PREDICTIVE MODELLING PROJECT REPORT

TEJAS PADEKAR
PGP-DSBA Online
June' 21
Date: 09/06/2021

CONTENTS:

List of Figures

List of Tables

PROBLEM 1

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 Dictionary:

Variable Name	Description
Carat	Carat weight of the cubic zirconia.
Cut	Describe the cut quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal.
Color	Colour of the cubic zirconia.With D being the best and J the worst.
Clarity	cubic zirconia Clarity refers to the absence of the Inclusions and Blemishes. (In order from Best to Worst, FL = flawless, I3= level 3 inclusions) FL, IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1, I2, I3
Depth	The Height of a 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.

Problem Statement 1:
Help the company in predicting the price for the stone on the bases of the details given in the dataset and provide them with the best 5 attributes that are most important.

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

Dataset Head:

	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

There are a total of 26967 rows and 11 columns in the dataset. The first column is just a label column and will not be used in the analysis. Hence, we have dropped it when loading the dataset. So, we are now left with 10 columns.

Null Values:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 26967 entries, 0 to 26966
Data columns (total 10 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   carat    26967 non-null  float64
 1   cut      26967 non-null  object
 2   color    26967 non-null  object
 3   clarity  26967 non-null  object
 4   depth    26270 non-null  float64
 5   table    26967 non-null  float64
 6   x        26967 non-null  float64
 7   y        26967 non-null  float64
 8   z        26967 non-null  float64
 9   price    26967 non-null  int64
dtypes: float64(6), int64(1), object(3)
memory usage: 2.1+ MB
```

There are 6 continuous and 3 categorical variables and 1 more continuous variable which is also our response/target variable of "Price". There are missing values in the variable "depth" which we will impute post further analysis.

Zero Value Count:

