Great Learning

PREDICTIVE MODELLING

BUSINESS REPORT

Submitted by:

N. Aishwarya
PGP-DSBA

May 2022

Table of Contents

List of Figures

List of Tables

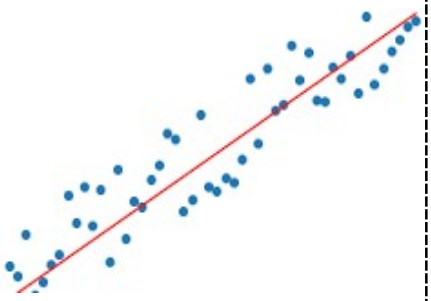
List of Formulas

Executive Summary – Predictive Modelling:

Predictive modeling is a commonly used statistical technique to predict future behavior. Predictive models analyze past performance to assess how likely a customer is to exhibit a specific behavior in the future.

Predictive modeling can be used to predict just about anything, from TV ratings and a customer's next purchase to credit risks and corporate earnings. It's one of the premier ways a business can see its path forward and make plans accordingly. This method tends to have high accuracy rates, which is why it is so commonly used.

Business problem 1 - Linear Regression



Problem Statement:

You are hired by a company Gem Stones co ltd, which is a cubic zirconia manufacturer. You are provided with the dataset containing the prices and other attributes of almost 27,000 cubic zirconia (which is an inexpensive diamond alternative with many of the same qualities as a diamond).

The company is earning different profits on different prize slots. You have to help the company in predicting the price for the stone on the bases of the details given in the dataset so it can distinguish between higher profitable stones and lower profitable stones so as to have better profit share. Also, provide them with the best 5 attributes that are most important.

Solution Approach:

The purpose of the solutioning exercise is to explore the dataset using Predictive analysis techniques to arrive at the customer segmentation, thus enabling business strategies customized to them. Below is the data dictionary for Linear Regression problem:

Variable Name	Description
Carat	Carat weight of the cubic zirconia.
Cut	Describe the cut quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal.
Color	Colour of the cubic zirconia. With D being the worst and J the best.
Clarity	Clarity refers to the absence of the Inclusions and Blemishes. (In order from Worst to Best in terms of avg price) IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1

Depth	The Height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter.
Table	The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter.
Price	the Price of the cubic zirconia.
X	Length of the cubic zirconia in mm.
Y	Width of the cubic zirconia in mm.
Z	Height of the cubic zirconia in mm.

Analysis 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.

Solution:

Firstly, import the necessary libraries required for the problem in the Jupiter Notebook file and run them. Read the " cubic_zirconia.csv " file for EDA.

- **Head of the data is obtained as below:**

	Unnamed: 0	carat	cut	color	clarity	depth	table	x	y	z	price
0	1	0.30	Ideal	E	SI1	62.1	58.0	4.27	4.29	2.66	499
1	2	0.33	Premium	G	IF	60.8	58.0	4.42	4.46	2.70	984
2	3	0.90	Very Good	E	VVS2	62.2	60.0	6.04	6.12	3.78	6289
3	4	0.42	Ideal	F	VS1	61.6	56.0	4.82	4.80	2.96	1082
4	5	0.31	Ideal	F	VVS1	60.4	59.0	4.35	4.43	2.65	779
5	6	1.02	Ideal	D	VS2	61.5	56.0	6.46	6.49	3.99	9502
6	7	1.01	Good	H	SI1	63.7	60.0	6.35	6.30	4.03	4836
7	8	0.50	Premium	E	SI1	61.5	62.0	5.09	5.06	3.12	1415
8	9	1.21	Good	H	SI1	63.8	64.0	6.72	6.63	4.26	5407
9	10	0.35	Ideal	F	VS2	60.5	57.0	4.52	4.60	2.76	706

Table 1: Description of Cubic Zirconia data

- **Output from shape command:**

No. of rows: 26967
No. of columns: 11

- **List of fields retrieval along with their data type:**

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

Table 2: Information of Cubic Zirconia data

Observation-1:

- The data set contains 26967 row, 11 columns.
- In the given data set, there are 2 Integer type features,6 Float type features. 3 Object type features. Where 'price' is the target variable and all other are predictor variable.
- The first column is an index ("Unnamed: 0")as this only serial no, we can remove it.
- Except depth, in all the column non-null count is 26967.

Output for missing values:

Unnamed	0
carat	0
cut	0
Color	0
clarity	0
depth	697
table	0
x	0
y	0
z	0
price	0

Observation-1:

- The data set contains 26967 row, 11 columns.
- In the given data set, there are 2 Integer type features,6 Float type features. 3 Object type features. Where 'price' is the target variable and all other are predictor variable.
- The first column is an index ("Unnamed: 0")as this only serial no, we can remove it.
- Except depth, in all the column non-null count is 26967.
- Depth variable has 697 missing values.
- We can also see there are no duplicates in the dataset.

Summary of the data, providing descriptive statistical variables:

Descriptive statistics help describe and understand the features of a specific data set by giving short summaries about the sample and measures of the data. The most recognized types of descriptive statistics are measures of center: the mean, median, and mode, which are used at almost all levels math and statistics.

We will drop the first column 'Unnamed: 0' column as this is not important for our study.

	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

Table 3: Summary Statistic of Cubic Zirconia data

Summary statistics info:

- **Carat**- This is an independent variable, and it ranges from 0.2 to 4.5. mean value is around 0.8 and 75% of the stones are of 1.05 carat value. Standard deviation is around 0.477 which shows that the data is skewed and has a right tailed curve. Which means that majority of the stones are of lower carat. There are very few stones above 1.05 carat.
- **Depth** - The percentage height of cubic zirconia stones is in the range of 50.80 to 73.60. Average height of the stones is 61.80 25% of the stones are 61 and 75% of the stones are 62.5. Standard deviation of the height of the stones is 1.4. Standard deviation is indicating a normal distribution.
- **Table** - The percentage width of cubic Zirconia is in the range of 49 to 79. Average is around 57. 25% of stones are below 56 and 75% of the stones have a width of less than 59. Standard deviation is 2.24. Thus, the data does not show normal distribution and is similar to carat with most of the stones having less width also this shows outliers are present in the variable.

- **Price** - Price is the Predicted variable. Prices are in the range of 3938 to 18818. Median price of stones is 2375, while 25% of the stones are priced below 945. 75% of the stones are in the price range of 5356. Standard deviation of the price is 4022. Indicating prices of majority of the stones are in lower range as the distribution is right skewed.

Summary statistics of the object variable:

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

Table 4: Summary Statistic of Object Variables

Observation-2:

- On the given data set the Mean and Median values does not have much difference.
- We can observe Min value of "x", "y", & "z" are zero this indicates that they are faulty values. As we know dimensionless or 2-dimensional diamonds are not possible. So, we need to filter out those as it clearly faulty data entries.
- There are three object data type 'cut', 'color' and 'clarity'.

Data Visualization:

Univariate analysis:

Helps us to understand the distribution of data in the dataset. With univariate analysis we can find patterns and we can summarize the data for.

1. CARAT VARIABLE:

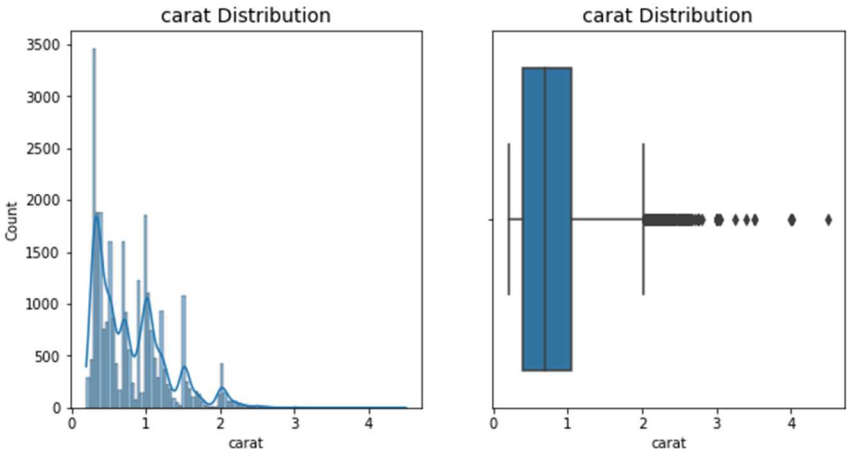


Figure 1: Univariate analysis of Carat variable

Observations:

- For Univariate Analysis of Carat, we are using histplot and boxplot to find information or patterns in the data.
- The Boxplot of Carat variable seems to have outliers.
- The distribution of the data is right skewed.

2. DEPTH VARIABLE:

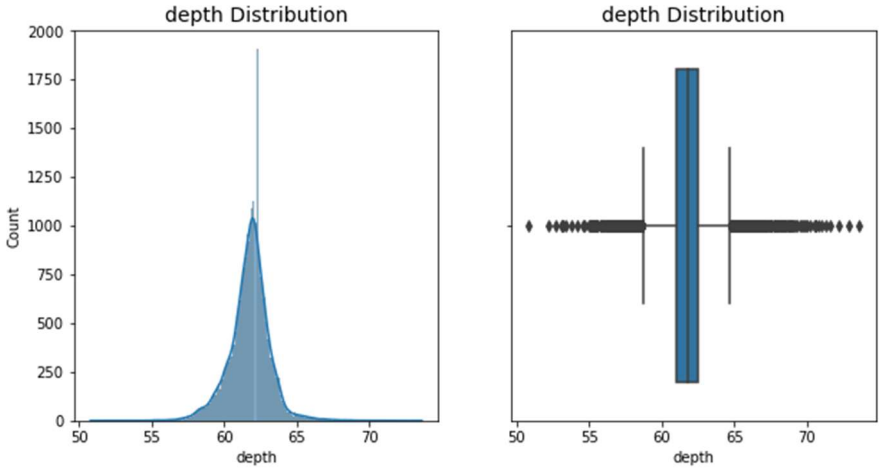


Figure 2: Univariate analysis of Depth Variable

Observations:

- The Boxplot of Depth variable seems to have outliers.
- The distribution of the data is highly skewed and normally distributed.

3. TABLE VARIABLE:

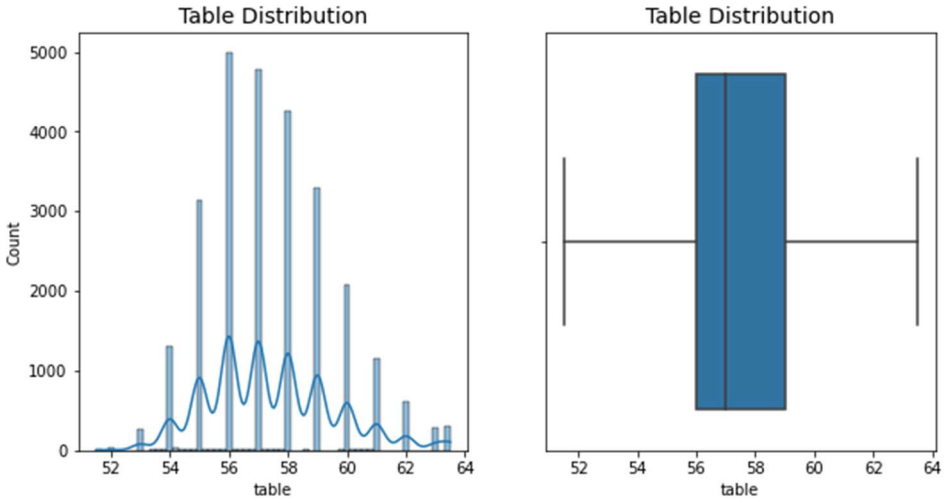


Figure 3: Univariate analysis of Table Variable

Observations:

- The Boxplot of Table variable seems to have no outliers.
- The distribution of the data is right skewed and normally distributed.

