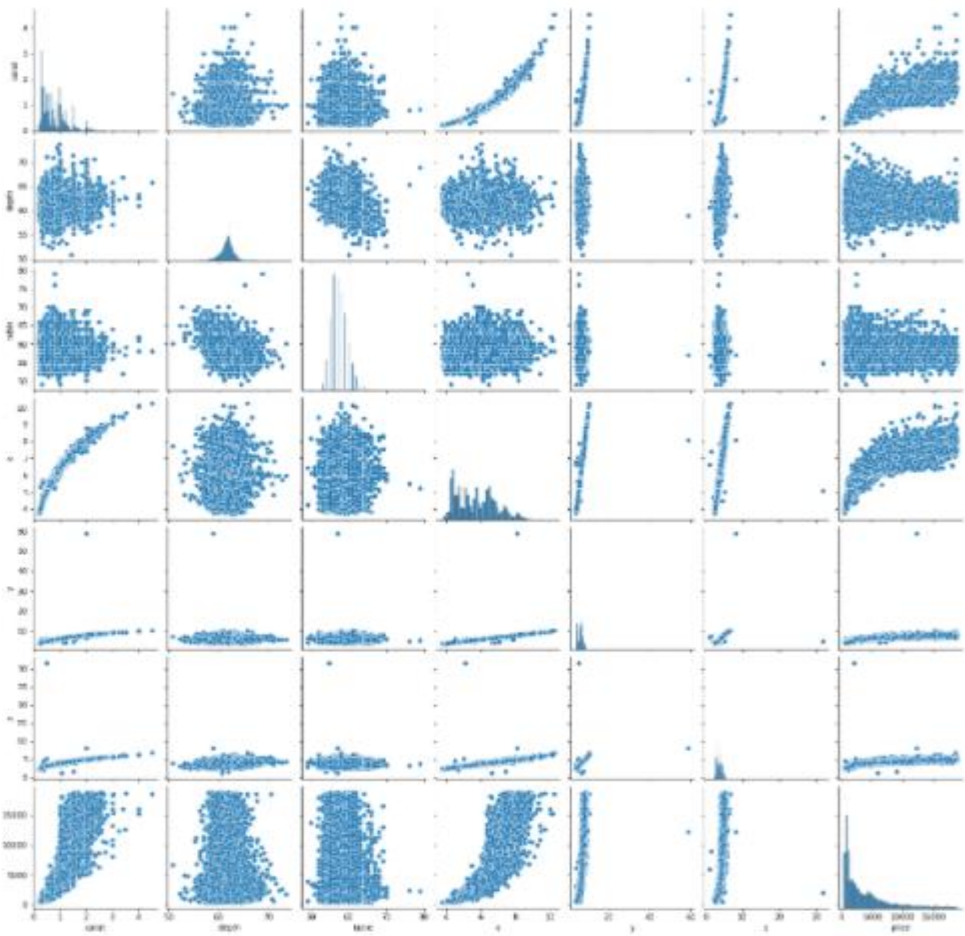

Figure 18: Pairplot

The pairplot also suggests the same as heatmap that there is good linear relation in carat and x,y,z.

To apply linear regression, the datatypes of variables have to be int or float. So, to convert them into numerical values one hot encoding is used and the variables have been increased and all have been converted to numeric variables.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 26933 entries, 0 to 26966
Data columns (total 24 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   carat          26933 non-null  float64
 1   depth          26236 non-null  float64
 2   table          26933 non-null  float64
 3   x              26931 non-null  float64
 4   y              26931 non-null  float64
 5   z              26925 non-null  float64
 6   price          26933 non-null  int64
 7   cut_Good       26933 non-null  uint8
 8   cut_Ideal      26933 non-null  uint8
 9   cut_Premium    26933 non-null  uint8
 10  cut_Very Good  26933 non-null  uint8
 11  clarity_IF     26933 non-null  uint8
 12  clarity_SI1    26933 non-null  uint8
 13  clarity_SI2    26933 non-null  uint8
 14  clarity_VS1    26933 non-null  uint8
 15  clarity_VS2    26933 non-null  uint8
 16  clarity_VVS1   26933 non-null  uint8
 17  clarity_VVS2   26933 non-null  uint8
 18  color_E        26933 non-null  uint8
 19  color_F        26933 non-null  uint8
 20  color_G        26933 non-null  uint8
 21  color_H        26933 non-null  uint8
 22  color_I        26933 non-null  uint8
 23  color_J        26933 non-null  uint8
dtypes: float64(6), int64(1), uint8(17)
memory usage: 3.1 MB
```

Figure 19: Columns after Label Encoding

Linear regression is very sensitive to outliers and all these outliers have been treated to the upper and lower whisker of the boxplot. The figure below shows that there are no outliers left in the dataset. However, the linear regression has been applied on the dataset with both the outliers present and without the outliers.

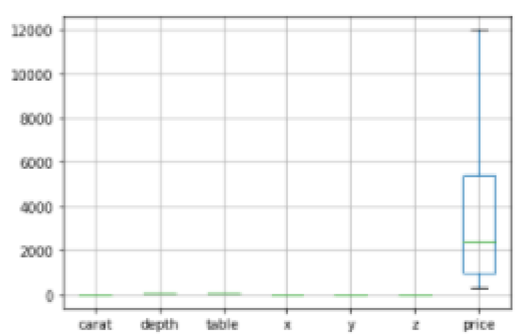

Figure 20: Boxplot After Outlier Treatment

Question 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 an ordinal variables and take actions accordingly. Explain why you are combining these sub levels with appropriate reasoning.

Answer 1.2

There are 697 null values present in depth variable.

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

Figure 21: Null Values

There are some values which are equal to zero in x,y,z variables. These have been replaced by null values as these values cannot be zero as each diamond would have a length, breadth and height.

```
carat              0
depth            697
table              0
x                  2
y                  2
z                  8
price              0
cut_Good           0
cut_Ideal          0
cut_Premium        0
cut_Very Good      0
clarity_IF         0
clarity_SI1        0
clarity_SI2        0
clarity_VS1        0
clarity_VS2        0
clarity_VVS1       0
clarity_VVS2       0
color_E            0
color_F            0
color_G            0
color_H            0
color_I            0
color_J            0
dtype: int64
```

Figure 22: Null Values after Data Modification

The null values in depth, x, y, z have been imputed by the mean values of these variables. The below figure shows that there are no null values present in the dataset

```
carat            0
depth            0
table            0
x                0
y                0
z                0
price            0
cut_Good         0
cut_Ideal        0
cut_Premium      0
cut_Very Good    0
clarity_IF       0
clarity_SI1      0
clarity_SI2      0
clarity_VS1      0
clarity_VS2      0
clarity_VVS1     0
clarity_VVS2     0
color_E          0
color_F          0
color_G          0
color_H          0
color_I          0
color_J          0
dtype: int64
```

Figure 23: Null values After Imputation

Some models have been built in order to increase the performance by combining certain sub levels of variables based on the domain knowledge about diamonds.

- Color D, E, F have been combined to a Colorless level.
- Color G,H,I,J have been combined to a Near Colorless level.
- Clarity I1, SI1, SI2 have been combined to an Impure level.
- Clarity VS1, VS2 have been combined to a Slightly-Impure level.
- Clarity VVS1, VVS2 have been combined to a Very_Slightly_Impure level.
- Clarity IF have been combined to a No_Impurities level.

Same technique of getting dummy variables using one hot encoding has been used to convert categorical variables to numeric variables.

Question 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.

Answer 1.3

The data having string values have been encoded using the one hot encoding method. This technique converts the categorical variables into numerical variables by making new columns equal to the levels in each original column and dropping the first column. The below figure shows the new columns that have been created and all the columns have been converted into numerical data types.

cut_Good	cut_Ideal	cut_Premium	...	clarity_VS1	clarity_VS2	clarity_VVS1	clarity_VVS2	color_E	color_F	color_G	color_H	color_I	color_J
0	1	0	...	0	0	0	0	1	0	0	0	0	0
0	0	1	...	0	0	0	0	0	0	1	0	0	0
0	0	0	...	0	0	0	1	1	0	0	0	0	0
0	1	0	...	1	0	0	0	0	1	0	0	0	0
0	1	0	...	0	0	1	0	0	1	0	0	0	0

Figure 24: One Hot Encoding

The data has been splitted into 70% train and 30% test data. Linear Regression analysis is applied on train data using scikit learn library. For further analysis Linear regression using statsmodel has been applied to get adjusted R squared and other parameters to decide the best model that can be used for the current dataset.

Model 1

The first model is built using all the columns and without dropping any of them.

Using scikit learn library applying Linear Regression Model.

Model score for a regression model is nothing but r squared.

- Score for Training data is 0.940
- Score for Test data is 0.942
- Root mean squared error for training data is 846.94
- Root mean squared error for test data is 836.75

This represents the best fit line. The model is right fit and train and test data are in sync

Using stats modelapplying Linear Regression model as it gives more widened performance metrics. R squared and adjusted r squared value is 0.940 which is very good.