```
carat      0
cut        0
color      0
clarity    0
depth      0
table      0
x          3
y          3
z          9
price      0
dtype: int64
```

We can observe that the variables of x, y and z have 0 as a value in some of the features. x variable has three 0's, y also has three 0's and z has nine o's in total. This is an incorrect feature as it is not possible in the real world to have any of the length, width or height as zero for a real cubic zirconia or any tangible product. Hence, we see these as either bad data or outliers and will treat them accordingly by imputing them appropriately. However, first we will relabel them appropriately to give them meaning. x is renamed x_length, y as y_width and z as z_height

Dataset Description (Continuous Variables):

	count	mean	std	min	25%	50%	75%	max
carat	26967.0	0.798375	0.477745	0.2	0.40	0.70	1.05	4.50
depth	26270.0	61.745147	1.412860	50.8	61.00	61.80	62.50	73.60
table	26967.0	57.456080	2.232068	49.0	56.00	57.00	59.00	79.00
x_length	26967.0	5.729854	1.128516	0.0	4.71	5.69	6.55	10.23
y_width	26967.0	5.733569	1.166058	0.0	4.71	5.71	6.54	58.90
z_height	26967.0	3.538057	0.720624	0.0	2.90	3.52	4.04	31.80
price	26967.0	3939.518115	4024.864666	326.0	945.00	2375.00	5360.00	18818.00

The mean and median values are very similar for all the independent variables except for the target variable of price. Price variable also has a very high standard deviation. We can also see that there seems to be outliers present in the dataset including for the target variable as well. We also see some bad data in terms of 0 values in x_length, y_width and z_height variables which we will treat appropriately as well.

Dataset Description (Categorical Variables):

	count	unique	top	freq
cut	26967	5	Ideal	10816
color	26967	7	G	5661
clarity	26967	8	SI1	6571

There are total of 3 categorical variables, cut, color and clarity. There are 5 unique types for the cut variable with 'Ideal' type sold the most with a count of 10816, for color there are 7 unique types with G type used most with 5661 observations and for clarity there are 8 unique types with SI1 having a count of 6571 observations.

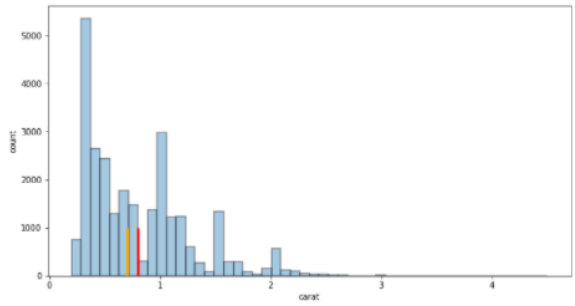
Duplicate Values:

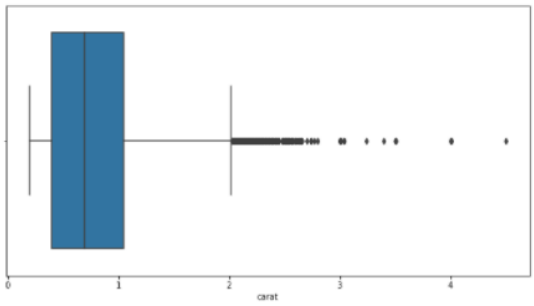
```
Number of duplicate rows = 34
Before (26967, 10)
After (26933, 10)
```

We found 34 duplicate entries which we removed from the dataset. Now we are left with a total of 26933 observations in our dataset.

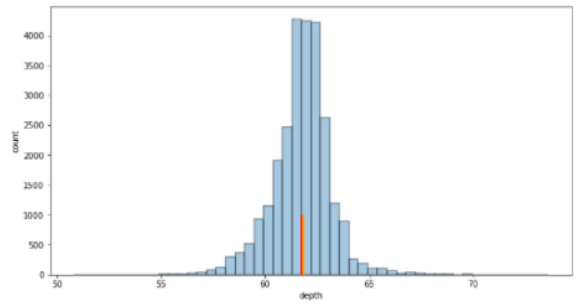
Univariate Analysis:

carat
Skew: 1.11





depth
Skew: -0.03



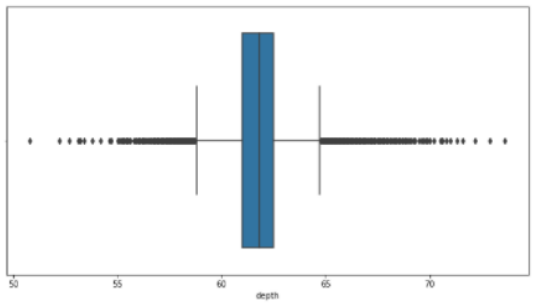
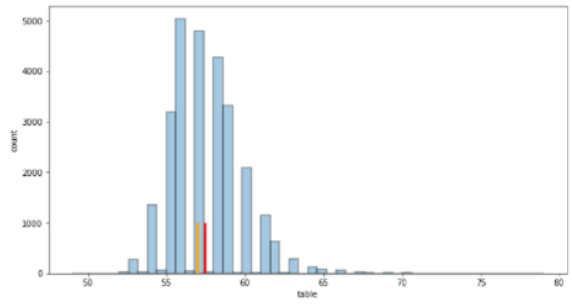
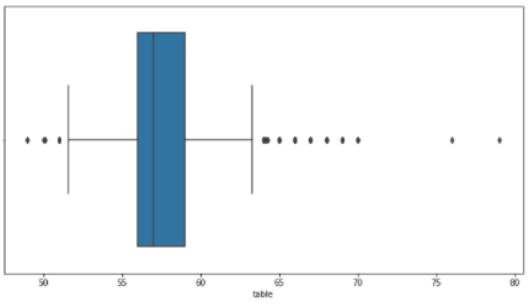
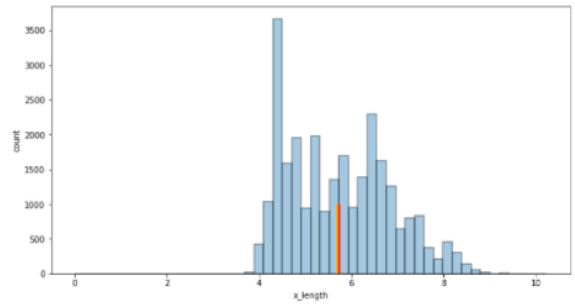


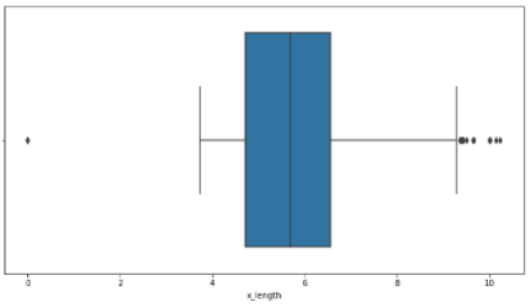
table
Skew: 0.77





x_length
Skew: 0.39





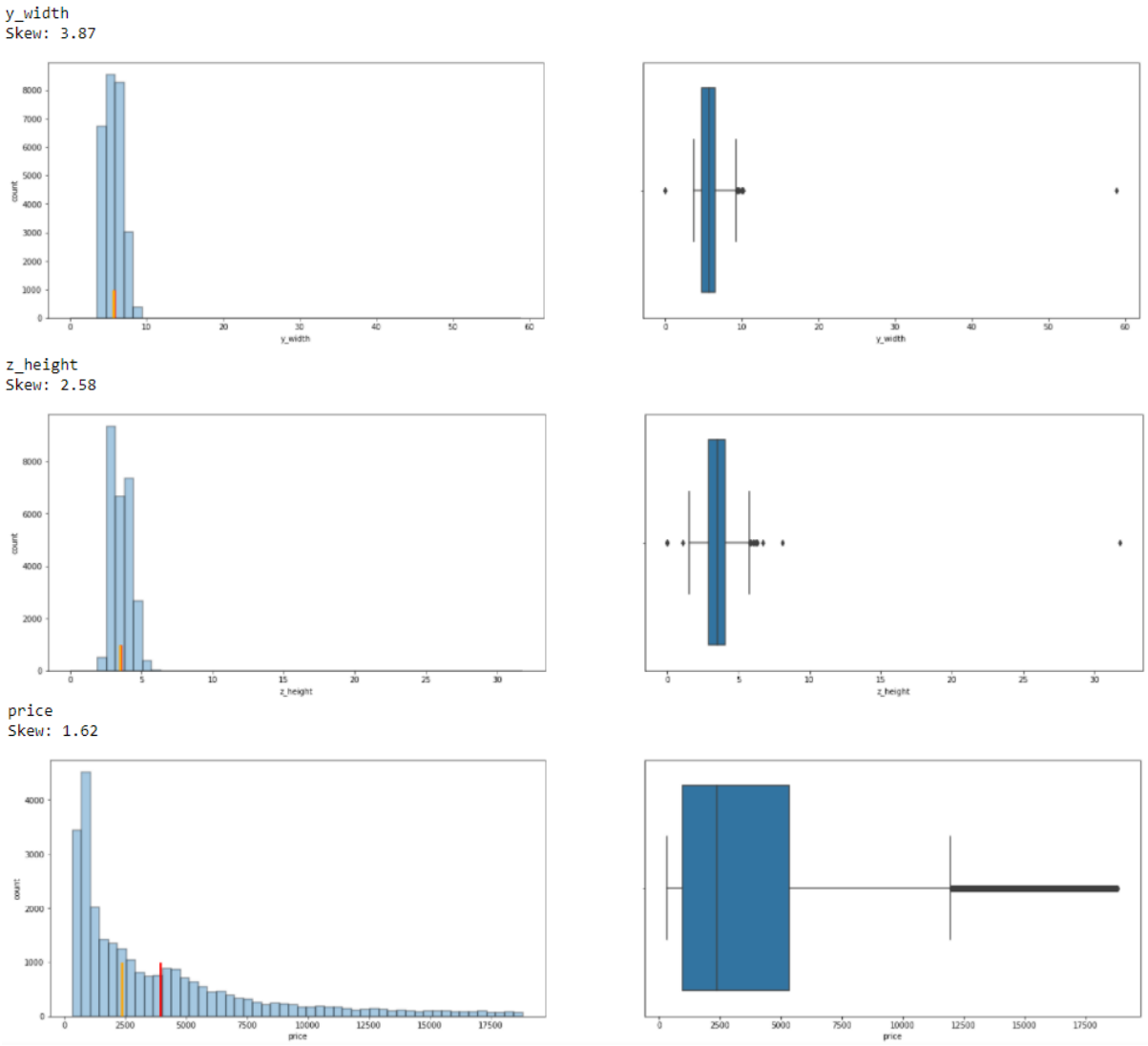
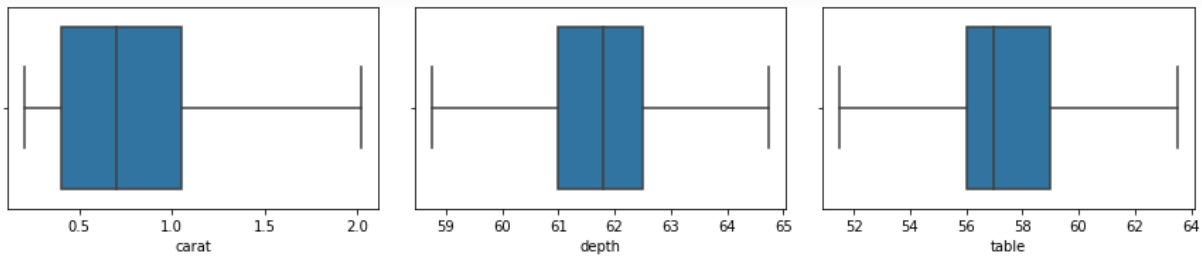


Figure 1: Univariate Analysis Linear Regression

The diagram shows that except for 'depth' variable all of the other continuous variables are right skewed with y_width, carat and price being highly right skewed. However, the means and medians of almost every independent variable are close to each other hence, they are nearly normally distributed. Also, there are outliers present in all of the continuous variables which can have a huge effect on regression. Therefore, we will be treating them appropriately.

Outlier Treatment:



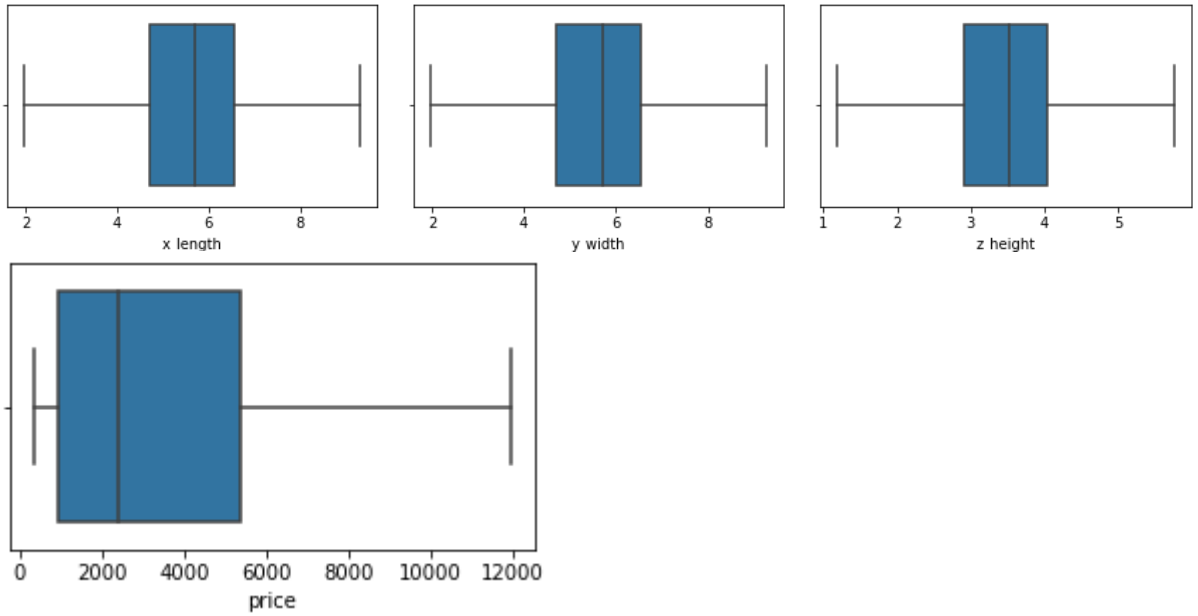
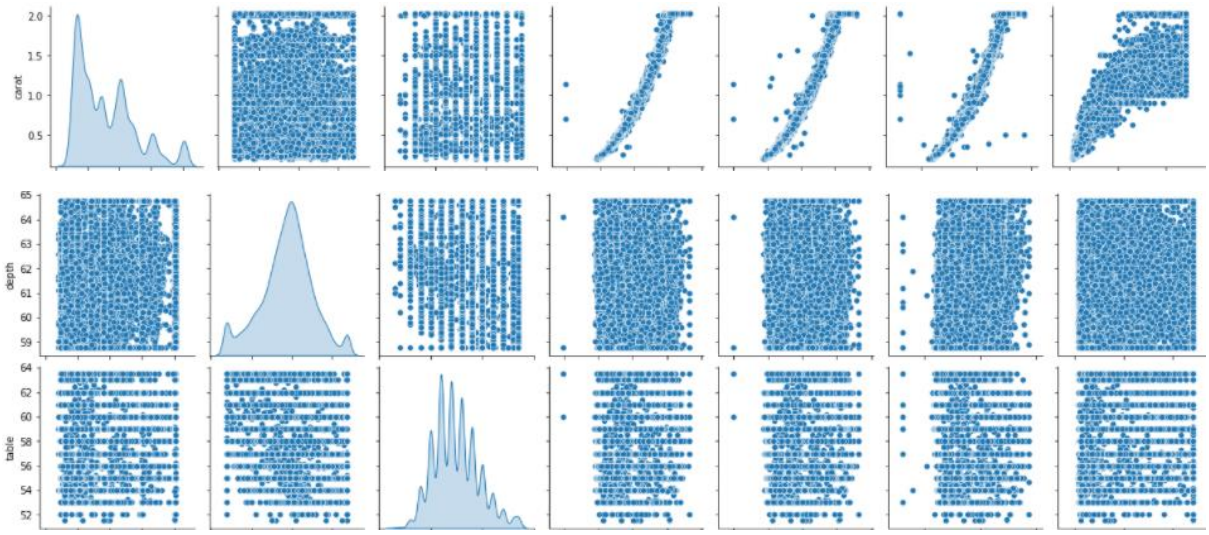


Figure 2 – Outlier Treatment

We can see that the outliers for all the continuous variables are treated successfully.

Bivariate Analysis:



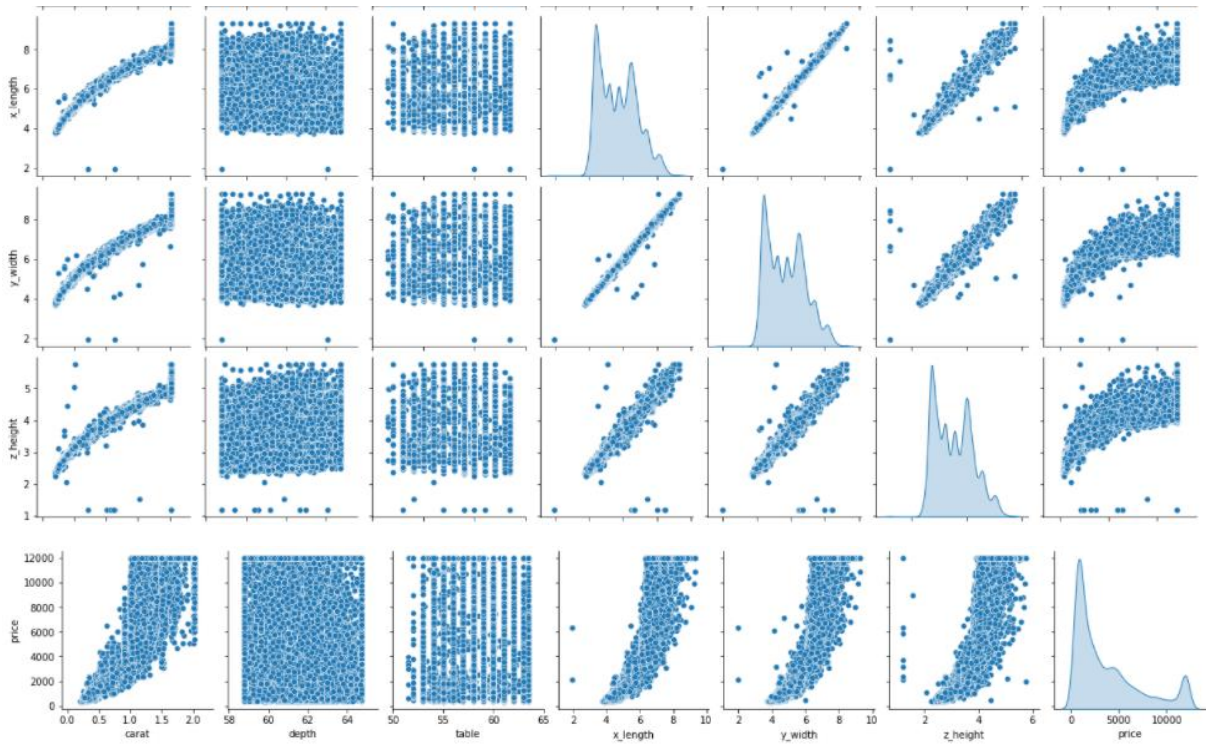


Figure 3 – Bivariate Analysis

As can be seen above, apart from 'depth' variable which has a normal distribution. Skewness for carat variable has reduced after treating the outliers and x_length, y_width and z_height can be seen having similar distribution features however with clusters in them.