4. X VARIABLE:

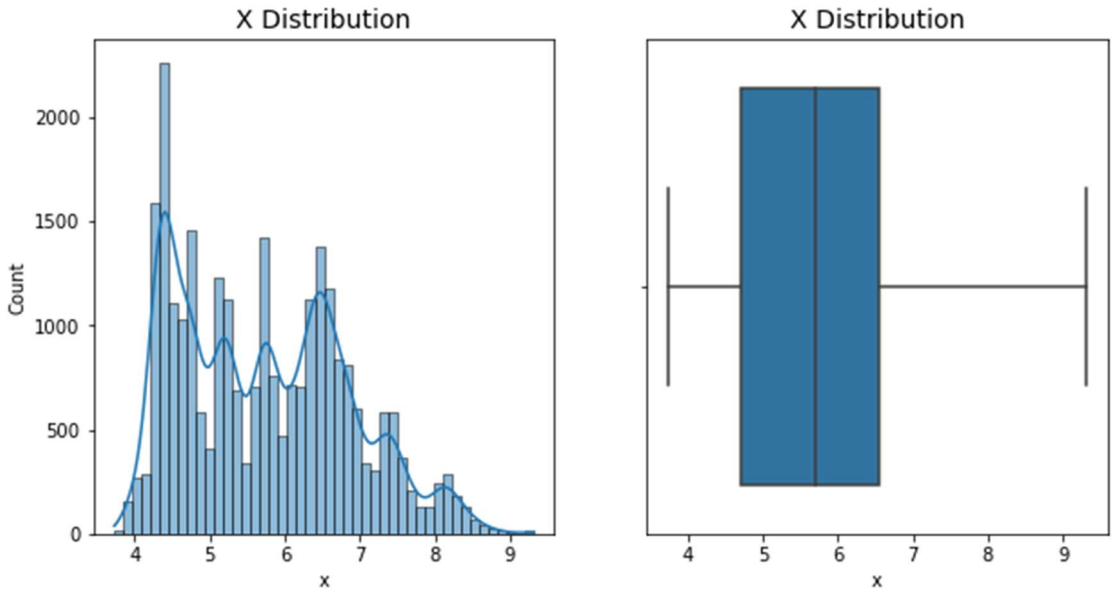


Figure 4: Univariate analysis of X Variable

Observations:

- The Boxplot of X variable seems to have no outliers.
- The distribution of the data is right skewed.

5. Y VARIABLE:

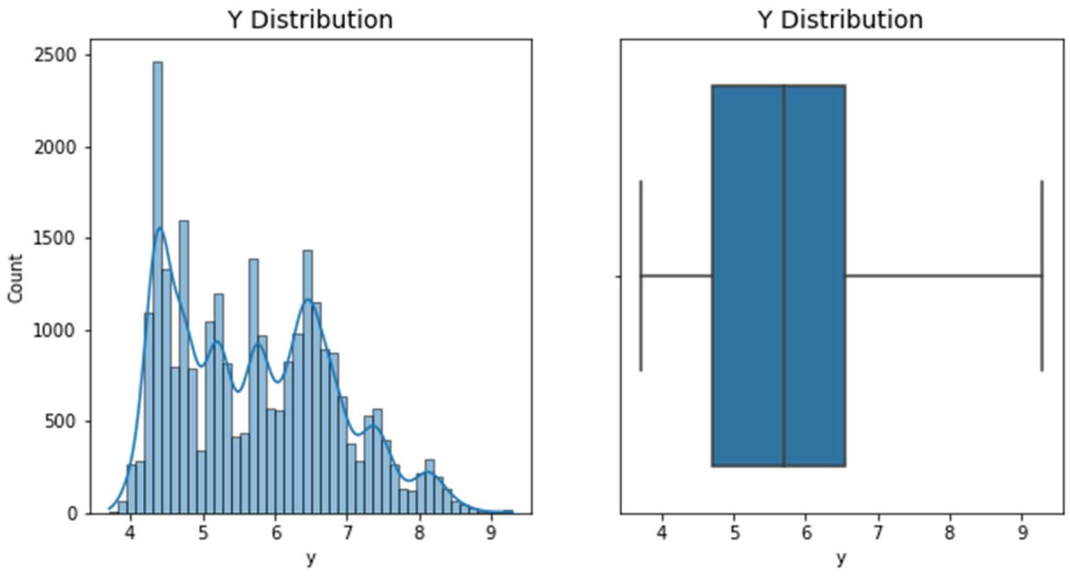


Figure 5: Univariate analysis of Y Variable

Observations:

- The Boxplot of Y variable seems to have no outliers.
- The distribution of the data is right skewed.

6. Z VARIABLE:

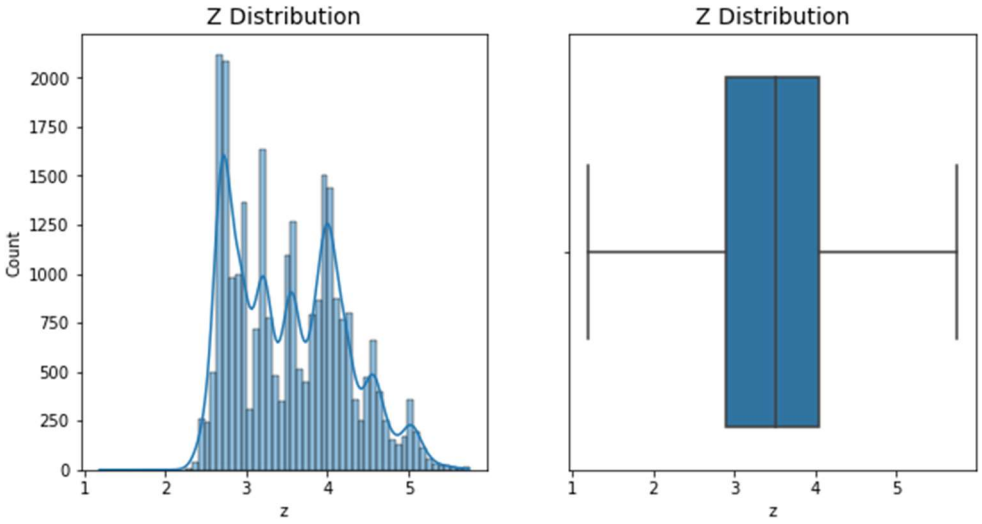


Figure 6: Univariate analysis of Z Variable

Observations:

- The Boxplot of Z variable seems to have no outliers.
- The distribution of the data is right skewed.

7. PRICE VARIABLE:

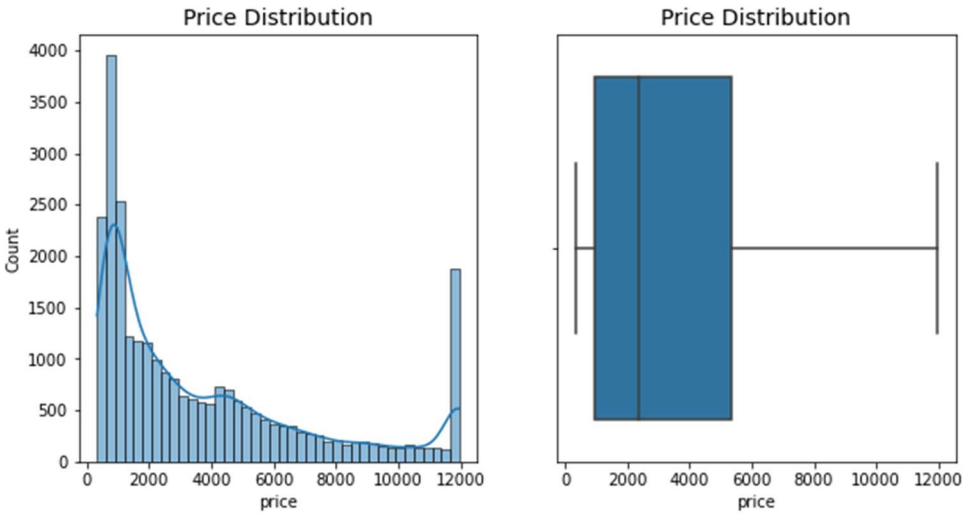
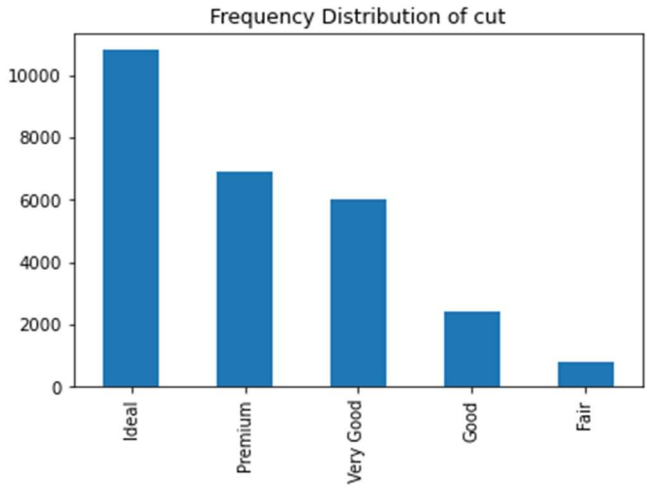


Figure 7: Univariate analysis of Price Variable

Univariate analysis of categorical variables:

1. CUT VARIABLE:



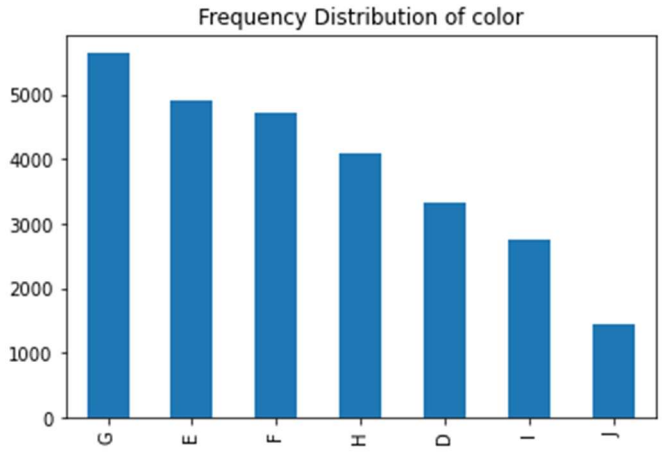

```
Details of cut
-------------------------
Ideal         10805
Premium        6880
Very Good      6027
Good           2434
Fair            779
Name: cut, dtype: int64
```

Looking at the above unique values for variable “Cut “ we see the ranking given for each unique value like “ Fair, Good, Ideal, Premium, Very Good “.

Observations:

- For the cut variable we see the most sold 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.