```
==============================================================================
Dep. Variable:                  price   R-squared:                       0.940
Model:                            OLS   Adj. R-squared:                  0.940
Method:                 Least Squares   F-statistic:                 1.289e+04
Date:                Wed, 12 Jan 2022   Prob (F-statistic):               0.00
Time:                        16:33:22   Log-Likelihood:            -1.5385e+05
No. Observations:               18853   AIC:                         3.078e+05
Df Residuals:                   18829   BIC:                         3.079e+05
Df Model:                          23
Covariance Type:            nonrobust
==================================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept     -3307.1169    748.747     -4.417      0.000   -4774.728   -1839.506
carat          9177.8162     77.380    118.607      0.000    9026.145    9329.488
depth            10.7415     10.394      1.033      0.301      -9.632      31.115
table           -18.9513      3.842     -4.933      0.000     -26.482     -11.421
x             -1086.8989    123.094     -8.830      0.000   -1328.174    -845.624
y               967.9215    124.393      7.781      0.000     724.101    1211.742
z              -577.3369    129.075     -4.473      0.000    -830.336    -324.337
cut_Good        460.9612     44.366     10.390      0.000     374.000     547.922
cut_Ideal       698.0794     43.141     16.181      0.000     613.520     782.639
cut_Premium     659.6221     41.422     15.925      0.000     578.432     740.812
cut_Very_Good   590.4956     42.407     13.924      0.000     507.373     673.618
clarity_IF     3992.6455     66.195     60.316      0.000    3862.897    4122.394
clarity_SI1    2510.2269     56.650     44.311      0.000    2399.188    2621.266
clarity_SI2    1674.8881     56.927     29.422      0.000    1563.305    1786.471
clarity_VS1    3333.5315     57.765     57.709      0.000    3220.308    3446.755
clarity_VS2    3029.8539     56.976     53.178      0.000    2918.176    3141.532
clarity_VVS1   3761.3204     61.011     61.650      0.000    3641.734    3880.907
clarity_VVS2   3746.8059     59.432     63.043      0.000    3630.314    3863.298
color_E        -181.2250     22.737     -7.971      0.000    -225.791    -136.659
color_F        -256.2452     23.206    -11.042      0.000    -301.730    -210.760
color_G        -428.1865     22.515    -19.018      0.000    -472.318    -384.055
color_H        -856.3686     24.094    -35.543      0.000    -903.594    -809.143
color_I       -1325.3521     26.791    -49.469      0.000   -1377.865   -1272.839
color_J       -1929.0572     33.026    -58.410      0.000   -1993.792   -1864.322
==============================================================================
Omnibus:                     4775.536   Durbin-Watson:                   1.983
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17747.449
Skew:                           1.234   Prob(JB):                         0.00
Kurtosis:                       7.062   Cond. No.                     1.04e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.04e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Figure 25: Model 1 Regression Analysis

Hypothesis testing for Linear Regression – The null hypothesis states that there is no relation between the dependent variable Price and other independent variables. Looking at the summary table above, all the P values are less than 0.05 or at 95% confidence level we can say that the variables have a direct impact on the price variable except the depth variable which hasp value greater than 0.05. For depth null hypothesis cannot be rejected and it is inferred that depth has no effect on depth variable. Carat and clarity variables seem to impact the price rise positively, surprisingly; color of the stones is reducing the price increase. We can study our model further to see if we can reduce multicollinearity, if present to get the correct coefficients.

The below scatter plot is between actual price on x axis and predicted price on y axis. The scatter plot shows a strong linear relationship between them meaning the model gives out a really good prediction.

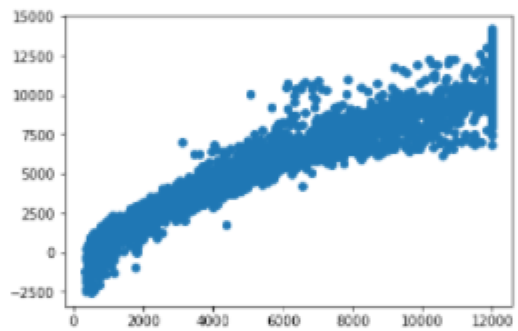

Figure 26: Scatter Plot Between Actual and Predicted Price

The below image is variance inflation factor which gives which factor is the reason for multicollinearity in the data. Here it is observed that depth, table, x, y, z are the most contributing factors to multicollinearity and hence further models have to be made such to address this multicollinearity. Multicollinearity affects the coefficients of independent variables and they may not be correct in predicting the target variable so this multicollinearity has tobe reduced to get the correct coefficients.

```
carat ---> 122.83209929169017
depth ---> 1348.5747037839517
table ---> 978.7910961905317
x ---> 11960.44592119137
y ---> 11486.794645087222
z ---> 3179.596094744831
cut_Good ---> 4.4914016280632785
cut_Ideal ---> 18.016728432469563
cut_Premium ---> 10.83115735472792
cut_Very Good ---> 10.01637987063325
clarity_IF ---> 3.655387429980751
clarity_SI1 ---> 19.688570072296287
clarity_SI2 ---> 13.82199801875791
clarity_VS1 ---> 12.746420794843807
clarity_VS2 ---> 18.44735046547165
clarity_VVS1 ---> 6.431913987513056
clarity_VVS2 ---> 8.379930444886524
color_E ---> 2.480829824573142
color_F ---> 2.448039716583097
color_G ---> 2.796050687532723
color_H ---> 2.305043749504993
color_I ---> 1.9312909945309655
color_J ---> 1.5142333068255323
```

Figure 27: Variance Inflation Factor

Model 2:

The second model has been built by dropping the Depth Variable as it had a p value greater than 0.05.

First using scikitlearn library Linear Regression Analysis is applied

- Score for Training data is 0.940
- Score for Test data is 0.942
- Root mean squared error for training data is 846.96
- Root mean squared error for test data is 836.81

There seems to no change in the performance metrics from the first model.

Using statsmodel to build the Linear Regression model. Although there is no change in r squared and adjusted r squared the f statistic has improved considerably.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.940
Model:                            OLS   Adj. R-squared:                  0.940
Method:                 Least Squares   F-statistic:                 1.347e+04
Date:                Wed, 12 Jan 2022   Prob (F-statistic):               0.00
Time:                        19:11:54   Log-Likelihood:            -1.5385e+05
No. Observations:               18853   AIC:                         3.077e+05
Df Residuals:                   18830   BIC:                         3.079e+05
Df Model:                          22
Covariance Type:            nonrobust
=================================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------
Intercept     -2583.1184    264.237     -9.776      0.000   -3101.046   -2065.191
carat          9186.0017     76.974    119.340      0.000    9035.126    9336.877
table           -19.7726      3.759     -5.260      0.000     -27.140     -12.405
x             -1112.1452    120.646     -9.218      0.000   -1348.622    -875.669
y               923.6574    116.786      7.909      0.000     694.746    1152.569
z              -470.0193     76.660     -6.131      0.000    -620.279    -319.760
cut_Good        463.4877     44.299     10.463      0.000     376.659     550.317
cut_Ideal       694.1161     42.970     16.154      0.000     609.891     778.341
cut_Premium     655.7942     41.256     15.896      0.000     574.929     736.659
cut_Very_Good   588.7693     42.375     13.894      0.000     505.711     671.827
clarity_IF     3992.6306     66.195     60.316      0.000    3862.882    4122.380
clarity_SI1    2511.9543     56.625     44.361      0.000    2400.963    2622.945
clarity_SI2    1676.2228     56.913     29.452      0.000    1564.669    1787.777
clarity_VS1    3334.3531     57.759     57.729      0.000    3221.140    3447.566
clarity_VS2    3031.1913     56.961     53.215      0.000    2919.542    3142.841
clarity_VVS1   3761.8847     61.009     61.662      0.000    3642.302    3881.467
clarity_VVS2   3747.7677     59.425     63.067      0.000    3631.289    3864.246
color_E        -181.4035     22.736     -7.979      0.000    -225.968    -136.839
color_F        -256.2485     23.206    -11.043      0.000    -301.734    -210.763
color_G        -427.9656     22.514    -19.009      0.000    -472.096    -383.836
color_H        -855.8721     24.089    -35.530      0.000    -903.089    -808.656
color_I       -1324.6763     26.783    -49.459      0.000   -1377.174   -1272.178
color_J       -1928.6259     33.024    -58.401      0.000   -1993.356   -1863.896
==============================================================================
Omnibus:                     4778.989   Durbin-Watson:                   1.983
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17785.860
Skew:                           1.234   Prob(JB):                         0.00
Kurtosis:                       7.068   Cond. No.                     2.57e+03
==============================================================================
```

Figure 28: Model 2 Regression Analysis

The scatter plot between actual and predicted price also shows a linear graph which tells model is a good model for predicting price.

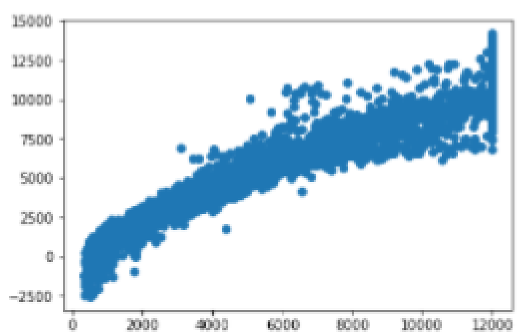

Figure 29: Scatter Plot Between Actual and Predicted Price

The Variance inflation factor below shows high value for x, y, z variable which means they are the contributing factors to multicollinearity in the data. X, y, z variables have a very strong relation with carat variable as carat variable increases it is bound that x, y, z will increase as they are the size measures.

