PREDICTIVE MODELLING PROJECT

Pooja Kabadi
PGP-DSBA Online
Batch- A4
23-01-2022

Table of Contents:

List of Figures:

List of Tables:

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.

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.
Colour	Colour of the cubic zirconia. With D being the worst and J the best.
Clarity	Clarity refers to the absence of the Inclusions and Blemishes. (In order from Worst to Best in terms of avg price) IF, VVS1, VVS2, VS1, VS2, 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.

The purpose of the report is to examine past information on cubic zirconia in order to assist the company in predicting price slots for the stone based on the information provided in the dataset. Understanding the data and examining the pattern of how pricing influences various variables. Providing business insights based on exploratory data analysis and predictions of price.

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.

Exploratory Data Analysis:

Read and view data after dropping 'Unnamed: 0' variable:

	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
5	1.02	Ideal	D	VS2	61.5	56.0	6.46	6.49	3.99	9502
6	1.01	Good	H	SI1	63.7	60.0	6.35	6.30	4.03	4836
7	0.50	Premium	E	SI1	61.5	62.0	5.09	5.06	3.12	1415
8	1.21	Good	H	SI1	63.8	64.0	6.72	6.63	4.26	5407
9	0.35	Ideal	F	VS2	60.5	57.0	4.52	4.60	2.76	706

Checking for the information of features:

```
<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
```

Checking the Skewness and Kurtosis:

```
zirconia.skew()

carat     1.116481
depth    -0.028618
table     0.765758
x         0.387986
y         3.850189
z         2.568257
price     1.618550
dtype: float64
```

```
zirconia.kurt()

carat        1.215364
depth        3.674431
table        1.582166
x           -0.657825
y          159.291616
z           87.006350
price        2.148617
dtype: float64
```

Checking the description of dataset:

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
carat	26967.0	NaN	NaN	NaN	0.798375	0.477745	0.2	0.4	0.7	1.05	4.5
cut	26967	5	Ideal	10816	NaN	NaN	NaN	NaN	NaN	NaN	NaN
color	26967	7	G	5661	NaN	NaN	NaN	NaN	NaN	NaN	NaN
clarity	26967	8	SI1	6571	NaN	NaN	NaN	NaN	NaN	NaN	NaN
depth	26270.0	NaN	NaN	NaN	61.745147	1.41286	50.8	61.0	61.8	62.5	73.6
table	26967.0	NaN	NaN	NaN	57.45608	2.232068	49.0	56.0	57.0	59.0	79.0
x	26967.0	NaN	NaN	NaN	5.729854	1.128516	0.0	4.71	5.69	6.55	10.23
y	26967.0	NaN	NaN	NaN	5.733569	1.166058	0.0	4.71	5.71	6.54	58.9
z	26967.0	NaN	NaN	NaN	3.538057	0.720624	0.0	2.9	3.52	4.04	31.8
price	26967.0	NaN	NaN	NaN	3939.518115	4024.864666	326.0	945.0	2375.0	5360.0	18818.0

Checking for duplicates in this dataset:

```
# Are there any duplicates?
dups = zirconia.duplicated()
print('Number of duplicate rows = %d' % (dups.sum()))
zirconia[dups]
```

```
Number of duplicate rows = 34
```

Checking the data types in the dataset:

```
zirconia.dtypes
```

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

Checking for number of rows and columns:

```
zirconia.shape
```

```
(26967, 10)
```

Observations:

- Dataset has 11 columns and 26967 rows including the 'unnamed:0' column.
- The first column "Unnamed: 0" has only serial numbers, so we can drop it as it is not useful.

- There are both categorical and continuous data. For categorical data, we have cut, colour and clarity for continuous data we have carat, depth, table, x. y, z and price.
- Price will be target variable.
- The dataset is used for predicting the price for the zirconia 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.
- There are around 697 missing values in the variable 'depth' which will be imputed during the data pre-processing stage.
- There are 34 duplicate values present in the dataset, although there is a probability that 2 or more stones can be of similar dimensions and features but we will drop the duplicates so avoid any overlapping.
- There is total 5 unique types of 'cut' out of which the highest number of cut is 'Ideal' one which accounts to almost 10816 of observations, which is approximately 50% of the dataset.
- There is total 7 types of 'color' out of which highest number of color is 'G', which is 5661, accounts to almost 25% of the dataset.
- There is total 8 types of 'clarity' in the dataset and the highest number of 'clarity' is 'SI1' which is 6571 which accounts to almost 30% of the dataset.
- Skewness and Kurtosis is also calculated for each column, Data with high skewness indicates lack of symmetry and high value of kurtosis indicates heavily tailed data.
- Based on summary descriptive, the data looks good, we see that for most of the variables the mean/medium are nearly equal.

Data Visualization:

Univariate Analysis for Numeric variables:

Let us define a function 'Univariate Analysis numeric' to display information as part of univariate analysis of numeric variables. The function will accept column name and number of bins as arguments. The function will display the statistical description of the numeric variable, histogram or distplot to view the distribution and the box plot to view 5-point summary and outliers if any

1 - Carat: Carat weight of the cubic zirconia

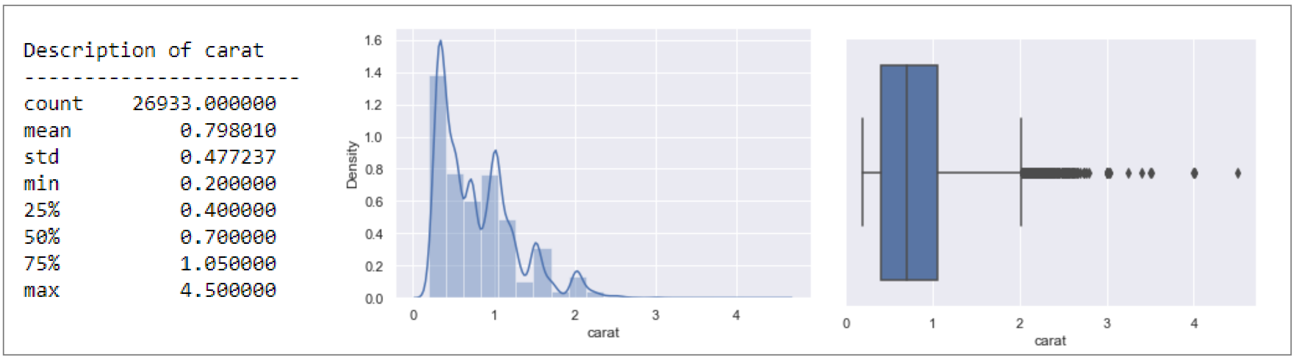


Figure 1. Boxplot and Distplot of Carat.

- From the above graphs, we can infer that mean 'carat' weight of the cubic zirconia is around 0.79 with the minimum of 0.20 and maximum of 4.50.
- The distribution of 'cart' is right skewed with skewness value of 1.1164.
- The distribution spikes at around 0.4 ,1, 1.5 and 2
- The distplot shows the distribution of most of data from 0 to 2.5.
- The box plot of the 'cart' variable shows presence of large number of outliers.

2 - Depth: The Height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter.

```
Description of depth
------------------------
count    26236.000000
mean        61.745285
std          1.412243
min         50.800000
25%         61.000000
50%         61.800000
75%         62.500000
max         73.600000
```

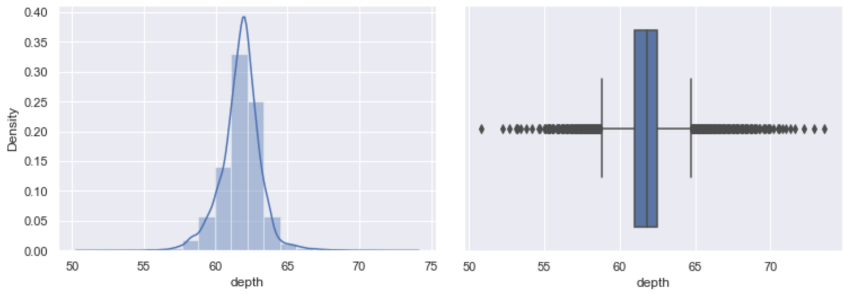


Figure 2. Boxplot and Distplot of Depth

- From the above graphs, we can infer that mean 'depth' height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter is around 61.74 with the minimum of 50.80 and maximum of 73.60.
- The distribution of 'depth' is slightly left skewed with skewness value of -0.0286.
- The distribution follows a near normal distribution with long tails both on the right side and the left side.
- The distplot shows the distribution of most of data from 55 to 70.
- The box plot of the 'depth' variable shows presence of large number of outliers.

3- Table: The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter.

```
Description of table
------------------------
count    26933.000000
mean        57.455950
std          2.232156
min         49.000000
25%         56.000000
50%         57.000000
75%         59.000000
max         79.000000
```

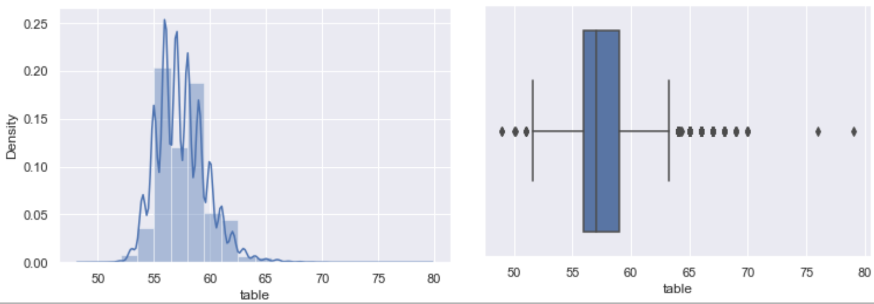


Figure 3. Boxplot and Distplot of Table.

- From the above graphs, we can infer that mean width of the cubic zirconia's 'Table' expressed as a Percentage of its Average Diameters around 57.45 with the minimum of 49.00 and maximum of 79.00.
- The distribution of 'table' is right skewed with skewness value of 0.7657.
- The distribution has multiple spikes at around 53, 55,60 and 62.5.
- The distplot shows the distribution of most of data from 50 to 65.
- The box plot of the 'table' variable shows presence of outliers.

4- X: Length of the cubic zirconia in mm.

```
Description of x
----------------------
count    26933.000000
mean         5.729346
std          1.127367
min          0.000000
25%          4.710000
50%          5.690000
75%          6.550000
max         10.230000
```

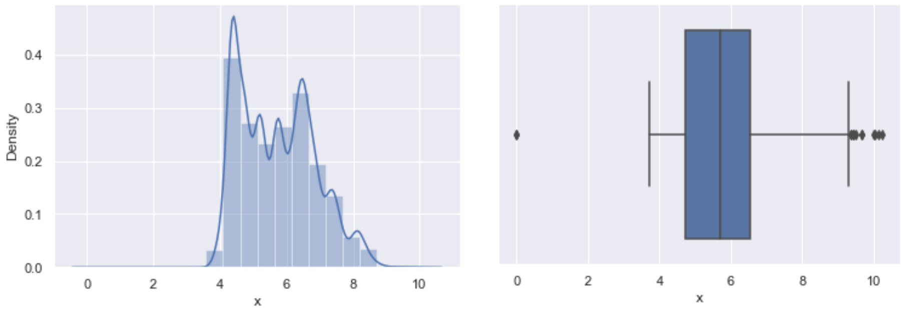


Figure 4. Boxplot and Distplot of 'X'

- From the above graphs, we can infer that mean 'X' length of the cubic zirconia in mm is around 5.72.
- The distribution of 'X' is slightly right skewed with skewness value of 0.3879.
- This distribution has various spikes.
- The distplot shows the distribution of most of data from 3 to 10.
- The box plot of the 'X' variable shows presence of few outliers.

5- Y: Width of the cubic zirconia in mm.

```
Description of y
----------------------
count    26933.000000
mean         5.733102
std          1.165037
min          0.000000
25%          4.710000
50%          5.700000
75%          6.540000
max         58.900000
```

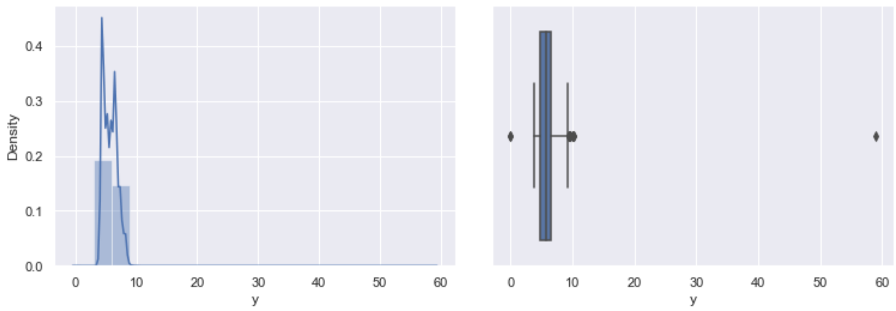


Figure 5. Boxplot and Distplot of 'Y'

- From the above graphs, we can infer that mean 'Y' Width of the cubic zirconia in mm is around 5.73.
- The distribution of 'Y' is right skewed with skewness value of 3.8501.
- The distribution has an extremely long right-side tail because of presence of one outlier at around 60.
- The distplot shows the distribution of most of data from 0 to 10.
- The box plot of the 'Y' variable shows presence of few outliers.

6 - Z: Height of the cubic zirconia in mm.

```
Description of z
----------------------
count    26933.000000
mean         3.537769
std          0.719964
min          0.000000
25%          2.900000
50%          3.520000
75%          4.040000
max         31.800000
```

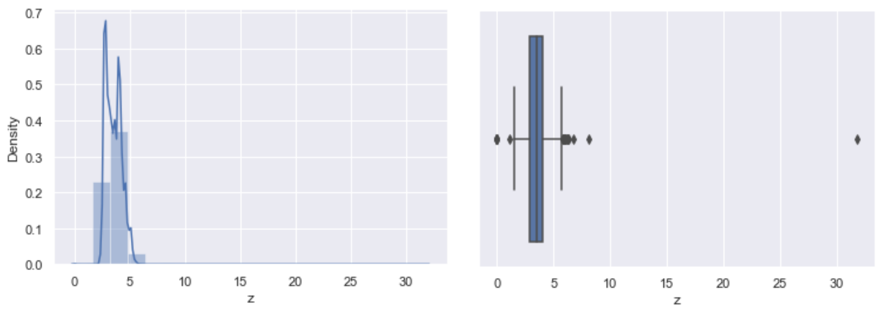


Figure 6. Boxplot and Distplot of 'Z'

- From the above graphs, we can infer that mean 'Z' Height of the cubic zirconia in mm is around 3.53.
- The distribution of 'Z' is right skewed with skewness value of 2.568.
- The distribution has an extremely long right-side tail because of presence of one outlier at around 30.
- The distplot shows the distribution of most of data from 0 to 5.
- The box plot of the 'Z' variable shows presence of few outliers.

7 - Price: The Price of the cubic zirconia

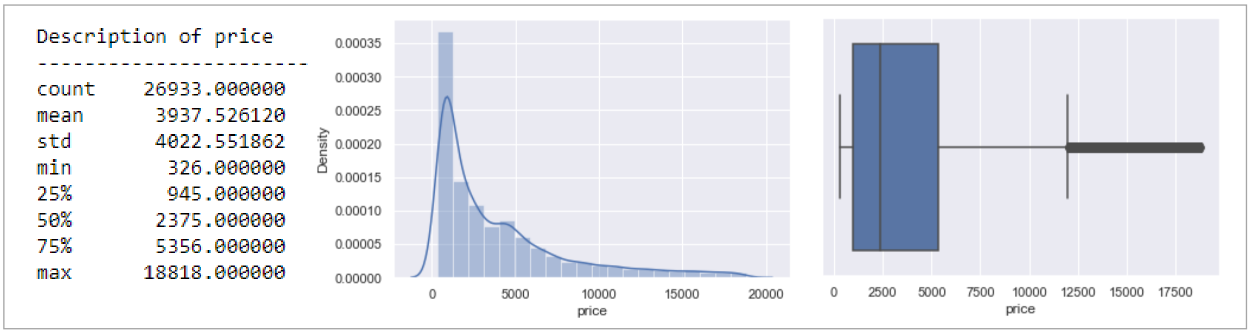


Figure 7. Boxplot and Distplot of Price

- From the above graphs, we can infer that mean the Price of the cubic zirconia is around 3939.51 with the minimum of 326.00 and maximum of 18818.00.
- The distribution of 'Price' is right skewed with skewness value of 1.6185.
- The distribution has an extremely long right-side tail because of presence of one outlier at around 30.
- The distplot shows the distribution of most of data from 325 to 15000.
- The box plot of the 'Price' variable shows presence of large number of outliers.

Observations:

Table 1. Inferences of Univariate Data visualization.

Sl. No	Features	Distribution	Skewness	Outliers
1	Carat	Right Skewed	+1.116	Yes
2	Depth	Almost Normal	-0.028	Yes
3	Table	Right Skewed	+0.765	Yes
4	X: Length	Right Skewed	+0.387	Yes
5	Y: Width	Right Skewed	+3.850	Yes
6	Z: Height	Right Skewed	+2.568	Yes
7	Price	Right Skewed	+1.618	Yes

- Mean and Median values are not very far away from each other.
- Data of all attributes are skewed (mostly right) except X (Length).
- Data for X (Length) is almost normal, outliers tend to make it a little left skewed.
- There are outliers in all numerical features of the cubic zirconia dataset.

Univariate Analysis for Categorical variables:

1. Cut- Describe the cut quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal.

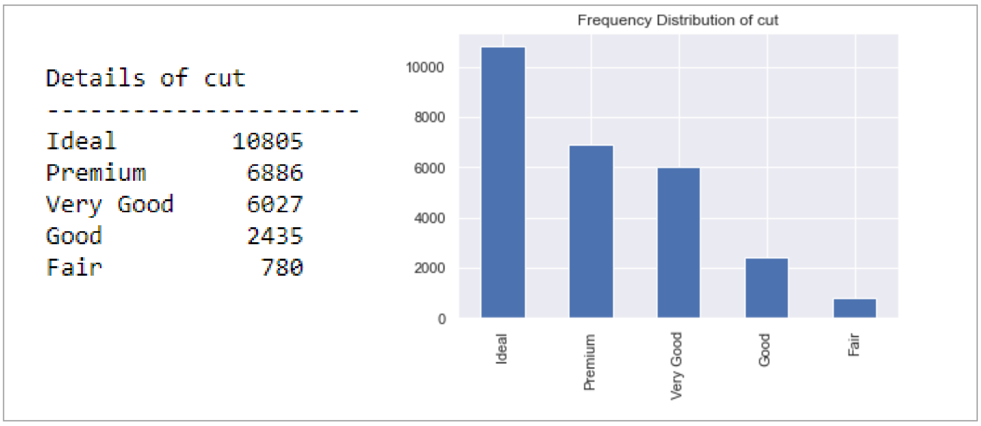


Figure 8. Frequency Distribution of Cut

2. Colour- Colour of the cubic zirconia. With D being the worst and J the best.

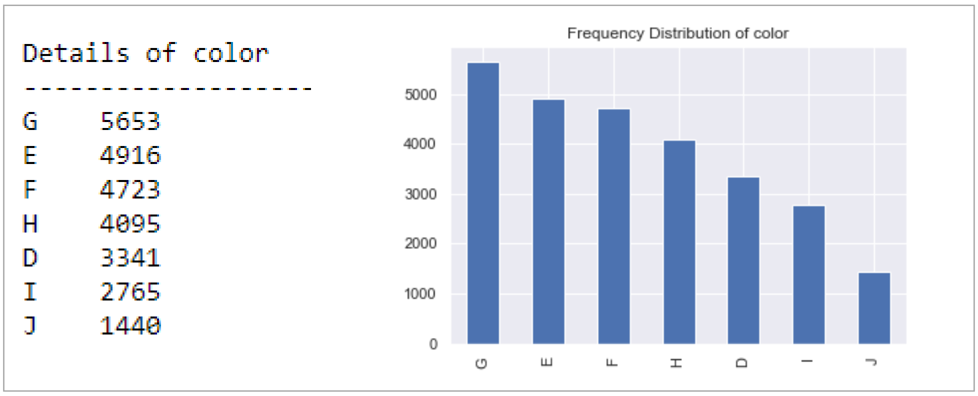


Figure 9. Frequency Distribution of colour.

3. 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

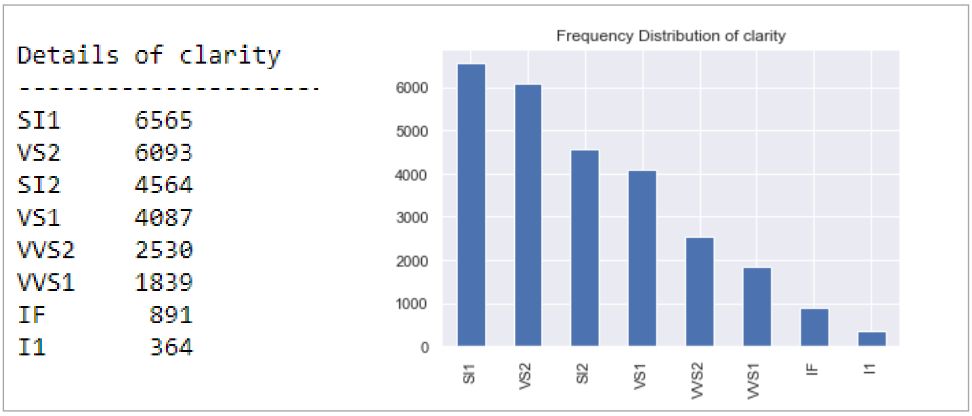


Figure 10. Frequency Distribution of Clarity.