2. COLOR VARIABLE:

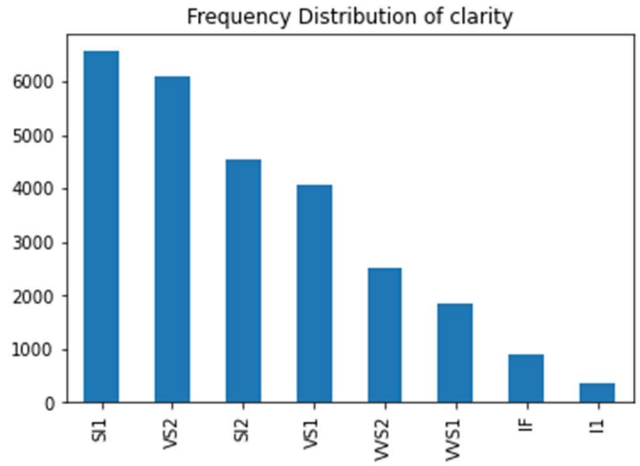



```
Details of color
----------------------------
G    5650
E    4916
F    4722
H    4091
D    3341
I    2765
J    1440
Name: color, dtype: int64
```

Observations:

- 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.

3. CLARITY VARIABLE:




```
Details of clarity
----------------------------
SI1     6564
VS2     6092
SI2     4561
VS1     4086
VVS2    2530
VVS1    1839
IF       891
I1       362
Name: clarity, dtype: int64
```

Observations:

- The Diamonds clarity with VS1 & VS2 are the most Expensive.
- All clarity type gems except VS1 & VS2 have outliers with respect to price.
- Slightly less priced seems to be I1 type; VS2 and SI1 clarity stones seems to be more expensive.

Histogram plot:

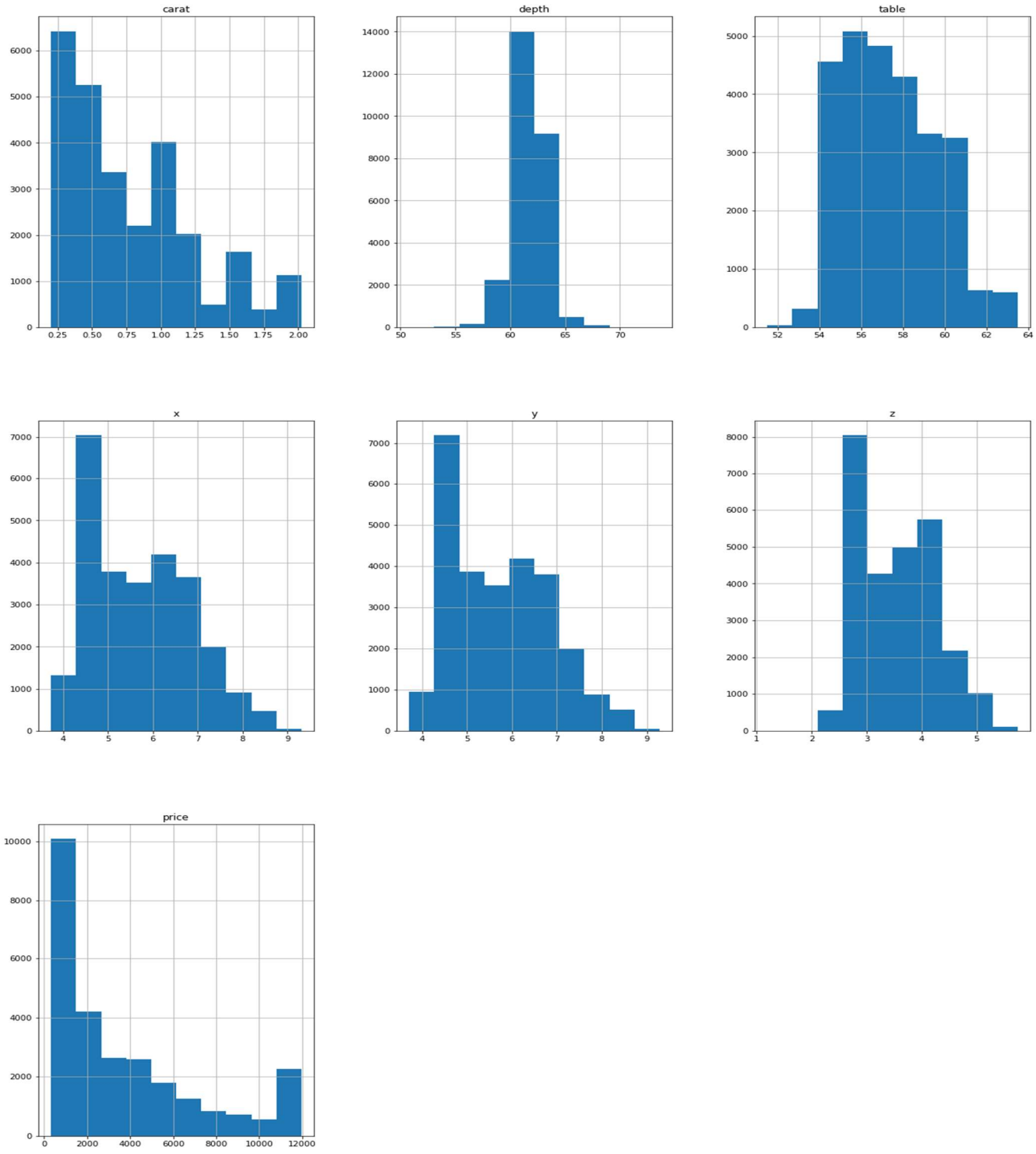


Figure 8: Histogram plot of various variables

Observations:

Independent Variables:

- Depth is the only variable which can be considered as normal distribution.
- Carat, Table, x, y, z these variables have multiple modes with the spread of data.

- Outliers: Large number of outliers are present in all the variables (Carat, Depth, Table, x, y, z).
- Except for carat and price variable, all other variables have mean and median values very close to each other, seems like there is no skewness in these variables. Whereas for carat and price we see some difference in value of mean and median, which slightly indicates existence of some skewness in the data.

Price will be the target variable or dependent variable.

- It is right skewed with large range of outlier.

Skewness values:

Formula 1: **Skewness = 3 * (Mean – Median) / Standard Deviation.**

Skewness values of 7 independent variables for our dataset is given below:

Price	1.157121
Carat	0.917214
Table	0.480476
X	0.397696
Z	0.394819
Y	0.394060
Depth	-0.025042

Table 5: Skewness value between 7 independent variables

Bivariate Analysis:

The Pair Plot helps us to understand the relationship between all the numerical values in the dataset. On comparing all the variables with each other we could understand the patterns. The pair plot function in seaborn makes it very easy to generate joint scatter plots for all the columns in the data.

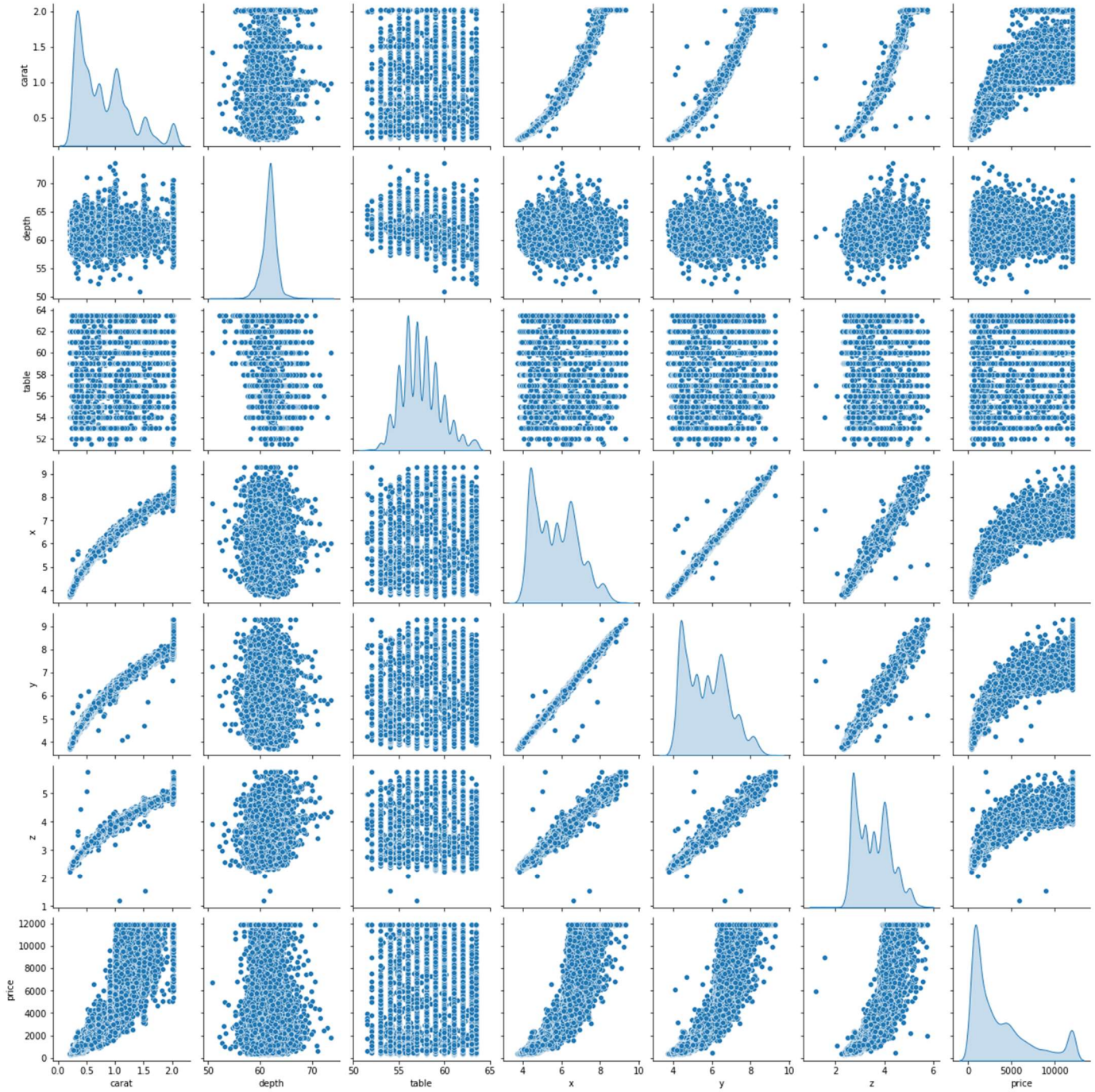


Figure 9: Pair Plot of independent variables

Correlation heatmap:

Correlation is a statistical measure that expresses the extent to which two variables are linearly related.