```
carat ---> 102.22998573256453
table ---> 343.67178696032374
x ---> 11907.66520065791
y ---> 11068.192853143442
z ---> 1592.6925974471997
cut_Good ---> 4.364430991175007
cut_Ideal ---> 16.032354095929502
cut_Premium ---> 10.552785297072413
cut_Very Good ---> 9.581139499530757
clarity_IF ---> 3.507794731999077
clarity_SI1 ---> 18.901145707067478
clarity_SI2 ---> 13.307124818568566
clarity_VS1 ---> 12.220903273804456
clarity_VS2 ---> 17.66874831948798
clarity_VVS1 ---> 6.149324444822197
clarity_VVS2 ---> 8.015137230464944
color_E ---> 2.478314015891665
color_F ---> 2.4444749919155604
color_G ---> 2.7914199796781376
color_H ---> 2.296962880031655
color_I ---> 1.92588279414264
color_J ---> 1.5119985802546922
```

Figure 30: Variance Inflation Factor

Model 3

The next model is built to reduce the multicollinearity in the data by further dropping variables x, y, z

First using scikitlearn library Linear Regression Analysis is applied

- Score for Training data is 0.939
- Score for Test data is 0.940
- Root mean squared error for training data is 853.72
- Root mean squared error for test data is 844.189

RMSE has increased by a little in this model but there is not any significant difference and train and test data are in line with each other.

Using statsmodel to build the Linear Regression model. Although there is no change in r squared and adjusted r squared the f statistic has improved considerably. All the p values are below 0 which shows that all the independent variables in this model have a significant impact on target variable price.

```
==============================================================================
Dep. Variable:                  price   R-squared:                       0.939
Model:                            OLS   Adj. R-squared:                  0.939
Method:                 Least Squares   F-statistic:                 1.534e+04
Date:                Wed, 12 Jan 2022   Prob (F-statistic):               0.00
Time:                        19:12:02   Log-Likelihood:            -1.5400e+05
No. Observations:               18853   AIC:                         3.080e+05
Df Residuals:                   18833   BIC:                         3.082e+05
Df Model:                          19
Covariance Type:            nonrobust
=================================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------
Intercept     -4717.6694    217.998    -21.641      0.000   -5144.965   -4290.373
carat          8039.2039     15.515    518.159      0.000    8008.793    8069.615
table           -16.9206      3.550     -4.766      0.000     -23.880      -9.961
cut_Good        551.9846     43.215     12.773      0.000     467.280     636.689
cut_Ideal       788.4697     40.630     19.406      0.000     708.831     868.109
cut_Premium     717.1073     39.820     18.009      0.000     639.057     795.157
cut_Very_Good   700.3553     40.311     17.374      0.000     621.343     779.368
clarity_IF     4103.3041     66.308     61.882      0.000    3973.334    4233.274
clarity_SI1    2549.2986     56.950     44.763      0.000    2437.671    2660.927
clarity_SI2    1711.2926     57.250     29.892      0.000    1599.077    1823.508
clarity_VS1    3391.3872     58.031     58.441      0.000    3277.642    3505.133
clarity_VS2    3086.3556     57.247     53.913      0.000    2974.146    3198.565
clarity_VVS1   3864.0629     61.159     63.180      0.000    3744.185    3983.941
clarity_VVS2   3829.7494     59.635     64.220      0.000    3712.859    3946.639
color_E        -183.4199     22.915     -8.004      0.000    -228.335    -138.504
color_F        -266.7587     23.379    -11.410      0.000    -312.585    -220.933
color_G        -438.3437     22.680    -19.327      0.000    -482.799    -393.888
color_H        -855.5083     24.265    -35.257      0.000    -903.070    -807.947
color_I       -1312.9075     26.973    -48.676      0.000   -1365.776   -1260.039
color_J       -1913.2713     33.262    -57.520      0.000   -1978.469   -1848.074
==============================================================================
Omnibus:                     4285.730   Durbin-Watson:                   1.980
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            13193.004
Skew:                           1.168   Prob(JB):                         0.00
Kurtosis:                       6.367   Cond. No.                     2.07e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.07e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Figure 31: Model 3 Regression Analysis

The scatter plot between actual and predicted price also shows a linear graph which tells model is a good model for predicting price.

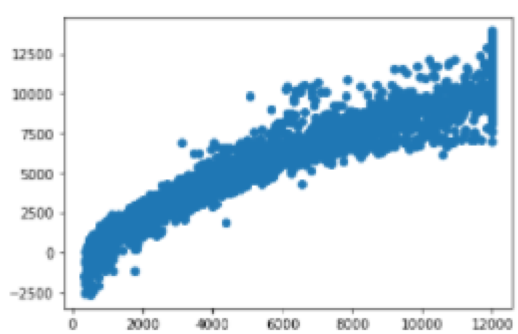

Figure 32: Scatter Plot Between Actual and Predicted Price

The Variance Inflation factor shows only table is the factor remaining which is showing some multicollinearity. All other variables contribute very little to the multicollinearity.

```
carat ---> 5.223684588164773
table ---> 97.56394186616357
cut_Good ---> 4.1312866621866124
cut_Ideal ---> 14.335956924499904
cut_Premium ---> 9.885733103057277
cut_Very Good ---> 8.70333130825681
clarity_IF ---> 3.481310724007501
clarity_SI1 ---> 18.559797736467978
clarity_SI2 ---> 13.097823682775097
clarity_VS1 ---> 12.053613938763373
clarity_VS2 ---> 17.42144406697902
clarity_VVS1 ---> 6.101273452791654
clarity_VVS2 ---> 7.9337889552988
color_E ---> 2.4758946280948533
color_F ---> 2.4352303532503305
color_G ---> 2.7781168170164494
color_H ---> 2.290612795680401
color_I ---> 1.9225371559098865
color_J ---> 1.509672238133545
```

Figure 33: Variance Inflation Factor

Model 4

This model is built by dropping the table variable further to get rid of the multicollinearity.

First using scikitlearn library Linear Regression Analysis is applied

- Score for Training data is 0.939
- Score for Test data is 0.940
- Root mean squared error for training data is 854.23
- Root mean squared error for test data is 844.833

RMSE has increased by a little in this model but there is not any significant difference. The train and test data are in line.

Using statsmodel to build the Linear Regression model. Although there is no change in r squared and adjusted r squared the f statistic has improved considerably. All the p values are below 0 which shows that all the independent variables in this model have a significant impact on target variable price.

```
==============================================================================
Dep. Variable:                  price   R-squared:                       0.939
Model:                            OLS   Adj. R-squared:                  0.939
Method:                 Least Squares   F-statistic:                 1.618e+04
Date:                Wed, 12 Jan 2022   Prob (F-statistic):               0.00
Time:                        19:12:03   Log-Likelihood:            -1.5401e+05
No. Observations:               18853   AIC:                         3.081e+05
Df Residuals:                   18834   BIC:                         3.082e+05
Df Model:                          18
Covariance Type:            nonrobust
=================================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------
Intercept     -5707.2444     66.430    -85.914      0.000   -5837.453   -5577.035
carat          8032.1135     15.452    519.799      0.000    8001.826    8062.402
cut_Good        552.8581     43.239     12.786      0.000     468.105     637.611
cut_Ideal       833.8164     39.523     21.097      0.000     756.348     911.285
cut_Premium     717.4752     39.843     18.008      0.000     639.380     795.570
cut_Very_Good   713.1301     40.245     17.720      0.000     634.247     792.013
clarity_IF     4107.9141     66.339     61.923      0.000    3977.883    4237.945
clarity_SI1    2551.6955     56.981     44.781      0.000    2440.008    2663.383
clarity_SI2    1712.3615     57.283     29.893      0.000    1600.082    1824.641
clarity_VS1    3394.0323     58.062     58.456      0.000    3280.226    3507.838
clarity_VS2    3088.7704     57.278     53.926      0.000    2976.501    3201.040
clarity_VVS1   3865.5425     61.194     63.169      0.000    3745.597    3985.488
clarity_VVS2   3830.9495     59.669     64.203      0.000    3713.993    3947.906
color_E        -185.2647     22.925     -8.081      0.000    -230.200    -140.330
color_F        -266.4984     23.393    -11.392      0.000    -312.351    -220.646
color_G        -437.3437     22.692    -19.273      0.000    -481.823    -392.864
color_H        -853.8460     24.276    -35.172      0.000    -901.430    -806.262
color_I       -1312.3437     26.988    -48.627      0.000   -1365.242   -1259.445
color_J       -1913.8403     33.281    -57.505      0.000   -1979.075   -1848.606
==============================================================================
Omnibus:                     4299.918   Durbin-Watson:                   1.980
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            13189.906
Skew:                           1.173   Prob(JB):                         0.00
Kurtosis:                       6.359   Cond. No.                         38.6
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Figure 34: Model 4 Regression Analysis

The problem of multicollinearity is removed as the VIF of this model indicates there is no multicollinearity in the data.