Observations:

- The distribution of the 'cut' which describe the cut quality of the cubic zirconia, in which 'Ideal' cut shows maximum frequency of 10816 and the least frequency cut observed is the 'Fair' one.
- The distribution of the 'Colour' of the cubic zirconia, shows 'G' colour with maximum frequency of 5661 and the least frequency one is J.
- The distribution of the 'clarity' of the cubic zirconia (Clarity refers to the absence of the Inclusions and Blemishes), shows 'SI1' type with maximum frequency of 6571 and the least frequently observed is 'I1'

Bivariate Analysis of Categorical variable with Price:

Cut with Price:

Statistical description of Cut variable with respective price.

cut	count	mean	std	min	25%	50%	75%	max
Fair	780.0	4568.096154	3745.800173	369.0	2117.0	3342.5	5430.0	18574.0
Good	2435.0	3926.336756	3621.197004	335.0	1157.0	3087.0	5111.5	18707.0
Ideal	10805.0	3454.820639	3869.198651	326.0	872.0	1762.0	4668.0	18804.0
Premium	6886.0	4544.558525	4320.888420	326.0	1038.5	3116.5	6268.5	18795.0
Very Good	6027.0	4032.267961	4016.865952	336.0	910.0	2633.0	5438.0	18818.0

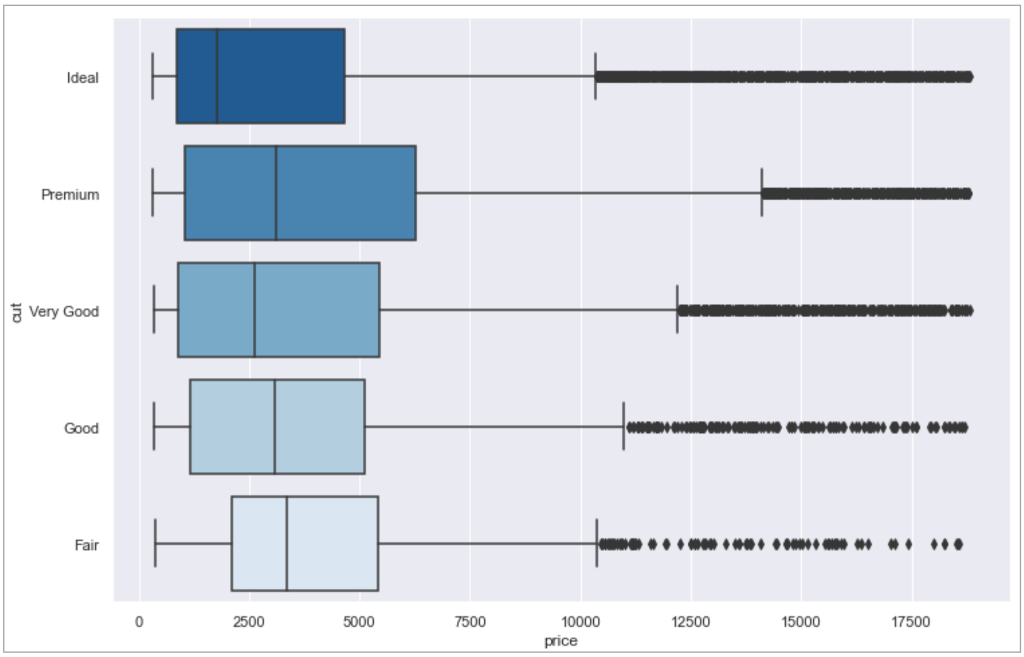


Figure 11. Boxplot of Cut with price variable.

- For the cut variable we see the most sold zirconia stone is 'Ideal' cut type gems and least sold is Fair cut gems
- All cut type gems have outliers with respect to price.
- Slightly less priced seems to be Ideal type and premium cut type to be slightly more expensive

Color with Price:

Statistical description of Color variable with respective Price:

color	count	mean	std	min	25%	50%	75%	max
D	3341.0	3184.827597	3419.875831	357.0	910.0	1799.0	4265.00	18526.0
E	4916.0	3073.940399	3397.600817	326.0	882.0	1698.0	3892.75	18731.0
F	4723.0	3699.944527	3807.933672	357.0	947.5	2281.0	4862.00	18791.0
G	5653.0	4005.046170	4057.515127	361.0	932.0	2274.0	6097.00	18818.0
H	4095.0	4477.932112	4249.859962	337.0	990.5	3398.0	5950.50	18795.0
I	2765.0	5124.816637	4728.462914	336.0	1145.0	3733.0	7292.00	18795.0
J	1440.0	5329.706250	4488.011962	335.0	1843.0	4234.5	7592.00	18701.0

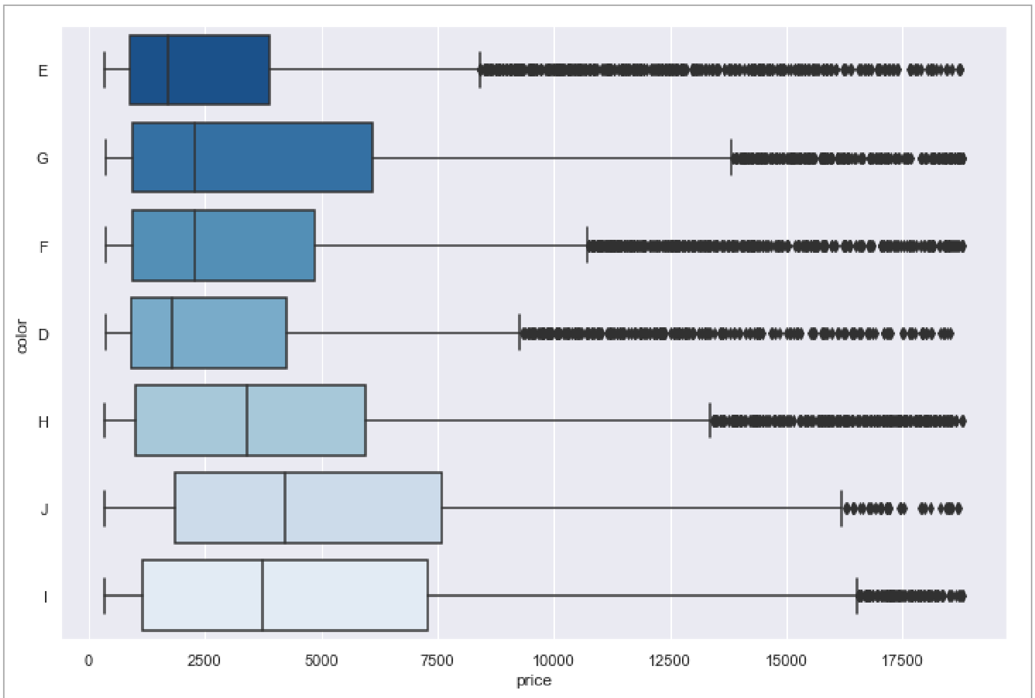


Figure 12. Boxplot of Color with Price variable.

- For the color variable we see the most sold is G colored gems and least is J colored gems
- All color type gems have outliers with respect to price
- However, the least priced seems to be E type; J and I colored gems seems to be more expensive

Clarity with Price:

Statistical description of Clarity variable with respective Price:

clarity	count	mean	std	min	25%	50%	75%	max
I1	364.0	3908.750000	2783.353422	345.0	2077.00	3471.5	5003.00	18531.0
IF	891.0	2739.534231	3738.032592	369.0	891.00	1063.0	2291.00	18552.0
SI1	6565.0	3998.635644	3829.728686	326.0	1090.00	2797.0	5266.00	18818.0
SI2	4564.0	5088.869413	4287.309747	326.0	2272.50	4077.0	5829.00	18804.0
VS1	4087.0	3838.752386	4051.412698	338.0	877.00	1949.0	6123.50	18795.0
VS2	6093.0	3965.496964	4118.691706	357.0	876.00	2066.0	6072.00	18791.0
VVS1	1839.0	2502.874388	3344.705599	336.0	814.00	1066.0	2217.50	18445.0
VVS2	2530.0	3263.042688	3829.353531	336.0	791.75	1253.0	3583.75	18718.0

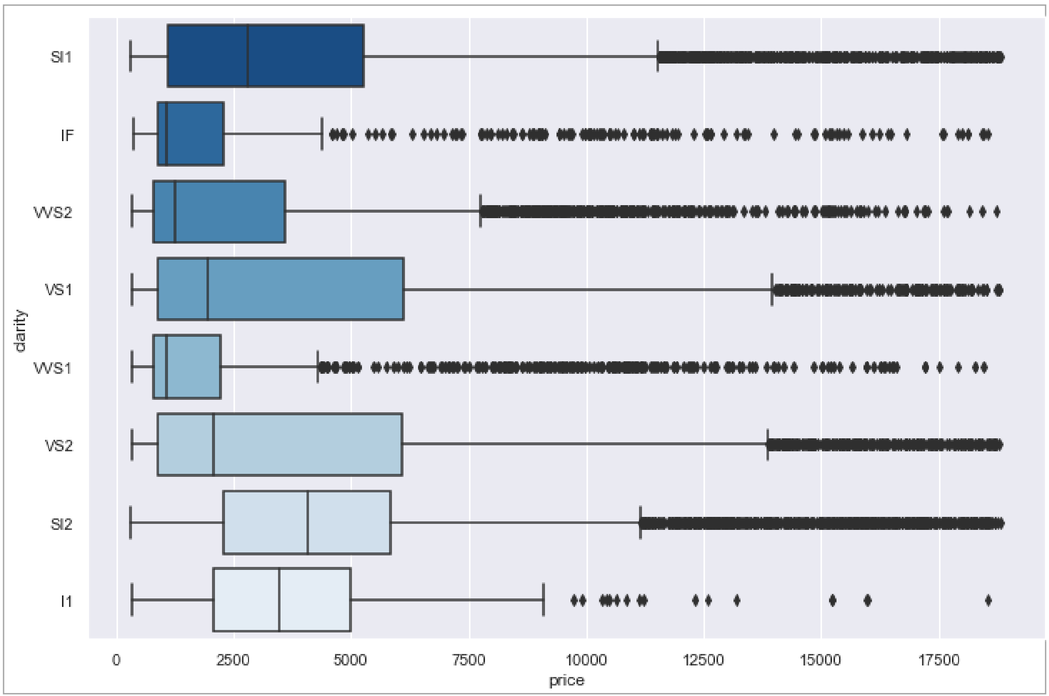


Figure 13. Boxplot of Clarity with Price

- For the clarity variable we see the most sold is SI1 clarity gems and least is I1 clarity gems
- All clarity type gems have outliers with respect to price
- Slightly less priced seems to be SI1 type; VS2 and SI2 clarity stones seems to be more expensive.

Count plot of Categorical variables with Target variable Price:

- 'Ideal' is the most selling cut type of zirconia stone and 'Fair' type being the least sold.
- We see that 'G' color is the most selling zirconia stone followed by 'E' and 'F' nearly following in same range and 'J' color gem is the least selling stone.
- S1 type of Clarity is most selling followed by VS2 and I1 being the least selling one.

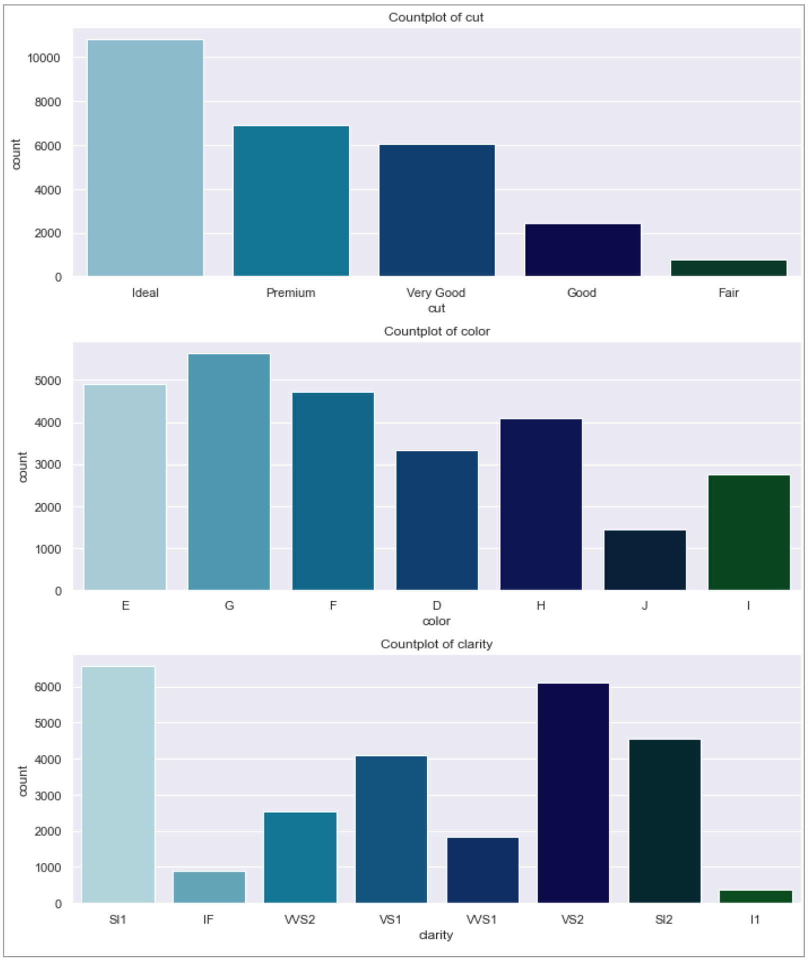


Figure 14. Count plot of Categorical variables with price.

Bar plot Categorical variables with Price:

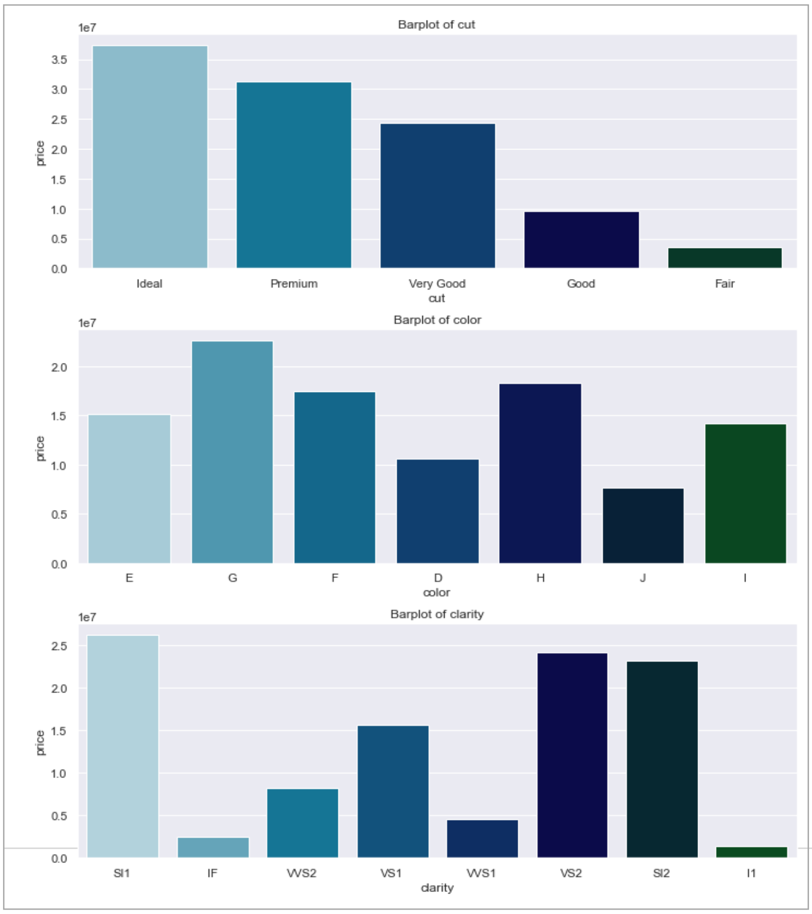


Figure 15. Bar plot of Categorical variables with Price.

- The price of ‘Ideal’ type cut is the most expensive and Fair is cheap one compared to all.
- G color gem is the costly one and also most liked by the people and are highest sold.
- J color gem price is less and also the least sold one
- S1 is the expensive one followed by the VS2 and S2 clarity which fall in the same price range and l1 and lF are the cheap gems.

Pair plot:

A pair plot gives us correlation graphs between all numerical variables in the dataset. Thus, from the graphs we can identify the relationships between all numerical variables.

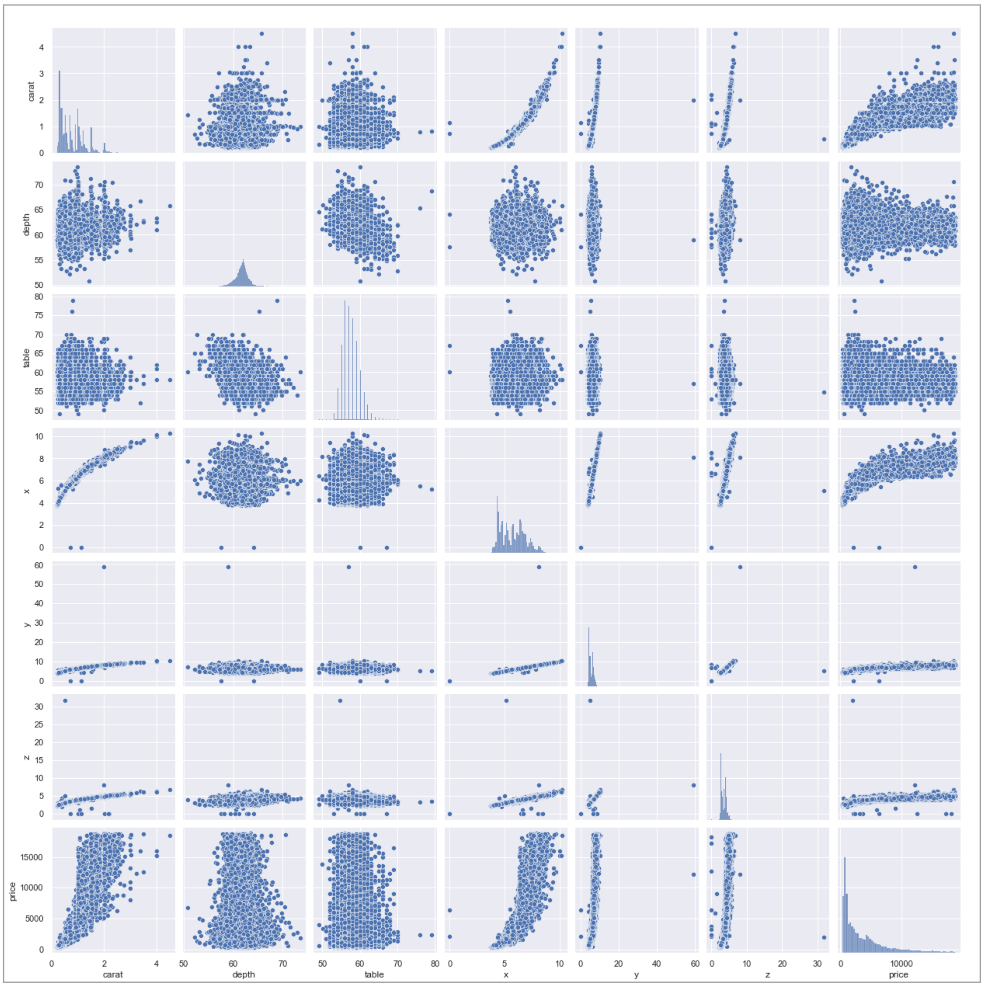
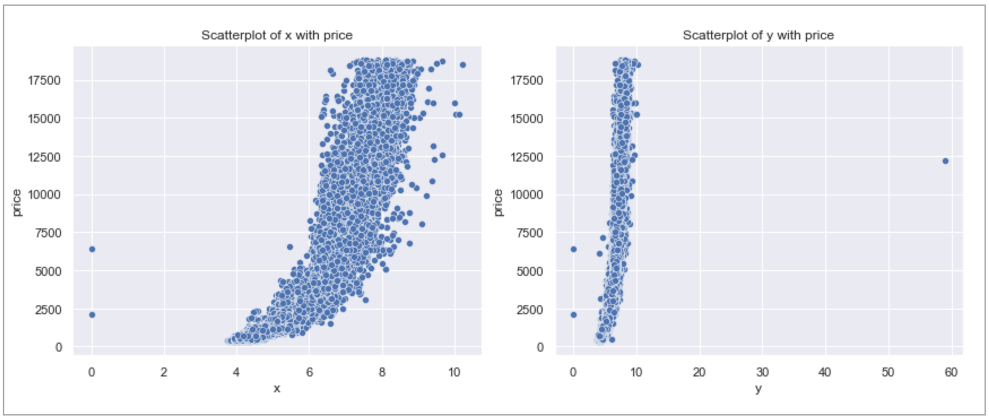


Figure 16. Pair plot of Zirconia dataset



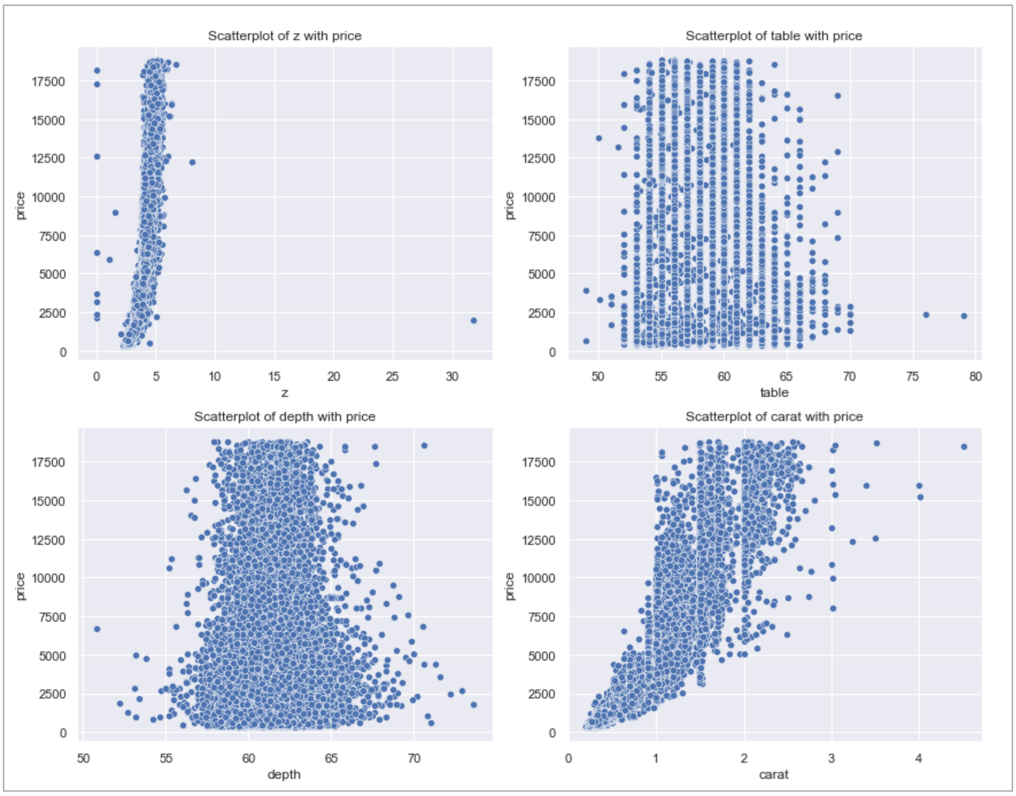


Figure 17. Scatter plots of all numeric variable with price.

Observations:

- From the above pair plot, we can see that 'Carat' and 'Price' are linearly correlated, which means the attribute carat influences the price of zirconia stone the most.
- We can see that X, Y and Z are having the linear relation with each other and also the target variable 'price'
- According to the assumptions for Linear regression model, the independent variables should not be linearly corelated with each other which leads to the high multicollinearity between the independent variables X, Y and Z which is length, width and height respectively.