Formula 2: Correlation

$$Correlation = \frac{Cov(x, y)}{\sigma_x * \sigma_y}$$

Where,

$Cov(x, y)$ = Covariance of x and y
$\sigma_x =$ Standard deviation of x
$\sigma_y =$ Standard deviation of y

Correlation matrix:

A correlation matrix displays the correlation coefficients for different variables. The matrix depicts the correlation between all the possible pairs of values in a table. The correlation matrix for the given set of variables is as follows:

	carat	depth	table	x	y	z	price
carat	1.000	0.034	0.187	0.983	0.982	0.981	0.937
depth	0.034	1.000	-0.294	-0.019	-0.022	0.101	0.000
table	0.187	-0.294	1.000	0.200	0.194	0.161	0.138
x	0.983	-0.019	0.200	1.000	0.998	0.991	0.913
y	0.982	-0.022	0.194	0.998	1.000	0.991	0.915
z	0.981	0.101	0.161	0.991	0.991	1.000	0.909
price	0.937	0.000	0.138	0.913	0.915	0.909	1.000

Figure 10: Correlation matrix

Correlation values:

Price	1.000000
Carat	0.936765
Y	0.914838
X	0.913409
Z	0.908599
Table	0.137915
Depth	0.000313

Table : feature affecting the price of diamonds.

Below is the heatmap output from Python, for the 7 variables -

Figure 11: Correlation heatmap

Observations:

- We can identify that there is a strong correlation between independent variables –like Carat, x, y, and z. All these variables are strongly correlated with the target variable – price.
- This indicates a strong case of our dataset struggling with multicollinearity.
- Depth does not show any strong relation with the price variable. For this case study, I would drop x, y, and z variables before creating the linear regression model.
- Similarly, Depth does not seem to be influencing my variable price and hence, at some point, I will be dropping this variable from my model building process as well.
- It can be inferred that most features correlate with the price of Diamond. The notable exception is "depth" which has a negligible correlation (<1%).

The inferences drawn from the above Exploratory Data analysis:

Observations-1:

- 'Price' is the target variable while all others are the predictors.
- The data set contains 26967 row, 11 column.
- In the given data set, there are 2 Integer type features,6 Float type features. 3 Object type features. Where 'price' is the target variable and all other are predictor variable.
- The first column is an index ("Unnamed: 0")as this only serial no, we can remove it.

Observation-2:

- On the given data set the mean and median values does not have much difference.
- We can observe Min value of "x", "y", "z" is zero this indicates that they are faulty values. As we know dimensionless or 2-dimensional diamonds are not possible. So, we have filter out those as it clearly faulty data entries.
- There are three object data type 'cut', 'color' and 'clarity'.

Observation-3:

- We can observe there are 697 missing values in the depth column. There are some duplicate rows presents (33 duplicate rows out of 26958). which is nearly 0.12 % of the total data. So, on this case we have dropped the duplicated row.

Observation-4:

- There is significant amount of outlier present in some variable, the features with datapoint that are far from the rest of dataset which will affect the outcome of our regression model. So, we have treated the outlier. We can see that the distribution of some quantitative features like "carat" and the target feature "price" are heavily "right-skewed".

Observation-5:

- It looks like most features do correlate with the price of Diamond. The notable exception is "depth" which has a negligible correlation (~1%). Observation on 'CUT': The Premium Cut on Diamonds are the most Expensive, followed by Very Good Cut.

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

Solution:

We have already check for 'Zero' value. and we can observe there are some amounts of 'Zero' value present on the data set on variable 'x', 'y', 'z'. This indicates that they are faulty values.

As we know dimensionless or 2-dimensional diamonds are not possible. So, we have filter out those as it clearly faulty data entries.

Values which are equal to zero:

```
Number of rows with x == 0: 3
Number of rows with y == 0: 3
Number of rows with z == 0: 9
Number of rows with depth == 0: 0
```

After removing 'zero value' from data set the data shape became as follows.

Number of Rows: 26925
Number of Columns : 10

There are missing values in the column “depth” – 697 cells or 2.6% of the total data set. We can choose to impute these values using a mean or median. We checked for both the values and the result for both is almost similar. For this case study, I have used median to impute the missing values.

Missing values after imputing the values:

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

Price Distribution of Cut Variable

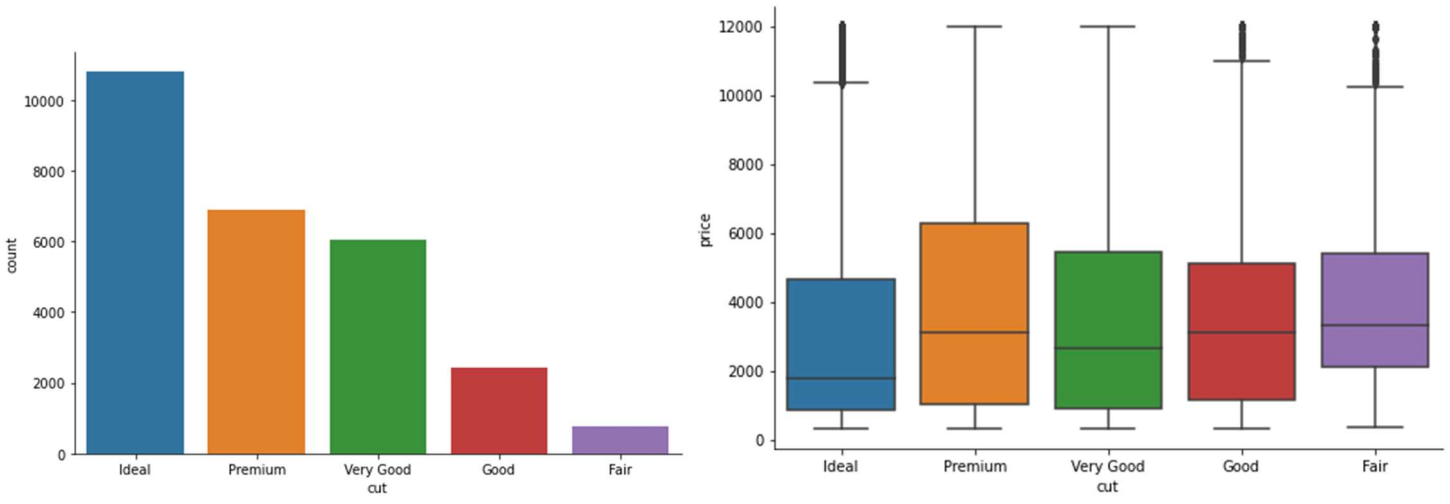


Figure 12: Price Distribution of Cut Variable

Observation on 'CUT':

- The Premium Cut on Diamonds are the most Expensive, followed by Very Good Cut.
- For the cut variable we see the most sold is Ideal cut type gems and least sold is Fair cut gems
- All cut type gems have outliers expect Premium and Very Good with respect to price.
- Slightly less priced seems to be Ideal type and Premium cut type to be slightly more expensive.

Price Distribution of Color Variable:

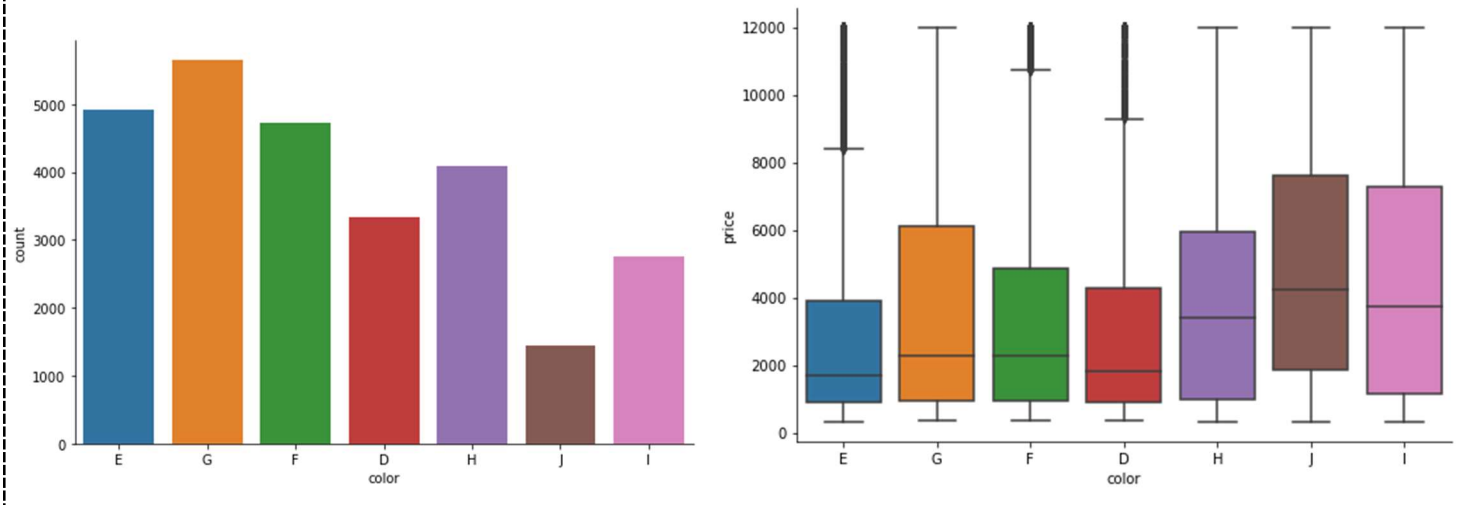


Figure 13: Price Distribution of Color Variable

Observation on 'Color':

- 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

Price Distribution of Clarity Variable:

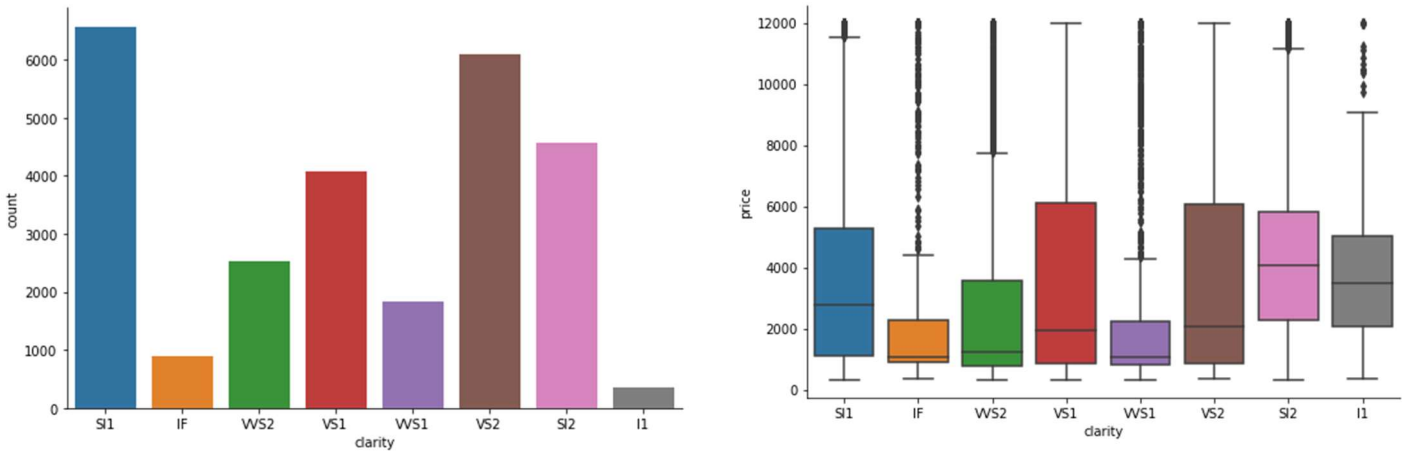


Figure 14: Price Distribution of Clarity Variable

Analysis 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.