```
carat ---> 4.823963552948521
cut_Good ---> 3.5020910476346123
cut_Ideal ---> 12.444657871405733
cut_Premium ---> 8.172657501145094
cut_Very Good ---> 7.362893118877587
clarity_IF ---> 2.111573704227221
clarity_SI1 ---> 8.582797349957549
clarity_SI2 ---> 6.341835689528017
clarity_VS1 ---> 5.81765676176599
clarity_VS2 ---> 8.132009944538451
clarity_VVS1 ---> 3.2360618789461135
clarity_VVS2 ---> 3.994479451757277
color_E ---> 2.3711536081820666
color_F ---> 2.340035283675498
color_G ---> 2.6842016288544035
color_H ---> 2.221751381261127
color_I ---> 1.8871709780128412
color_J ---> 1.4941185157588375
```

Figure 35: Variance Inflation Factor

Model 5:

This model is built using feature modeling and combining different ordinal sub levels into 1 to check if there is any improvement in the performance metrics.The combined levels have been discussed in the previous question.

```
carat                           0
depth                           0
table                           0
x                               0
y                               0
z                               0
price                           0
cut_Good                        0
cut_Ideal                       0
cut_Premium                     0
cut_Very Good                   0
clarity_No_Impurities           0
clarity_Slightly_Impure         0
clarity_Very_Slightly_Impure    0
color_Near_Colorless            0
dtype: int64
```

Figure 36: Modified Columns

First using scikitlearn library Linear Regression Analysis is applied

- Score for Training data is 0.920
- Score for Test data is 0.920
- Root mean squared error for training data is 979.76
- Root mean squared error for test data is 979.80

RMSE has increased and Rsquared value has dropped which tells this is not a good model.

```
==============================================================================
Dep. Variable:                  price   R-squared:                       0.920
Model:                            OLS   Adj. R-squared:                  0.920
Method:                 Least Squares   F-statistic:                 1.549e+04
Date:                Sun, 16 Jan 2022   Prob (F-statistic):               0.00
Time:                        21:57:35   Log-Likelihood:            -1.5660e+05
No. Observations:               18853   AIC:                         3.132e+05
Df Residuals:                   18838   BIC:                         3.133e+05
Df Model:                          14
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                    -1825.5543    864.108     -2.113      0.035   -3519.284    -131.825
carat                         8672.2225     88.950     97.495      0.000    8497.872    8846.573
depth                           18.2815     12.007      1.523      0.128      -5.253      41.816
table                          -25.6301      4.441     -5.772      0.000     -34.334     -16.926
x                            -1178.1551    142.155     -8.288      0.000   -1456.792    -899.518
y                             1245.1260    143.490      8.677      0.000     963.873    1526.379
z                             -731.4832    149.058     -4.907      0.000   -1023.651    -439.316
cut_Good                       698.6566     51.005     13.698      0.000     598.683     798.631
cut_Ideal                      951.3730     49.527     19.209      0.000     854.295    1048.451
cut_Premium                    927.1785     47.499     19.520      0.000     834.077    1020.280
cut_Very_Good                  845.2556     48.655     17.372      0.000     749.887     940.624
clarity_No_Impurities         1873.9209     42.471     44.123      0.000    1790.674    1957.168
clarity_Slightly_Impure       1003.5859     16.470     60.935      0.000     971.304    1035.868
clarity_Very_Slightly_Impure  1615.4711     22.269     72.543      0.000    1571.821    1659.121
color_Near_Colorless          -660.8148     14.838    -44.534      0.000    -689.899    -631.730
==============================================================================
Omnibus:                     3020.949   Durbin-Watson:                   2.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            12340.520
Skew:                           0.750   Prob(JB):                         0.00
Kurtosis:                       6.669   Cond. No.                     1.04e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.04e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Figure 37: Model 5 Regression Analysis

Using statsmodel to build the Linear Regression model. R squared and adjusted r squared value has decreased. Depth has p value greater than alpha so there is not enough evidence to reject null hypothesis.

This model is not a good model as compared to the previous models.

For deciding best model in linear regression, r squared value should be high and RMSE should be low as possible. Further the adjusted r squared and other metrics are checked using statsmodel.To get the correct coefficients multicollinearity should be reduced.

Model Comparison:

	Model1 Train	Model1 Test	Model2 Train	Model2 Test	Model3 Train	Model3 Test	Model4 Train	Model4 Test	Model5 Train	Model5 Test
R Squared	0.940	0.941	0.940	0.941	0.939	0.940	0.939	0.940	0.920	0.920
RMSE	846.94	836.75	846.96	836.81	853.72	844.18	854.23	844.23	979.76	979.80
Adjusted R squared	0.94		0.94		0.939		0.939		0.92	
Multicollinearity	Yes		Yes		Yes		No		Yes	

Table 1:Model comparison

Model 4 has very good r squared and adjusted r squared and it has the highest f statistic. The coefficients can be well explained for this model as there is no multicollinearity present for this model.

Question 1.4

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

Answer 1.4

The 5 most Important factors affecting the price of diamonds are:

Carat, clarity_IF, clarity_VVS1, clarity_VVS2, clarity_VS1 are the 5 attributes that are most important for this dataset.

The final regression equation is:

```
(-5707.24) * Intercept + (8032.11) * carat + (552.86) * cut_Good + (833.82) * cut_Ideal + (717.48) * cut_Premium + (713.13) * c
ut_Very_Good + (4107.91) * clarity_IF + (2551.7) * clarity_SI1 + (1712.36) * clarity_SI2 + (3394.03) * clarity_VS1 + (3088.77)
* clarity_VS2 + (3865.54) * clarity_VVS1 + (3830.95) * clarity_VVS2 + (-185.26) * color_E + (-266.5) * color_F + (-437.34) * co
lor_G + (-853.85) * color_H + (-1312.34) * color_I + (-1913.84) * color_J +
```

Figure 38: Linear Regression Equation

When carat increases by 1 unit, price increases by 8032.11 units, keeping all other predictors constant. Clarity is a strong predictor as clarity_IF,clarity_SI1, clarity_SI2,clarity_VS1, clarity_VS2, clarity_VVS1, clarity_VVS2 have high coefficients.

There are also some negative co-efficient values, for instance, color has all its coefficients negative. The colorless colors like D,E,F contribute very less negatively comparatively to colors G, H, I, J.

Cut which are of premium and very good quality seem to have a higher positive impact. Although Ideal cut also has a very high coefficient.

Recommendations

- The colors H, I, J of the diamonds should be avoided as they contribute very negatively to the price so colors D, E, F, G should be used.
- The company should focus on clarity and carat of the diamonds so as to increase the price of the diamond.
- Company can introduce a lottery system for people who buy expensive diamonds and give them a chance to receive some gifts.
- The customers will be of different financial backgrounds so all type of diamonds from low range to high range should be available in stock. A further analysis can be done by the company based on the customer's financial backgrounds to understand which diamonds they are likely to buy.

Problem 2: Logistic Regression and LDA

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.

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

Question 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.

Answer 2.1

The dataset consists of 872 rows and 8 columns. 2 columns are of datatype object namely Holliday_Package and foreign. All other columns are of continuous nature and datatype as int or float.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 872 entries, 0 to 871
Data columns (total 8 columns):
 #   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)
memory usage: 54.6+ KB
```

Figure 39: Dataset Information

The figure below shows the description of the data. The range of each variable is the min value to the max value. There seems to be a large difference in 75 percentile and max values for Salary, no_young_children, no_older_children variable which suggests there are outliers in the data. Number of older children varies from 0 to 6 and number of younger children vary from 0 to 3.

	Unnamed: 0	Salary	age	educ	no_young_children	no_older_children
count	872.000000	872.000000	872.000000	872.000000	872.000000	872.000000
mean	436.500000	47729.172018	39.955275	9.307339	0.311927	0.982798
std	251.869014	23418.668531	10.551675	3.036259	0.612870	1.086786
min	1.000000	1322.000000	20.000000	1.000000	0.000000	0.000000
25%	218.750000	35324.000000	32.000000	8.000000	0.000000	0.000000
50%	436.500000	41903.500000	39.000000	9.000000	0.000000	1.000000
75%	654.250000	53469.500000	48.000000	12.000000	0.000000	2.000000
max	872.000000	236961.000000	62.000000	21.000000	3.000000	6.000000

Figure 40: Dataset Description

There are no null values present in the dataset, therefore no imputation is required.

```
Unnamed: 0           0
Holliday_Package     0
Salary               0
age                  0
educ                 0
no_young_children    0
no_older_children    0
foreign              0
dtype: int64
```

Figure 41: Null Values

There are no duplicates in the data.

The Unnamed: 0 column is of no use and therefore it has been dropped from the dataset.

The percentage of employees who have opted for the package is 45%. The data is a balanced between people who have opted and not opted for the package.