Multivariate Analysis:

Heat map

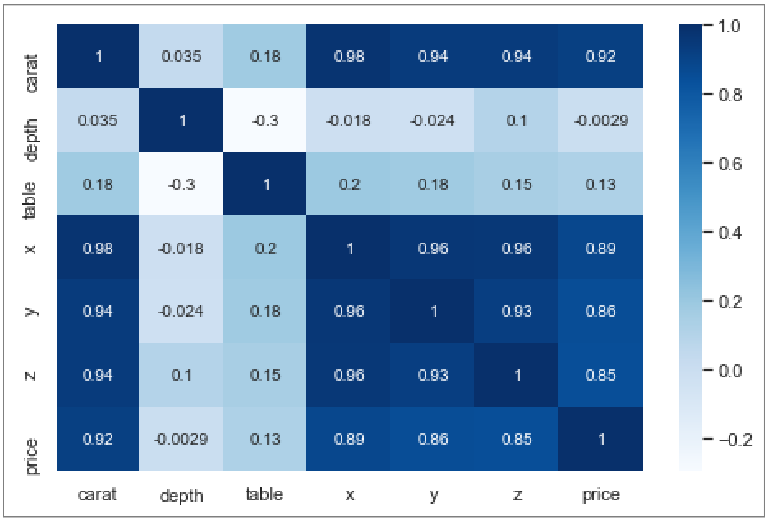


Figure 18. Heatmap for Zirconia dataset.

A heatmap gives us the correlation between numerical variables. If the correlation value is tending to 1, the variables are highly positively correlated whereas if the correlation value is close to 0, the variables are not correlated. Also, if the value is negative, the correlation is negative. That means, higher the value of one variable, the lower is the value of another variable and vice-versa.

Observations:

- Carat is highly correlated with price. Carat attribute is the best predictor of price.
- Depth is not related with price, so it depth attribute does not play major role in prediction of price.
- X (Length), Y (Width) and Z (Height) are highly correlated with price.
- X (Length), Y (Width) and Z (Height) are highly correlated with each other and are responsible for high multicollinearity.
- Multicollinearity is a setback for the linear regression model. The highly correlated values can be dropped in one of the model buildings and check the model performance.

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

Imputing Null values:

Following table shows the total number of missing values for all the variables.

```
zirconia.isnull().sum()

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

```
zirconia.isnull().sum()

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

We can see that there are 697 missing values in the depth variable, The **missing values are imputed by the median values** of the variable. The above table shows the total number of missing values before and after imputation.

Checking the values which are equal to zero:

As we saw in the Describe function earlier that ‘x’, ‘y’ and ‘z’ attributes have 0 values which implies that either the length, width or height of the stone is 0. This is practically not possible and this must be some kind of manual error.

Checking the data points where we have 0 value for dimensions:

	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

We can see that, there are 8 observations with values as 0. Since, the number of observations are very less in number compared to the total number of observations that is 26967, so dropping these won't affect much.

Checking duplicate data points:

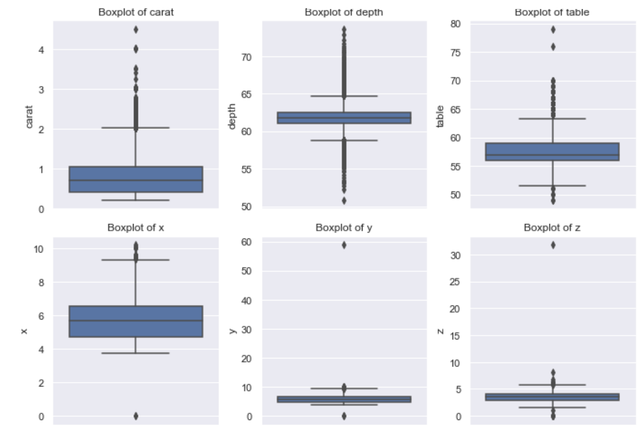
```
# Are there any duplicates?
dups = zirconia.duplicated()
print('Number of duplicate rows = %d' % (dups.sum()))
zirconia[dups]
```

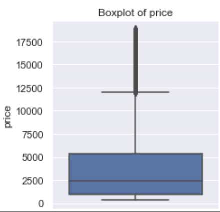
```
Number of duplicate rows = 34
```

We can see that, the total number of duplicates are 34 values, dropping the duplicates since the values are very less compared to size of dataset. **The total number of data points after dropping the duplicates and the values which are equal to zero are 26925.**

Outlier Treatment:

Figure 19. Boxplot before outlier treatment.





Check the outliers of all variables by plotting the box plot. According the assumptions, the outlier's impact on the model building, so the outliers are treated and the boxplots of all variables are plotted to check the treatment.

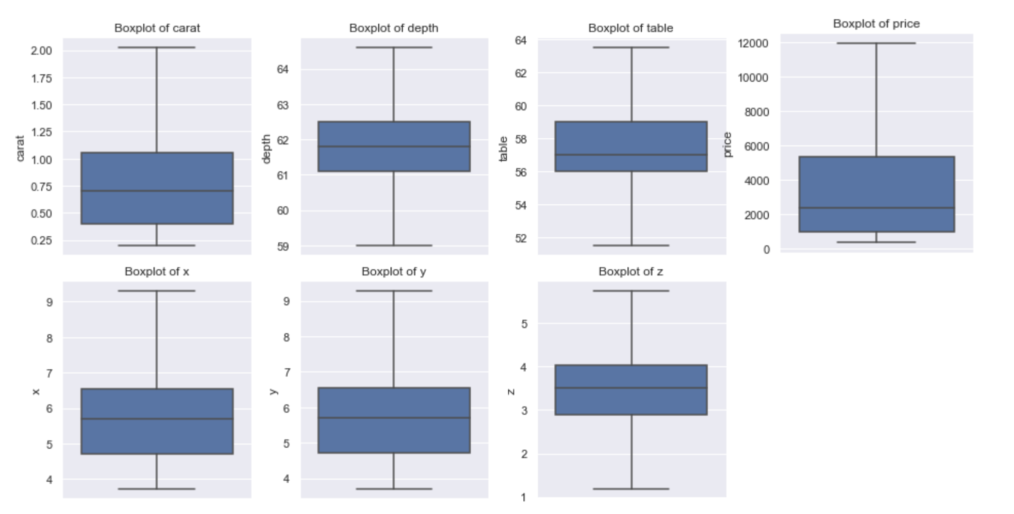


Figure 20. Boxplot after Outlier Treatment.

Checking for the possibility of combining the sub levels of Categorical variables attribute, an ordinal variable and take actions accordingly:

There are 3 different categorical variables. ‘Cut’, ‘Color’ and ‘Clarity’.

1. Checking the possibility of combining Cut variable sub-categories:

The variable ‘cut’ describes quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal. Checking the brief summary of cut attribute across different categories with respect to ‘Price’, which is our target variable.

cut	count	mean	std	min	25%	50%	75%	max
Fair	780.0	4568.096154	3745.800173	369.0	2117.0	3342.5	5430.0	18574.0
Good	2435.0	3926.336756	3621.197004	335.0	1157.0	3087.0	5111.5	18707.0
Ideal	10805.0	3454.820639	3869.198651	326.0	872.0	1762.0	4668.0	18804.0
Premium	6886.0	4544.558525	4320.888420	326.0	1038.5	3116.5	6268.5	18795.0
Very Good	6027.0	4032.267961	4016.865952	336.0	910.0	2633.0	5438.0	18818.0

- There are 5 sub categories in the ‘Cut’ variable.
- From above summary we can see that the mean and median price of ‘Good’ and ‘Very Good’ are close to each other.
- The stones of these 2 categories have similar description with respect to price.
- Combining these two sun categories ‘Good’ and ‘Very Good’ and naming it as ‘Good’.
- **Final sub-categories of ‘Cut’ are ‘Fair’, ‘Good’, ‘Premium’ and ‘Ideal’.**

2. Color refers to the color of the stone.

Although we can see a lot of possibilities of grouping this field. But we will choose to ignore it. No grouping is done of color attribute sub-category. This is because colors are different and cannot be grouped.

3. Checking the possibility of combining Clarity variable sub-categories:

'Clarity' is the absence of the inclusions and blemishes. Summary of clarity attribute with respect to price is as below:

clarity	count	mean	std	min	25%	50%	75%	max
I1	364.0	3908.750000	2783.353422	345.0	2077.00	3471.5	5003.00	18531.0
IF	891.0	2739.534231	3738.032592	369.0	891.00	1063.0	2291.00	18552.0
SI1	6565.0	3998.635644	3829.728686	326.0	1090.00	2797.0	5266.00	18818.0
SI2	4564.0	5088.869413	4287.309747	326.0	2272.50	4077.0	5829.00	18804.0
VS1	4087.0	3838.752386	4051.412698	338.0	877.00	1949.0	6123.50	18795.0
VS2	6093.0	3965.496964	4118.691706	357.0	876.00	2066.0	6072.00	18791.0
VVS1	1839.0	2502.874388	3344.705599	336.0	814.00	1066.0	2217.50	18445.0
VVS2	2530.0	3263.042688	3829.353531	336.0	791.75	1253.0	3583.75	18718.0

- The sub-categories VS1 and VS2's mean and median prices are very close. The both stones prices lie in similar range. Let the category be called as VS.
- The next categories which can be grouped are VVS1 and VVS2. The mean and median of price range is little different but still close enough to be grouped. Let the category be called as VVS.
- The third grouping involves SI1 and SI2. The price range of these categories is almost same. Given that SI1 has a larger number of stones but a lower mean price, and SI2 has a lower number of stones but a higher mean price, we may conclude that the two are balanced and can be grouped together. Let the category be called as SI
- **Final categories of Clarity variable are I1, SI, VS, VVS and IF.**

The groping of sub-categories of the above variables are considered in the new copy of dataset. The model is built based on this dataset and the model performance is checked based on non-grouped and other models as well.

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 stats model. Create multiple models and check the performance of Predictions on Train and Test sets using Rsquare, RMSE & Adj-Rsquare. Compare these models and select the best one with appropriate reasoning.

Encoding the categorical variables:

The given dataset categorical variables are having the defined ordinal sub-categories, so Ordinal encoding is appropriate and best suitable for the model building. Mapping the sub-categories of variables from 1 to n as mentioned in the data dictionary. An ordinal encoding involves mapping each unique label to an integer value. This type of encoding is really only appropriate, in this situation where the relationship or order is already known between the categories, which is clearly mentioned in Data dictionary.

Encoding/ Mapping Cut variable:

'Cut' variable describes the quality of the stone. According to data dictionary, the quality is increasing in order from Fair, Good, Very Good, Premium, Ideal. Mapping the numbers such the 1 being the best and 5 as the cheap quality cut.

- **Ideal: 1**
- **Premium: 2**
- **Very Good: 3**
- **Good: 4**
- **Fair: 5**

Encoding/ Mapping Clarity variable:

'Clarity' is the absence of the inclusions and blemishes. The order is given from Worst to Best in terms of average price it is IF, VVS1, VVS2, VS1, VS2, Sl1, Sl2, l1. That is l1 being the best clarity stone and IF being the worst. The mapping is done such that 1 being the best clarity and 8 being the worst.

- **I1: 1**
- **Sl2: 2**
- **Sl1: 3**
- **VS2: 4**
- **VS1: 5**
- **VVS2: 6**
- **VVS1: 7**
- **IF: 8**

Encoding/ Mapping Color variable:

'Color' refers to the color of the stone. With D being the worst and J the best. The mapping is done such that 1 being the best color and 7 being the worst.

- **J: 1**
- **I: 2**
- **H: 3**
- **G: 4**
- **F: 5**
- **E: 6**
- **D: 7**

Checking the head of Dataset after encoding the Categorical variables

	carat	depth	table	x	y	z	price	cut_c	color_c	clarity_c
0	0.30	62.1	58.0	4.27	4.29	2.66	499.0	1	6	3
1	0.33	60.8	58.0	4.42	4.46	2.70	984.0	2	4	8
2	0.90	62.2	60.0	6.04	6.12	3.78	6289.0	3	6	6
3	0.42	61.6	56.0	4.82	4.80	2.96	1082.0	1	5	5
4	0.31	60.4	59.0	4.35	4.43	2.65	779.0	1	5	7

Now the dataset is cleaned, encoded and ready to use for model building.

Linear Regression model:

Linear Regression is the **supervised Machine Learning model** in which the model finds the best fit linear line between the independent and dependent variable i.e., it finds the linear relationship between the dependent and independent variable. A Linear Regression model's main aim is to find the best fit linear line and the optimal values of intercept and coefficients such that the error is minimized.

Multiple Linear Regression models are built and check their model performance metrics. In the end, the models are compared and best fit model is selected. The selected model will be used to create the final equation

y (price) = m0 + m1 * carat + m2 * depth + m3 * table + m4 * X + m5 * Y + m6 * Z + m7 * cut_c + m8 * color_c + m9 * clarity_c.

The objective is building different models and make predictions of price slots and check the performance of each model using different performance matrices. Finally, comparing all the models and select the best one with appropriate reasoning. The data is analysed and following models are built with appropriate reasoning.

Model 1: Considering all the variables as it is and fitting the linear regression model.

Model 2: Dropping the attributes 'x', 'y' & 'z' and fitting the linear regression model.

Model 3: Dropping the attributes 'x', 'y', 'z' & 'depth' and fitting the linear regression model.

Model 4: Dropping the attributes 'x', 'y','z', 'depth' and grouping sub categories of attributes and fitting the linear regression model.

Model 5: Dropping the attributes 'x', 'y', 'z' & 'depth' and fitting the linear regression model for scaled data.

Model 1: Considering all the variables as it is and fitting the linear regression model.

In this model, all the attributes are considered as it is and the dataset is not scaled since the accuracy and model performance does not get influence by scaling the dataset.

Linear Regression Model - Sklearn

1. Capturing the target column into separate vectors for training set and test set.
2. X variable with independent attributes and y variable with the target variable which is 'price' in our case.
3. Splitting the dataset in to train and test in the ratio of 70:30 using train test split from sklearn, keeping the random state as 1.
4. Checking the shape of the split data.
```
X_train_mod1 (18847, 9)
X_test_mod1 (8078, 9)
y_train_mod1 (18847, 1)
y_test_mod1 (8078, 1)
```
5. Fitting the Linear regression model from sklearn linear models to Training set.
6. Finding the coefficient of determinants for each of independent attributes.

- The coefficient for carat is 8887.182245900442
- The coefficient for depth is 35.446432597917344
- The coefficient for table is -15.069203823159084
- The coefficient for x is -1348.7213850676303
- The coefficient for y is 1561.8443409182516
- The coefficient for z is -970.5030385552958

- The coefficient for cut_c is -113.33064005373288
- The coefficient for color_c is 273.22599181271306
- The coefficient for clarity_c is 436.8984753150906

  - From the above following coefficient of determinants of all independent attributes we can infer that **'Carat' variable has the most weightage and acts as the best predictor for price.**
  - We can see that; on other hand the 'Depth' and 'Table' variable do not have that much weightage in the prediction.
  - The coefficient of determination is a measurement used to explain how much variability of one factor can be caused by its relationship to another related factor
  - For example, unit change in the value of carat will bring 8887.18 change in price.

7. The intercept for our model is -5164.440069032453
8. Model performance of regression model built is calculated by the coefficient of determinant (R square). R square determines the fitness of a linear model. R square value ranges from 0 to 1. The closer the data point is to the best fit plane; the coefficient of determinant value tends to 1 and the better the model.
   - **R square of training data is 0.93122**
   - **R square of testing data is 0.93162**

   We can see that the score of Train and Test is almost similar, **the model is a good fit model.**

9. Calculating root mean square error (RMSE) value for checking model performance i.e. RMSE value is standard deviation of the prediction errors (residual). Residual errors or sum of squared errors are the measure of how far the data point is from the best fit plane. So basically, RMSE tells the spread out of these residuals. That means lower the RMSE is, closer are the data points to the best fit plane.
   - **RMSE of training data is 906.899**
   - **RMSE of testing data is 911.29**
10. Checking the plot between the original price and the predicted price for linear relationship.

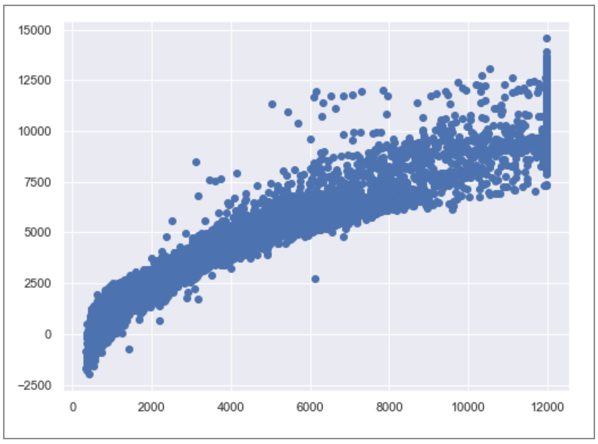


Figure 21. Scatter plot for model 1.

The model performance is limited for Linear regression model using sci-kit learn library have limited performance parameters to measure. Therefore, we perform Linear Regression by using a statsmodel.

Linear Regression Model- statsmodels

Statsmodel uses OLS (ordinary least square method) to predict the best fit plane. OLS also minimizes the sum of squared differences between the observed and predicted values by estimating coefficients and bias.

The difference between sci-kit Learn Linear Regression and statsmodel Linear Regression is that the stat model gives a more detailed summary of the model. Statsmodel also provides with Adjusted R square values and probabilities to check if the model is reliable or not. It also provides the probabilities of all variables depicting if their coefficients are reliable or not. Statsmodel is a good statistical analysis of the model and get information on which attributes we can drop and which we can keep and better compared to sklearn.

Adjusted R-square metric accounts for the spurious correlations. The above analysis infers that there is Multicollinearity in some extent

In OLS model, to establish the reliability of the coefficients, we need hypothesis testing. **The null hypothesis (H0) claims that there is no relation between dependent and independent variables.**

At 95% confidence level if the p value is > = 0.5, we do not have enough evidence to reject H0. Therefore, no relation between dependent and independent variable.

Similarly, if p value is < 0.5, we reject null hypothesis. Therefore, there is relationship between dependent and independent variables.

Checking the OLS summary for the model:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.940
Model:                            OLS   Adj. R-squared:                  0.940
Method:                 Least Squares   F-statistic:                 1.293e+04
Date:                Thu, 30 Dec 2021   Prob (F-statistic):               0.00
Time:                        23:22:34   Log-Likelihood:            -1.5373e+05
No. Observations:               18847   AIC:                         3.075e+05
Df Residuals:                   18823   BIC:                         3.077e+05
Df Model:                          23
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept      -4325.0165    742.591     -5.824      0.000   -5780.562   -2869.471
cut_c[T.2]       -31.2146     19.185     -1.627      0.104     -68.819       6.389
cut_c[T.3]      -127.5366     18.360     -6.946      0.000    -163.525     -91.549
cut_c[T.4]      -242.7045     26.138     -9.285      0.000    -293.938    -191.471
cut_c[T.5]      -629.9503     42.846    -14.703      0.000    -713.932    -545.969
color_c[T.2]     531.4839     32.902     16.154      0.000     466.993     595.975
color_c[T.3]    1030.0967     31.186     33.030      0.000     968.969    1091.225
color_c[T.4]    1450.5361     30.361     47.777      0.000    1391.026    1510.046
color_c[T.5]    1630.3862     31.029     52.543      0.000    1569.566    1691.207
color_c[T.6]    1672.7471     31.120     53.751      0.000    1611.749    1733.745
color_c[T.7]    1861.6261     32.764     56.819      0.000    1797.406    1925.847
clarity_c[T.2]  1712.1434     55.883     30.638      0.000    1602.609    1821.678
clarity_c[T.3]  2535.8724     55.574     45.630      0.000    2426.942    2644.803
clarity_c[T.4]  3072.1311     55.924     54.934      0.000    2962.514    3181.748
clarity_c[T.5]  3355.0983     56.782     59.087      0.000    3243.800    3466.397
clarity_c[T.6]  3766.7712     58.502     64.387      0.000    3652.102    3881.441
clarity_c[T.7]  3776.8836     60.182     62.758      0.000    3658.921    3894.846
clarity_c[T.8]  3995.2161     64.905     61.555      0.000    3867.997    4122.435
carat           9200.1934     77.389    118.883      0.000    9048.505    9351.882
depth             12.5864     10.463      1.203      0.229      -7.922      33.095
table            -23.0697      3.834     -6.018      0.000     -30.584     -15.555
x              -1176.9474    136.485     -8.623      0.000   -1444.471    -909.424
y               1083.2014    138.148      7.841      0.000     812.419    1353.984
z               -642.4803    131.066     -4.902      0.000    -899.381    -385.580
==============================================================================
Omnibus:                     4642.635   Durbin-Watson:                   2.002
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17342.056
Skew:                           1.197   Prob(JB):                         0.00
Kurtosis:                       7.043   Cond. No.                     1.03e+04
==============================================================================
```

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.03e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

From the above summary we can infer that:

- **R- squared and Adjusted R-squared** vales are same which is equal to **0.940**
- Overall p value of model is 0.00 (< 0.05) which means model is reliable.
- Considering individual variable and its p-value, we can see cut_c [T.2] and depth have values greater than 0.05. Therefore, these variables are not good predictors of price and can be dropped to get a better performing model.
- The condition number is large which means it indicates the presence of multicollinearity in dataset, which was clearly seen in the above analysis between X, Y and Z variables.
- **The RMSE score for our OLS model is 844.29**
- **Checking Multi-collinearity using VIF**
  We test for multicollinearity with Variation Inflation Factor (VIF). VIF identifies correlation between independent variables and the strength of that correlation. VIF starts from 1 and has no upper value.

  VIF equal to 1 indicates no correlation between independent variables.
  VIF between 1 to 5 indicates moderate correlation but not severe.
  VIF greater than 5 indicates critical levels of multicollinearity.

  - **carat** ---> 122.65490394147022
  - **depth** ---> 1126.3143618911165
  - **table** ---> 892.2124758097101
  - **x** ---> 10638.27854893691
  - **y** ---> 9419.13075753421
  - **z** ---> 3226.9583455469033
  - **cut_c** ---> 6.138962724380723
  - **color_c** ---> 8.53348426777295
  - **clarity_c** ---> 8.66162674295014

- High levels of Multicollinearity are present in data. This model is not reliable based on the high multicollinearity. Making changes in data and dropping highly correlated variables may overcome the problem of Multicollinearity.
- **Linear Equation:**