Solution:

Encoding the data (having string values):

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

Converting the'cut','color','clarity' column from object / string type to float we get:

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

We can establish that there is an order in ranking such as mean price is increasing from ideal then good to very good, premium and fair. Fair segment has the highest median value as well.

Train-Test Split:

First, copy all the predictor variables into X data frame and copy target into the y data frame. Let us break the X and y data frames into training set and test set in 70:30 ratio. For this we will use “Sklearn package's data splitting function” which is based on random function. This is the dependent variable. We will get the following output:

	carat	cut	color	clarity	depth	table	x	y	z
0	0.30	4.0	5.0	2.0	62.1	58.0	4.27	4.29	2.66
1	0.33	3.0	3.0	7.0	60.8	58.0	4.42	4.46	2.70
2	0.90	2.0	5.0	5.0	62.2	60.0	6.04	6.12	3.78
3	0.42	4.0	4.0	4.0	61.6	56.0	4.82	4.80	2.96
4	0.31	4.0	4.0	6.0	60.4	59.0	4.35	4.43	2.65

We Will invoke the Linear Regression function and find the best fit model on training data.

Coefficients for each of the independent attributes:

Y=mx +c (m= m1, m2, m3...m9) here 9 different co-efficient will learn along with the intercept which is "c" from the model.

We can conclude that:

- The one unit increase in carat increases price by 8901.941.
- The one unit increase in cut increases price by 109.188.
- The one unit increase in color increases price by 272.921.
- The one unit increase in clarity increases price by 436.441.
- The one unit increase in y increases price by 1464.827.
- The one unit increase in depth increases price by 8.236,
- But the one unit increase in table decreases price by -17.345,
- The one unit increase in x decreases price by -1417.908,
- The one unit increase in z decreases price by -711.225.

The intercept for the model:

The intercept for our model is -3171.950447307667

Observation:

- The intercept (often labelled the constant) is the expected mean value of Y when all X=0. If X never equals 0, then the intercept has no intrinsic meaning.
- The intercept for our model is -3171.950. In preset case when the other predictor variable is zero i.e like carat, cut, color, clarity all are zero then the C=-3172. that means price is -3172. which is meaningless. We can do Z score or scaling the data and make it nearly zero.

Formula 3: Linear Regression :

$$Y = m_1X_1 + m_2X_2 + \cdots + m_nX_n + C + e$$

Y = Dependent / target / predicted variable
X_i= Independent/ predictor variable
m_i= coefficients for the i^{th} independent / predictor variable.
C = constant / intercept / bias
e = residual error / unexplained variance / difference between actual and prediction.

R square Method:

Formula 4 : R^2

$$R^2 = 1 - \frac{SSE}{SST}$$

Where, $SST = \sum_{i=1}^{n}(Y_i - \bar{Y})^2$ and $SSE = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$

R square on training data:

Model score on the training data set is 0.9312

R square on testing data:

Model score on the training data set is 0.9315

Observation:

R-square is the percentage of the response variable variation that is explained by a linear model.

R-squared is always between 0 and 100%: 0% indicates that the model explains none of the variability of the response data around its mean. 100% indicates that the model explains all the variability of the response data around its mean.

In this regression model we can see the R-square value on Training and Test data respectively 0.9312 and 0.9315.

Root mean square error (RMSE) Method:

Formula 5: RMSE $RMSE = \sqrt{\left[\frac{(e_1^2+e_2^2+\cdots+e_n^2)}{n}\right]}$

Where $e_i = y_i - \hat{y}_i$

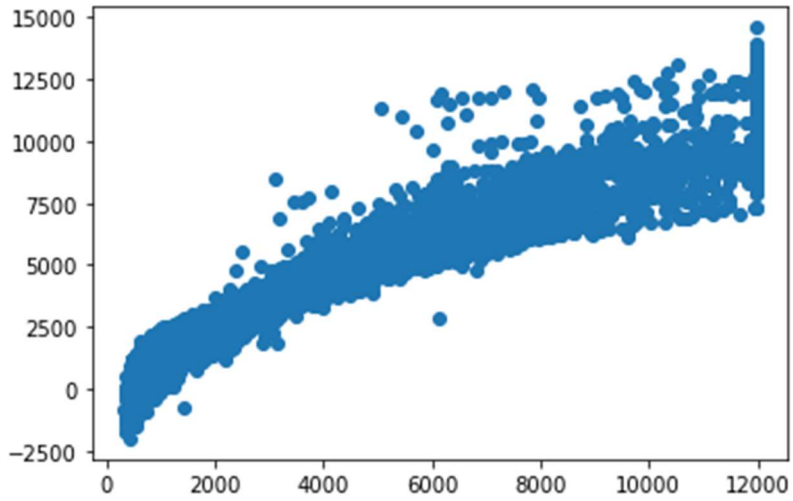
RMSE on Training data:

RMSE on Training data: 907.1312

RMSE on Testing data:

RMSE on Testing data: 911.8447

Plot of predicted y value vs actual y values for the test data:



Observation:

- We can see that above is a linear plot, strong correlation between the predicted y and actual y. But there are lots of spread. That indicated some kind noise present on the data set i.e Unexplained variances on the output.

Linear regression Performance Metrics:

Intercept for the model: -3171.9504

R square on training data: 0.9312

R square on testing data: 0.9315

RMSE on Training data: 907.1312

RMSE on Testing data: 911.8447

As the training data & testing data score are almost inline, we can conclude this model is a Right-Fit Model.

Applying Z score Stats models:

The coefficients for each of the independent attributes:

```
The coefficient for carat is 1.1837737061779434
The coefficient for cut is 0.035125000655297195
The coefficient for color is 0.13449269287641535
The coefficient for clarity is 0.20809779325621866
The coefficient for depth is 0.003326293718838773
The coefficient for table is -0.010815851633643403
The coefficient for x is -0.4596898424125279
The coefficient for y is 0.47166270917924047
The coefficient for z is -0.14249737973827098
```

Observations:

Now we can observe by applying z score the intercept for our model has become -5.87961525130473e-16. Earlier it was -3171.950447307667. The coefficient of determinant is 0.9315 which is changed, the bias became nearly zero but the overall accuracy still same.

Multi-collinearity using Variance Inflation Factor (VIF):

The variance inflation factor(VIF) identifies correlation between independent variables and the strength of that correlation.

Formula 6: VIF $$\mathbf{VIF} = \frac{1}{1-r^2}$$

We can observe there are very strong multi collinearity present in the data set. Ideally it should be within 1 to 5. We are exploring the Linear Regression using Stats models as we are interested in some more statistical metrics of the model.

```
carat ---> 121.96543302739589
cut ---> 10.388738909800345
color ---> 5.546407587131625
clarity ---> 5.455999699082339
depth ---> 1218.3824913329145
table ---> 878.3985698779234
x ---> 10744.05623520385
y ---> 9482.053091580401
z ---> 3697.5688286012546
```

Formula 7: Adj R2

$$\mathbf{Adj\ R^2} = 1 - \frac{\mathbf{SSE}/(n-p)}{\mathbf{SST}/(n-1)}$$

Linear Regression using Stats models:

Concatenating X and y into a single data frame, we get

	carat	cut	color	clarity	depth	table	x	y	z	price
5030	1.10	1.0	5.0	1.0	63.3	56.0	6.53	6.58	4.15	4065.0
12108	1.01	2.0	6.0	1.0	64.0	56.0	6.30	6.38	4.06	5166.0
20181	0.67	1.0	1.0	3.0	60.7	61.4	5.60	5.64	3.41	1708.0
4712	0.76	1.0	3.0	2.0	57.7	63.0	6.05	5.97	3.47	2447.0
2548	1.01	3.0	3.0	4.0	62.8	59.0	6.37	6.34	3.99	6618.0

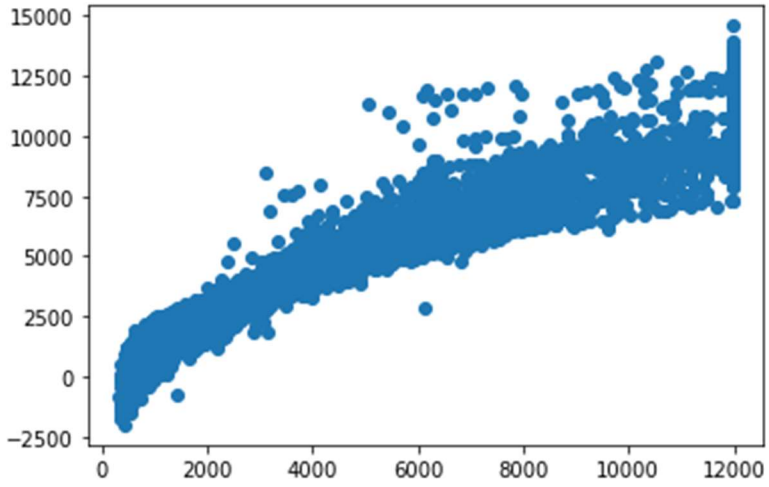
```
Intercept   -3171.950447
carat        8901.941225
cut           109.188125
color         272.921330
clarity       436.441104
depth           8.236972
table         -17.345170
x           -1417.908930
y            1464.827270
z            -711.225033
dtype: float64
```