```
no     0.540138
yes    0.459862
Name: Holliday_Package, dtype: float64
```

Figure 42: Holliday_Package Value Count

There are 216 employees out of the total records who are foreigners.

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

Figure 43: Foreign Value Count

Univariate Analysis

Salary

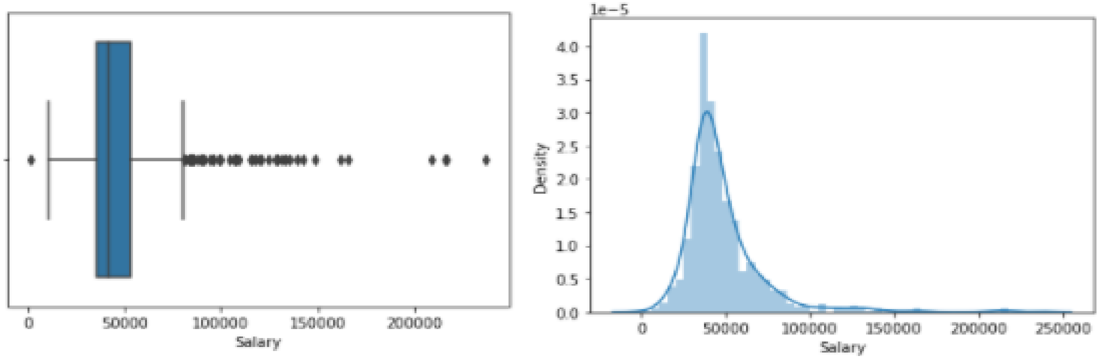

Figure 44: Variable 'Salary'

The boxplot for salary variable shows that there are a lot of outliers. The distribution plot shows that salary variable is right skewed.

Age

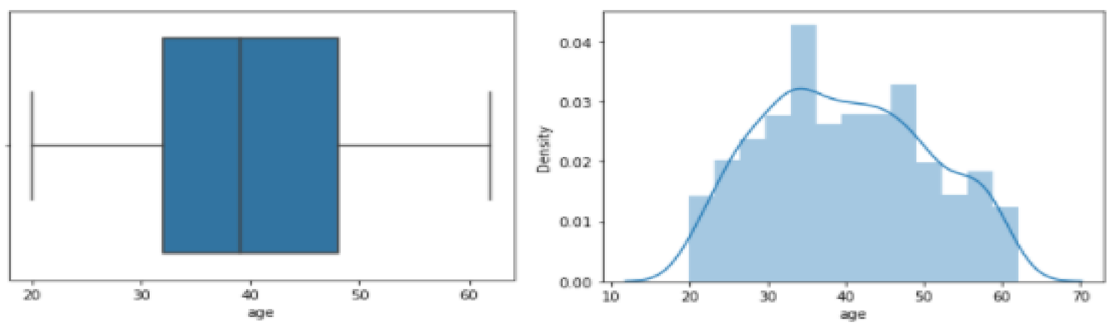

Figure 45: Variable 'Age'

The boxplot for Age variable shows that there are no outliers. The distribution plot shows that age variable is almost normally distributed.

Educ

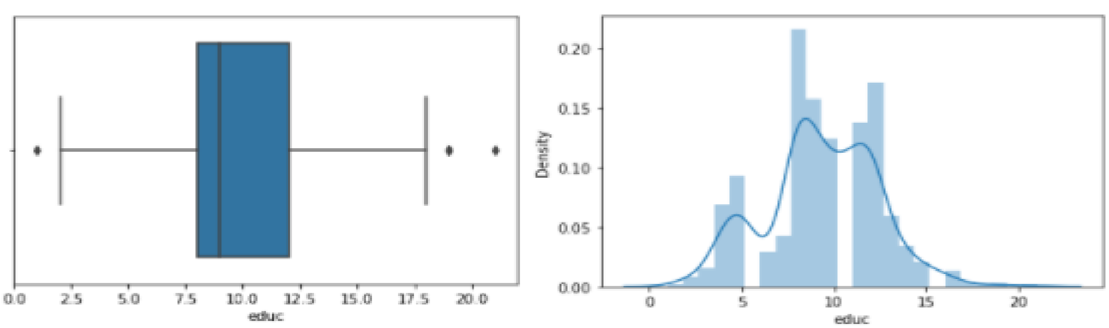

Figure 46: Variable 'Educ'

The boxplot of Educ variable shows that there are outliers on both ends. The distribution plot shows that it is left skewed. Educ is a discrete numeric variable

No_young_children

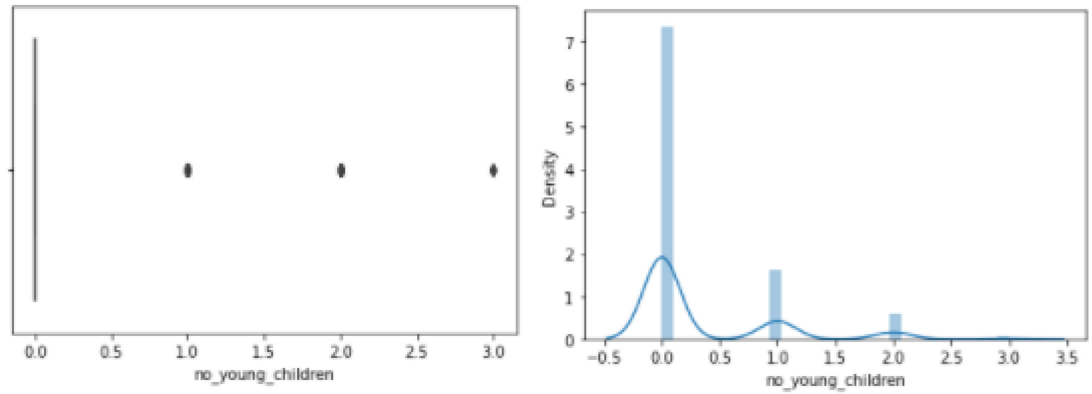

Figure 47: Variable ‘No_young_children’

The boxplot for no_young_children variable shows that there are a lot of outliers(anything other than value 0 is considered an outlier for this variable) The distribution plot shows that no_young_children variable is right skewed. This no_young_children is a discrete numeric variable.

No_older_children

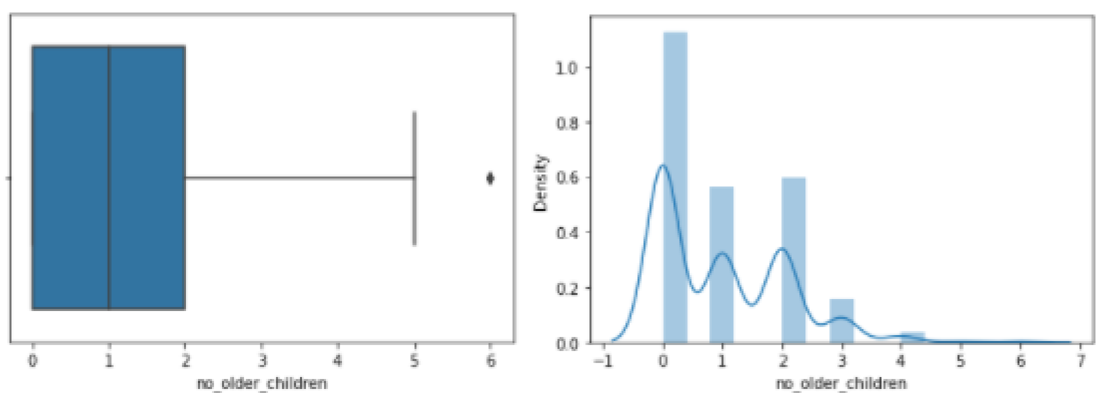

Figure 48; Variable 'No_older_children'

The boxplot for no_older_children variable shows that there are a few outliers. The distribution plot shows that no_older_children variable is right skewed. This no_older_children is a discrete numeric variable.

Bivariate Analysis

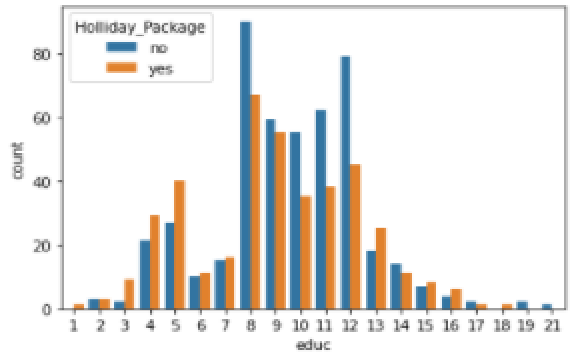

Figure 49: Count plot of Educ

The countplot for educ variable with a hue of Holliday_Package variable shows that it widely distributed and most of the employees have educ between 4 to 13 years and ratio between opted and not opted is almost 60/40.

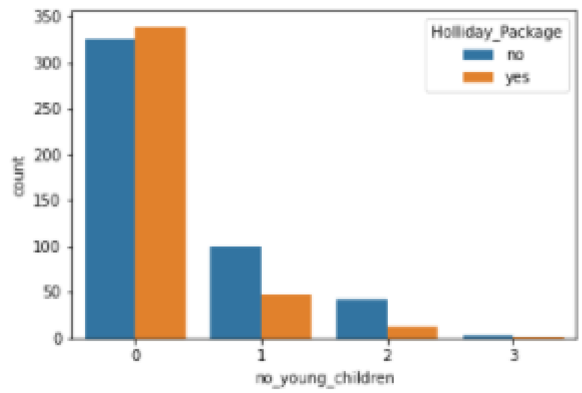

Figure 50: Count Plot of no_young_children

The countplot for no_young_children shows that people with 0 children have chosen the package more and people with children have hardly chosen the package which is expected as people do not tend to travel with young children.