```
(-4325.02) * Intercept + (-31.21) * cut_c[T.2] + (-127.54) * cut_
c[T.3] + (-242.7) * cut_c[T.4] + (-629.95) * cut_c[T.5] + (531.48
) * color_c[T.2] + (1030.1) * color_c[T.3] + (1450.54) * color_c[
T.4] + (1630.39) * color_c[T.5] + (1672.75) * color_c[T.6] + (186
1.63) * color_c[T.7] + (1712.14) * clarity_c[T.2] + (2535.87) * c
larity_c[T.3] + (3072.13) * clarity_c[T.4] + (3355.1) * clarity_c
[T.5] + (3766.77) * clarity_c[T.6] + (3776.88) * clarity_c[T.7] +
(3995.22) * clarity_c[T.8] + (9200.19) * carat + (12.59) * depth
+ (-23.07) * table + (-1176.95) * x + (1083.2) * y + (-642.48)*z
```

- **Inferences:** Carat with a coefficient of 9200.19 is the best predictor of price. For 1 unit change in ‘carat’, the price will change by 9200.19 units keeping all other variables 0.
  Based on all the analysis of model, we see that, this is not the best model for predicting price slots for zirconia stones. But from this model, we came to know what are the changes to be done to create a best fit model.

Model 2 - Dropping the attributes 'x', 'y' & 'z' and fitting the linear regression model

In this model, considering all the variables except ‘x’, ‘y’ and ‘z’ and also the data is not scaled. As we saw in model 1 analysis that ‘x’, ‘y’ and ‘z’ contributes in high multicollinearity. The VIF scores was large indicating for the cause of high multicollinearity, so building a model by drooping the x, y and z variables and checking the model performance and compare.

Linear Regression Model - Sklearn

1. Capturing the target column into separate vectors for training set and test set.
2. X variable with independent attributes where x, y & z variables are not considered and y variable with the target variable which is ‘price’ in our case.
3. Splitting the dataset in to train and test in the ratio of 70:30 using train test split from sklearn, keeping the random state as 1.
4. Checking the shape of the split data.
5.
```
X_train_mod1 (18847, 6)
X_test_mod1 (8078, 6)
y_train_mod1 (18847, 1)
y_test_mod1 (8078, 1)
```
6. Fitting the Linear regression model from sklearn linear models to Training set.
7. Finding the coefficient of determinants for each of independent attributes.

    - The coefficient for carat is 7957.233009701646
    - The coefficient for depth is -17.60091599461744
    - The coefficient for table is -20.090752371797244
    - The coefficient for cut_c is -105.77168383079943
    - The coefficient for color_c is 271.88660130178056
    - The coefficient for clarity_c is 450.3790478005037

8. From the above following coefficient of determinants of all independent attributes we can infer that ‘Carat’ variable even in this model has the most weightage and acts as the best predictor for price.
9. We can see that; on other hand the ‘Depth’ and ‘Table’ variable do not have that much weightage in the prediction. There are changes in coefficient of determinant values from model 1 to model 2. Model 2 is performing better compared to model 1.
10. The coefficient of determination is a measurement used to explain how much variability of one factor can be caused by its relationship to another related factor
11. For example, unit change in the value of carat will bring 7957.23 change in price.
12. The intercept for our model is -3136.18073. The absolute value of intercept is lower than Model 1 but it is more than Model 2.
13. Model performance of regression model built is calculated by the coefficient of determinant (R square). R square determines the fitness of a linear model. R square value ranges from 0 to 1. The closer the data point is to the best fit plane; the coefficient of determinant value tends to 1 and the better the model.

    R square of training data is 0.93016

    R square of testing data is 0.93055

We can see that the score of Train and Test is almost similar, **the model is a good fit model.**

Calculating root mean square error (RMSE) value for checking model performance i.e. RMSE value is standard deviation of the prediction errors (residual). Residual errors or sum of squared errors are the measure of how far the data point is from the best fit plane. So basically, RMSE tells the spread out of these residuals. That means lower the RMSE is, closer are the data points to the best fit plane.

RMSE of training data is 913.878

RMSE of testing data is 918.433

The RMSE values have increased a bit as compared to our previous model.

14. Checking the plot between the original price and the predicted price for linear relationship.

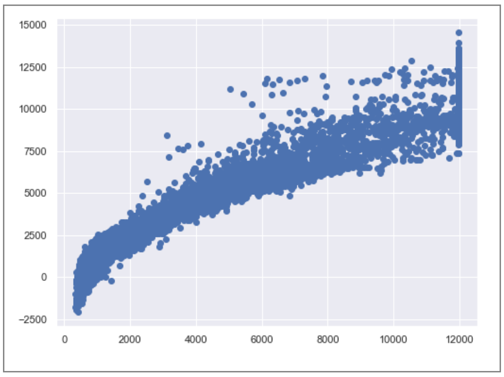


Figure 22. Scatter plot for model 2.

Checking the OLS summary for the model:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.939
Model:                            OLS   Adj. R-squared:                  0.939
Method:                 Least Squares   F-statistic:                 1.461e+04
Date:                Thu, 30 Dec 2021   Prob (F-statistic):               0.00
Time:                        23:22:35   Log-Likelihood:            -1.5389e+05
No. Observations:               18847   AIC:                         3.078e+05
Df Residuals:                   18826   BIC:                         3.080e+05
Df Model:                          20
Covariance Type:            nonrobust
====================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
Intercept       -4823.8371    497.883     -9.689      0.000   -5799.733   -3847.941
cut_c[T.2]        -67.1698     18.701     -3.592      0.000    -103.825     -30.515
cut_c[T.3]       -100.0986     18.241     -5.487      0.000    -135.853     -64.344
cut_c[T.4]       -232.5789     26.219     -8.871      0.000    -283.971    -181.187
cut_c[T.5]       -706.7314     42.370    -16.680      0.000    -789.781    -623.682
color_c[T.2]      529.3658     33.171     15.959      0.000     464.348     594.384
color_c[T.3]     1015.5930     31.431     32.311      0.000     953.985    1077.201
color_c[T.4]     1423.1022     30.570     46.553      0.000    1363.183    1483.021
color_c[T.5]     1602.3689     31.239     51.293      0.000    1541.137    1663.601
color_c[T.6]     1655.9967     31.364     52.799      0.000    1594.520    1717.473
color_c[T.7]     1844.9583     33.021     55.873      0.000    1780.235    1909.682
clarity_c[T.2]   1747.3177     56.212     31.084      0.000    1637.136    1857.499
clarity_c[T.3]   2575.5267     55.858     46.108      0.000    2466.039    2685.014
clarity_c[T.4]   3131.4981     56.189     55.731      0.000    3021.362    3241.634
clarity_c[T.5]   3415.8589     57.049     59.875      0.000    3304.037    3527.681
clarity_c[T.6]   3853.6884     58.700     65.650      0.000    3738.630    3968.746
clarity_c[T.7]   3882.4037     60.342     64.340      0.000    3764.128    4000.679
clarity_c[T.8]   4106.8825     65.071     63.114      0.000    3979.338    4234.427
carat            8027.4643     15.500    517.908      0.000    7997.083    8057.845
depth             -10.4188      5.914     -1.762      0.078     -22.010       1.173
table             -22.9543      3.856     -5.953      0.000     -30.512     -15.397
==============================================================================
Omnibus:                     4140.504   Durbin-Watson:                   2.003
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            12713.680
Skew:                           1.131   Prob(JB):                         0.00
Kurtosis:                       6.328   Cond. No.                     6.79e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 6.79e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

From the above summary we can infer that:

- R- squared and Adjusted R-squared vales are same which is equal to 0.939
- Overall p value of model is 0.00 (< 0.05) which means model is reliable.
- Considering each variable and its p value, we can see cut_c [T.2] has become 0 in this model. But ‘depth’ has p value greater than 0.05. Keeping ‘depth’ variable in model is not necessary. Next model is built by dropping depth variable too.
- The condition number is large which means it indicates the presence of multicollinearity in dataset, which is because of other variables of dataset. Condition number is reduced from model 1, dropping depth variable may eliminated the problem of multicollinearity and improve the performance of model.
- **The RMSE score for our OLS model is 851.307.**
- **Checking Multi-collinearity using VIF for model 2.**

  We test for multicollinearity with Variation Inflation Factor (VIF). VIF identifies correlation between independent variables and the strength of that correlation. VIF starts from 1 and has no upper value.

  VIF equal to 1 indicates no correlation between independent variables.
  VIF between 1 to 5 indicates moderate correlation but not severe.
  VIF greater than 5 indicates critical levels of multicollinearity.

    - carat ---> 5.138632848517505
    - depth ---> 480.09395166004856
    - table ---> 500.29465902799984
    - cut_c ---> 5.165922749512226
    - color_c ---> 8.477234121974698
    - clarity_c ---> 8.33395177559938

  - VIF scores has decreased to a great extent. So, the problem of multicollinearity is getting treated to a very good extent by removing ‘x’, ‘y’ and ‘z’ variables.
  - **Linear Equation for model 2:**

```
(-4823.84) * Intercept + (-67.17) * cut_c[T.2] + (-100.1) * cut_
c[T.3] + (-232.58) * cut_c[T.4] + (-706.73) * cut_c[T.5] + (529.
37) * color_c[T.2] + (1015.59) * color_c[T.3] + (1423.1) * color
_c[T.4] + (1602.37) * color_c[T.5] + (1656.0) * color_c[T.6] + (
1844.96) * color_c[T.7] + (1747.32) * clarity_c[T.2] + (2575.53)
* clarity_c[T.3] + (3131.5) * clarity_c[T.4] + (3415.86) * clari
ty_c[T.5] + (3853.69) * clarity_c[T.6] + (3882.4) * clarity_c[T.
7] + (4106.88) * clarity_c[T.8] + (8027.46) * carat + (-10.42) *
depth + (-22.95) * table
```

  - **Inferences:** ‘Carat’ is still the best predictor with a coefficient of 8027.46. For 1 unit change in ‘carat’, the price will change by 8027.46 units keeping all other variables 0. From this model we can infer that, the strong multicollinearity which was due to x, y & z is reduced to greater extend. The remaining collinearity is due to the ‘depth’ variable. The depth variable can also be dropped since its not a good predictor for model building. The next model is built by dropping even the depth variable and compare its affect on the model performance and compare with the previous models and choose the best fit model for prediction of price slots for a company.

Model 3: Dropping the attributes 'x', 'y', 'z' & 'depth' and fitting the linear regression model.

In this model, considering all the variables except ‘x’, ‘y’, ‘z’ and ‘depth’ and the data is unscaled. As we saw in model 2 analysis that dropping ‘x’, ‘y’ and ‘z’ contributes in reducing the high multicollinearity in the model. In this model along with ‘x’, ‘y’ and ‘z’ dropping the ‘depth’ variable to which is not good predictor and also contributes for some amount of multicollinearity and enhancing the model for better performances.

Linear Regression Model – Sklearn

1. Capturing the target column into separate vectors for training set and test set.
2. X variable with independent attributes where x, y , z & depth variables are not considered and y variable with the target variable which is ‘price’ in our case.
3. Splitting the dataset in to train and test in the ratio of 70:30 using train test split from sklearn, keeping the random state as 1.
4. Checking the shape of the split data.
```
X_train_mod3 (18847, 5)
X_test_mod3 (8078, 5)
y_train_mod3 (18847, 1)
y_test_mod3 (8078, 1)
```
5. Fitting the Linear regression model from sklearn linear models to Training set.
6. Finding the coefficient of determinants for each of independent attributes.

   - The coefficient for carat is 7956.146166877565
   - The coefficient for table is -15.226520064985072
   - The coefficient for cut_c is -113.92419740534123
   - The coefficient for color_c is 272.479510071415
   - The coefficient for clarity_c is 451.1570717705239

7. No difference in coefficient of discriminant is seen compared to previous models.
8. From the above following coefficient of determinants of all independent attributes we can infer that **‘Carat’ variable has the most weightage and acts as the best predictor for price.** The number of predictors is less, yet the model is better comparatively.
9. The coefficient of determination is a measurement used to explain how much variability of one factor can be caused by its relationship to another related factor
10. **The intercept for our model is -4490.273.** The absolute value of intercept is more compared to other two models.
11. Model performance of regression model built is calculated by the coefficient of determinant (R square). R square determines the fitness of a linear model. R square value ranges from 0 to 1. The closer the data point is to the best fit plane; the coefficient of determinant value tends to 1 and the better the model.

    R square of training data is 0.93013

    R square of testing data is 0.930502

We can see that the score of Train and Test is almost similar, **the model is a good fit model.**

12. Calculating root mean square error (RMSE) value for checking model performance i.e. RMSE value is standard deviation of the prediction errors (residual). Residual errors or sum of squared errors are the measure of how far the data point is from the best fit plane. So basically, RMSE tells the spread out of these residuals. That means lower the RMSE is, closer are the data points to the best fit plane.

RMSE of training data is 914.069

RMSE of testing data is 918.750

No changes in RMSE scores compared to last model.

13. Checking the plot between the original price and the predicted price for linear relationship.

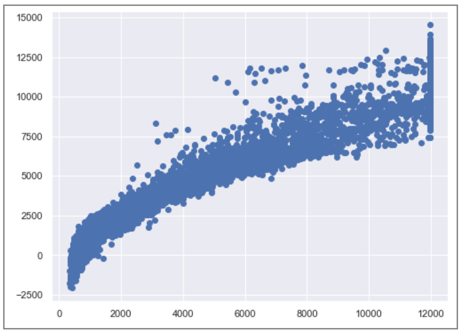


Figure 23. Scatter plot for model 3.

Checking the OLS summary for the model:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.939
Model:                            OLS   Adj. R-squared:                  0.939
Method:                 Least Squares   F-statistic:                 1.538e+04
Date:                Thu, 30 Dec 2021   Prob (F-statistic):               0.00
Time:                        23:22:36   Log-Likelihood:            -1.5389e+05
No. Observations:               18847   AIC:                         3.078e+05
Df Residuals:                   18827   BIC:                         3.080e+05
Df Model:                          19
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept      -5621.2524    207.433    -27.099      0.000   -6027.839   -5214.665
cut_c[T.2]       -70.0183     18.632     -3.758      0.000    -106.538     -33.499
cut_c[T.3]      -106.5924     17.866     -5.966      0.000    -141.611     -71.573
cut_c[T.4]      -247.6196     24.792     -9.988      0.000    -296.213    -199.026
cut_c[T.5]      -729.9098     40.278    -18.122      0.000    -808.859    -650.960
color_c[T.2]     529.2500     33.173     15.954      0.000     464.229     594.271
color_c[T.3]    1015.9841     31.432     32.323      0.000     954.374    1077.594
color_c[T.4]    1423.8188     30.569     46.578      0.000    1363.901    1483.736
color_c[T.5]    1603.8502     31.230     51.356      0.000    1542.637    1665.064
color_c[T.6]    1657.3889     31.356     52.857      0.000    1595.929    1718.849
color_c[T.7]    1846.7137     33.008     55.948      0.000    1782.016    1911.412
clarity_c[T.2]  1749.4844     56.202     31.128      0.000    1639.323    1859.646
clarity_c[T.3]  2576.9059     55.856     46.135      0.000    2467.423    2686.389
clarity_c[T.4]  3133.6350     56.179     55.779      0.000    3023.518    3243.752
clarity_c[T.5]  3418.9457     57.026     59.955      0.000    3307.170    3530.721
clarity_c[T.6]  3856.7233     58.678     65.726      0.000    3741.708    3971.738
clarity_c[T.7]  3885.5959     60.318     64.419      0.000    3767.367    4003.825
clarity_c[T.8]  4112.1575     65.006     63.258      0.000    3984.741    4239.574
carat           8026.7430     15.495    518.013      0.000    7996.371    8057.115
table            -20.2457      3.536     -5.726      0.000     -27.177     -13.315
==============================================================================
Omnibus:                     4137.461   Durbin-Watson:                   2.003
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            12707.437
Skew:                           1.130   Prob(JB):                         0.00
Kurtosis:                       6.328   Cond. No.                     1.99e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.99e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

From the above summary we can infer that:

- R- squared and Adjusted R-squared vales are same which is equal to 0.939
- Overall p value of model is 0.00 (< 0.05) which means model is reliable.
- Considering each variable and its p value, we can see no variable has p value more than 0.05. So, all variables in this model have good relationship with dependent variable. Considering all these variables in our model.
- Condition number is large which means there is string multicollinearity in the dataset or other numerical problems are present in dataset. But we can see that, condition number has reduced from 1.03e+04 in Model 1 to 1.99e+03 in model 3.
- **The RMSE score for our OLS model is 851.355.**
- **Checking Multi-collinearity using VIF for model 3.**

  We test for multicollinearity with Variation Inflation Factor (VIF). VIF identifies correlation between independent variables and the strength of that correlation. VIF starts from 1 and has no upper value.

  VIF equal to 1 indicates no correlation between independent variables.
  VIF between 1 to 5 indicates moderate correlation but not severe.
  VIF greater than 5 indicates critical levels of multicollinearity.

  - carat ---> 5.137344729116566
  - table ---> 34.373679445889046
  - cut_c ---> 5.036820239177246
  - color_c ---> 8.425690234941365
  - clarity_c ---> 8.165917118475038

- VIF scores for variables have reduced considerably.
- **Linear Equation for model 3:**

```
(-5621.25) * Intercept + (-70.02) * cut_c[T.2] + (-106.59) * cut_c
[T.3] + (-247.62) * cut_c[T.4] + (-729.91) * cut_c[T.5] + (529.25)
* color_c[T.2] + (1015.98) * color_c[T.3] + (1423.82) * color_c[T.
4] + (1603.85) * color_c[T.5] + (1657.39) * color_c[T.6] + (1846.7
1) * color_c[T.7] + (1749.48) * clarity_c[T.2] + (2576.91) * clari
ty_c[T.3] + (3133.64) * clarity_c[T.4] + (3418.95) * clarity_c[T.5
] + (3856.72) * clarity_c[T.6] + (3885.6) * clarity_c[T.7] + (4112
.16) * clarity_c[T.8] + (8026.74) * carat + (-20.25) * table
```

Inferences:

‘Carat’ is still the best predictor with a coefficient of 7956.14. For 1 unit change in ‘carat’, the price will change by 7956.14 units keeping all other variables 0.

VIF scores have reduced to almost 5 for most of the variables. RMSE has not shown any major change till in this model. Variables which are good predictors are understood through this model and all p values for all the variables are under 0.05. Condition number is also reduced considerably.

Now we see that our RMSE is high and coefficients are not balanced. So, we should bring the variables to a balanced state. We can achieve that by bringing all variables to a comparable form. We can achieve that by scaling the data. Performing the same on scaled model and comparing.

Model 4: Dropping the attributes 'x', 'y', 'z', 'depth' and grouping sub categories of attributes and fitting the linear regression model.

In this model considering the same attributes as the previous one and also grouping the sub categories of the clarity variable. The data frame which was copied in which the sub categories are grouped is used in the model building. To check if the performance while altering the data. It is compared with the original data performance and the suggestions for company can be given based on the results.

Linear Regression Model – Sklearn

1. Capturing the target column into separate vectors for training set and test set.
2. X variable with independent attributes where x, y , z, depth are dropped and grouping the sub categories of clarity variables and y variable with the target variable which is 'price' in our case.
3. Splitting the dataset in to train and test in the ratio of 70:30 using train test split from sklearn, keeping the random state as 1.
4. Checking the shape of the split data.

```
X_train_mod4 (18847, 5)
X_test_mod4 (8078, 5)
y_train_mod4 (18847, 1)
y_test_mod4 (8078, 1)
```

5. Fitting the Linear regression model from sklearn linear models to Training set.
6. Finding the coefficient of determinants for each of independent attributes.

    - The coefficient for carat is 7861.428694376872
    - The coefficient for table is -17.17005364009355
    - The coefficient for cut_c is -117.61394278203296
    - The coefficient for color_c is 260.6060552289523
    - The coefficient for clarity_c is 407.7462698772857

7. No difference in coefficient of discriminant is seen compared to previous models.
8. The coefficient of determination is a measurement used to explain how much variability of one factor can be caused by its relationship to another related factor
9. **The intercept for our model is -3878.169.**
10. Model performance of regression model built is calculated by the coefficient of determinant (R square). R square determines the fitness of a linear model. R square value ranges from 0 to 1. The closer the data point is to the best fit plane; the coefficient of determinant value tends to 1 and the better the model.

    R square of training data is 0.92477

    R square of testing data is 0.92506

We can see that the score of Train and Test is almost similar, **the model is a good fit model.**

11. Calculating root mean square error (RMSE) value for checking model performance i.e. RMSE value is standard deviation of the prediction errors (residual). Residual errors or sum of squared errors are the measure of how far the data point is from the best fit plane. So basically, RMSE tells the spread out of these residuals. That means lower the RMSE is, closer are the data points to the best fit plane.

    RMSE of training data is 948.47

    RMSE of testing data is 953.85

The RMSE is more than the previous models even after altering and combining the sub categories of the attributes is not working fine. Its better to keep the all the sub categories same as original data.

12. Checking the plot between the original price and the predicted price for linear relationship.

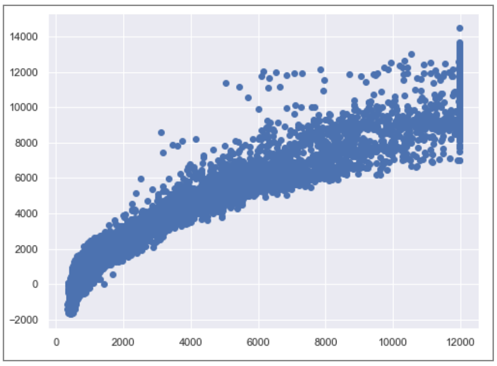


Figure 24. Scatter plot for model 4.

Checking the OLS summary for the model:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.933
Model:                            OLS   Adj. R-squared:                  0.933
Method:                 Least Squares   F-statistic:                 1.646e+04
Date:                Thu, 30 Dec 2021   Prob (F-statistic):               0.00
Time:                        23:22:37   Log-Likelihood:            -1.5481e+05
No. Observations:               18847   AIC:                         3.096e+05
Df Residuals:                   18830   BIC:                         3.098e+05
Df Model:                          16
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept      -5326.1738    217.636    -24.473      0.000   -5752.760   -4899.588
cut_c[T.2]       -75.2710     19.556     -3.849      0.000    -113.602     -36.940
cut_c[T.3]       -94.8981     18.749     -5.062      0.000    -131.648     -58.148
cut_c[T.4]      -242.0297     26.024     -9.300      0.000    -293.039    -191.020
cut_c[T.5]      -775.7239     42.266    -18.353      0.000    -858.569    -692.879
color_c[T.2]     515.2533     34.821     14.797      0.000     447.001     583.505
color_c[T.3]     973.0107     32.979     29.504      0.000     908.370    1037.652
color_c[T.4]    1379.2998     32.068     43.011      0.000    1316.443    1442.156
color_c[T.5]    1551.0458     32.757     47.351      0.000    1486.840    1615.252
color_c[T.6]    1591.2733     32.868     48.415      0.000    1526.850    1655.697
color_c[T.7]    1775.2859     34.587     51.328      0.000    1707.492    1843.079
clarity_c[T.2]  2202.4753     57.880     38.053      0.000    2089.026    2315.924
clarity_c[T.4]  3188.5949     58.349     54.647      0.000    3074.225    3302.965
clarity_c[T.6]  3788.5122     60.130     63.006      0.000    3670.652    3906.372
clarity_c[T.8]  4021.8144     68.203     58.969      0.000    3888.131    4155.498
carat           7916.8565     16.031    493.836      0.000    7885.434    7948.279
table            -22.1108      3.712     -5.957      0.000     -29.386     -14.836
==============================================================================
Omnibus:                     3808.402   Durbin-Watson:                   2.003
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            10790.422
Skew:                           1.069   Prob(JB):                         0.00
Kurtosis:                       6.028   Cond. No.                     1.96e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.96e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

From the above summary we can infer that:

- R- squared and Adjusted R-squared vales are same which is equal to 0.933. The R- square value is slightly decreased compared to previous models
- Overall p value of model is 0.00 (< 0.05) which means model is reliable.
- Considering each variable and its p value, we can see no variable has p value more than 0.05. So, all variables in this model have good relationship with dependent variable. Considering all these variables in our model.
- Condition number is large, +1.96e+03. This indicates that is some numerical problems since we have removed the variables contributing for the multicollinearity.
- **The RMSE score for our OLS model is 893.715**
- **Checking Multi-collinearity using VIF for model 4.**

  We test for multicollinearity with Variation Inflation Factor (VIF). VIF identifies correlation between independent variables and the strength of that correlation. VIF starts from 1 and has no upper value.

  VIF equal to 1 indicates no correlation between independent variables.
  VIF between 1 to 5 indicates moderate correlation but not severe.
  VIF greater than 5 indicates critical levels of multicollinearity.

  - carat ---> 5.014781525999582
  - table ---> 30.721836494082076
  - cut_c ---> 5.030393865852007
  - color_c ---> 8.39160381584299
  - clarity_c ---> 6.38478635319889

- VIF scores for variables have reduced considerably.
- **Linear Equation for model 3:**