Price variable can be seen having strong correlation with the variables of carat, x_length, y_width and z_height but near no correlation with the other two variables of 'depth' and 'table'. Moreover, we can observe strong correlation between a few independent variables among themselves.

We will explore these further in the below correlation heatmap. Similar to outliers, multicollinearity among independent variables is a threat to the performance of the regression model.

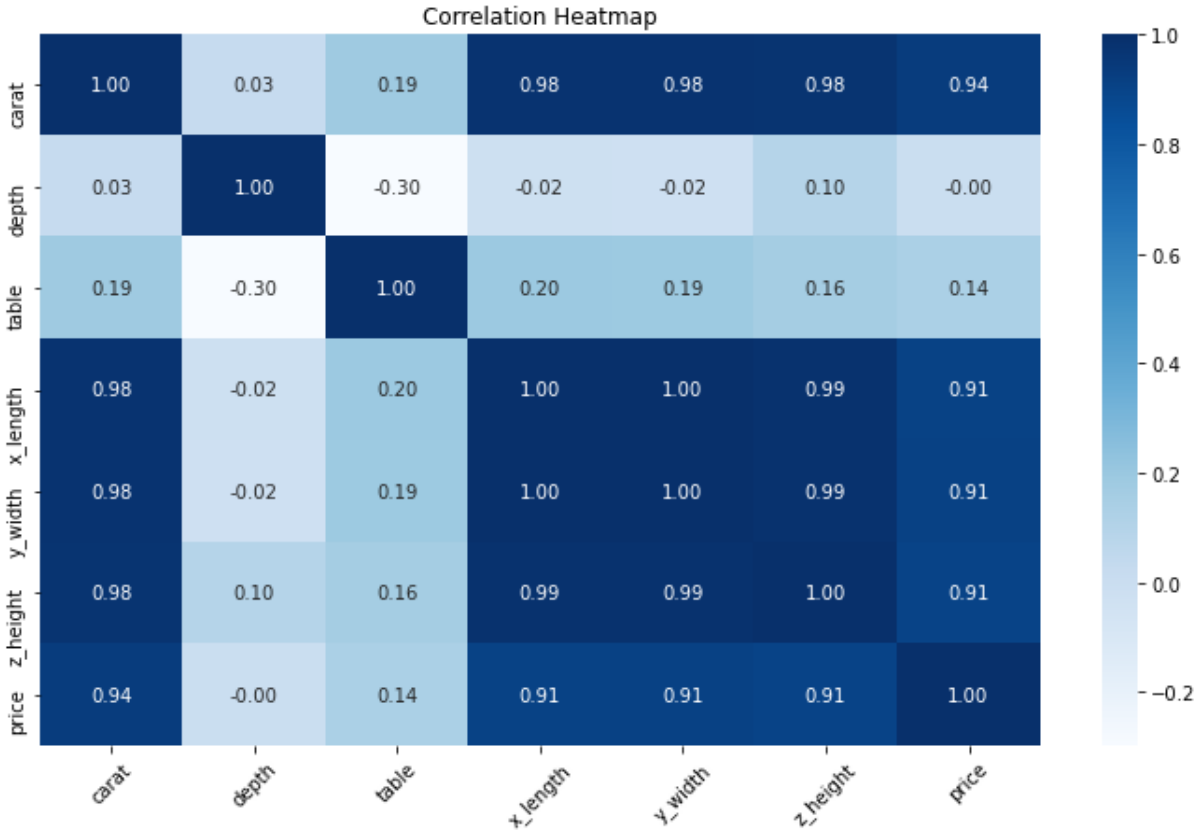


Figure 4 – Correlation Heatmap Linear Regression

From the above figure, we can observe strong positive correlations between the following variables which clearly indicates presence of multicollinearity in them:

<u>For Dependent 'price' variable and Independent Variables:</u>

- price and carat with correlation coefficient 0.94
- price and x_length with correlation coefficient 0.91
- price and y_width with correlation coefficient 0.91
- price and z_height with correlation coefficient 0.91

<u>Between Independent Variables:</u>

- carat and x_length with correlation coefficient 0.98
- carat and y_width with correlation coefficient 0.98
- carat and z_height with correlation coefficient 0.98
- x_length and z_height with correlation coefficient 0.99
- y_width and z_height with correlation coefficient 0.99

From the above figure, we can observe perfect correlation between the following variables:
- x_length and y_width with correlation coefficient 1.00.

From the above figure, we can also observe inverse correlation between the following variables:
- depth and table with correlation coefficient (-0.30)

From the above figure, we can observe no linear relationship between the following variables:

- price and depth (correlation coefficient = 0.00) & Hence, 'depth' variable seems to come across as a poor predictor for 'price' variable.

Collinearity check of Target variable with Independent variables:

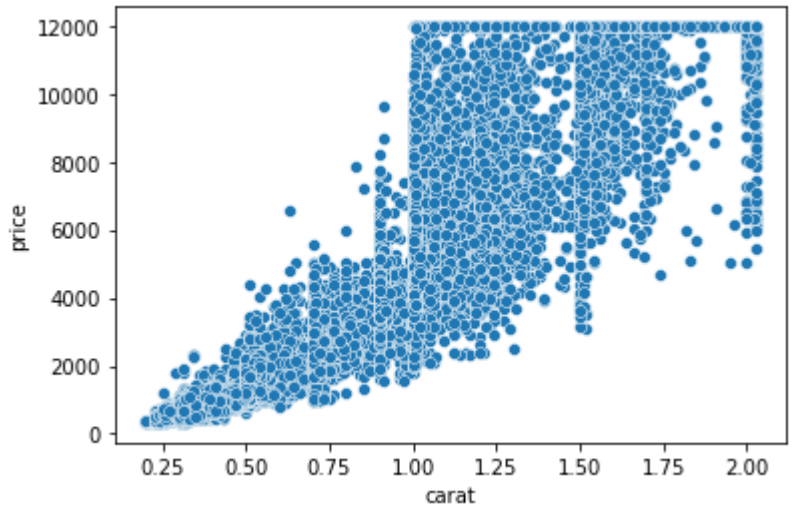


Figure 5 – Correlation Carat and price

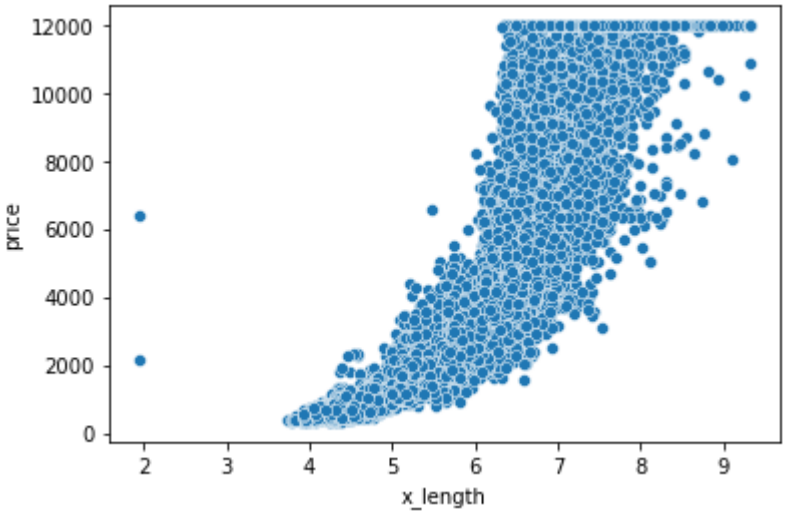


Figure 6 – Correlation x_length and price

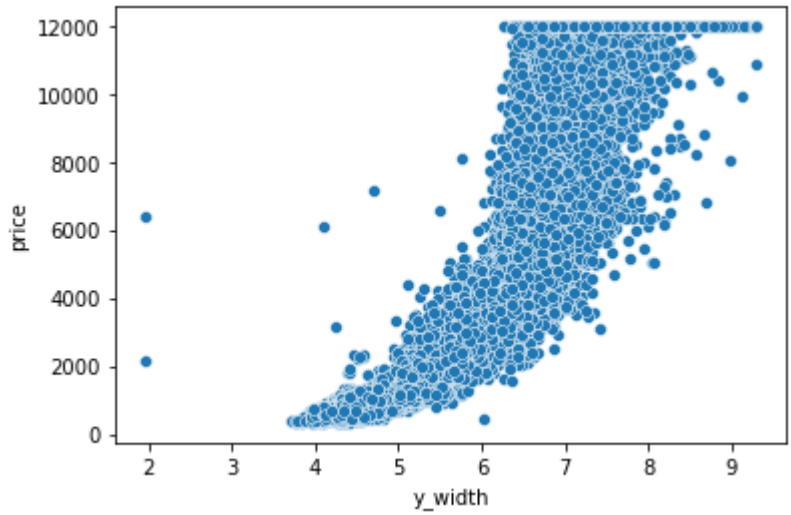


Figure 7 – Correlation y_width and price

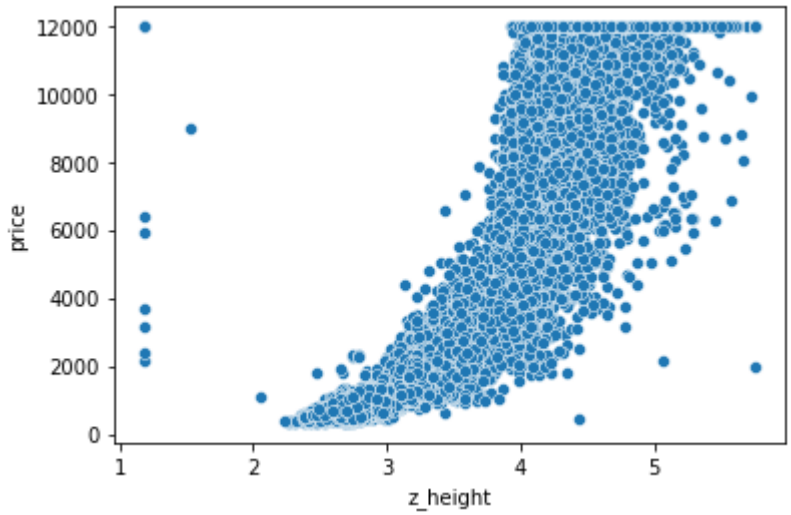


Figure 8 – Correlation z_height and price

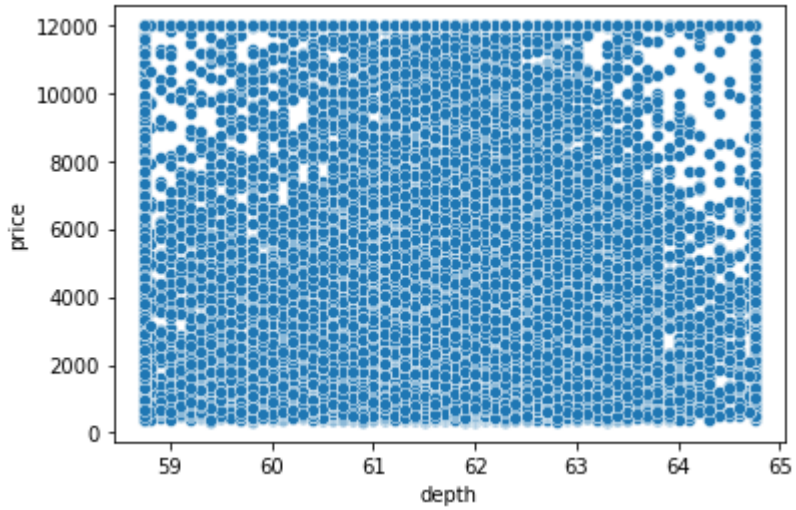


Figure 9 – Correlation depth and price

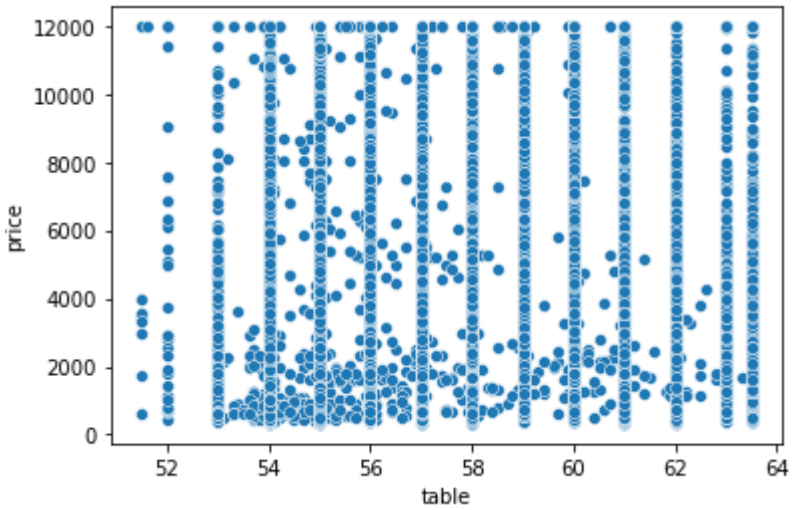


Figure 10 – Correlation table and price

We also see that the target variable 'price' is positively correlated to the 'x_length', 'y_width' and 'z_height' variables but not with 'depth' and 'table'.

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? Do you think scaling is necessary in this case?

Imputing Missing Values:

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

We have imputed the missing Nan values with the median values for 'depth' variable appropriately and there are no more missing values in our dataset as can be seen above. Also, we had also imputed the variables of x_length, y_width and z_height with 0 value with the .25 quantile value for their respective columns when treating the outliers earlier.

Do you think scaling is necessary in this case?

No, scaling is not necessary here as scaling will not affect the train and test results. We do have variables in the dataset of different magnitudes in the dataset and it is always a good practice to scale the data for any type of analysis as models usually get inclined towards variables with higher magnitudes which could bring drastic difference in a models performance.

1.3 Encode the data (having string values) for Modelling. Data Split: Split the data into train and test (70:30). Apply Linear regression. Performance Metrics: Check the performance of Predictions on Train and Test sets using Rsquare, RMSE.

Categorical Variable Features:

```
---- cut ---- 5
            Count  Percent
Fair        780.0     2.90
Good       2435.0     9.04
Very Good  6027.0    22.38
Premium    6886.0    25.57
Ideal     10805.0    40.12
```

```
---- color ---- 7
    Count  Percent
J  1440.0     5.35
I  2765.0    10.27
D  3341.0    12.40
H  4095.0    15.20
F  4723.0    17.54
E  4916.0    18.25
G  5653.0    20.99
```

```
---- clarity ---- 8
       Count  Percent
I1     364.0     1.35
IF     891.0     3.31
VVS1  1839.0     6.83
VVS2  2530.0     9.39
VS1   4087.0    15.17
SI2   4564.0    16.95
VS2   6093.0    22.62
SI1   6565.0    24.38
```

Above we can see the count of unique features in our three categorical variables of 'cut', 'color' & 'clarity'. We observe that cut variable has 5 unique features, color has 7 and clarity has 8.

In the cut variable, 'Ideal' feature which also happens to be the best quality of the lot, has the highest proportion of approx 40% and the worst quality of 'Fair' is also least in proportion of only approx 3%. Hence, maximum proportion of the quality of the cubic zirconia is of the highest standard with features of 'Very Good' (22.38%), 'Premium' (25.57%) and 'Ideal' (40.12%) qualities making up of approx 88% of the dataset. Whereas, poor features of 'Good' (9.04%) and 'Fair' (2.90%) make up for the rest of the approx only 12% of the proportion.

In the color variable, similar to the observation made for the cut variable earlier, the worst feature 'J' counts for only approx 5% of the dataset and the top 4 features of 'D' (12.40%), 'E' (18.25%), 'F' (17.54%) & 'G' (20.99%) cover approx 70% of dataset. 'G', an average feature has the highest proportion in the dataset and most of the features have similar proportions in the dataset.

The clarity variable has only 8 unique features out of the total 13 for this dataset with no record of the best feature of 'FL'. Also, this variable has more proportion of average quality features of clarity such as SI1 with highest proportion of 24.38% followed by VS2 with 22.62%, SI2 with 16.95% and 'VS1' (15.17), which in total comes upto approx 80%. Whereas, the top quality features of 'IF' (3.31%), 'VVS1' (6.83%) & 'VVS2'(9.39%) comprise of most of the balance proportion of only approx 20% of the dataset.

Encoding Categorical Variables:

	carat	cut	color	clarity	depth	table	x_length	y_width	z_height	price
0	0.30	4	5	2	62.1	58.0	4.27	4.29	2.66	499.0
1	0.33	3	3	7	60.8	58.0	4.42	4.46	2.70	984.0
2	0.90	2	5	5	62.2	60.0	6.04	6.12	3.78	6289.0
3	0.42	4	4	4	61.6	56.0	4.82	4.80	2.96	1082.0
4	0.31	4	4	6	60.4	59.0	4.35	4.43	2.65	779.0

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

Above we assigned appropriate numerical values in increasing order starting from 0 to Ordinal Variables and converted them in numeric datatype to include in our analysis as linear regression only takes values which are int or float datatype. Hence, we need to encode the categorical variable to run the regression models.

Linear Regression Model Building and Results

Intercept & Coefficient Values of Independent Variables:

```
Intercept   -2653.694605
carat        8830.478833
cut           110.855270
color         278.002561
clarity       440.556903
depth          -7.494044
table         -12.369931
x_length    -1413.216178
y_width      1246.778618
z_height     -310.685170
```

We can see a very high value for intercept at -2653.69 which seems meaningless. This clearly indicate that we need to scale the data to reduce the intercept closer to 0 for a better result.

From the above, we can also conclude that the coefficients for each of the independent variables will affect the target variable price as below:

One unit increase in carat will increase price by 8830.478
One unit increase in cut will increase the price by 110.855
One unit increase in color will increase the price by 278.002
One unit increase in clarity will increase the price by 440.556
One unit increase in y_width will increase the price by 1246.778

One unit increase in depth will decrease the price by -7.494
One unit increase in table will decrease the price by -12.369
One unit increase in x_length will decrease the price by -1413.216
One unit increase in z_height will decrease the price by -310.685

Training and Testing Scores:

```
regression_model.score(X_train, y_train)
```

```
0.9311671714263209
```

```
regression_model.score(X_test, y_test)
```

```
0.93121145382335
```

RSME Scores:

```
math.sqrt(mse)
```

```
911.1428282209592
```

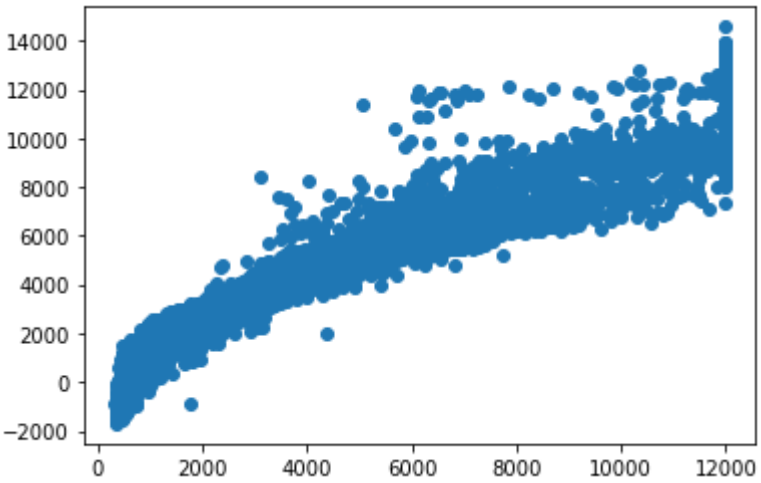


Figure 11 – Correlation Actual Vs Predicted Y

We see a strong correlation between the actual y and predicted y from the above figure with some error scene as well as even though most of the datapoints are concentrated close to each other there are a few which can are scattered.

Ordinary Least Square method:

Hypothesis Testing

Ho = All regression coefficients are 0

Ha = Atleast one regression coefficient in non-zero

OLS Regression Results

Dep. Variable:	price	R-squared:	0.931
Model:	OLS	Adj. R-squared:	0.931
Method:	Least Squares	F-statistic:	2.832e+04
Date:	Wed, 09 Jun 2021	Prob (F-statistic):	0.00
Time:	05:24:00	Log-Likelihood:	-1.5519e+05
No. Observations:	18853	AIC:	3.104e+05
Df Residuals:	18843	BIC:	3.105e+05
Df Model:	9		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-2653.6946	697.034	-3.807	0.000	-4019.943	-1287.446
carat	8830.4788	81.252	108.680	0.000	8671.217	8989.740
cut	110.8553	7.339	15.105	0.000	96.471	125.240
color	278.0026	4.111	67.617	0.000	269.944	286.061
clarity	440.5569	4.461	98.754	0.000	431.813	449.301
depth	-7.4940	8.927	-0.839	0.401	-24.991	10.003
table	-12.3699	3.908	-3.165	0.002	-20.030	-4.710
x_length	-1413.2162	119.863	-11.790	0.000	-1648.158	-1178.275
y_width	1246.7786	118.417	10.529	0.000	1014.670	1478.887
z_height	-310.6852	96.480	-3.220	0.001	-499.795	-121.575

Omnibus:	2721.747	Durbin-Watson:	1.988
Prob(Omnibus):	0.000	Jarque-Bera (JB):	9114.715
Skew:	0.730	Prob(JB):	0.00
Kurtosis:	6.078	Cond. No.	8.99e+03

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.99e+03. This might indicate that there are
strong multicollinearity or other numerical problems.

Figure 12 – OLS Regression Result

INTERPRETATION:

The overall P value is less than alpha, so rejecting H0 and accepting Ha that atleast 1 regression co-efficient is not 0. Here all regression coefficients are not 0.

Moreover, the p-value for depth is 0.401 which is > alpha (0.05) which indicates it to be a poor predictor as we saw earlier during the correlation check as well where it was seen to have 0.00 correlation with the target variable of price. Hence, we can also try building a model without 'depth' variable.

Also, the R2 and adjusted R2 values are high at 0.931 which can strongly suggest that the model built is a good one.

Variance Inflation Factor:

```
carat ---> 112.84069217546832
cut ---> 9.74968052092832
color ---> 5.543448028180082
clarity ---> 5.422847428303261
depth ---> 949.7845695975695
table ---> 753.4137009972687
x_length ---> 10290.781663994765
y_width ---> 9327.239640925243
z_height ---> 1999.358484518994
```

VIF is very high among all the independent variables which clearly indicates strong multicollinearity among the independent variables. All of the variables are above the VIF of 5 and a VIF above 5 indicates strong multicollinearity among the variables.

Conclusion:
The final Linear Regression equation is:

price = b0 + b1 * carat + b2 * cut + b3 * color + b4 * clarity + b5 * depth + b6 * table + b7 * x_length + b8 * y_width + b9 * z_height

price = (-2653.69) * Intercept + (8830.48) * carat + (110.86) * cut + (278.0) * color + (440.56) * clarity + (-7.49) * depth + (-12.37) * table + (-1413.22) * x_length + (1246.78) * y_width + (-310.69) * z_height

When carat increases by 1 unit, price increases by 8830.48 units, keeping all other predictors constant. Similarly, when cut increases by 1 unit, price increases by 110.86 units, keeping all other predictors constant. This increase by 1 unit increasing the price as per coefficient values is applicable to all positive valued attributes.

Also, we have attributes with negative co-efficient values, which too have a significant impact on the price of the diamonds however, this time an increase in 1 unit for these attributes, the price decreases instead of increasing. For eg; when depth increases by 1 unit, price decreases by -7.49 units, keeping all other predictors constant. Similarly, you can see that when x_length increases by 1 unit, price decreases by a huge -1413.22 units, keeping all other predictors constant.

Therefore, we can conclude that the top 5 attributes that are most important to business that affect the price and in turn affects the profitability are:
1. Carat (+8830.48)
2. x_length (-1413.22)
3. y_width (+1246.78)
4. clarity (+440.56)
5. z_height (-310.69)

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

Business Insights and Recommendations:

- Depth is a poor predictor for price and hence, the least amount of production expense should be spent on it and allocation should be diverted to more meaningful and good predictors for price such as any of the above given 5 attributes.

- As we saw earlier the major portion of diamonds have clarity feature which is of lower standards with the best standard FL (Flawless) completely missing from the production itself. This is a crucial area where the business can focus. We saw that a large portion around 80% for clarity is covered by lower to average class where SI1 had highest proportion of 24.38% followed by VS2 with 22.62%, SI2 with 16.95% and 'VS1' (15.17). This concentration needs to be diverted to class having better clarity features such as 'VVS2', 'VVS1', 'IF' and 'FL'

- Given that x-length, y_width and z-height are important attributes, the size and volume of the stones need to be monitored and produced carefully in such a way that the coefficient value impacts are utilized aptly to price the gems accurately.

- Lastly, introduction of diamonds having a good balanced combination of all important attributes will help increase the profits significantly.

****

PROBLEM 2

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 Dictionary:

Variable Name	Description
Holiday_Package	Opted for Holiday Package yes/no?
Salary	Employee salary
age	Age in years
edu	Years of formal education
no_young_children	The number of young children (younger than 7 years)
no_older_children	Number of older children
foreign	foreigner Yes/No

Problem Statement 2:
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.

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.

Dataset Head:

	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

The dataset contents 872 observations across 8 columns in total. The first column is just a label and will not be used in the analysis. Hence, we have dropped it when loading the dataset. So, we remain with 7 columns to perform regression.

Null Values:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 872 entries, 0 to 871
Data columns (total 7 columns):
 #   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)
memory usage: 47.8+ KB
```

There are 5 continuous and 1 categorical variable of binary type apart from target variable of 'holliday_Package' which is also a binary type categorical variable. We will correct the spelling of our target variable to Holiday_package for further analysis. There are no missing values in the dataset as well.

Dataset Description (Continuous Variables):

	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

We can see that Salary variable is having some difference between the mean and median due to which it can appear to be rightly skewed and also it looks to contain outliers and so does years of education. Rest of the data is observed to have almost similar mean and median values without any outliers which we will further evaluate during our analysis. There are max 3 children below age of 7 years and for older than 7 years there are 6 children whereas 25% of the employees are yet to have any children.

Dataset Description (Categorical Variables):

```
---- Holiday_package ----
     Count  Percent
no   471.0    54.01
yes  401.0    45.99

---- foreign ----
     Count  Percent
no   656.0    75.23
yes  216.0    24.77
```

The class is divided as such where in the response variable of 'Holiday-package' the class of interest is 54% and the other class is of 46%. Hence, we can confirm that there is no class imbalance in the data

Also, the employees who have been to foreign are only 25% whereas about 75% are yet to travel overseas which seems like a good opportunity for the employer.

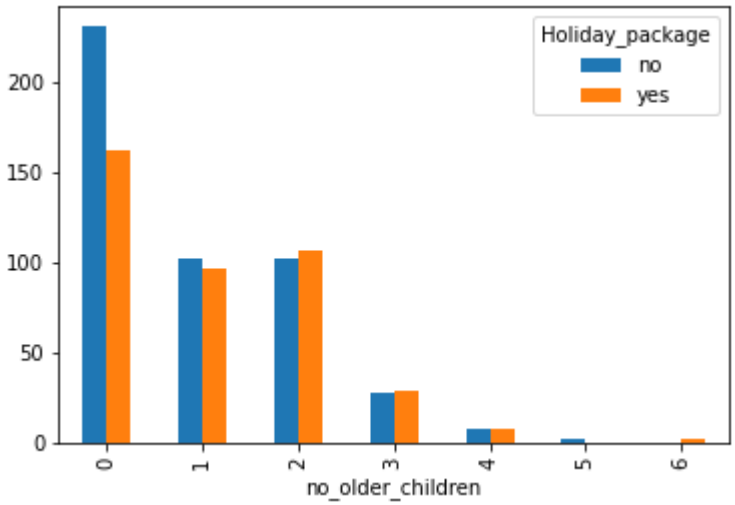


Figure 13 – Crosstab Holiday_package & no_older_children

As we can see with increasing age of children the chances to take a holiday decreases especially beyo nd 2 children. Also, the magnitude of taking a holiday drastically reduces when we move from 0 child older than 7 years to 1 child and beyond.