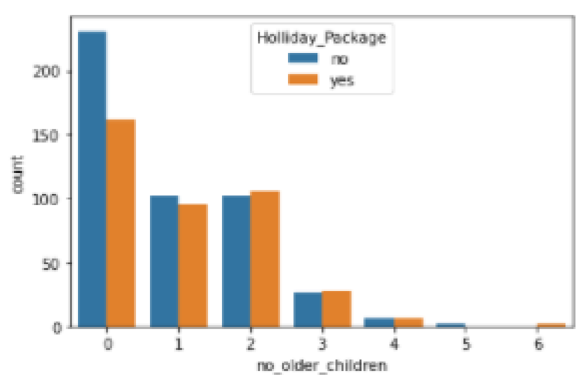

Figure 51: Count plot of no_older_children

The countplot for no_older_children with hue of Holliday_Package shows how many employees have chosen and not chosen the package. From the plot it seems that proportion that most employees have a smaller number of older children and the proportion between chosen/not chosen improves as the number of children increases which can be due to the reason that older children being independent and the employees can go for holiday.

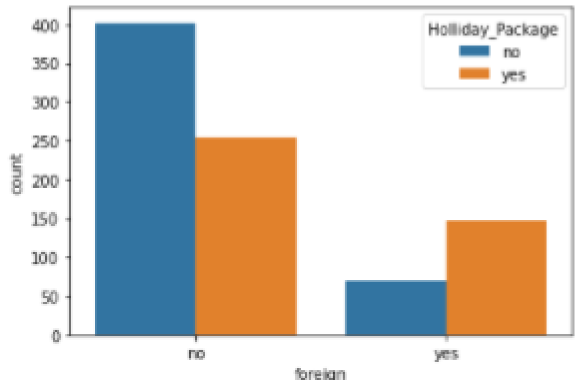

Figure 52: Count Plot of foreign

The counplot for foreign with hue of Holliday_Package shows that employees who are foreigners have opted for the holiday package more that the employees who are not foreigners.

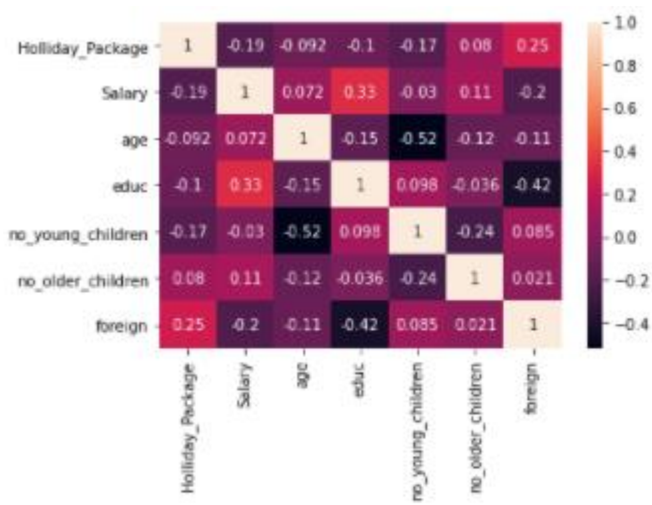

Figure 53: Heatmap

The heatmap shows there is not any high correlation in any of the variables. However, there is some correlation between no_young_children and age variable.

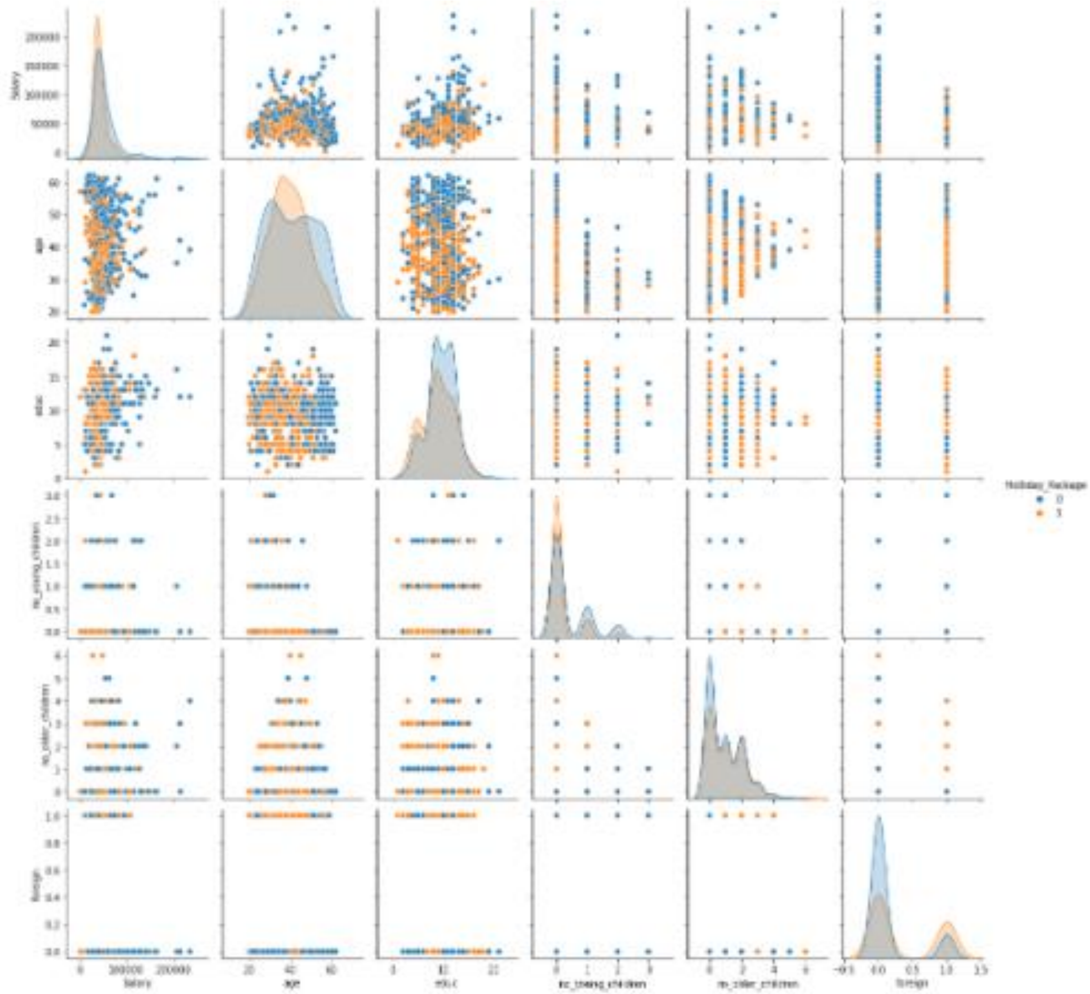

Figure 54: Pairplot

The pairplot above shows that the data points of opted and not opted overlap which means that none of the variables is a good predictor for the target column Holliday_Package. Foreign variable seems to be able to distinguish between opted and not opted better than most variables through this pairplot. More analysis can be withdrawn after the model building.

Question 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).

Answer 2.2

Using the Label Encoder from sklearn.preprocessing library the string values of Holliday_package and foreign have been changed to numerical values as Logistic regression and LDA uses only numerical inputs. (Foreign: yes = 1, no = 0), (Holliday_Package: yes=1, no=0)

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

Figure 55: Label Encoding

The data has been splitted into two parts which is one has the independent variables and other has the target variable(Holliday_Package).

Using train_test_split function the data has been splitted into 70% train and 30% test.

Logistic Regression has been applied to the train data to train the model and further the same models are used to predict on the test data.

Logistic Regression

A Grid Search CV has been used to find out the best parameters for logistic regression