```
(-5326.17) * Intercept + (-75.27) * cut_c[T.2] + (-94.9) * cut_c[T
.3] + (-242.03) * cut_c[T.4] + (-775.72) * cut_c[T.5] + (515.25) *
color_c[T.2] + (973.01) * color_c[T.3] + (1379.3) * color_c[T.4] +
(1551.05) * color_c[T.5] + (1591.27) * color_c[T.6] + (1775.29) *
color_c[T.7] + (2202.48) * clarity_c[T.2] + (3188.59) * clarity_c[
T.4] + (3788.51) * clarity_c[T.6] + (4021.81) * clarity_c[T.8] + (
7916.86) * carat + (-22.11) * table
```

Inferences:

'Carat' is still the best predictor with a coefficient of 7916.86. For 1 unit change in 'carat', the price will change by 7916.86 units keeping all other variables 0.

VIF scores have reduced to almost 5 for most of the variables. RMSE has not shown any major change till in this model. Variables which are good predictors are understood through this model and all p values for all the variables are under 0.05. Condition number is also reduced considerably.

The R squared values is decreased slightly from the original dataset and also there is fair amount of increase in RMSE values which indicates that combining sub categories of variables is not contributing for better performance. So, it is better to consider the previous model i.e model 3.

Model 5: Dropping the attributes 'x', 'y', 'z' & 'depth' and fitting the linear regression model for scaled data.

In all the above four models, model 3 is performing better compared all other model, but the data is not balanced, hence scaling the dataset using z-score, where mean is closer to 0 and standard deviation to 1. Checking the impact of scaling on the model and comparing model 3 and 5 and finally selecting the best model for prediction of price slots.

Linear Regression Model – Sklearn

1. Capturing the target column into separate vectors for training set and test set.
2. X variable with independent attributes where x, y , z & depth variables are not considered and y variable with the target variable which is 'price' in our case.
3. Splitting the dataset in to train and test in the ratio of 70:30 using train test split from sklearn, keeping the random state as 1.
4. Checking the shape of the split data.
```
X_train_mod5 (18847, 5)
X_test_mod5 (8078, 5)
y_train_mod5 (18847, 1)
y_test_mod5 (8078, 1)
```
5. Fitting the Linear regression model from sklearn linear models to Training set.
6. Finding the coefficient of determinants for each of independent attributes.

   - The coefficient for carat is 1.0580025633434904
   - The coefficient for table is -0.009494734169229253
   - The coefficient for cut_c is -0.03664855966671572
   - The coefficient for color_c is 0.13427496894739407
   - The coefficient for clarity_c is 0.2151144567745126

7. From the above following coefficient of determinants of all independent attributes we can infer that **'Carat' variable has the most weightage and acts as the best predictor for price.** The number of predictors is less, yet the model is better comparatively.
8. The coefficient of determination is a measurement used to explain how much variability of one factor can be caused by its relationship to another related factor
9. **The intercept for our model is -2.725e-16.** After scaling the intercept becomes almost equal to zero.
10. Model performance of regression model built is calculated by the coefficient of determinant (R square). R square determines the fitness of a linear model. R square value ranges from 0 to 1. The closer the data point is to the best fit plane; the coefficient of determinant value tends to 1 and the better the model.

    R square of training data is 0.93013

    R square of testing data is 0.930502

Scaling of dataset, does not affect the R square values. We can see that the score of Train and Test is almost similar, **the model is a good fit model.**

11. Calculating root mean square error (RMSE) value for checking model performance i.e. RMSE value is standard deviation of the prediction errors (residual). Residual errors or sum of squared errors are the measure of how far the data point is from the best fit plane. So basically, RMSE tells the spread out of these residuals. That means lower the RMSE is, closer are the data points to the best fit plane.

RMSE of training data is 0.2643

RMSE of testing data is 0.2636

This means we have almost 26% variance of residual error or unexplained error in our model. It allows better interpretability for us to study the model. After scaling our RMSE value has been cut down to a large extent. The data points are concentrated close to the best fit plane.

12. Checking the plot between the original price and the predicted price for linear relationship.

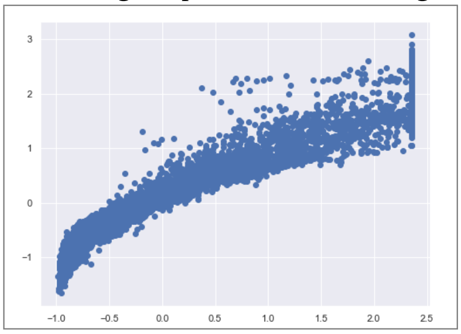


Figure 25. Scatter plot for model 5.

Checking the OLS summary for the model:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.930
Model:                            OLS   Adj. R-squared:                  0.930
Method:                 Least Squares   F-statistic:                 5.017e+04
Date:                Thu, 30 Dec 2021   Prob (F-statistic):               0.00
Time:                        23:22:37   Log-Likelihood:                -1664.7
No. Observations:               18847   AIC:                             3341.
Df Residuals:                   18841   BIC:                             3389.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept  -4.033e-17      0.002  -2.09e-14      1.000      -0.004       0.004
carat          1.0580      0.002    481.982      0.000       1.054       1.062
table         -0.0095      0.002     -4.382      0.000      -0.014      -0.005
cut_c         -0.0366      0.002    -16.968      0.000      -0.041      -0.032
color_c        0.1343      0.002     66.009      0.000       0.130       0.138
clarity_c      0.2151      0.002    102.260      0.000       0.211       0.219
==============================================================================
Omnibus:                     2374.863   Durbin-Watson:                   2.007
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             7550.189
Skew:                           0.653   Prob(JB):                         0.00
Kurtosis:                       5.812   Cond. No.                         1.87
==============================================================================

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

From the above summary we can infer that:

- R- squared and Adjusted R-squared vales are same which is equal to 0.930
- Overall p value of model is 0.00 (< 0.05) which means model is reliable.

- Considering each variable and its p value, we can see no variable has p value more than 0.05. So, all variables in this model have good relationship with dependent variable. Considering all these variables in our model.
- In this model, condition number has reduced to 1.87. This shows that multicollinearity and other mathematical problems are not there anymore in our model.
- **The RMSE score for our OLS model is 0.2643.**
- **Checking Multi-collinearity using VIF for model 3.**

  We test for multicollinearity with Variation Inflation Factor (VIF). VIF identifies correlation between independent variables and the strength of that correlation. VIF starts from 1 and has no upper value.

  VIF equal to 1 indicates no correlation between independent variables.
  VIF between 1 to 5 indicates moderate correlation but not severe.
  VIF greater than 5 indicates critical levels of multicollinearity.

  - carat ---> 1.30155395589329
  - table ---> 1.2632339639911656
  - cut_c ---> 1.2570332894546756
  - color_c ---> 1.1153449236143709
  - clarity_c ---> 1.193835288526028

- All variables have VIF score of almost 1 suggesting negligible correlation among the independent variables is present in the dataset
- **Linear Equation for model 5:**

```
(-0.0) * Intercept + (1.06) * carat + (-0.01) * table + (-0.04) *
cut_c + (0.13) * color_c + (0.22) * clarity_c
```

Inferences:

'Carat' is still the best predictor with a coefficient of 1.06. For 1 unit change in 'carat', the price will change by 1.06 units keeping all other variables 0. By looking at all the performance matrices of this model, we can say this model fulfils all criteria to be the best fit model.

Model comparison:

Model comparison	Intercept	Sk_learn_Rsq_train	Sk_learn_Rsq_test	Sk_learn_RMSE_train	Sk_learn_RMSE_test	Stat_Rsq	Stat_Adj_Rsq	Stat_RMSE	VIF_max	VIF_min
Model 1	-5.164440e+03	0.9312	0.9316	906.8900	911.2900	0.940	0.940	844.290	10638.270	6.138
Model 2	-3.136180e+03	0.9301	0.9305	914.8700	918.4300	0.939	0.939	851.300	500.290	5.165
Model 3	-4.490270e+03	0.9301	0.9301	914.0600	918.7500	0.939	0.939	851.350	34.373	5.135
Model 4	-3.871690e+02	0.9240	0.9250	948.4700	953.8100	0.933	0.933	839.710	30.720	5.014
Model 5	-2.720000e-16	0.9301	0.9304	0.2643	0.2636	0.930	0.930	0.264	1.301	1.115

Table 2. Model comparison table.

Model 1: Considering all the variables as it is and fitting the linear regression model.

Model 2: Dropping the attributes 'x', 'y' & 'z' and fitting the linear regression model.

Model 3: Dropping the attributes 'x', 'y', 'z' & 'depth' and fitting the linear regression model.

Model 4: Dropping the attributes 'x', 'y', 'z', 'depth' and grouping sub categories of attributes and fitting the linear regression model.

Model 5: Dropping the attributes 'x', 'y', 'z' & 'depth' and fitting the linear regression model for scaled data

Inferences:

- Accuracy (R square) is same for all models for both sklearn as well as stat models.
- Model 1, 3 and 5 give us the best RMSE values.
- VIF max and VIF min values are lowest for models 5, since that data is scaled.
- In model 4, even after combining the sub categories of attributes the RMSE score for train and test is more compared to other model, so the idea of combining sub categories is dropped.
- Model 3 and 5 using same attributes while one model is built using scaled attributes and other is original dataset.
- **Model 5 is our best fit model and most viable for the given set based on the performance measures of other models.**

Final linear equation is as given below:

(-0.0) * Intercept + (1.06) * carat + (-0.01) * table + (-0.04) * cut_c + (0.13) * color_c + (0.22) * clarity_c

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

According to the problem statement, Gem stones co ltd, a cubic zirconia manufacturer earns varying profits on different pricing slots. The company wants to predict the stone's price based on the data provided so that the company may distinguish between higher profitable and lower profitable stones and maximise the company’s profit share. Also require the top five attributes which are most essential in price prediction.

In our extensive analysis so far, we have thoroughly examined historical data and developed a model that predicts different price slots based on the characteristics in our dataset. Let us now look at the key points in our past data first and try to suggest some recommendations for the firm.

To have a better profit share, the business value is to distinguish between higher profitable stones and lower profitable stones. Our model has an accuracy score of more than 90%, which may be acceptable in this business, and will properly predict the price for more than 90% of the stones.

Following are the insights and recommendations to help the firm to solve the business objective:

1. Carat: Carat weight of the cubic zirconia

Insights:

- Carat is the best predictor for the price.
- It has the positive linear relation with price. The price increases with increase in carat of zirconia stone.
- Carat is measure of weight which has direct correlation with physical dimensions (x, y, z)

Recommendations:

- Carat is the best predictor of price, according to the best fit model.
- The firm should favour more stones with a higher carat value, stones with larger carat values are priced higher
- The significance of higher carat stones should be advertised to people.

- Marketing should be done in such a way that clients are aware of the significance of higher carat values.
- Customers should receive varied presentations depending on their financial capabilities. Customers with a higher financial status should be offered higher quality carat stones, while those with a lesser paying ability should be offered lower carat stones.
- The marketing can be done educating customers about the significance of a better carat score and quality.

2. Cut: Describe the cut quality of the cubic zirconia.

Insights:

- For cut attribute, we see that Ideal cut type is the most selling and the average price of Ideal is slightly less prices compared to premium cut type which is slightly more expensive.
- ‘Fair’ and ‘Good’ have a lower count of sales and have a relatively higher average price.
- The ideal, premium, very good cut types have better profits.

Recommendations:

- The ideal, premium, very good cut types are the one which are bringing more profits, proper marketing of the products may increase the sales to greater extend.
- The best quality cut, 'Ideal,' has a lower average price comparatively. However, 'Ideal' has a high count at this pricing. The firm might try increasing the price of the ideal category a little to see whether it affects sales. If sales are reduced, they should return to the current market price.
- Although we know that 'Fair' and 'Good' are of the lowest cut quality and are sold in small quantities, their average price is still rather substantial. The firm can attempt to lower its average price or increase the quality of these cuts so that customers are willing to pay the higher price.
- ‘Fair’ and ‘Good’ cut types is advisable to eschew as the number of sales and profits are very less.

3. ‘X’, ‘Y’ & ‘Z’:

Insights:

- X, Y and Z are the length, width and height of the cubic zirconia. All are having the linear relation with each other and also the target variable 'price'.
- All three have a strong relation to the price variable. That is, changes in the values of x, y, and z cause price values to change.
- At the same time, there is a significant association between these three. This indicates that these variables end up causing a high multicollinearity, which affect the performance of our price prediction

Recommendations:

- The dimensions are having negative effect on the stones, smaller the dimension’s mostly balanced size is more expensive.
- If a stone with smaller dimensions has a larger carat value and superior clarity, it will be valued higher than a huge stone with lower carat and clarity.
- Firm can focus more on the balanced different sizes with higher quality stones.

4. Depth and Table:

- Depth and table both are poor predictors of price.
- From the EDA of depth and price & table & price, we can see that there is a minimal relationship of depth and table with price, there is no defined relationship its spread like could, which is not useful for model building.

5. Clarity:

Insights:

- Clarity refers to the absence of the Inclusions and Blemishes and has emerged as a strong predictor of price as well.
- S1 is the expensive one followed by the VS2 and S2 clarity which fall in the same price range and l1 and lF are the cheap stones.
- S1 type of Clarity is most selling followed by VS2 and I1 being the least selling one.
- Clarity of stone types Sl1, VS2 and Sl2 are helping the firm put an expensive price cap on the stones and also have most selling counts.

Recommendations:

- Price of 'I1' could be reduced as it is having very low sales.
- I1' is of the highest quality and may reduce earnings, but a little risk may be taken by the firm by lowering its price for a period of time, and if sales grow, the price can be raised to its former level.
- 'IF,' 'VVS1', and 'VVS2' are more helpful in price prediction than other clarity categories. In comparison to other areas, the firm should put greater emphasis on them.

6. Color

Insights:

- G color gem is the costly one and also most liked by the people and are highest sold.
- J color gem price is less and also the least sold one
- We see that 'G' color is the most selling zirconia stone followed by 'E' and 'F' nearly following in same range and 'J' color gem is the least selling stone.

Recommendations:

- The color of the stones, such as H, I, and J, will not help the company in putting a high price cap on such stones.
- Instead, the firm should concentrate on stones in the color D, E, and F in order to fetch greater prices and boost sales.
- This might also signal that the firm should be exploring for unique color stones, such as transparent stones to help boost the pricing.
- 'J' and 'I' color stones should be priced lower. Maybe the customers get attracted by the lower price and the sales is increased.

The best 5 attributes which are good predictors for prediction of price are as follows:

1. Carat
2. Clarity
3. Color
4. Cut
5. Table

Key performance indicators:

- Sales promotion: Special deals stimulates demand. Sales promotion can be effective in changing short term behaviour of buyer.
- Advertising is the efficiency way for reaching many people and the potential buyers. For example, Advertising campaign can be done in around month of Jan and Feb, when the Valentine’s Day is near, or the occasions like Mother’s Day, etc.
- The company can make segments, and target the customer based on their income/paying capacity etc, which can be further studied.
- Customers can be educated about the value of a higher carat score and the clarity index through marketing initiatives.
- Customization of products can be initiated for better sales.

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.

Data Dictionary:

Variable Name	Description
Holiday_Package	Opted for Holiday Package yes/no?
Salary	Employee salary
age	Age in years
educ	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

The purpose of the report is to examine past information on selling holiday packages in order to assist 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. Understanding the data and examining the pattern. Providing business insights based on exploratory data analysis and predictions of classes.

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.

Exploratory Data Analysis:

Read and view data after dropping 'Unnamed: 0' variable:

	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
5	yes	61590	42	12	0	1	no
6	no	94344	51	8	0	0	no
7	yes	35987	32	8	0	2	no
8	no	41140	39	12	0	0	no
9	no	35826	43	11	0	2	no

Checking for the information of features:

```
<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
```

Checking the Skewness and Kurtosis:

```
Holiday_df.skew()

Salary               3.103216
age                  0.146412
educ                -0.045501
no_young_children    1.946515
no_older_children    0.953951
dtype: float64
```

```
Holiday_df.kurt()

Salary               15.852557
age                  -0.909962
educ                  0.005558
no_young_children     3.109892
no_older_children     0.676017
dtype: float64
```

Checking the description of dataset:

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

Checking for duplicates in this dataset:

```
# Are there any duplicates?
dups = Holiday_df.duplicated()
print('Number of duplicate rows = %d' % (dups.sum()))
Holiday_df[dups]

Number of duplicate rows = 0
```

Checking for number of rows and columns:

```
Holiday_df.shape

(872, 7)
```

Checking for Null and missing values in the dataset:

```
Holiday_df.isnull().sum()

Holliday_Package     0
Salary               0
age                  0
educ                 0
no_young_children    0
no_older_children    0
foreign              0
```

Observations:

- Dataset has 7 columns and 872 rows excluding the 'unnamed:0' column.
- The first column "Unnamed: 0" has only serial numbers, so we can drop it as it is not useful.
- There are both categorical and continuous data. For categorical data, we have 'Holiday_Package' and 'foreign', for continuous data we have salary, age, educ, no_young_children, no_older_children.
- Holliday Package will be target variable.
- The dataset is used in predicting whether an employee will opt for the Holiday_package or not on the basis of the information given in the data set.
- There are no missing and duplicate values in the dataset.
- There is total 5 unique types of 'cut' out of which the highest number of cut is 'Ideal' one which accounts to almost 10816 of observations, which is approximately 50% of the dataset.
- Skewness and Kurtosis is also calculated for each column, Data with high skewness indicates lack of symmetry and high value of kurtosis indicates heavily tailed data.
- Based on summary descriptive, the data looks good, we see that for most of the variables the mean/medium are nearly equal.
- We have a balanced dataset where 54% yes values and 45% no values of Target variable.

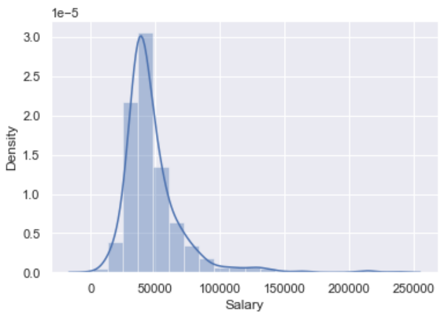
Data Visualization:

Univariate Analysis for Numeric Variables:

Let us define a function 'univariateAnalysis_numeric' to display information as part of univariate analysis of numeric variables. The function will accept column name and number of bins as arguments.

1 - Salary

```
Description of Salary
------------------------
count        872.000000
mean       47729.172018
std        23418.668531
min         1322.000000
25%        35324.000000
50%        41903.500000
75%        53469.500000
max       236961.000000
```



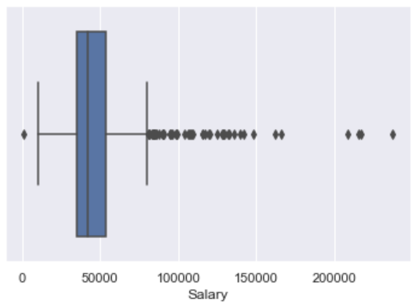


Figure 26. Boxplot and Distplot for Salary

- From the above graphs, we can infer that mean 'Salary' of employee is around 47729.17 with the minimum of 1322.0 and maximum of 236961.0.
- The distribution of 'Salary' is right skewed with skewness value of 3.103216.
- The distplot shows the distribution of most of data from 1000 to 10,000 approximately.
- The box plot of the 'Salary' variable shows presence of large number of outliers.

2 – Age

```
Description of age
--------------------
count    872.000000
mean      39.955275
std       10.551675
min       20.000000
25%       32.000000
50%       39.000000
75%       48.000000
max       62.000000
```

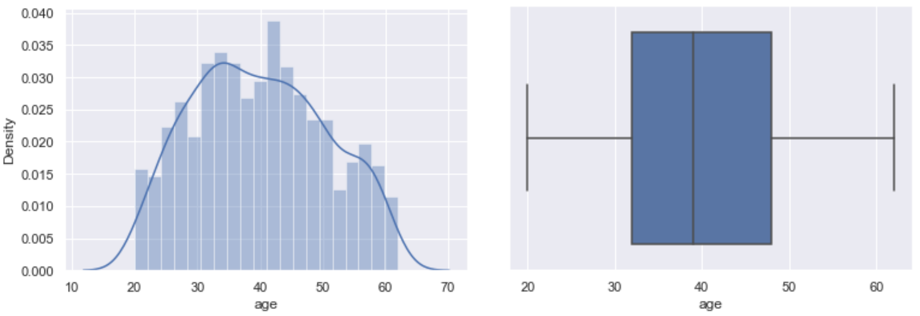


Figure 27. Boxplot and Distplot for Age.

- From the above graphs, we can infer that mean 'Age' of employee is around 39 years with the minimum of 20yrs and maximum of 62yrs old in company.
- The distribution of 'Age' looks almost normally distributed with skewness value of 0.146412
- The distplot shows the distribution of most of data from 20 to 60 approximately.
- The box plot of the 'Age' variable does not have any outlier.

3 - Educ: Years of formal education

```
Description of educ
--------------------
count    872.000000
mean       9.307339
std        3.036259
min        1.000000
25%        8.000000
50%        9.000000
75%       12.000000
max       21.000000
```

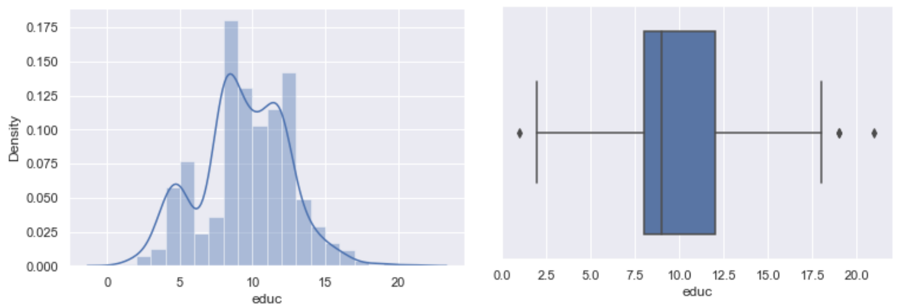


Figure 28. Boxplot and Distplot for Education.

- From the above graphs, we can infer that mean 'Educ' years of formal education of employee is around 9 years with the minimum of 1yr and maximum of 21yrs.
- The distribution of 'Educ' is slightly left skewed with skewness value of -0.045501.
- The distplot shows the distribution of most of data from 1 to 20 approximately.
- The box plot of the 'Educ' variable shows presence of few outliers.

4 - no_young_children: The number of young children below the age of 7yrs.