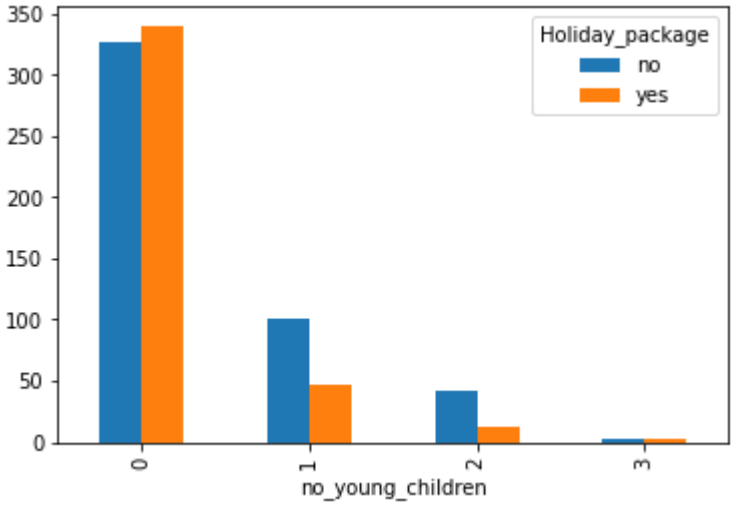


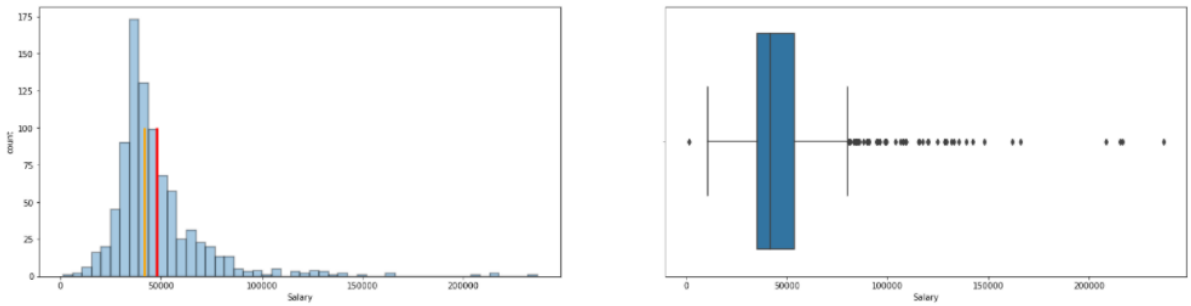
Figure 14 – Crosstab Holiday_package & no_young_children

There is not much difference for preference of holiday for 0 child as yes and no are almost the same. Again similar to no_older_children, here too the magnitude drastically reduces when we move from 0 to 1 child and beyond.

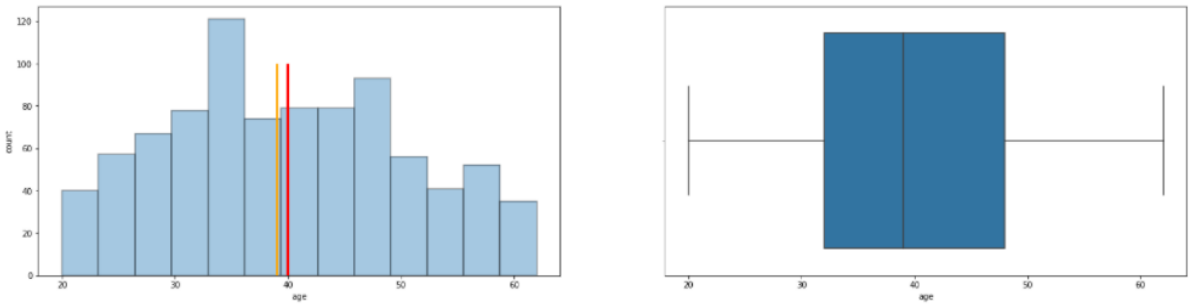
UNIVARIATE ANALYSIS

Please note: Mean – Red and Median – Orange in below plots

Salary
Skew: 3.1



age
Skew: 0.15



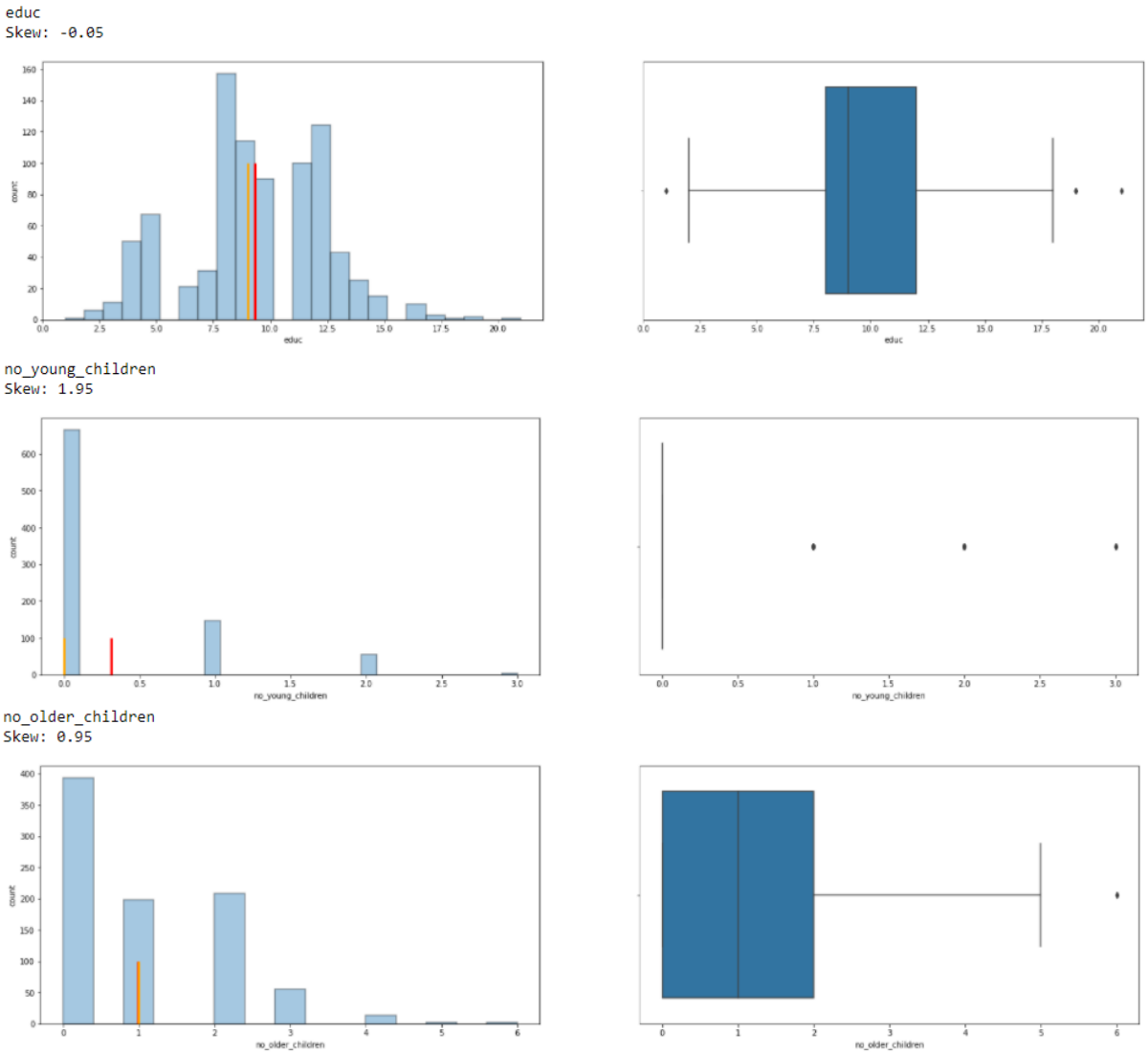
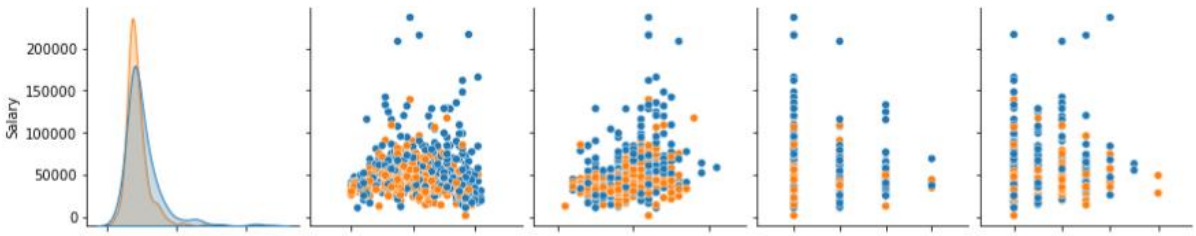


Figure 15 – Univariate Analysis Logistic Regression

From above it can be seen clearly that Salary variable is rightly skewed and contains outliers. Also, both variables for no of children can also be seen to be right skewed and containing outliers. Variables of age and educ are near normally distributed with only age being the variable without any outlier. However, even though there is a presence of outliers as per dataset, they seem to be justifiable and hence, we will not treat the outliers for this case as of now.