```
GridSearchCV(cv=3, estimator=LogisticRegression(max_iter=100000, n_jobs=2),
             n_jobs=-1,
             param_grid={'penalty': ['l1', 'l2', 'elasticnet', 'none'],
                         'solver': ['newton-cg', 'lbfgs', 'liblinear', 'sag',
                                    'saga'],
                         'tol': [0.001, 0.0001]},
             scoring='f1')
```

Figure 56:Grid Search CV

The best parameters that Grid Search CV keeping the scoring parameter as F1 score gives are in the image below:

```
{'penalty': 'l2', 'solver': 'newton-cg', 'tol': 0.001}

LogisticRegression(max_iter=100000, n_jobs=2, solver='newton-cg', tol=0.001)
```

Figure 57: Best Parameters

Linear Discriminant Analysis:

Similarly, like logistic regression data has been splitted to independent variables and target variable.

After this the data has been splitted into 70% train and 30% test data. Linear Discriminant Analysis has been applied to the train data to train the model and further the same models are used to predict on the test data.

```
clf=LinearDiscriminantAnalysis()
model=clf.fit(X_train,Y_train)
model
```

```
LinearDiscriminantAnalysis()
```

Figure 58: Linear Discriminant Analysis

Question 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.

Answer 2.3

Models without treating any outliers:

Logistic Regression:

Train data

The Area under the curve is 0.742 for training dataset. Higher the AOC value better is the model so let's understand all other performance metrics.

```
0.7428497364555432
```

```
[<matplotlib.lines.Line2D at 0x1efcbea2160>]
```

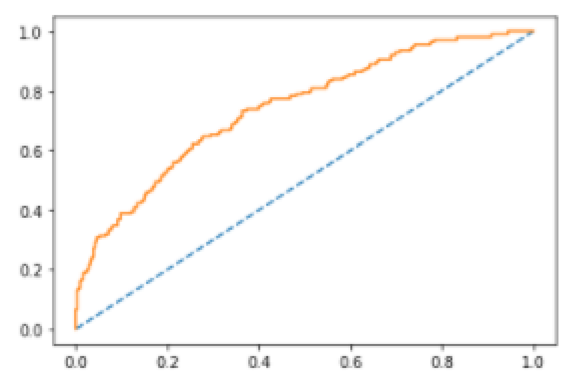

Figure 59: ROC Curve

The accuracy is 68% but the recall for 1 is average. The parameters for 1 are more important because it tells us about the employees that have opted for the holiday package.

```
              precision    recall  f1-score   support

           0       0.68      0.77      0.72       326
           1       0.69      0.57      0.63       284

    accuracy                           0.68       610
   macro avg       0.68      0.67      0.67       610
weighted avg       0.68      0.68      0.68       610
```

Figure 60: Classification Report

Confusion matrix cells are populated by the terms:

True Positive(TP)- The values which are predicted as True and are actually True.

True Negative(TN)- The values which are predicted as False and are actually False.

False Positive(FP)- The values which are predicted as True but are actually False.

False Negative(FN)- The values which are predicted as False but are actually True.

The False negatives in this case is high which is the reason for a low recall score of 1.

163 records are the ones predicted correctly for employees who have opted. 121 records are the employees who had opted but the model has predicted it wrong which is not good. 252 records are the employees who have not opted and model also predicted them correctly. 74 records are who have not opted and model has predicted them as opted.

```
array([[252,  74],
       [121, 163]], dtype=int64)
```

Figure 61: Confusion Matrix

Test Data

The area under the curve score for train data is 0.704 which is almost in line with the training dataset.

```
0.7046861184792219

[<matplotlib.lines.Line2D at 0x1efcc492c40>]
```

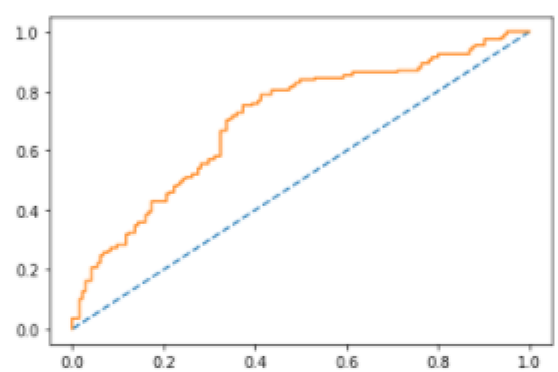

Figure 62: ROC Curve

Accuracy, recall, precision and f1 score are almost inline with the training data.

	precision	recall	f1-score	support
0	0.67	0.70	0.69	145
1	0.61	0.57	0.59	117
accuracy			0.65	262
macro avg	0.64	0.64	0.64	262
weighted avg	0.64	0.65	0.64	262

Figure 63: Classification Report

67 records are the ones predicted correctly for employees who have opted. 50 records are the employees who had opted but the model has predicted it wrong which is not good. 102 records are the employees who have not opted and model also predicted them correctly. 43 records are who have not opted and model has predicted them as opted.

```
array([[102,  43],
       [ 50,  67]], dtype=int64)
```

Figure 64: Confusion Matrix

LDA:

Train data

The Area under the curve is 0.742 for training dataset. Higher the AOC value better is the model so let's understand all other performance metrics.

```
0.7421152682968979

[<matplotlib.lines.Line2D at 0x1efcf5928b0>]
```

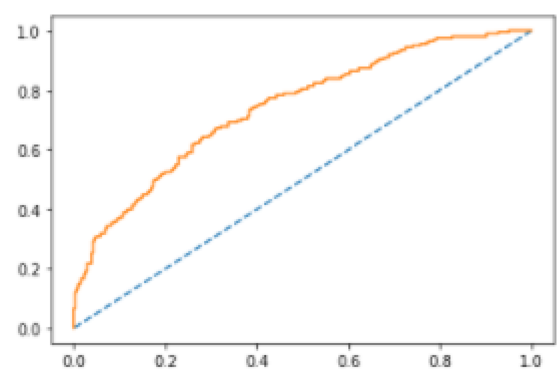

Figure 65: ROC Curve

Accuracy for training dataset is 67%. Precision and f1 score also seems to be good. Recall for 1 is little low.

```
              precision    recall  f1-score   support

           0       0.67      0.77      0.72       326
           1       0.68      0.56      0.61       284

    accuracy                           0.67       610
   macro avg       0.67      0.66      0.66       610
weighted avg       0.67      0.67      0.67       610
```

Figure 66: Classification Report

158 records are the ones predicted correctly for employees who have opted. 126 records are the employees who had opted but the model has predicted it wrong which is not good. 252 records are the employees who have not opted and model also predicted them correctly. 74 records are who have not opted and model has predicted them as opted.

```
array([[252,  74],
       [126, 158]], dtype=int64)
```

Figure 67: Confusion Matrix

Test data

Area under the curve for test data is 0.702. This is inline with the train data.

```
0.7029177718832891
[<matplotlib.lines.Line2D at 0x1efcee464c0>]
```

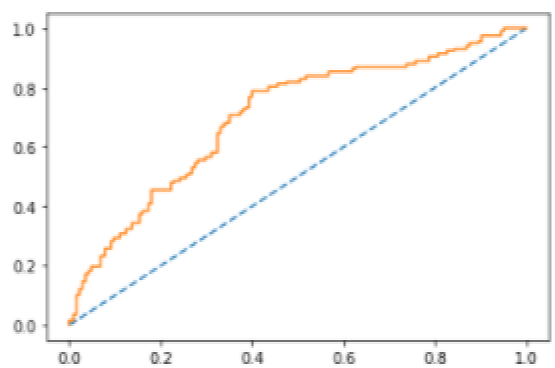

Figure 68: ROC Curve

Accuracy for test data is 64% and other parameters also are in line with the train data.

```
              precision    recall  f1-score   support

           0       0.66      0.71      0.69       145
           1       0.61      0.56      0.58       117

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

Figure 69: Classification Report

65 records are the ones predicted correctly for employees who have opted. 52 records are the employees who had opted but the model has predicted it wrong which is not good. 103 records are the employees who have not opted and model also predicted them correctly. 42 records are who have not opted and model has predicted them as opted.

```
array([[103,  42],
       [ 52,  65]], dtype=int64)
```

Figure 70: Confusion Matrix

The LDA model has been trained and tested by changing the threshold from 0.1 to 1 at an interval of 0.1. But the best results are seen at a threshold of 0.5 which is the default threshold. It gives the best combination of accuracy, f1 score, precision and recall

	Logistic Regression		LDA	
	Train	Test	Train	Test
Accuracy	0.68	0.65	0.67	0.64
AUC	0.74	0.70	0.74	0.70
F1 score	0.63	0.59	0.61	0.58
Recall	0.57	0.57	0.56	0.56
Precision	0.69	0.61	0.68	0.61

Table 2:Model comparison

The Recall, Accuracy, F1 score for logistic regression is better than LDA so the chosen model for this dataset is Logistic Regression.

Models after outlier Treatment:

The Salary variable has a lot of outliers whereas educ, no_young_children, no_older_children are discrete variables so the outliers are only treated for the salary variable and the models are applied. The boxplot below shows that there are no more outliers left for salary variable.