Inferential statistics:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.931
Model:                            OLS   Adj. R-squared:                  0.931
Method:                 Least Squares   F-statistic:                 2.833e+04
Date:                Thu, 19 May 2022   Prob (F-statistic):               0.00
Time:                        21:11:00   Log-Likelihood:            -1.5510e+05
No. Observations:               18847   AIC:                         3.102e+05
Df Residuals:                   18837   BIC:                         3.103e+05
Df Model:                           9
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept  -3171.9504    787.532     -4.028      0.000   -4715.583   -1628.318
carat       8901.9412     82.792    107.521      0.000    8739.661    9064.222
cut          109.1881      7.268     15.024      0.000      94.943     123.433
color        272.9213      4.105     66.478      0.000     264.874     280.968
clarity      436.4411      4.473     97.581      0.000     427.674     445.208
depth          8.2370     10.876      0.757      0.449     -13.080      29.554
table        -17.3452      3.904     -4.443      0.000     -24.998      -9.693
x          -1417.9089    136.590    -10.381      0.000   -1685.637   -1150.181
y           1464.8273    136.068     10.765      0.000    1198.122    1731.533
z           -711.2250    156.187     -4.554      0.000   -1017.366    -405.084
==============================================================================
Omnibus:                     2652.028   Durbin-Watson:                   2.005
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             9642.429
Skew:                           0.687   Prob(JB):                         0.00
Kurtosis:                       6.223   Cond. No.                     1.03e+04
==============================================================================
```

Root Mean Squared Error – RMSE (For training data): 907.1312

Root Mean Squared Error – RMSE (For testing data): 911.8447



The final Linear Regression equation is:

Price = b0 + b1 *carat [T. True] + b2 * cut + b3 * color + b4 * clarity+ b5 * depth + b6 * table + b7 * x + b8 * y + b9 *z True

Price = (-3171.95) * Intercept + (8901.94) * carat + (109.19) * cut + (272.92) * color + (436.44) * clarity + (8.24) * depth + (-17.35) * table + (-1417.91)) * x + (1464.83) * y + (-711.23) * z _True

Observation - 1:

- When carat increases by 1 unit, diamond price increases by 8901.94 units, keeping all other predictors constant.
- When cut increases by 1 unit, diamond price increases by 109.19 units, keeping all other predictors constant.
- When color increases by 1 unit, diamond price increases by 272.92 units, keeping all other predictors constant.
- When clarity increases by 1 unit, diamond price increases by 436.44 units, keeping all other predictors constant.
- When y increases by 1 unit, diamond price increases by 1464.83 units, keeping all other predictors constant.

As per model these five attributes that are most important attributes 'Carat', 'Cut', 'color', 'clarity' and width i.e 'y' for predicting the price.

There are also some negative co-efficient values, for instance, corresponding co-efficient (-1417.91) for 'x', (-711.23) for z and (-17.35) for table This implies, these are inversely proportional with diamond price.

Observation - 2:

- On the given data set we can see the 'X' i.e Length of the cubic zirconia in mm. having negative co-efficient. And the p value is less than 0.05, so can conclude that as higher the length of the stone is a lower profitable stone.
- Similarly, for the 'z' variable having negative co-efficient i.e -711.23. And the p value is less than 0.05, so we can conclude that as higher the 'z' of the stone is a lower profitable stone.
- We can see the 'y' width in mm having positive co-efficient. And the p value is less than 0.05, so we can conclude that higher the width of the stone is a higher profitable stone.
- Finally, we can conclude that best 5 attributes that are most important are 'Carat', 'Cut', 'color', 'clarity' and width i.e 'y' for predicting the price.

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

Inference:

We can see that the from the linear plot, very strong corelation between the predicted y and actual y. But there are lots of spread. That indicates some kind noise present on the data set i.e Unexplained variances on the output.

- **Linear regression Performance Metrics:**

  - Intercept for the model: -3171.9504
  - R square on training data: 0.9312
  - R square on testing data: 0.9315
  - RMSE on Training data: 907.1312
  - RMSE on Testing data: 911.8447

As the training data & testing data score are almost inline, we can conclude this model is a Right-Fit Model.

- **Impact of scaling:**

Now we can observe by applying z score the intercept became -5.87961525130473e-16. Earlier it was -3171.9504. the co-efficient has changed, the bias became nearly zero but the overall accuracy still same.

- **Multi collinearity:**

We can observe there are very strong multi collinearity present in the data set.

- **From stats models:**

We can see R-squared:0.931 and Adj. R-squared: 0.931 are same. The overall P value is less than alpha.

Finally, we can conclude that **Best 5 attributes that are most important** are 'Carat', 'Cut', 'color', 'clarity' and width i.e 'y' for predicting the price.

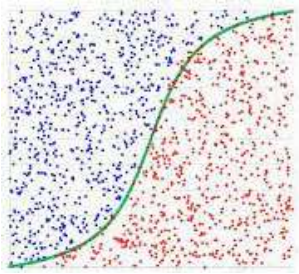
- When 'carat' increases by 1 unit, diamond price increases by 8901.94 units, keeping all other predictors constant.
- When 'cut' increases by 1 unit, diamond price increases by 109.19 units, keeping all other predictors constant.
- When 'color' increases by 1 unit, diamond price increases by 272.92 units, keeping all other predictors constant.
- When 'clarity' increases by 1 unit, diamond price increases by 436.44 units, keeping all other predictors constant.
- When 'y' increases by 1 unit, diamond price increases by 1464.83 units, keeping all other predictors constant.
- we can see the p value is showing 0.449 for depth variable, which is much greater than 0.05. That means this attribute is useless.
- There are also some negative co-efficient values, we can see the 'X' i.e Length of the cubic zirconia in mm. having negative co-efficient -1417.9089. And the p value is less than 0.05, so can conclude that as higher the length of the stone is a lower profitable stone.
- Similarly, for the 'z' variable having negative co-efficient i.e -711.23. And the p value is less than 0.05, so we can conclude that as higher the 'z' of the stone is a lower profitable stone.

Recommendations:

- We can see that the from the linear plot, very strong correlation between the predicted y and actual y. But there are lots of spread. That indicates some kind noise present on the data set i.e Unexplained variances on the output.
- The Gem Stones company should consider the features ‘Carat', 'Cut', 'color', 'clarity' and width i.e 'y' as most important for predicting the price.
- To distinguish between higher profitable stones and lower profitable stones so as to have better profit share.
- As we can see from the model Higher the width('y') of the stone is higher the price. So, the stones having higher width('y') should consider in higher profitable stones.
- The 'Premium Cut' on Diamonds are the most Expensive, followed by 'Very Good' Cut, these should consider in higher profitable stones.
- The Diamonds clarity with 'VS1' &'VS2' are the most Expensive. So, these two categories also consider in higher profitable stones.
- There are also some negative co-efficient values, we can see the 'X' i.e Length of the cubic zirconia in mm. having negative co-efficient -1417.9089. And the p value is less than 0.05, so can conclude that as higher the length of the stone is a lower profitable stone.

- As we see for 'X' i.e Length of the stone, higher the length of the stone is lower the price. So higher the Length('x') of the stone is lower is the profitability.
- Similarly, for the 'z' variable having negative co-efficient i.e -711.23. And the p value is less than 0.05, so we can conclude that as higher the 'z' of the stone is a lower profitable stone. This is because if a Diamond's Height is too large Diamond will become 'Dark' in appearance because it will no longer return an Attractive amount of light. That is why Stones with higher 'z' is also are lower in profitability.

Business Problem 2: Logistic Regression and LDA



Problem Statement:

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

Solution Approach:

The purpose of the solutioning exercise is to explore the dataset using Logistic Regression and LDA techniques to predict the claim status. Below is the data dictionary for the problem:

Attribute Information:

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

Analysis 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.

Firstly, import the necessary libraries required for the problem in the Jupiter Notebook file and run them. Read the "Holiday_Package.csv" file for EDA.

- **Head of the data is obtained as below**:

	Unnamed: 0	Holliday_Package	Salary	age	educ	no_young_children	no_older_children	foreign
0	1	no	48412	30	8	1	1	no
1	2	yes	37207	45	8	0	1	no
2	3	no	58022	46	9	0	0	no
3	4	no	66503	31	11	2	0	no
4	5	no	66734	44	12	0	2	no
5	6	yes	61590	42	12	0	1	no
6	7	no	94344	51	8	0	0	no
7	8	yes	35987	32	8	0	2	no
8	9	no	41140	39	12	0	0	no
9	10	no	35826	43	11	0	2	no

Table 6: Description of Tour and Travel agency data

- **Output from shape command:**

Number of rows:	872
Number of Columns:	8

- **List of fields retrieval along with their data type:**

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 872 entries, 0 to 871
Data columns (total 8 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Unnamed: 0          872 non-null    int64
 1   Holliday_Package    872 non-null    object
 2   Salary              872 non-null    int64
 3   age                 872 non-null    int64
 4   educ                872 non-null    int64
 5   no_young_children   872 non-null    int64
 6   no_older_children   872 non-null    int64
 7   foreign             872 non-null    object
dtypes: int64(6), object(2)
memory usage: 61.3+ KB
```

Output for missing values:

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

We will drop the first column 'Unnamed: 0' column as this is not important for our study. The shape would be – 872 rows and 7 columns

Summary of the data, providing descriptive statistical variables:

Descriptive statistics help describe and understand the features of a specific data set by giving short summaries about the sample and measures of the data. The most recognized types of descriptive statistics are measures of center: the mean, median, and mode, which are used at almost all levels math and statistics.

	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

Table 7: Summary Statistic of Tour and Travel agency data

Inferences:

- The data set contains 872 rows and 7 columns.
- In the given data set, there are 5 Integer type features, 2 Object type features. Where 'Holliday_Package' is the target variable and all other are predictor variable.
- We have dropped the first column 'Unnamed: 0' column as this is not important.
- There are No duplicate rows in the dataset.
- Salary ranges from 1322 to 236961. Average salary of employees is around 47729 with a standard deviation of 23418. Standard deviation indicates that the data is not normally distributed.

- Skewness of 0.71 indicates that the data is right skewed and there are few employees earning more than an average of 47729. - - 75% of the employees are earning below 53469 while 255 of the employees are earning 35324.
- Age of the employee ranges from 20 to 62. Median is around 39. 25% of the employees are below 32 and 25% of the employees are above 48. Standard deviation is around 10. Standard deviation indicates almost normal distribution.
- Years of formal education ranges from 1 to 21 years. 25% of the population has formal education for 8 years, while the median is around 9 years. 75% of the employees have formal education of 12 years. Standard deviation of the education is around 3. This variable is also indicating skewness in the data.
- No missing values or null values.

Univariate Analysis:

1. SALARY VARIABLE:

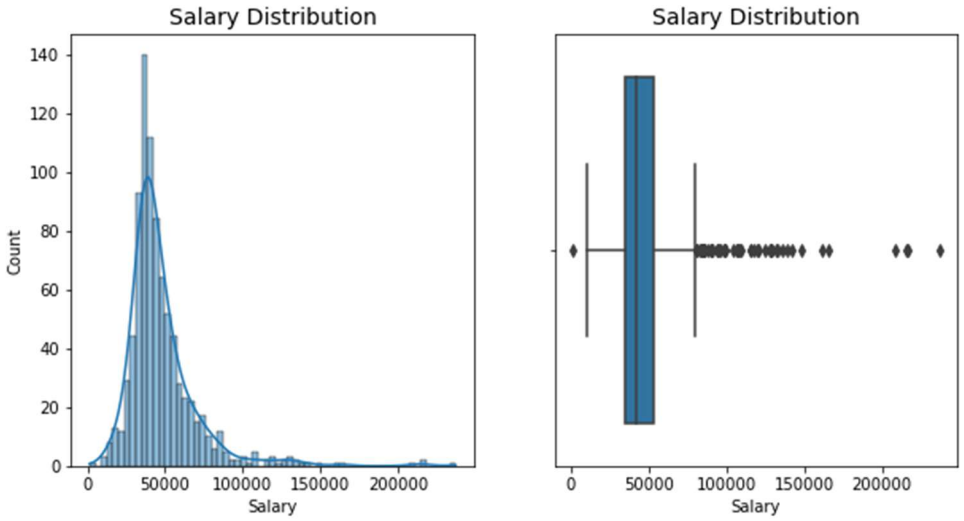


Figure 15: Univariate analysis of Salary variable

Observations:

- For Univariate Analysis of Salary, we are using histplot and boxplot to find information or patterns in the data.
- The Boxplot of Salary variable seems to have outliers.
- The distribution of the data is right skewed.