Bivariate Analysis



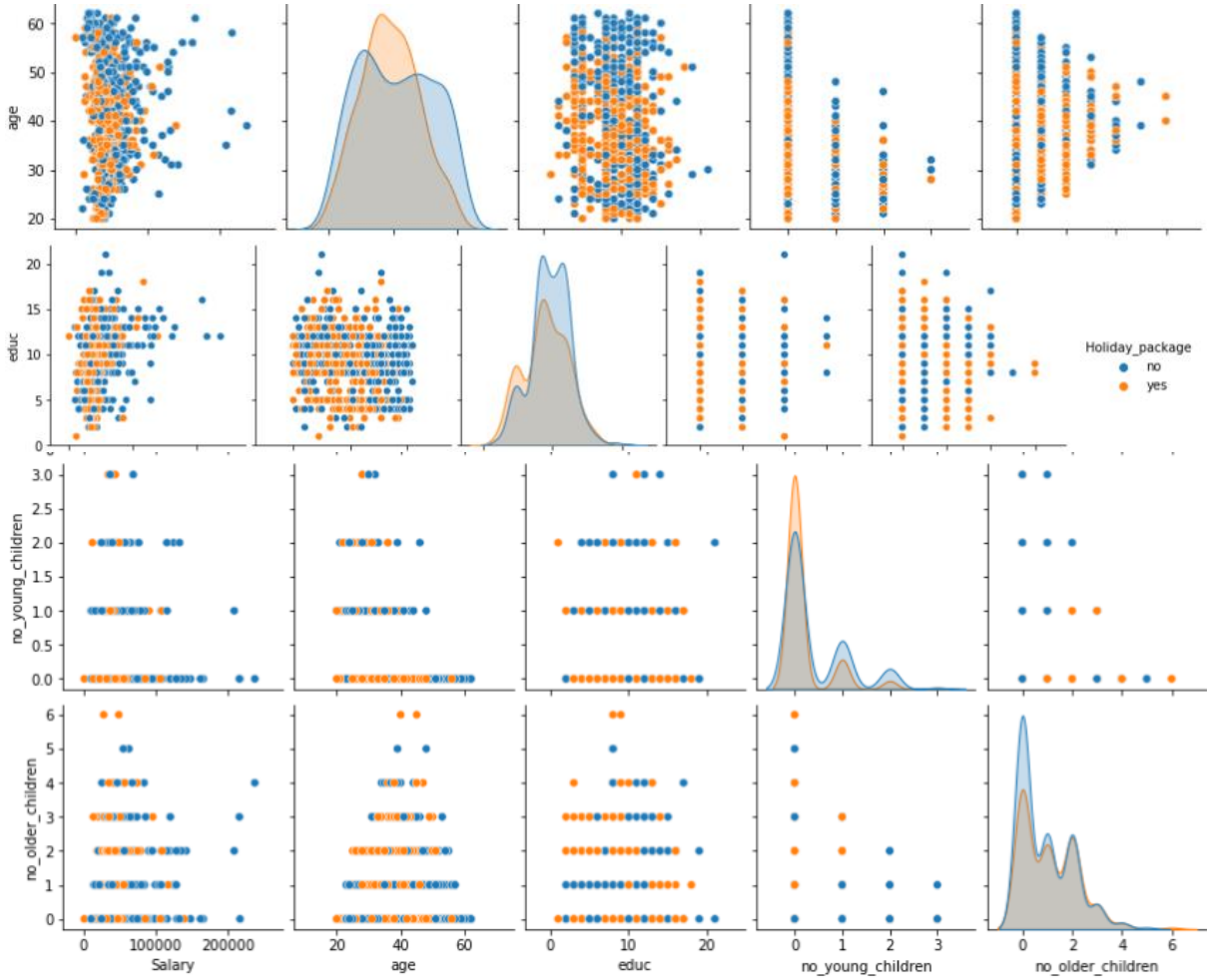


Figure 16 – Bivariate Analysis Logistic Regression

For the diagonals above we can observe that the classes are overlapping each other i.e., they are not well separated and thus the dataset may not prove to have enough good predictors for the model and an overall high f score as well. Also, there are a few peaks in educ, no_young_children and no_older_children which could be appearing due to the outliers present in the data. Let us further check on the correlation of the independent variables as from the above figure it does not seem to have any strong correlations among them.

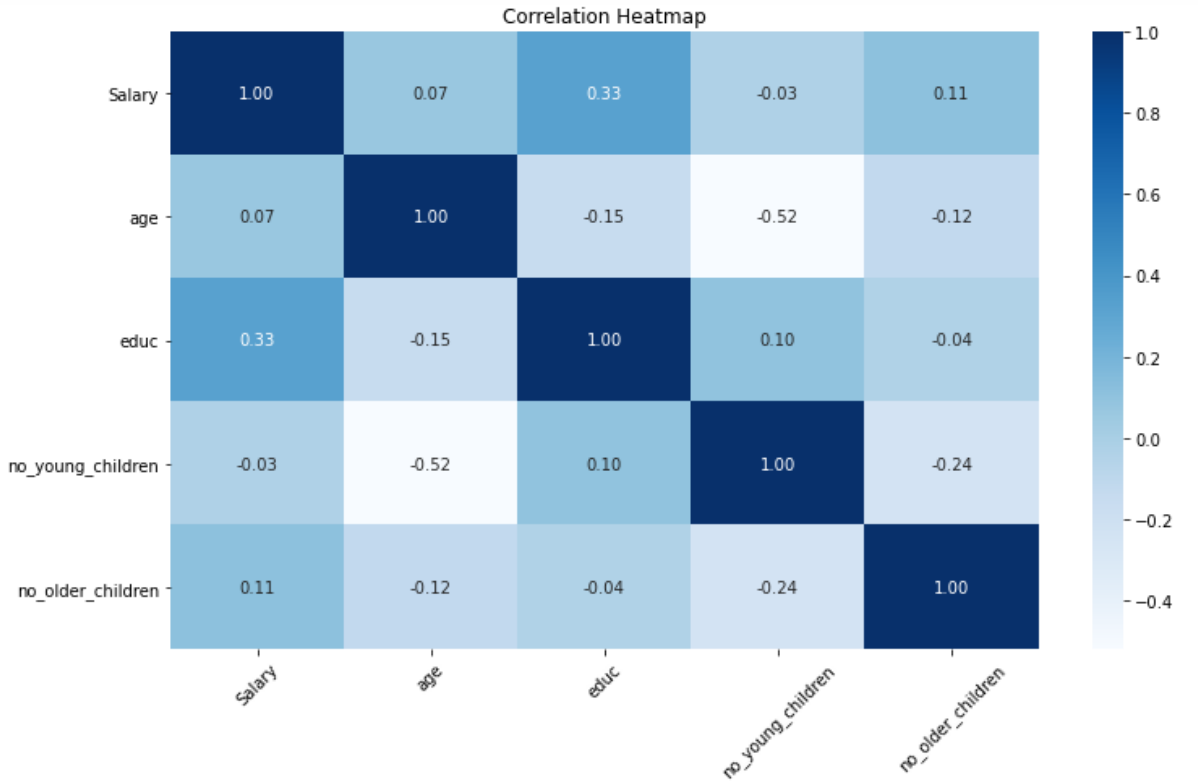


Figure 17 – Correlation Heatmap Logistic Regression

As stated earlier, we do not see any strong correlation among the independent variables which is a good sign for regression. We can only see a strong inverse relation between age and no_young_children.

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).

Encoding Categorical Variables:

	Holiday_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

We have converted the categorical variables of Holiday_package and foreign into numerical data type by assigning 0 in place of no and 1 in place of yes to them.

Train-Test Split: 70:30

```
y_train.value_counts(normalize=True)
```

```
Holiday_package
0                 0.534426
1                 0.465574
dtype: float64
```

```
y_test.value_counts(normalize=True)
```

```
Holiday_package
0                 0.553435
1                 0.446565
dtype: float64
```

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.

Model Evaluation:

```
# Accuracy - Training Data
model.score(X_train, y_train)
```

```
0.6786885245901639
```

```
# Accuracy - Test Data
model.score(X_test, y_test)
```

```
0.6374045801526718
```

AUC and ROC for the training and testing data:

```
AUC: 0.743
```

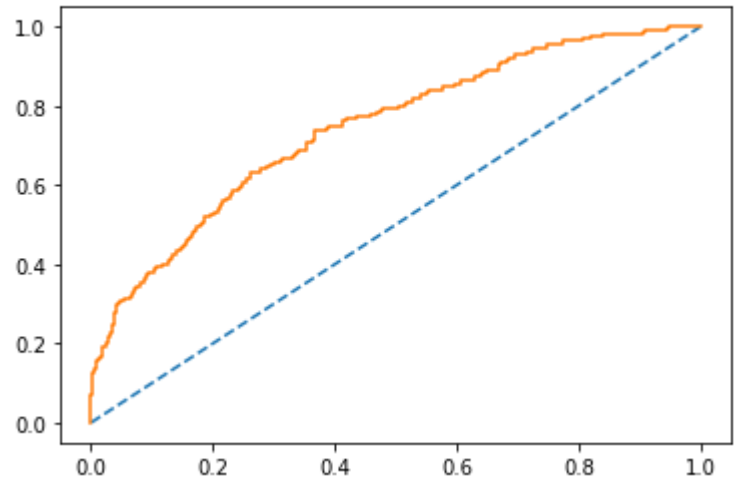


Figure 18 - AUC and ROC for Training Data of Logistic Regression

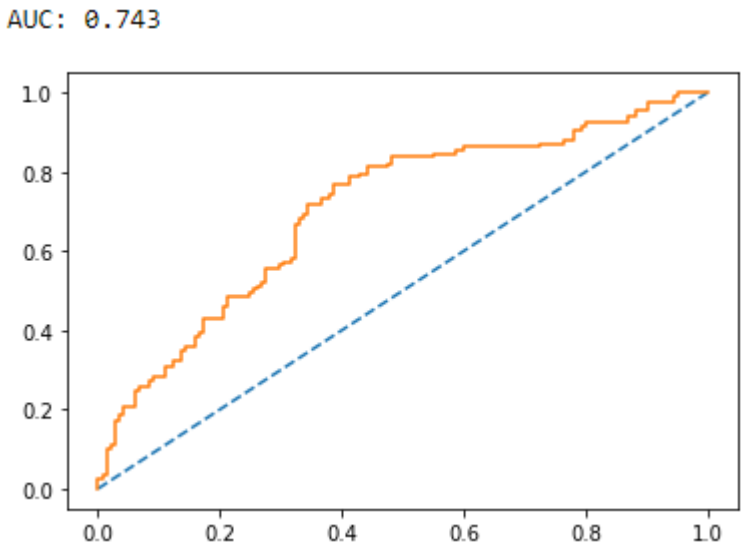


Figure 19 - AUC and ROC for Test Data of Logistic Regression

Confusion Matrix:

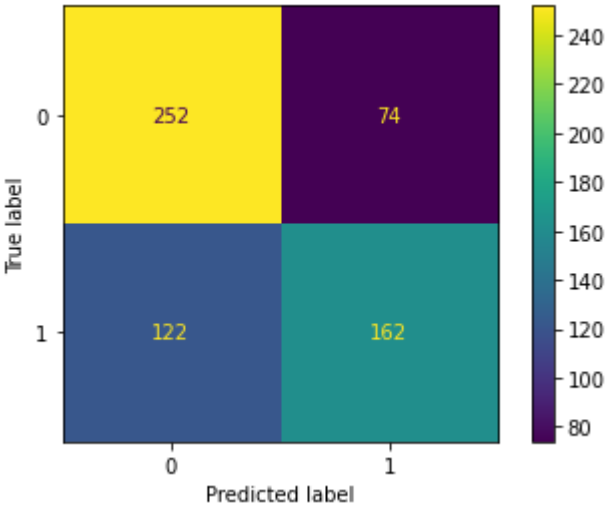


Figure 20 - Confusion Matrix on Training Data for Logistic Regression

The confusion matrix indicates that there are a total of 414 correct predictions (252+162) where 252 are True Negatives and 162 are True Positives and 196 (122+74) incorrect predictions where 74 are False Positives and 122 are False Negatives.

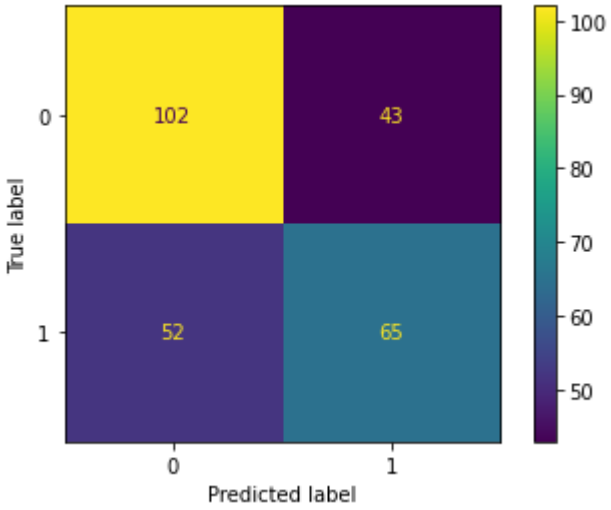


Figure 21: Confusion Matrix on Test Data for Logistic Regression

The confusion matrix indicates that there are a total of 167 correct predictions (102+65) where 102 are True Negatives and 65 are True Positives and 95 (52+43) incorrect predictions where 43 are False Positives and 52 are False Negatives.

Classification Report:

	precision	recall	f1-score	support
0	0.67	0.77	0.72	326
1	0.69	0.57	0.62	284
accuracy			0.68	610
macro avg	0.68	0.67	0.67	610
weighted avg	0.68	0.68	0.67	610

Figure 22: Classification Report of Training Data for Logistic Regression

	precision	recall	f1-score	support
0	0.66	0.70	0.68	145
1	0.60	0.56	0.58	117
accuracy			0.64	262
macro avg	0.63	0.63	0.63	262
weighted avg	0.64	0.64	0.64	262

Figure 23: Classification Report of Testing Data for Logistic Regression

We can see a sharp fall in the recall in the test report compared to the training report. That means the test could not separate false positives as True negative (Precision = True Positives / (True Positive + False Positives)

Also, the overall accuracy has also declined from 68% to 64% and the F score has dropped from 62% to 58%.

Logistic Regression Result:

```
Optimization terminated successfully.
         Current function value: 0.612003
         Iterations 5
                           Logit Regression Results
==============================================================================
Dep. Variable:        Holiday_package   No. Observations:                  872
Model:                          Logit   Df Residuals:                      866
Method:                           MLE   Df Model:                            5
Date:                Wed, 09 Jun 2021   Pseudo R-squ.:                 0.1129
Time:                        10:25:15   Log-Likelihood:                -533.67
converged:                       True   LL-Null:                       -601.61
Covariance Type:            nonrobust   LLR p-value:                1.337e-27
==============================================================================
                         coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Salary            -1.584e-05   4.07e-06     -3.896      0.000   -2.38e-05   -7.87e-06
age                  -0.0173      0.005     -3.370      0.001      -0.027      -0.007
educ                  0.1105      0.024      4.578      0.000       0.063       0.158
no_young_children    -0.9674      0.152     -6.373      0.000      -1.265      -0.670
no_older_children     0.0924      0.068      1.362      0.173      -0.041       0.225
foreign               1.6075      0.189      8.502      0.000       1.237       1.978
==============================================================================
```

Figure 24: Logistic Regression Result

The model converges after 5 iterations. The model is significant with model deviance as below
DN = −2(log LN - log LG) = −2(-601.61 – (-533.67))=1203.22 – 1067.34 = 135.88 on 5 df.
Also, no_older_children is not significant as it is > alpha 0.05

LDA Model Evaluation:

```
# Accuracy - Training Data
model.score(X_train, y_train)
```

```
0.6721311475409836
```

```
# Accuracy - Test Data
model.score(X_test, y_test)
```

```
0.6412213740458015
```

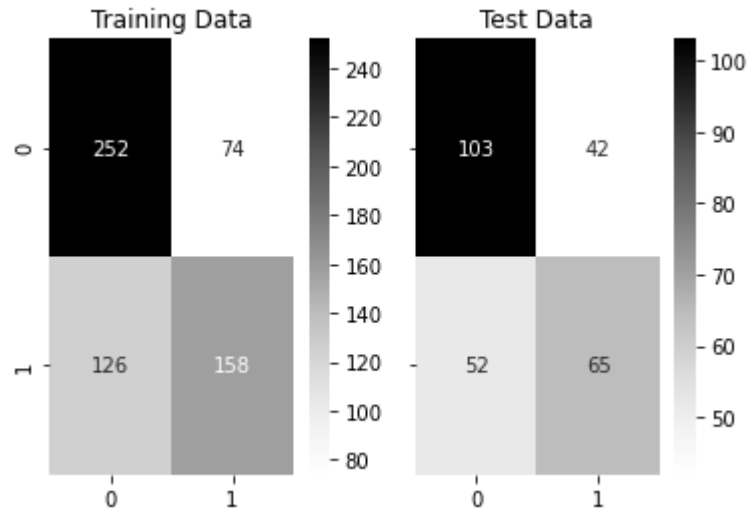


Figure 25 - Confusion Matrix for Training and testing data of LDA

We do not see much difference in the confusion matrix of training and testing data set for Logit and LDA. They are almost very similar.

Classification Report of the training data as per LDA:

	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 26 - Classification Report of Training Data for LDA

Classification Report of the test data as per LDA:

	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 27 - Classification Report of Test Data for LDA

AUC and ROC for the training and testing data:

```
AUC for the Training Data: 0.742
AUC for the Test Data: 0.703
```

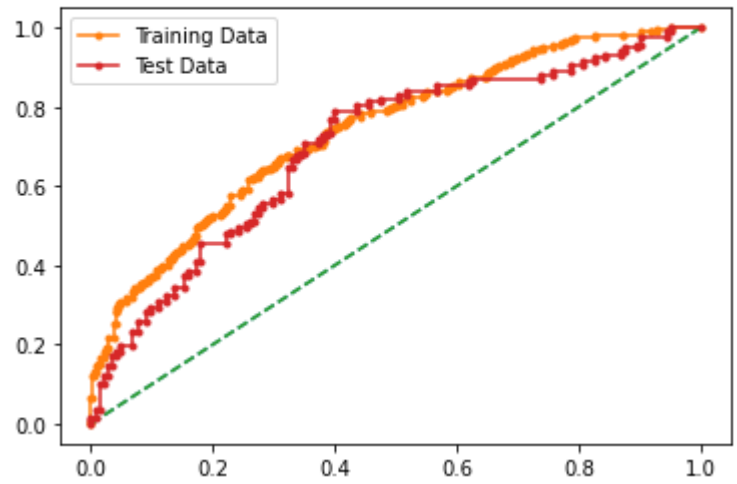


Figure 28 - AUC and ROC for the training and testing data of LDA

Conclusion:

When compared both the Logistic Regression model and LDA model show similar results for the train and test. In the below table we have provided all the scores. However, the F scores are not very high for both these models.

	LR TRAIN	LR TEST	LDA TRAIN	LDA TEST
ACCURACY	**0.68**	**0.64**	**0.67**	**0.64**
PRECISION	**0.69**	**0.60**	**0.68**	**0.61**
RECALL	**0.57**	**0.56**	**0.56**	**0.56**
AUC / ROC	**0.743**	**0.743**	**0.742**	**0.703**
F SCORE	**0.62**	**0.58**	**0.61**	**0.58**

Table 1 - Comparison Chart LR Vs LDA

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

Both the models have performed similarly and we also saw no_older_children is not a good predictor in the logistic regressing result.

It was also seen in our EDA that increase in no of older children decreases the probability of the employee taking a holiday. Hence, it is important that the company target its 25% of employees who do not have any children yet and they can also target the employees who have a child below 7 years as it is very unlikely that these employees will travel once their children turn 8 years old.

Salary and Age also had negative coefficients apart from no_older_children. Therefore, the employer can target younger employees and who are not at senior management positions which usually have high salary packages.