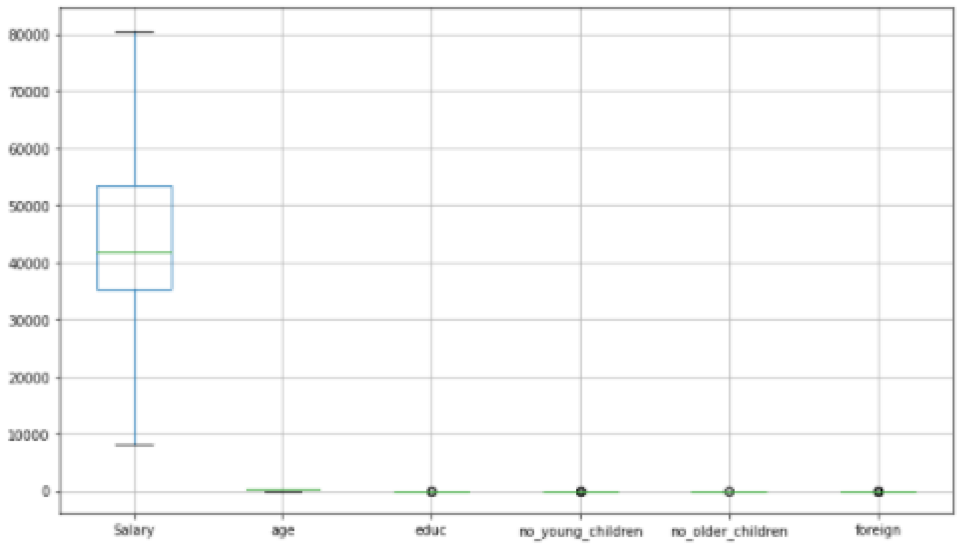

Figure 71: Boxplot

Logistic Regression Model:

A Grid Search CV is used to find the best parameters

```
GridSearchCV(cv=3, estimator=LogisticRegression(max_iter=100000, n_jobs=2),
             n_jobs=-1,
             param_grid={'penalty': ['l1', 'l2', 'elasticnet', 'none'],
                         'solver': ['newton-cg', 'lbfgs', 'liblinear', 'sag',
                                    'saga'],
                         'tol': [0.0001, 1e-05]},
             scoring='f1')
```

Figure 72: Grid Search CV

The below image gives the best parameters for Logistic Regression

```
{'penalty': 'l2', 'solver': 'newton-cg', 'tol': 0.0001}

LogisticRegression(max_iter=100000, n_jobs=2, solver='newton-cg')
```

Figure 73: Best Parameters

Train data:

Area under the curve is 0.74.

```
0.7409595610472651

[<matplotlib.lines.Line2D at 0x28d7e14a610>]
```

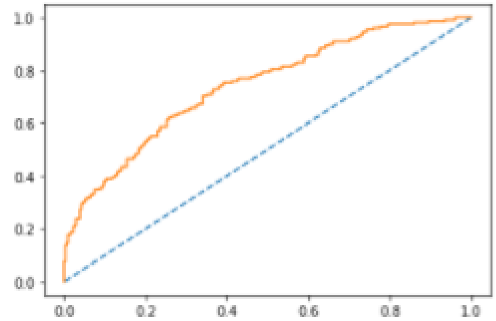

Figure 74: ROC Curve

The accuracy of the train data is 67%. Recall for 1 is little less but all other performance metrics are giving good results.

	precision	recall	f1-score	support
0	0.67	0.77	0.72	326
1	0.68	0.56	0.62	284
accuracy			0.67	610
macro avg	0.68	0.67	0.67	610
weighted avg	0.67	0.67	0.67	610

Figure 75: Classification Report

160 records are the ones predicted correctly for employees who have opted. 124 records are the employees who had opted but the model has predicted it wrong which is not good. 251 records are the employees who have not opted and model also predicted them correctly. 75 records are who have not opted and model has predicted them as opted.

```
array([[251,  75],
       [124, 160]], dtype=int64)
```

Figure 76: Confusion Matrix

Test data:

Area under the curve for test data is 0.704. This is almost in line with test data.

```
0.7049218980253462

[<matplotlib.lines.Line2D at 0x28d7e186850>]
```

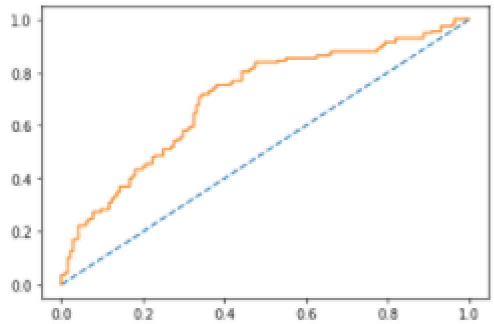

Figure 77: ROC Curve

The accuracy for train data is 65%. Recall has improved for test data and all metrics are in line with train data which the model is right fit.

```
              precision    recall  f1-score   support

           0       0.67      0.70      0.69       145
           1       0.61      0.57      0.59       117

    accuracy                           0.65       262
   macro avg       0.64      0.64      0.64       262
weighted avg       0.64      0.65      0.64       262
```

Figure 78: Classification Report

67 records are the ones predicted correctly for employees who have opted. 50 records are the employees who had opted but the model has predicted it wrong which is not good. 102 records are the employees who have not opted and model also predicted them correctly. 43 records are who have not opted and model has predicted them as opted.

```
array([[102,  43],
       [ 50,  67]], dtype=int64)
```

Figure 79: Confusion Matrix

Linear Discriminant Analysis:

Train data:

Area under the curveis 0.739

0.7394798237276419

[<matplotlib.lines.Line2D at 0x28d7e1e1400>]

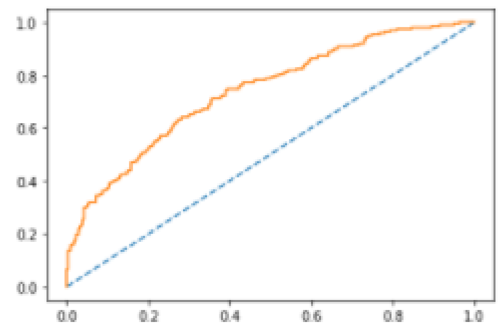

Figure 80: ROC Curve

The accuracy of the train data is 68%. Recall for 1 is a little less suggesting there are more false negatives in the data but all other performance metrics look good.

	precision	recall	f1-score	support
0	0.67	0.78	0.72	326
1	0.69	0.56	0.61	284
accuracy			0.68	610
macro avg	0.68	0.67	0.67	610
weighted avg	0.68	0.68	0.67	610

Figure 81: Classification Report

158 records are the ones predicted correctly for employees who have opted. 126 records are the employees who had opted but the model has predicted it wrong which is not good. 254 records are the employees who have not opted and model also predicted them correctly. 72 records are who have not opted and model has predicted them as opted.

```
array([[254,  72],
       [126, 158]], dtype=int64)
```

Figure 82: Confusion Matrix

Test data:

Area under the curve is 0.702

```
0.7029767167698201

[<matplotlib.lines.Line2D at 0x28d7e22df40>]
```

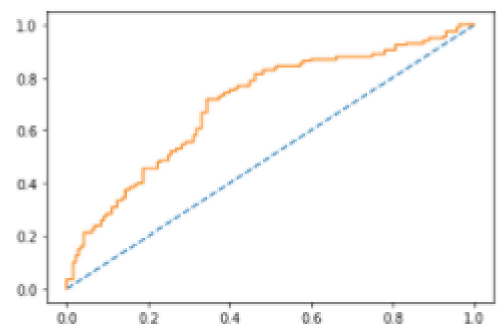

Figure 83: ROC Curve

The accuracy of the test data is 64%. Train and test data are in line with each other and model is right fit.

```
              precision    recall  f1-score   support

           0       0.66      0.71      0.69       145
           1       0.61      0.56      0.58       117

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

Figure 84: Classification Report

65 records are the ones predicted correctly for employees who have opted. 52 records are the employees who had opted but the model has predicted it wrong which is not good. 103 records are the employees who have not opted and model also predicted them correctly. 42 records are who have not opted and model has predicted them as opted.

```
array([[103,  42],
       [ 52,  65]], dtype=int64)
```

Figure 85: ConfusionMatrix

	Logistic Regression		LDA	
	Train	Test	Train	Test
Accuracy	0.67	0.65	0.67	0.64
AUC	0.74	0.70	0.73	0.70
F1 score	0.62	0.59	0.61	0.58
Recall	0.56	0.57	0.56	0.56
Precision	0.68	0.61	0.69	0.61

Table 3:Model comparison

The better model is Logistic regression in this case as well. Accuracy, f1 score, recall is better for Logistic Regression and it is a right fit model.

Question 2.4

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

Answer 2.4

Logistic Regression is the better model so checking the coefficients for each factor.

```
array([[-1.74320515e-05, -5.29508007e-02,  7.15166689e-02,
        -1.45899971e+00, -4.63494204e-02,  1.47629800e+00]])
```

Figure 86: Coefficients

1. Coefficient values for Salary is: -1.74320515e-05

2. Coefficient values for age is: -5.29508007e-02

3. Coefficient values for is educ: 7.15166689e-02

4. Coefficient values for is no_young_children: -1.45899971e+00

5. Coefficient values for no_older_children: -4.63494204e-02

6. Coefficient values for foreign: 1.47629800e+00

The most important factors affecting the target variable Holliday_Package are no_young_children and foreign. Salary seemed to be one of the important factors but after model building Salary does not seem to affect the target variable.

No_young_children and foreign have emerged out to be strong predictors. Salary, age, educ and no_older_children are bad predictors.

No_young_children have negative coefficient which means that more the number of young children employee has it is more unlikely for him to opt for the package.

Foreign has a positive coefficient meaning more foreign employees are opting for the package.

Recommendations:

1. Company to should focus on foreign employees to drive more sales.
2. Employees with young children do not seem to opt for the package so the company can come up with a package for such employees where they can take their children also for the holiday. Although employees with young children avoid going to trips so the company should not focus more on them but can target employees who do not have any young children.

3. Company can plan some marketing and better offers to convert more employees to opt for holiday package.
4. They can offer some discounts to employees with less salary.