2. AGE VARIABLE:

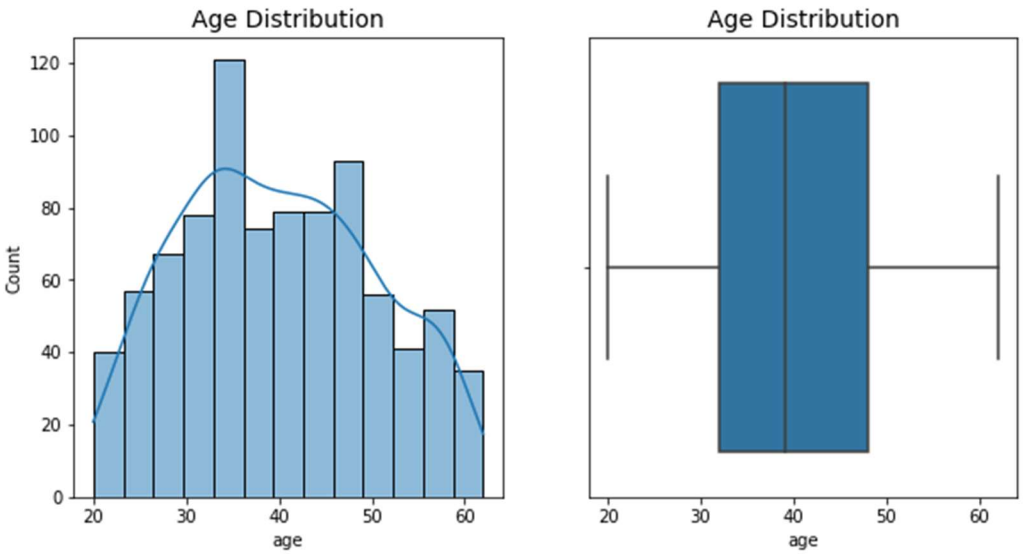


Figure 16: Univariate analysis of Age variable

Observations:

- The Boxplot of Age variable has no outliers.
- The distribution of the data is normally distributed and right skewed.
- Most of the employees are between the age limit 33 to 36 years.

3. EDUCATION VARIABLE:

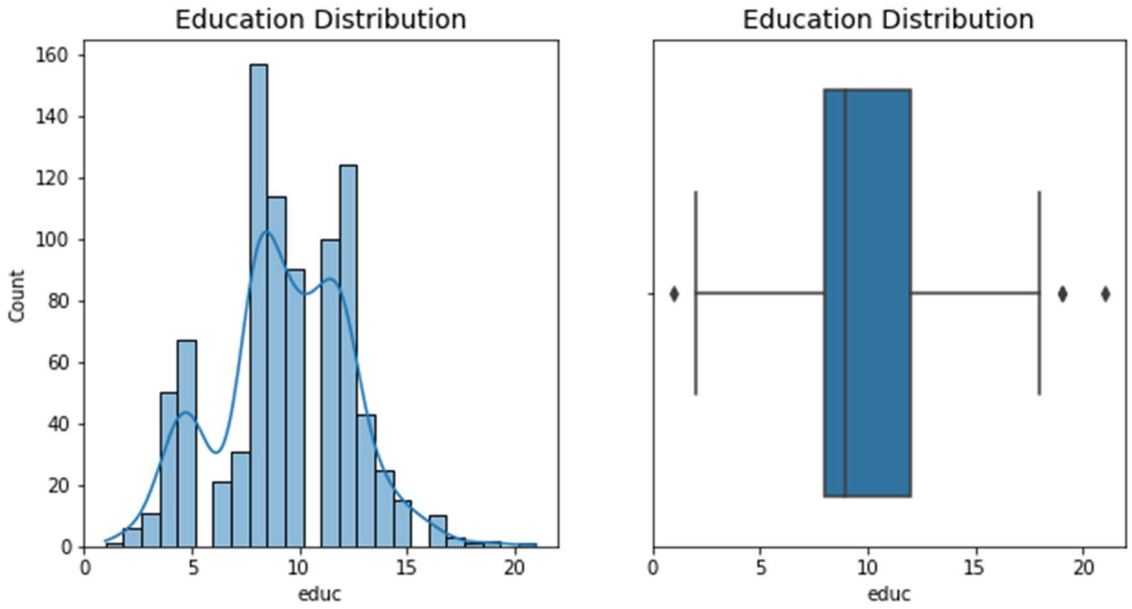


Figure 17: Univariate analysis of Education variable

Observations:

- The Boxplot of Education variable has few outliers.
- The distribution of the data is left skewed.

4. NUMBER OF YOUNG CHILDREN VARIABLE :

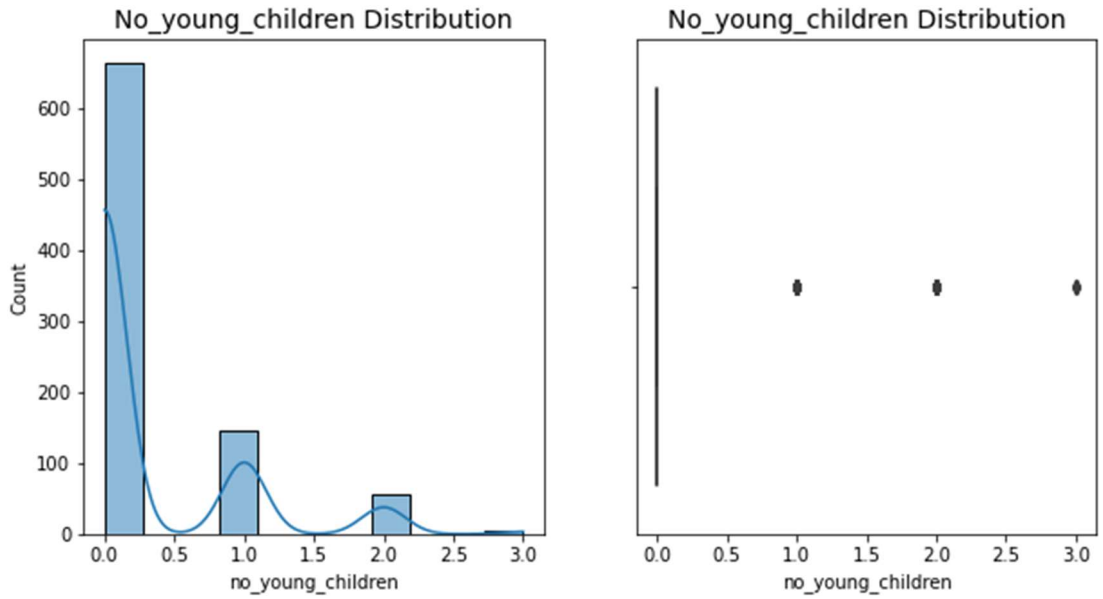


Figure 18: Univariate analysis of No_young_children variable

Observations:

- The Boxplot of no_young_children variable has few outliers.
- The distribution of the data is right skewed.

5. NUMBER OF OLDER CHILDREN VARIABLE:

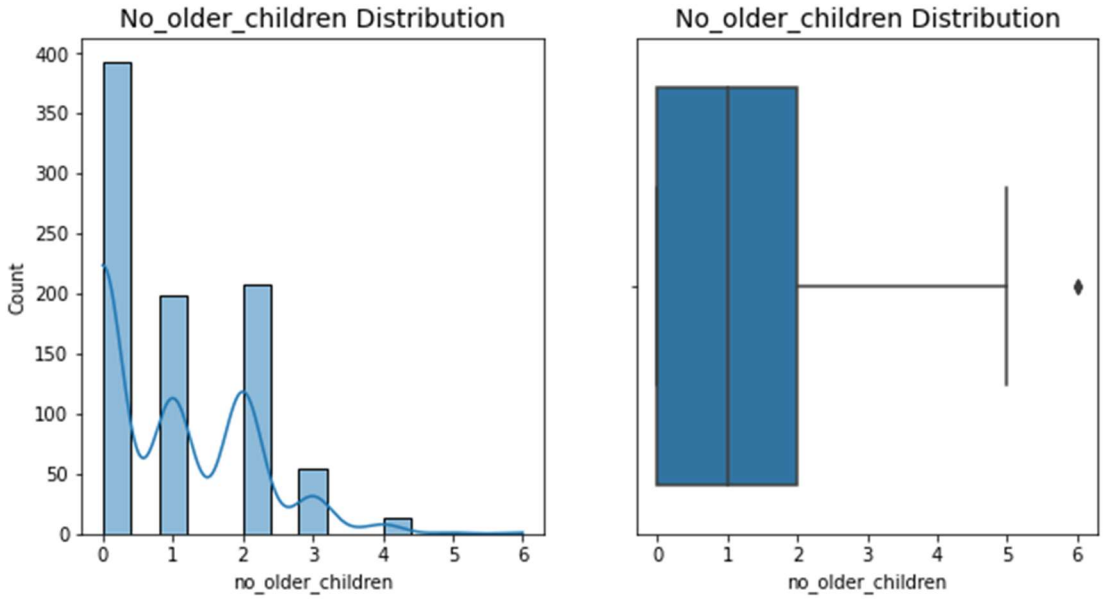


Figure 19: Univariate analysis of No_older_children variable

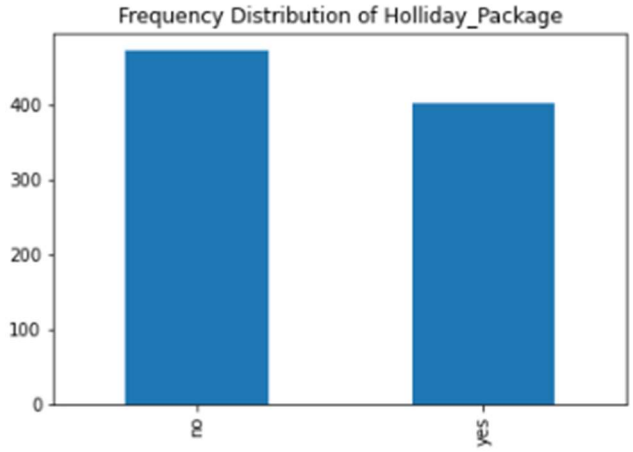
Observations:

- The Boxplot of No_older_children variable has very few outliers.
- The distribution of the data is right skewed.

Univariate analysis of categorical variables :

1. Holliday_Package Variable:

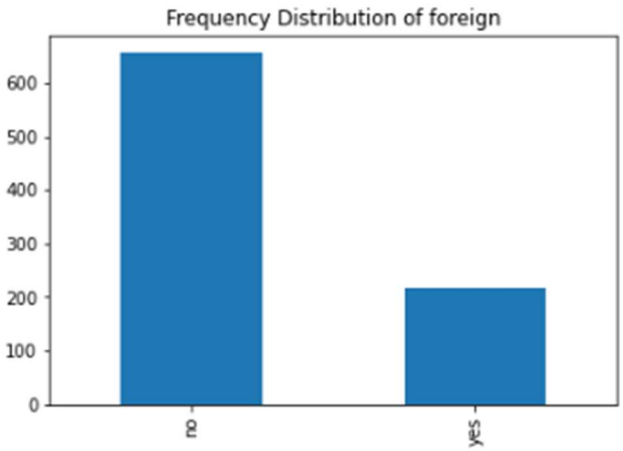
```
Details of Holliday_Package
-------------------------------------------------------
no     471
yes    401
Name: Holliday_Package, dtype: int64
```



Holliday_Package	Salary	age	educ	no_young_children	no_older_children
no	51739.443737	40.853503	9.594480	0.409766	0.902335
yes	43018.852868	38.900249	8.970075	0.197007	1.077307

2. Foreign Variable:

```
Details of foreign
-------------------------------------------------------
no     656
yes    216
Name: foreign, dtype: int64
```



foreign	Salary	age	educ	no_young_children	no_older_children
no	50429.248476	40.603659	10.038110	0.282012	0.969512
yes	39528.939815	37.986111	7.087963	0.402778	1.023148

Inference:

Foreign: The data is imbalanced with more skewed towards no and relatively a smaller shared for yes.