```
Description of no_young_
------------------------
count    872.000000
mean       0.311927
std        0.612870
min        0.000000
25%        0.000000
50%        0.000000
75%        0.000000
max        3.000000
```

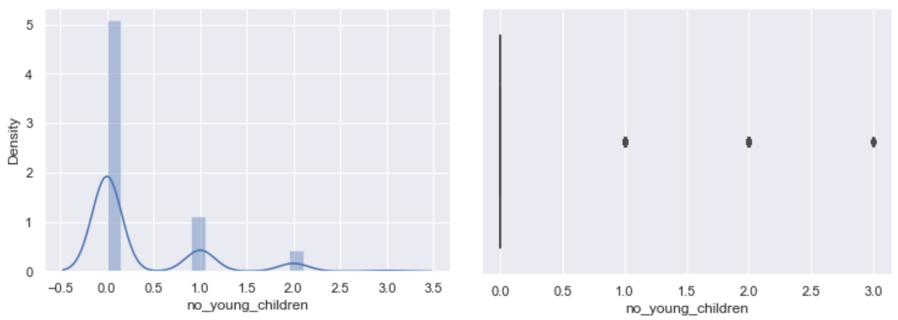


Figure 29. Boxplot and Distplot for no_young_children

- From the above graphs, we can infer that mean 'no_young_children' number of young children below the age of 7yrs is around 0.3119 with the minimum of 0 and 3.
- The distribution of 'no_young_children' is slightly left skewed with skewness value of 1.9465.
- The distplot shows the distribution of most of data from 0-3.
- The box plot of the 'no_young_children' variable shows presence of few outliers.

5 - no_older_children: The number of older children.

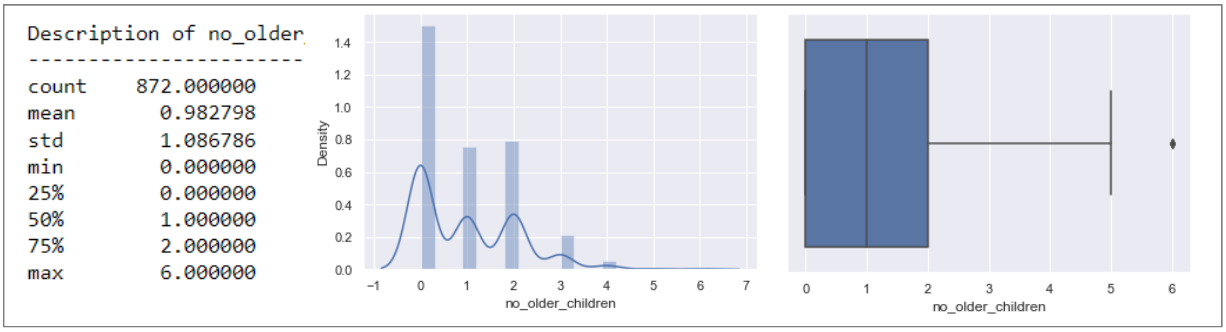


Figure 30. Boxplot and Displot for no_older_children

- From the above graphs, we can infer that mean 'no_older_children' the number of older children is around 0.9827 with the minimum of 0 and maximum of 6.
- The distribution of 'no_older_children' is slightly right skewed with skewness value of 0.953951
- The distplot shows the distribution of most of data 0-4 approximately.
- The box plot of the 'no_older_children' variable shows presence of one outlier at 6.

Observations:

Table 3. Inferences of Univariate Data visualization for problem 2.

Sl. No	Features	Distribution	Skewness	Outliers
1	Salary	Right Skewed	+3.103	Yes
2	Age	Left Skewed	-0.146	No
3	Education	Left Skewed	-0.045	Very few
4	no_young_children	Right Skewed	+1.946	Very few
5	no_older_children	Right Skewed	+3.850	Very few

- There are outliers just in Salary variable, and the outliers in other variable are just 1 or 2 which does not effect.
- Treating of Outlier might not be feasible option as the data can be original and genuine.
- Foreigners accepting the holiday package have mean of years of formal education lesser than natives accepting the holiday package.
- If employee is foreigner and employee not having young children, chances of opting for Holiday Package is good.

Univariate Analysis for Categorical variables:

1. Holliday_Package

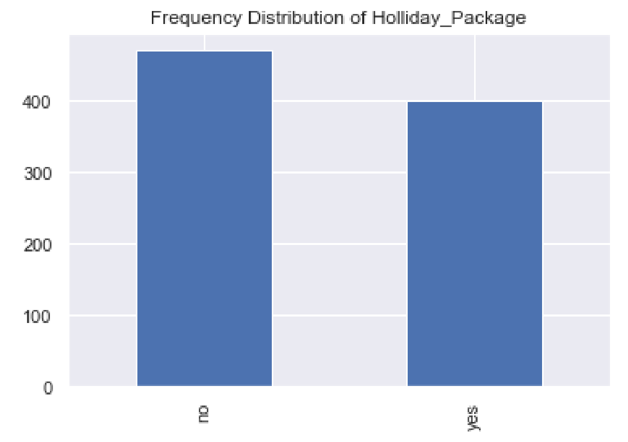


Figure 32. Count plot for Holiday package

2. Foreign: foreigner Yes/No

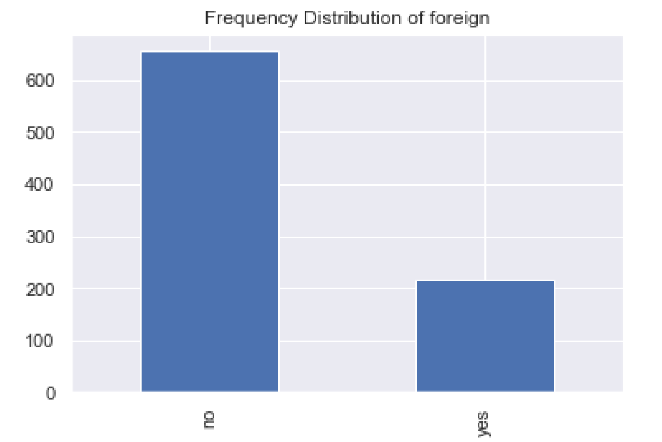


Figure 31. Count plot for foreign.

Observations:

- The distribution of the 'Holiday_package' is one where employee opt for package or no, we can see that frequency distribution of 'No' is more which is around 471 and the employees who opted are slight less which is 401 in count.
- We can observe that 54% of the employees are not opting for the holiday package and 46% are interested in the package. This implies we have a dataset which is fairly balanced
- The frequency distribution of foreign implies that the employees are mostly from the same country which is around 75% of employees and foreigners are around 25% of them.

Bivariate Analysis:

Salary vs Holiday_Package:

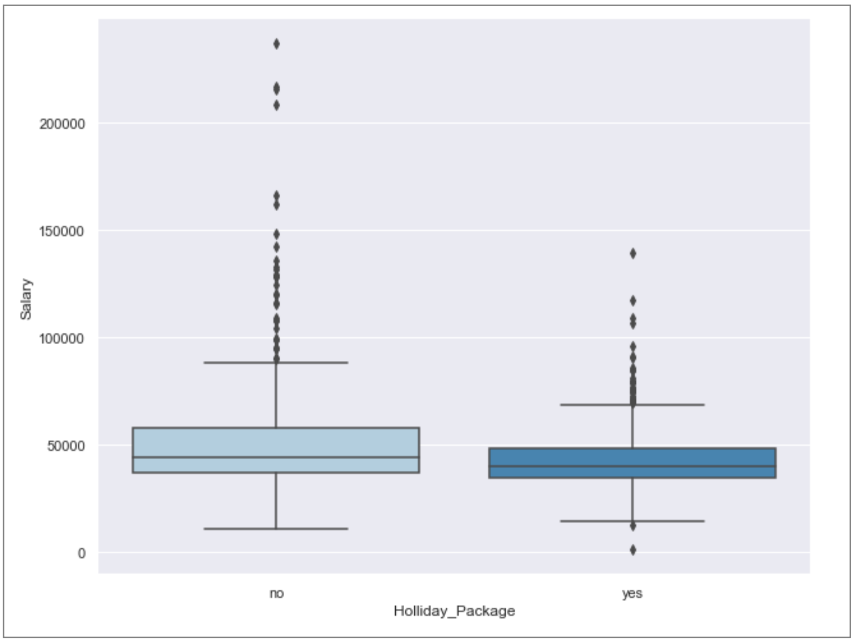


Figure 33. Boxplot of Salary vs Holiday package

We can see that the average 'Salary' of employees opting for holiday package and not opting for holiday package is similar in nature. However, the distribution is fairly more spread out for people not opting for holiday packages.

Age vs Holiday package:

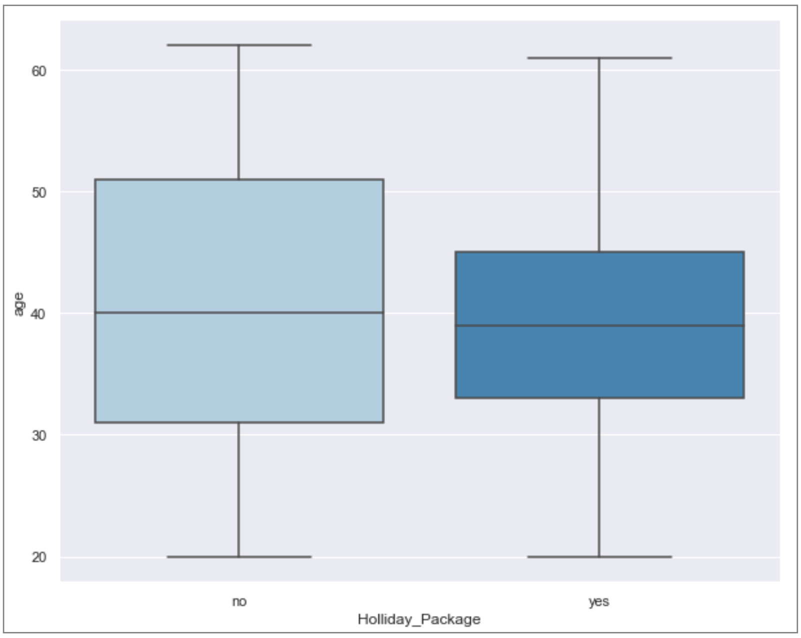


Figure 34. Boxplot of Age vs Holiday package

We can see that, the age distribution for employees who are opting for holiday package and not opting are similar in nature, though the number of people opting are less in number and mostly fall in range of 35-45 age group.

Count plot of Age with Holiday package as hue

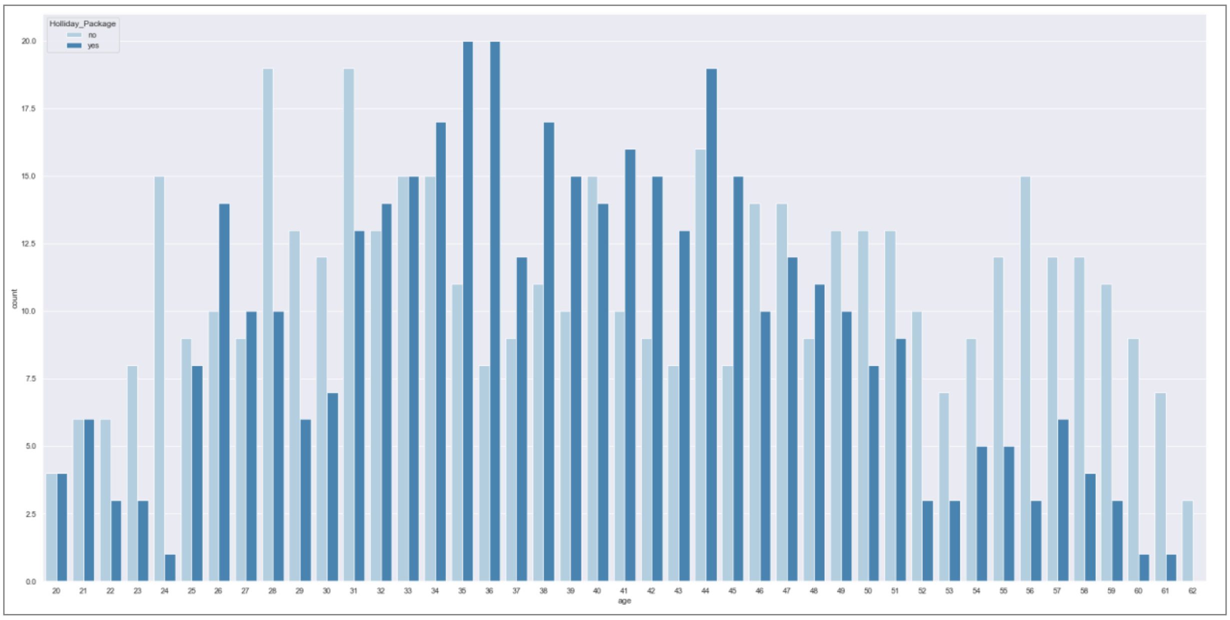


Figure 35. Count plot of Age against Holiday package

We can clearly see that frequency of employees in middle range (34 to 45 years) are opting for holiday package are more as compared to older and younger employees.

Education vs Holiday package

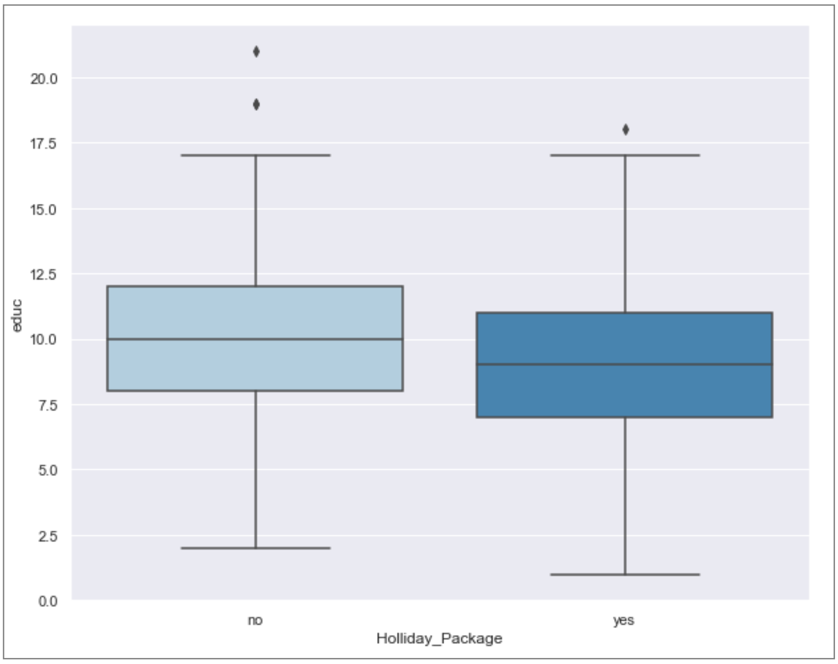


Figure 37. Boxplot of Education vs Holiday package

The variable 'educ' the number of years of formal education is showing a similar pattern. This means education is likely not a variable that influences for opting of holiday packages for employees.

Count plot of Education with Holiday package as hue

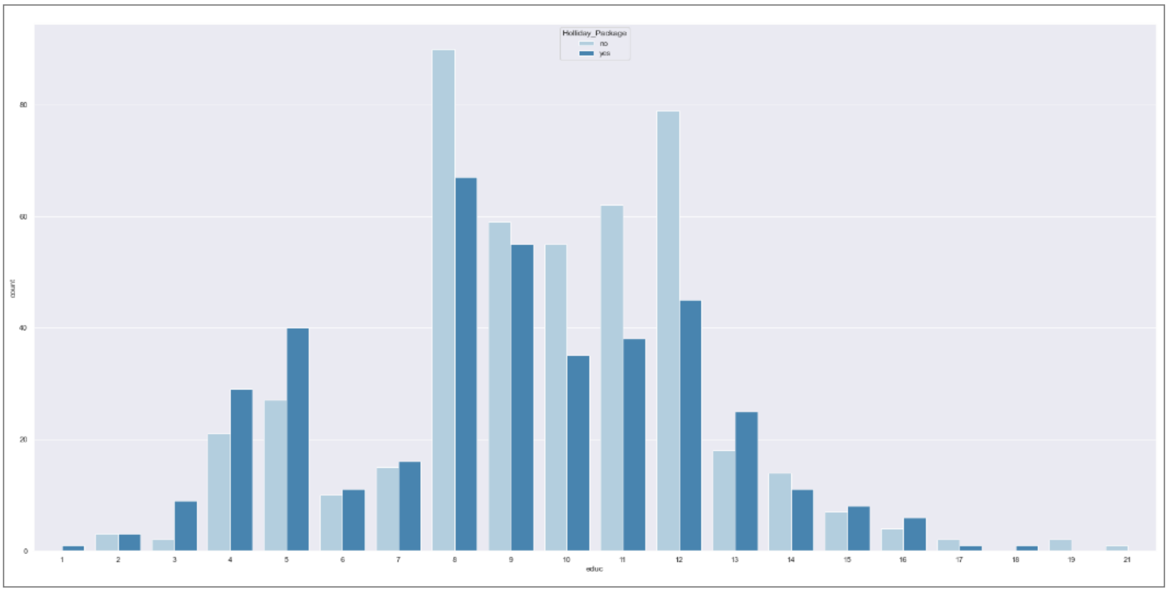


Figure 38. Count plot of Education against Holiday package

We can see that employee with less years of formal education (1 to 7 years) and higher education are not opting for the Holiday package as compared to employees with formal education of 8 year to 12 years

No of young children vs Holiday package

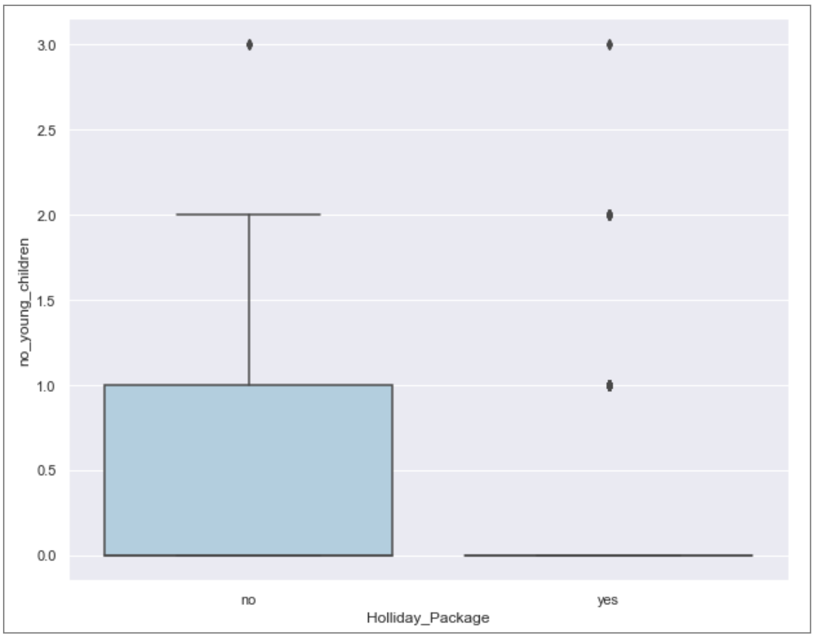


Figure 39. Boxplot of no of young children vs Holiday package

We can see that there is a significant difference in employees with younger children who are opting for holiday package and employees who are not opting for holiday package, this attribute is good predictor as there is significant difference in them.

Count plot of no of young children with Holiday package as hue

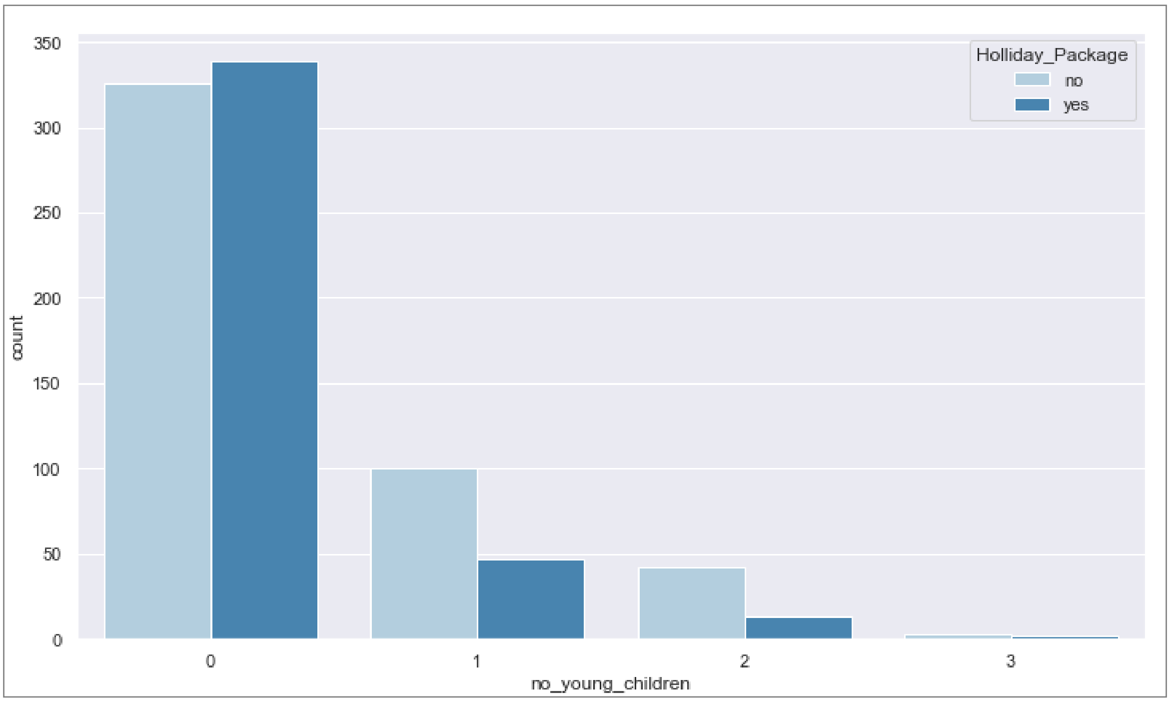


Figure 40. Count plot of no of young against Holiday package

We can see clearly that people with younger children are opting for holiday packages are very few in number compared to employees who do not have young children.

No of older children vs Holiday Package

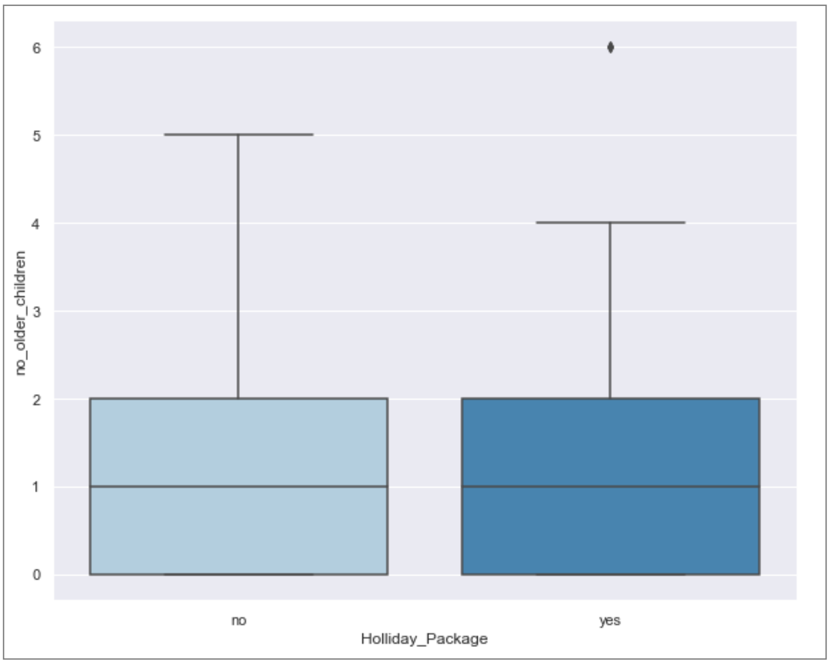


Figure 41. Boxplot of no of older children vs Holiday package

The distribution for opting or not opting for holiday packages looks same for employees with older children. At this point, this might not be a good predictor for model building.

Count plot of no of older children with Holiday package as hue

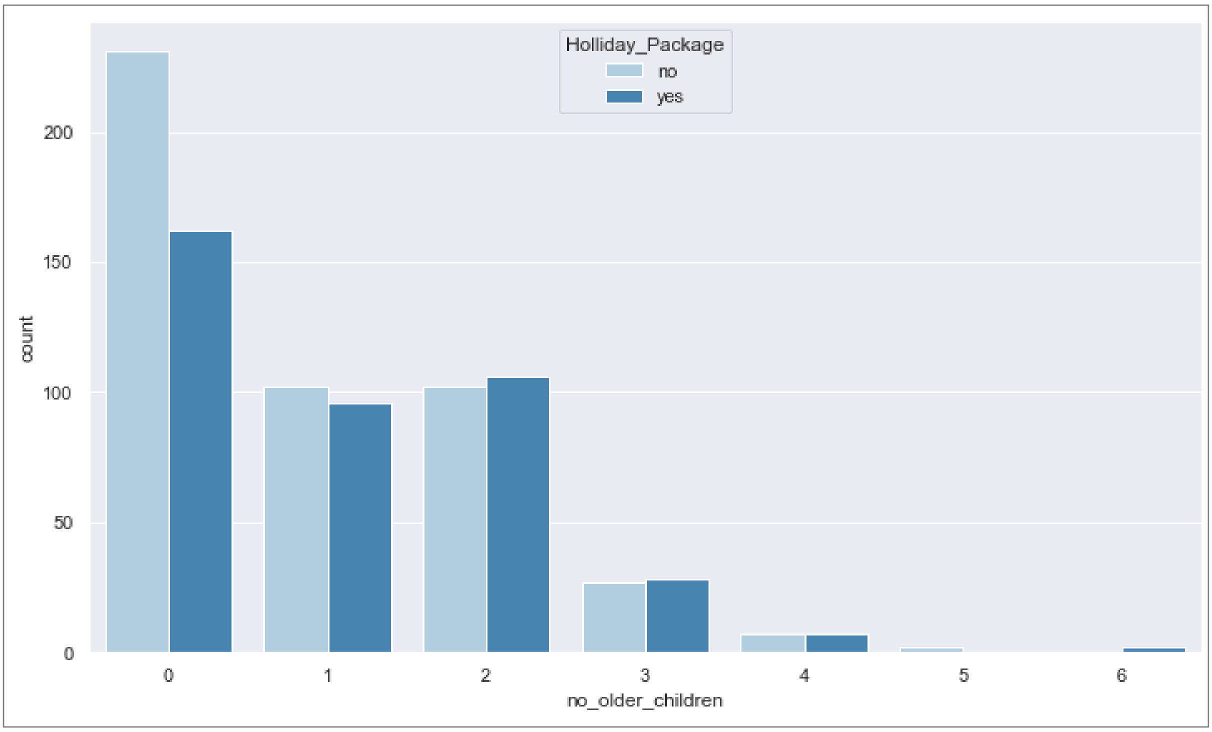


Figure 42. Count plot of no of older against Holiday package

Almost same distribution for both the scenarios when dealing with employees with older children.

Foreign vs Holiday package

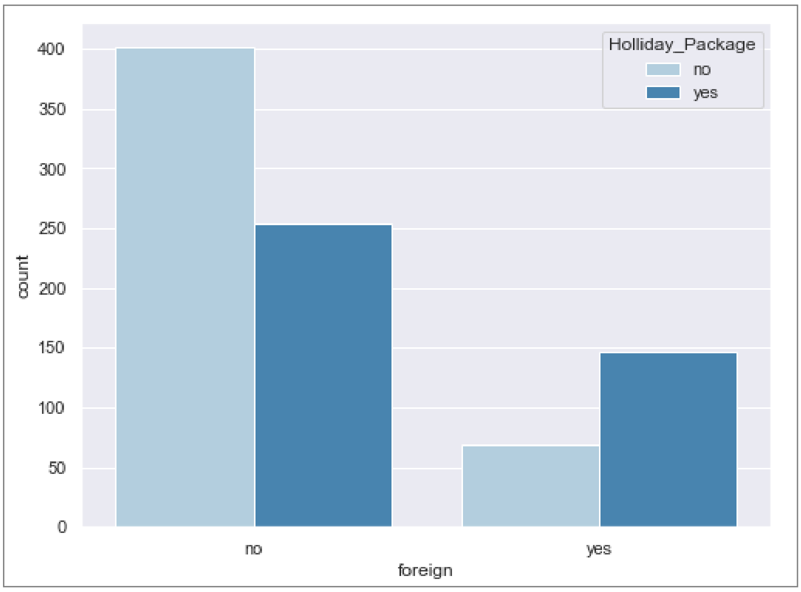


Figure 43. Count plot of foreign vs Holiday package

We can see that the percentage of foreigners accepting the holiday package is substantially higher compared to the citizens with considering the ratio of foreigners and the citizens.

Box plot of foreign vs Salary with Holiday package as hue

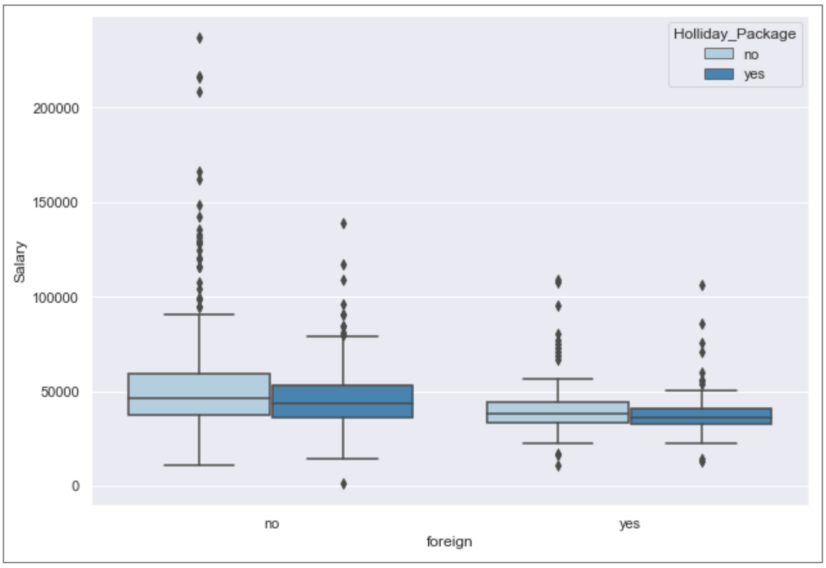


Figure 44. Boxplot of foreign vs Salary with Holiday package as hue

- In both foreigner and non-foreigner, the people who did not opt for the Holiday package are more in number that the people who have opted.
- The average of people who didn't opt for Holiday package is slightly more than who have opted.
- The mean salary of foreign people is slightly less than natives.
- There are outliers in all the combinations.

Pair plot:

The Pair plot helps us to visualize how the features numerical in nature interact with each other. The pair plot further helps us visualize how the distribution of the target variables differs within each individual the feature itself.

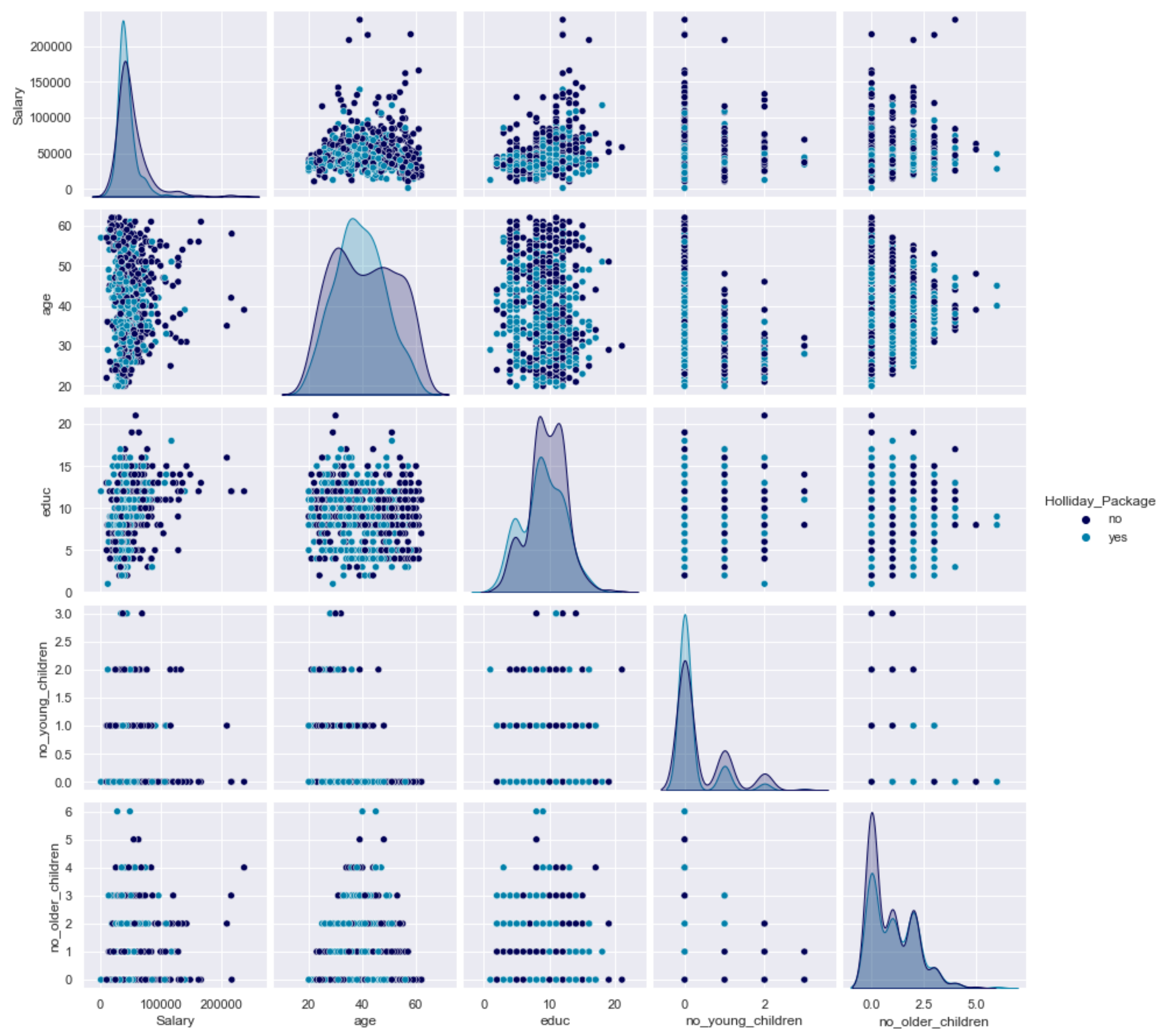


Figure 45. Pair plot of problem 2

Observations:

- There is no obvious defined correlation between the attributes and Holiday package, the data seems to be fine.
- There is no considerable difference between data distribution of holiday package. No clear and considerable difference is observed.
- Looking at the distribution of age, we can deduce that the employees who accept the holiday package usually tend to be in the middle of their careers (late 30s).
- Across education we can observe that the employees with higher number of years of formal education have a lower tendency to opt for the holiday package relative to employees with lesser years of formal education

Multivariate Analysis:

Heatmap

A heatmap gives us the correlation between numerical variables. If the correlation value is tending to 1, the variables are highly positively correlated whereas if the correlation value is close to 0, the variables are not correlated. Also, if the value is negative, the correlation is negative. That means, higher the value of one variable, the lower is the value of another variable and vice-versa.

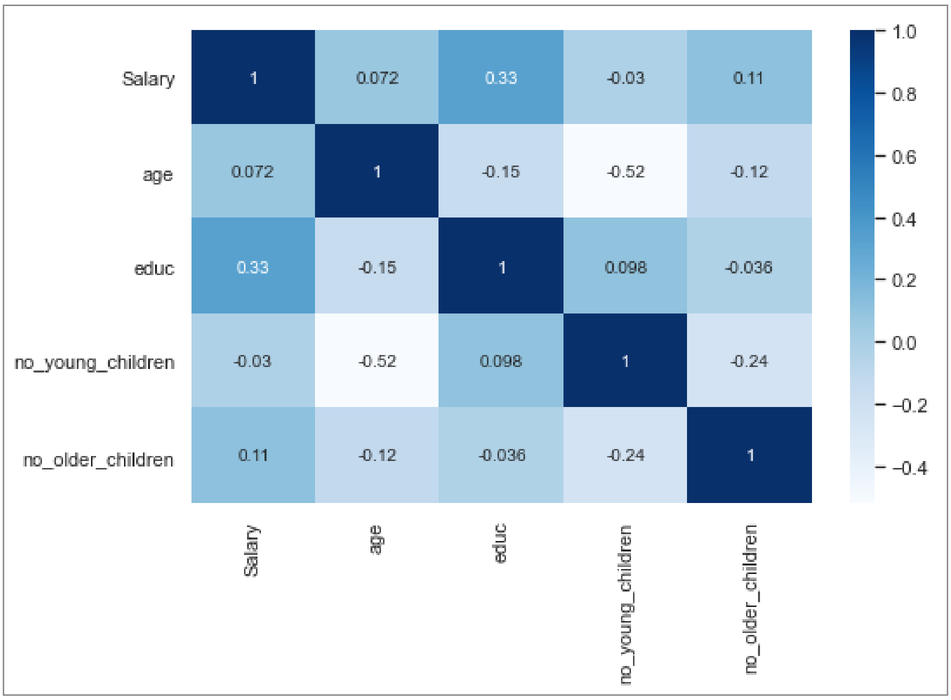


Figure 46. Heatmap for Problem 2.

Observations:

- There is no strong correlation between the variables, hence we do not face the issue of multicollinearity.
- Observing the heatmap we can see that the there is some positive correlation is among number of years of formal education and the salary received.
- There some negative correlation between age and the employees with no of young children below age 7.

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 the categorical variables:

Encoding categorical data is a process of converting categorical data into integer format so that the data with converted categorical values can be used to build the models to give the predictions. The object type variables are converted to integer using pandas categorical to codes of 0 and 1.After the encoding the variables should be converted in to integer type data types for the model building.

Checking datatypes after encoding:

```
Holiday_df.dtypes

Holliday_Package      int32
Salary                int64
age                   int64
educ                  int64
no_young_children     int64
no_older_children     int64
foreign               int32
```

Checking the head of Dataset after encoding the Categorical variables

	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

Now the dataset is cleaned, encoded and ready to use for model building.

Split: Split the data into train and test (70:30)

1. Capture the target column into separate vectors for training set and test set

X variable with independent attributes and y variable with the target variable which is 'Holiday_Package' in our case.

```
# Copy all the predictor variables into X dataframe
X = Holiday_df.drop('Holliday_Package', axis=1)

# Copy target into the y dataframe.
y = Holiday_df['Holliday_Package']
```

2. Splitting the dataset in to train and test in the ratio of 70:30 using train test split from sklearn, keeping the random state as 1.

3. Checking the shape of the split data.

```
X_train (610, 6)
X_test (262, 6)
y_train (610,)
y_test (262,)
```

The data is now read to fit the models on train and check the performance of test data. The data is divided in 70% of train and 30% of test.

Logistic Regression Model

Logistic Regression is a Machine Learning algorithm which is used for the classification problems, it is a predictive analysis algorithm to understand the relationship between the dependent variable and one or more independent variables by estimating probabilities using a logistic regression equation. The classification algorithm Logistic Regression is used to predict the likelihood of a categorical dependent variable. The dependant variable in logistic regression is a binary variable with data coded as 1.

To build a Logistic Regression model,

- Fitting the Logistic Regression model which is imported from Sklearn linear model.
- Predicting on Training and Testing dataset
- Getting the Predicted Classes and Probabilities and creating a data frame.
- Model evaluation through Accuracy, Confusion Matrix, Classification report, AUC, ROC curve.

Initially, we fit the train data and labels in the Logistic Regression model, based on the model performance the model is tunned using Grid search, the best parameters are used and the model is re-built and model performance is calculated which includes Classification report of accuracy, recall, precision and F1 score for both train and test data.

Grid Search: Grid search divides the hyperparameter domain into distinct grids. Then, using cross-validation, it attempts every possible combination of values in this grid, computing some performance measures. The ideal combination of values for the hyperparameters is the point on the grid that maximizes the average value in cross-validation. Grid search is a comprehensive technique that considers all possible combinations in order to locate the best point in the domain

Hyperparameter Tuning:

- ***'penalty':*** *['L2', 'none'],*
- ***'solver'****: ['sag', 'lbfgs', 'liblinear', 'newton-cg'],*
- ***'tol':*** *[0.0001, 0.00001],*
- ***'Max_iter':*** *[10000, 5000, 15000]*
- ***Cross validation (cv)****:5*
- ***Scoring****: 'f1'*

Penalized logistic regression imposes a penalty to the logistic model for having too many variables. This results in shrinking the coefficients of the less contribute variables toward zero. This is also known as regularization. In our grid search, we take 'L2' and 'none' as our arguments and check which is preferred by grid search.

The solver is the process that runs for the optimization of the weights in the model. The solver uses a Coordinate Descent (CD) algorithm that solves optimization problems by successively performing approximate minimization along coordinate directions or coordinate hyperplanes. Different solvers take a different approach to get the best fit model. In our case, we have taken 'sag', 'lbfgs', 'liblinear' and 'newton-cg' as our arguments. We will check which is preferred by grid search.

Tol is the tolerance of optimization. When the training loss is not improved by at least the given tol on consecutive iterations, convergence is considered to be reached and the training stops. We will be checking for tolerance of 0.0001 and 0.00001.

The logistic regression uses an iterative maximum likelihood algorithm to fit the data. There are no set criteria for **maximum iterations**. The solver will run the model till it reaches convergence or till the max iterations, you have provided. In this case, we have given 5000, 10000 and 15000 as inputs. We will see which fits better.

We have taken cross-validation as 3 and scoring as F1 for our grid search.

The final best parameters are:

- ***Max_iter*** *is '10000'*
- ***Penalt****y is 'None'*
- ***Solver*** *used is 'newton-cg'*
- ***Tol*** *is 0.0001*

Our new model, which is based on the grid search algorithm's best parameters and the model's performance is tested using these parameters is then saved in a distinct variable as best_model. This model is used to predict the values of the target variable, and then the model's performance is evaluated using these parameters.

Checking the Coefficients:

- The coefficient for Salary is -1.646142121152848e-05
- The coefficient for age is -0.05707255243551053
- The coefficient for educ is 0.06034737348280886
- The coefficient for no_young_children is -1.3488352961597043
- The coefficient for no_older_children is -0.04894374035375453
- The coefficient for foreign is 1.2664799760127905

LDA Model (linear discriminant analysis)

Linear Discriminant Analysis is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling differences in groups i.e., separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.

LDA works when the measurements made on independent variables for each observation are continuous quantities. When dealing with categorical independent variables, the equivalent technique is discriminant correspondence analysis.

On the train data set, we fit our Linear Discriminant model. By default, LDA uses a custom cut-off probability of 0.5. So, initially, we'll create our LDA model with a cut-off probability of 0.5 and see how it performs, then we'll see how it performs with multiple cut-off probabilities to see which one performs the best.

We obtain an LDA model based on a default custom cut-off probability (i.e., 0.5). To get the best results, we'll need to test our model with several cut-off probabilities and choose the one that produces the greatest results. To do so, we'll start with probability 0.1 and work our way up to 0.9 with a 1 interval, checking each probability recall and F1 score value along the way. We will use the likelihood that we will get the best recall and F1 score balance as our final probability value.

Cut off probability	Recall	F1 Score
0.1	0.9964	0.6393
0.2	0.9644	0.6499
0.3	0.8932	0.6693
0.4	0.7580	0.6762
0.5	0.5765	0.6125
0.6	0.4235	0.5336
0.7	0.2989	0.4398
0.8	0.1103	0.1981
0.9	0.0071	0.0141

Table 4. LDA cut off probability performance table

We can see from the table above that cut off probability 0.4 provides the optimal balance of recall and F1 score. As a result, we'll discuss about the performance of our LDA model using both the default and the 0.4 cut-off probability.

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 performance helps to understand how good the model that we have trained using the dataset is so that we have confidence in the performance of the model for future predictions.

We evaluate our models' performance on train and test datasets once they've been constructed. We try to determine if the model is underfitting or overfitting by checking for accuracy, precision, and other factors. We have specific scores and matrices for our model's performance. Following are the methods used to evaluate the model performance:

1. Confusion Matrix
2. Classification Report
    - Accuracy
    - Precision
    - Recall
    - F1 Score
3. ROC curve
4. AUC score

1. Confusion Matrix:

This gives us how many zeros (0s) i.e. (class = No claim) and ones (1s) i.e. (class = Yes claim) were correctly predicted by our model and how many were wrongly predicted.

		Predicted Class	
Actual class		Class = No	Class = Yes
	Class = No	True Negative	False Positive
	Class = yes	False Negative	True Positive

I. Accuracy:

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.

<u>Accuracy = (TP + TN) / (TP + TN + FP + FN)</u>

II. Precision:

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

Precision = TP/(TP + FP)

III. **Recall** (Sensitivity):

Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes.

Recall = TP/(TP + FN)

IV. F1 Score:

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. That is, a good F1 score means that you have low false positives and low false negatives, so you're correctly identifying real threats and you are not disturbed by false alarms. An F1 score is considered perfect when it's 1 , while the model is a total failure when it's 0

F1 score = 2 x [(Precision x Recall) / (Precision + Recall)]

2. ROC Curve:

ROC curve is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: True Positive Rate. False Positive Rate.

3. AUC Score:

AUC score gives the area under the ROC curve built. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative.

Employees who choose a holiday package denoted as 1 while those who do not opt are denoted as 0 in our dependent variable 'Holliday Package.' In this scenario**, True Positives** are workers who chose a vacation package and our model correctly anticipated their decision, whereas **True Negatives** are employees who did not choose a vacation package and our model correctly predicted their decision.

False positives, on the other hand, are those who did not choose a package but were predicted to do so by our model. **False Negatives**, on the other side, are those who choose a vacation package despite our model's prediction that they would not.

If an employee chose to choose a package that was not anticipated by the algorithm, the company would suffer more losses. As a result, false negatives should be kept to a minimum. As a result, **recall should be enhanced.**

False positives, on the other hand, will result in some loss. As a result, precision is important. As a result, there should be a balance between recall and precision. As a result, the **F1 score should also be considered.**

Checking the Model performance of Logistic Regression model:

Classification report:

Classification report for train data

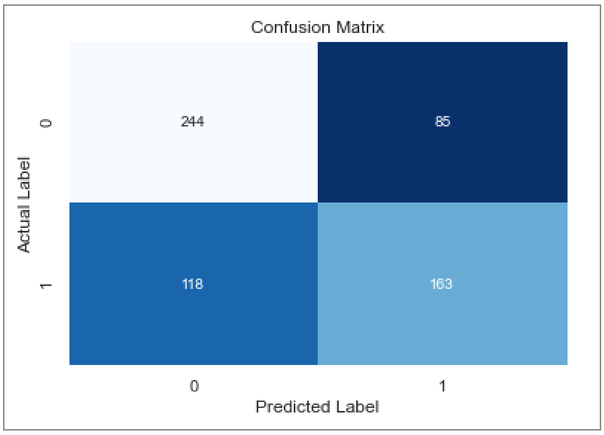
	precision	recall	f1-score	support
0	0.67	0.74	0.71	329
1	0.66	0.58	0.62	281
accuracy			0.67	610
macro avg	0.67	0.66	0.66	610
weighted avg	0.67	0.67	0.66	610

Classification report for test data

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

Figure 47. Classification report of training and testing data for Logistic Regression model

Confusion Matrix for training and testing data:



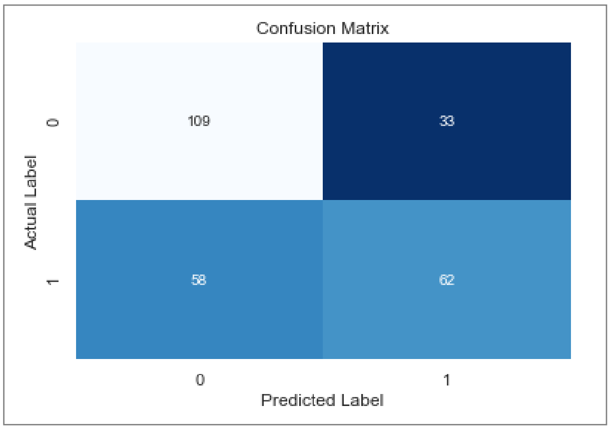


Figure 48. Confusion Matrix of train (left) and Test (right) for Logistic Regression

ROC Curve and ROC_AUC score

Figure 49. ROC curve for training and testing data for Logistic Regression

- AUC for the Training Data: 0.735
- AUC for the Test Data: 0.717

Logistic Regression Model			
Sl. No		**Train Data**	**Test Data**
1.	True Positive	163	62
2.	True Negative	244	109
3.	False Positive	85	33
4.	False Negative	118	58
5.	Accuracy	67%	65%
6.	Precision	66%	65%
7.	Recall	58%	52%
8.	F1 score	62%	58%
9.	AUC score	73.5%	71.7%

Table 5. Model Performance for Logistic Regression Model.

- Test data Accuracy, AUC, precision, and recall are nearly identical to training data and test data.
- This shows that there was neither overfitting or underfitting, and that the model is a good classification model overall.
- Overall, the metrics are high and good fit.

Inferences:

We must comprehend the meaning of False Positives and False Negatives as stated in the issue description. False positives, are those who did not choose a package but were predicted to do so by our model. False Negatives are those who choose a vacation package despite our model's prediction that they would not.

As a result, False positive impacts in small extents. False negatives will impact the firm. **Sensitivity or recall will be the important in this instance.** And also, **F1 score** should be considered.

checking the coefficients of each variable for this model:

```
The coefficient for Salary is -1.646142121152848e-05
The coefficient for age is -0.05707255243551053
The coefficient for educ is 0.06034737348280886
The coefficient for no_young_children is -1.3488352961597043
The coefficient for no_older_children is -0.04894374035375453
The coefficient for foreign is 1.2664799760127905
```

- The coefficients for no young children and foreign are the highest.
- That is, a unit change in these variables will cause the log function of the Logistic Regression model to change the most.
- With the lowest coefficient, salary is the weakest predictor.
- The coefficients for age, education, and no older children are all quite low.

Checking the Model performance of Linear discriminant analysis model:

LDA model performance based on a default cut-off probability (i.e., 0.5).

Classification report:

Classification Report of the training data:

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

Classification Report of the test data:

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

Figure 50. Classification report for LDA with default probability cut-off of 0.5

Confusion Matrix for training and testing data:

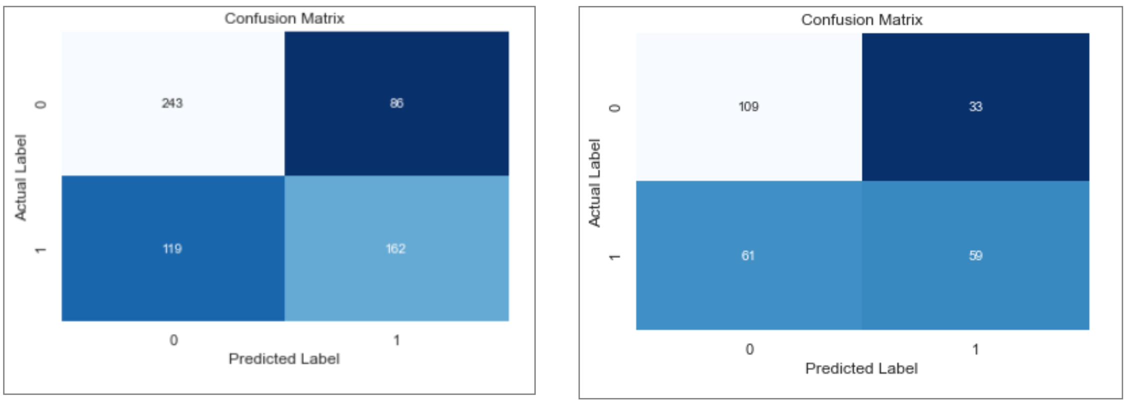


Figure 51. Confusion matrix of train (left) and test(right) for LDA :0.5

ROC Curve and ROC_AUC score:

Figure 52. ROC curve for train and test for LDA:0.5

- AUC for the Training Data: 0.733
- AUC for the Test Data: 0.714

Linear discriminant analysis model – 0.5			
Sl. No		**Train Data**	**Test Data**
1.	True Positive	162	59
2.	True Negative	243	109
3.	False Positive	86	33
4.	False Negative	119	58
5.	Accuracy	66%	64%
6.	Precision	65%	64%
7.	Recall	58%	49%
8.	F1 score	61%	56%
9.	AUC score	73.3%	71.4%

Table 6. Model performance for LDA [0.5]

- Test data Accuracy, AUC, precision, and recall are nearly identical to training data and test data.
- This shows that there was neither overfitting or underfitting, and that the model is a good classification model overall.
- Overall, the metrics are high and good fit.

The model accuracy on the training as well as the test set is about 63% and 65% respectively, which is roughly the same proportion as the class 0 observations in the dataset. This model is affected by a class imbalance problem. Since we only have 872 observations, if re-build the same LDA model with a greater number of data points, an even better model could be built.

Further changing the cut-off values for maximum recall, since recall is important and the performance of model is regularized. We saw that at probability of 0.4, the recall is increasing to greater extend and without impacting much on accuracy. At 0.4 the F1 score is also best fit. Now next checking the model performance at 0.4 and considering the best one.

LDA model performance based on a custom cut-off probability (i.e., 0.4).

Classification report:

Classification Report of the custom cut-off train data:

	precision	recall	f1-score	support
0	0.74	0.59	0.65	329
1	0.61	0.76	0.68	281
accuracy			0.67	610
macro avg	0.67	0.67	0.67	610
weighted avg	0.68	0.67	0.66	610

Classification Report of the custom cut-off test data:

	precision	recall	f1-score	support
0	0.71	0.58	0.64	142
1	0.59	0.72	0.65	120
accuracy			0.65	262
macro avg	0.65	0.65	0.64	262
weighted avg	0.66	0.65	0.64	262

Figure 53. Classification report for LDA with default probability cut-off of 0.4

Confusion Matrix for training and testing data:

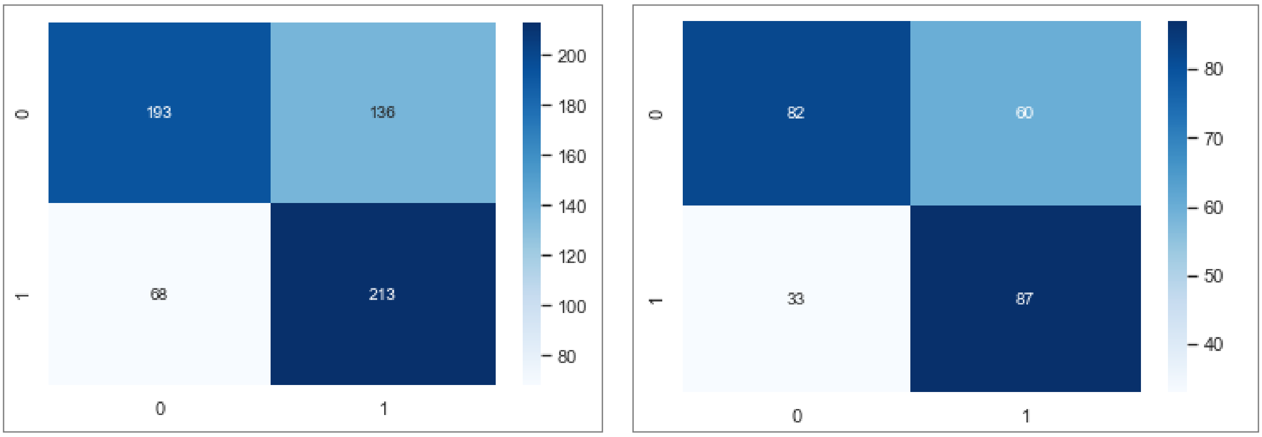


Figure 54. Confusion matrix of train (left) and test(right) for LDA :0.4

ROC Curve and ROC_AUC score:

Figure 55. ROC curve for train and test for LDA:0.4

- AUC for the Training Data: 0.733
- AUC for the Test Data: 0.714

Linear discriminant analysis model – 0.5			
Sl. No		**Train Data**	**Test Data**
1.	True Positive	213	87
2.	True Negative	193	82
3.	False Positive	68	33
4.	False Negative	136	60
5.	Accuracy	67%	65%
6.	Precision	61%	59%
7.	Recall	76%	72%
8.	F1 score	68%	65%
9.	AUC score	73.3%	71.4%

Table 7. Model performance for LDA [0.4]

- Test data Accuracy, AUC, precision, and recall are nearly identical to training data and test data.
- This shows that there was neither overfitting or underfitting, and that the model is a good classification model overall.
- Overall, the metrics are high and good fit.

We see that the Recall and F1 score is increased to greater extend in the custom probability cut-off of 0.4. This is our best fit model for the LDA. Considering this model for further comparison of Logistic Regression model and LDA to check the best fit model for the firm.

Checking the coefficients of each variable for this model:

```
The coefficient for Salary is -1.3803065402589292e-05
The coefficient for age is -0.05779485342767467
The coefficient for educ is 0.058604307804757796
The coefficient for no_young_children is -1.282791270742752
The coefficient for no_older_children is -0.03756728141585798
The coefficient for foreign is 1.3206019493992331
```

We see a similar result with no_young_children and foreign as good predictors and salary being the worst predictor.

Comparison of performance metrics between models:

So far, we've developed models for Logistic regression and Linear discriminant analysis, and we've used a confusion matrix, classification report, AUC scores, and ROC curves to evaluate their performance. Now we'll compare the models based on their results to see which one is best for classification.

As previously stated, **recall value is quite important for our problem statement. To some extent, precision is also vital. The F1 score and recall value should be concentrated.**

For model comparison of Logistic Regression, the best fit model after applying grid search is used and for Liner discriminant analysis, we saw that the custom probability cut-off of 0.4 is giving the better results, so the best model of custom probability performance is considered for model comparison.

Table 8. Metrices comparison table between models.

	Logistic reg Train	Logistic reg Test	LDA Train	LDA Test
Accuracy	0.67	0.65	0.67	0.64
AUC	0.74	0.72	0.73	0.71
Recall	0.58	0.52	0.76	0.72
Precision	0.66	0.65	0.61	0.59
F1 Score	0.62	0.58	0.68	0.65

In this table, we have Accuracy, Recall, Precision, F1 score and AUC scores for 2 different models. The models are as follows:

1) **Logistic Regression Model** – Best fit model after grid search.
2) **LDA with custom cut- off probability (0.4).**

Inferences:

- We can see that, the Accuracy for both models for both train and test is almost similar.
- The AUC and Precision of Logistic is slightly greater than the LDA for both test and train
- However, for our model, Recall and F1 score being the important measure of model performance, we can see that LDA model is performing much better compared to Logistic regression model. We can say that LDA is best fit model.
- **Linear discriminant analysis model with custom probability of 0.4 is the best fit model.**

Comparing the ROC curves and AUC scores for LDA and Logistic Regression models.

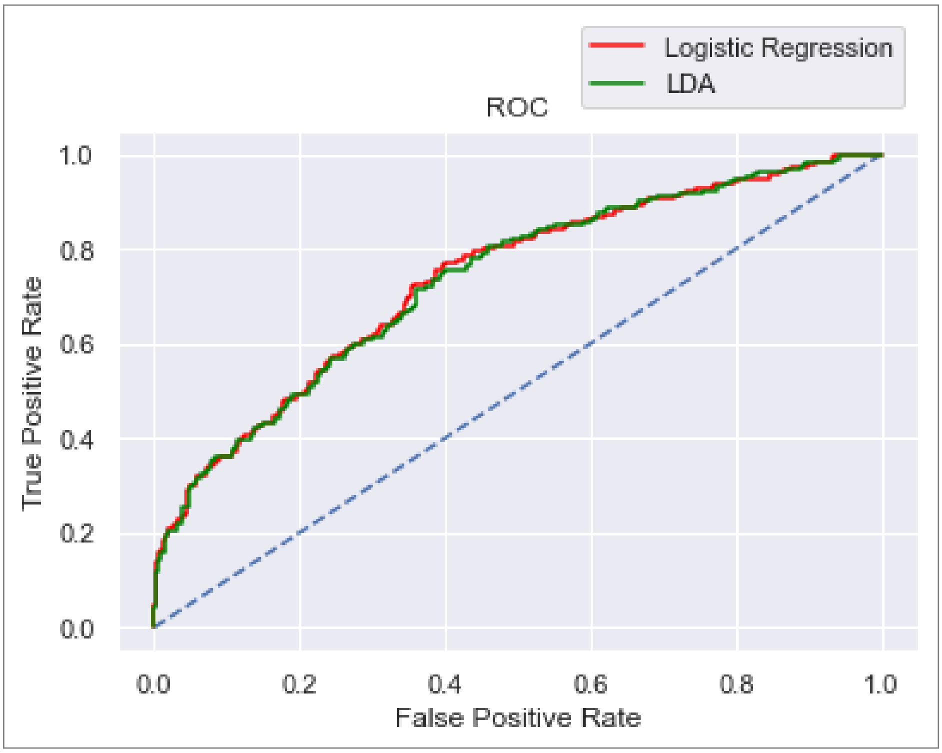


Figure 56. ROC of model comparison for Train data.

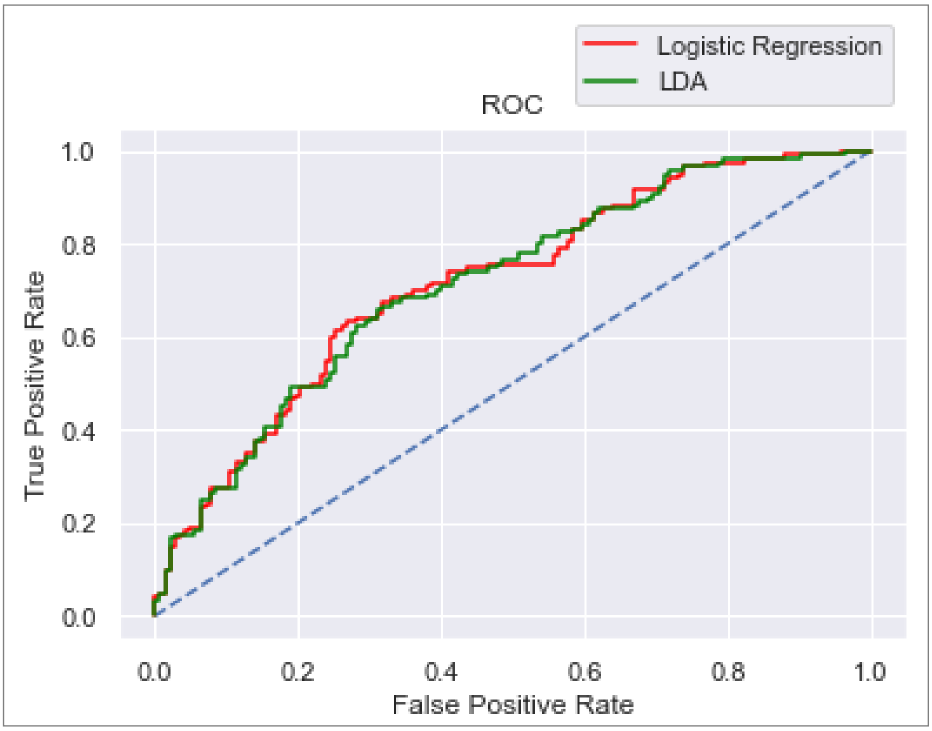


Figure 57. ROC of model comparison for Test data.

We can see from the graphs that Logistic Regression and LDA perform approximately identically for both the train and test data sets. The logistic regression model, on the other hand, performs somewhat better ROC.

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

We had a business problem where we need to predict whether an employee would opt for a holiday package or not. For this problem we had predicted the results using both logistic regression and linear discriminant analysis.

In our extensive analysis so far, we have thoroughly examined given data and developed a model that predicts the classification of whether the employee opts for holiday package or no, based on the attributes in our dataset. Let us now look at the key points in our past data first and try to suggest some recommendations for the firm.

Insights from the Graphs and Analysis from EDA:

Holiday package:

- We can observe that 54% of the employees are not opting for the holiday package and 46% are interested in the package. This implies we have a dataset which is fairly balanced.

Salary

- The average 'Salary' of employees opting for holiday package and not opting for holiday package is similar in nature.
- The coefficient for Salary is -1.3803 e-05. There is almost no relation with the Holiday package, so we can say that Salary is not a good predictor for model building.
- Higher salary employees are more prone to not opt for holiday package.

Foreign

- Foreign is a good predictor of dependent variable with a high positive coefficient.
- The frequency distribution of foreign implies that the employees are mostly from the same country which is around 75% of employees and foreigners are around 25% of them.
- We can see that the percentage of foreigners accepting the holiday package is substantially higher compared to the citizens while considering the ratio of foreigners and the citizens.
- The mean salary of foreign people is slightly less than natives.

Age

- We can see that, the age distribution for employees who are opting for holiday package and not opting are similar in nature, though the number of people opting are less in number and mostly fall in range of 35-45 age group.
- We can see that, employees in middle range (34 to 45 years) are opting for holiday package are more as compared to older and younger employees.

Education

- The variable 'educ' the number of years of formal education is showing a similar pattern. This means education is likely not a variable that influences for opting of holiday packages for employees.
- We can see that employee with less years of formal education (1 to 7 years) and higher education are not opting for the Holiday package as compared to employees with formal education of 8 year to 12 years
- Across education we can observe that the employees with higher number of years of formal education have a lower tendency to opt for the holiday package relative to employees with lesser years of formal education

No. of young children

- No_young_children have a -1.29 approximately coefficient. This can be treated as a good predictor of dependent variable.
- We can see that there is a significant difference in employees with younger children who are opting for holiday package and employees who are not opting for holiday package, this attribute is good predictor as there is significant difference in them.
- We can see that people with younger children are opting for holiday packages are very few in number compared to employees who do not have young children.

No. of older children

- The distribution for opting or not opting for holiday packages looks same for employees with older children. At this point, this might not be a good predictor for model building.
- Almost same distribution for both the scenarios when dealing with employees with older children
- For the employees with older children, it’s hard to differentiate between the 2 different classes of dependent variable. The employees who opt for package and the ones who do not do not have much difference between them.
- This is not a good variable for model building.

Recommendations:

- The firm should concentrate its efforts on foreigners in order to increase sales of vacation packages, as this is where the majority of conversions will occur.
- The firm might try to target their marketing efforts or offers at foreigners in order to increase the number of people who choose vacation packages.
- Focus on Foreign variable for good prediction while building the classification model.
- To improve the likelihood of lower-wage employees selecting for a vacation package, the firm might provide certain incentives or discounts to them.
- The company should not target employees with younger children. The employees with younger children have more chances of not opting for holiday package.
- Employees with older children who do not opt for vacation package might be targeted using some marketing strategies. The organisation can conduct a deep dive or conduct a survey to determine why the rest of the employees are not taking advantage of the holiday package. The corporation may be able to come up with some suggestions or offers to convert the remaining employees.
- The employer can provide references of workers with older children who have chosen the package to those who have not chosen it, in order to persuade them to do so.

Key performance indicators:

- Highlight the benefits of Holiday package and services and educate the employees about it.
- Company can come up with lucrative enchantments in holiday packages
- Customer satisfaction should be utmost priority.
- Engage with employees through social media.
- New destinations can be added.
- Video is a great way to engage and inspire potential travellers.
- Travel influencers can promote destinations, activities, and businesses by using their social media influence.
- Get feedback from employees who took the holiday package and work on the betterment of package accordingly.