Bi-Variate Analysis :

Bi-Variate Analysis with Target variable:

1. Holiday Package & Salary:

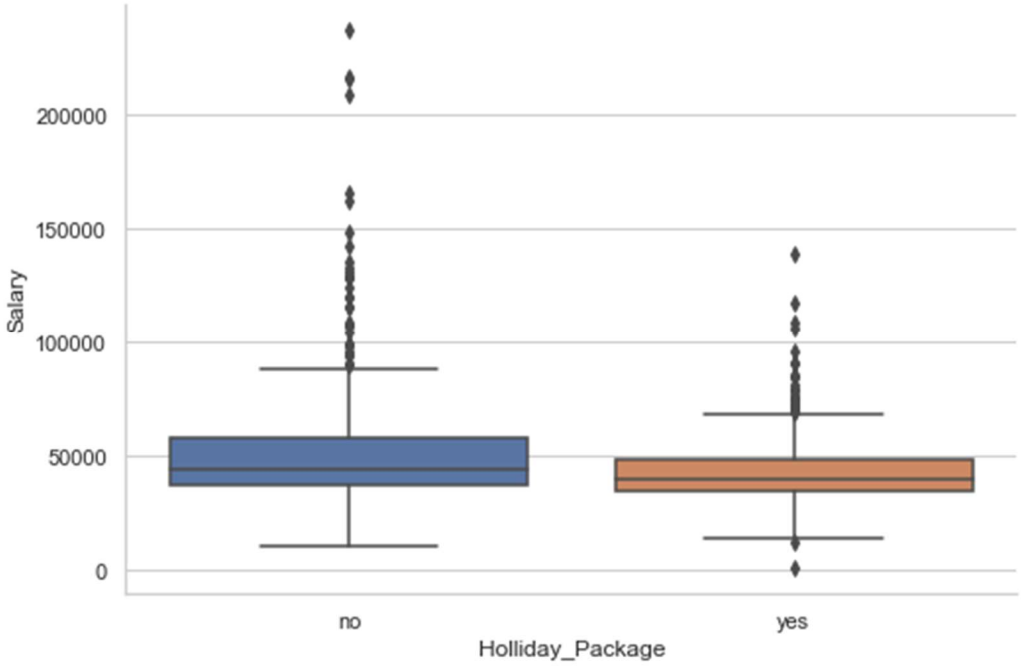


Figure 20: Bivariate analysis with Holiday Package & Salary variables

While performing the bivariate analysis we observe that Salary for employees opting for holiday package and for not opting for holiday package is similar in nature. However, the distribution is fairly spread out for people not opting for holiday packages.

2. Holiday Package & Age :

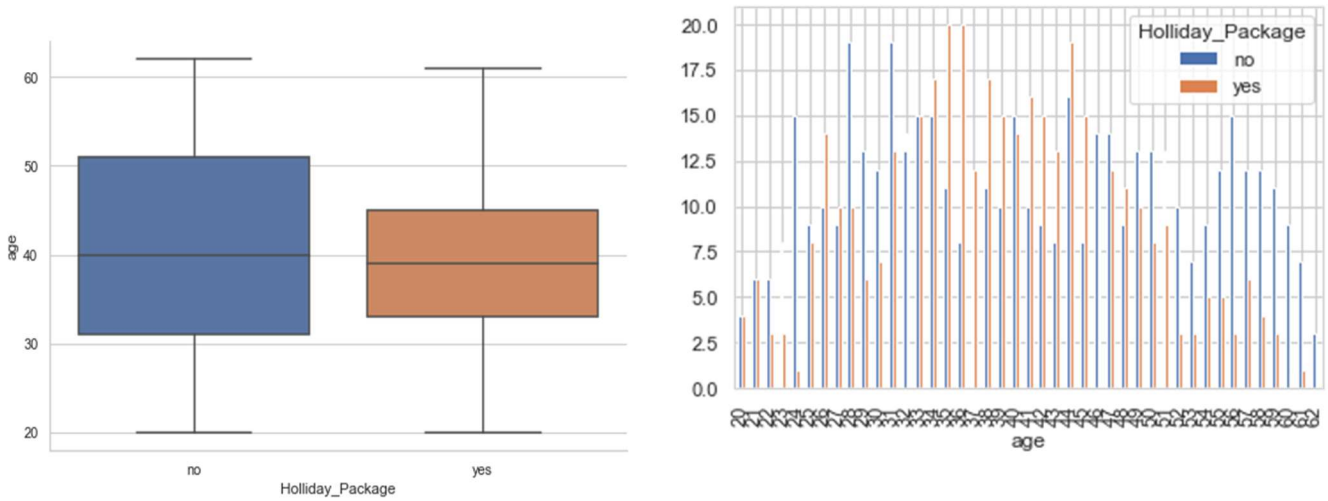


Figure 21: Bivariate analysis with Holiday Package & Age variables

There are no outliers present in age variable. The distribution of data for age variable with holiday package is also similar in nature.

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

3. Holiday Package & Education:

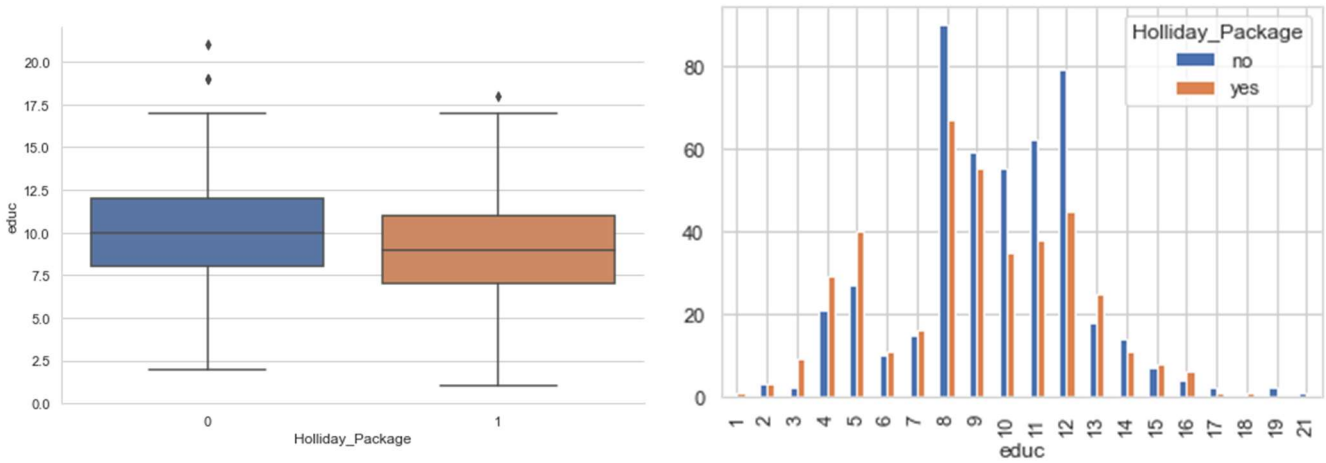


Figure 22: Bivariate analysis with Holiday Package & Education variables

This variable is also showing a similar pattern. This means education is likely not to be a variable for influencing holiday packages for employees.

We observe that employees 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.

4. Holiday Package & No_young_children :

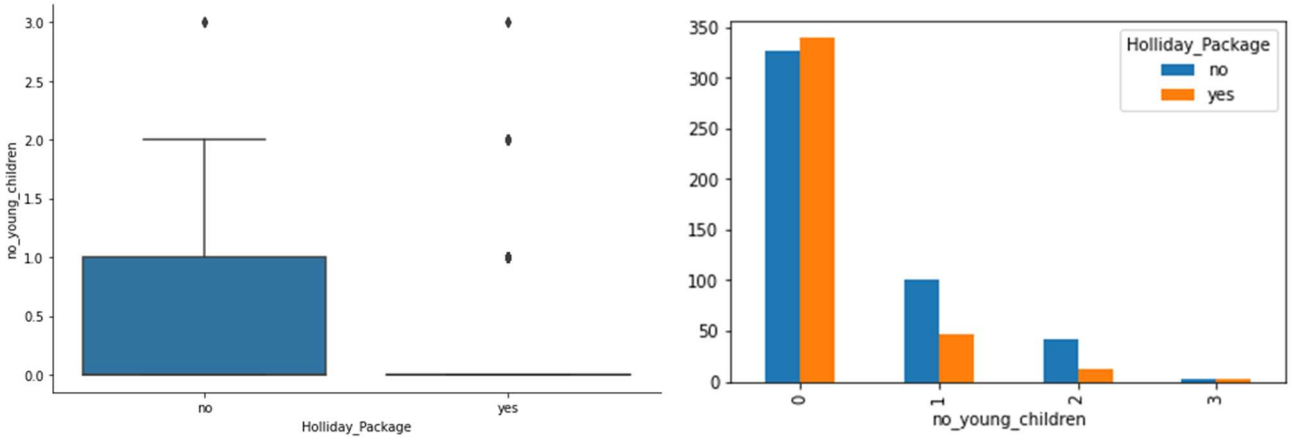


Figure 23: Bivariate analysis with Holiday Package & No_young_children variables

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. We can clearly see that people with younger children are not opting for holiday packages.

5. Holiday Package & No_older_children:

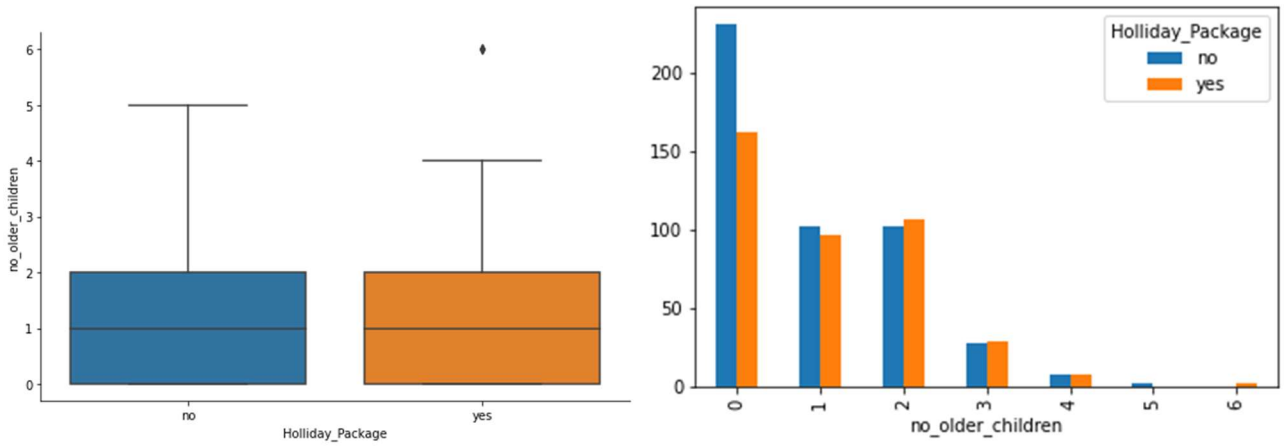


Figure 24: Bivariate analysis with Holiday Package & No_older_children variables

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 while creating our logistics model. Almost same distribution for both the scenarios when dealing with employees with older children.

Histogram:

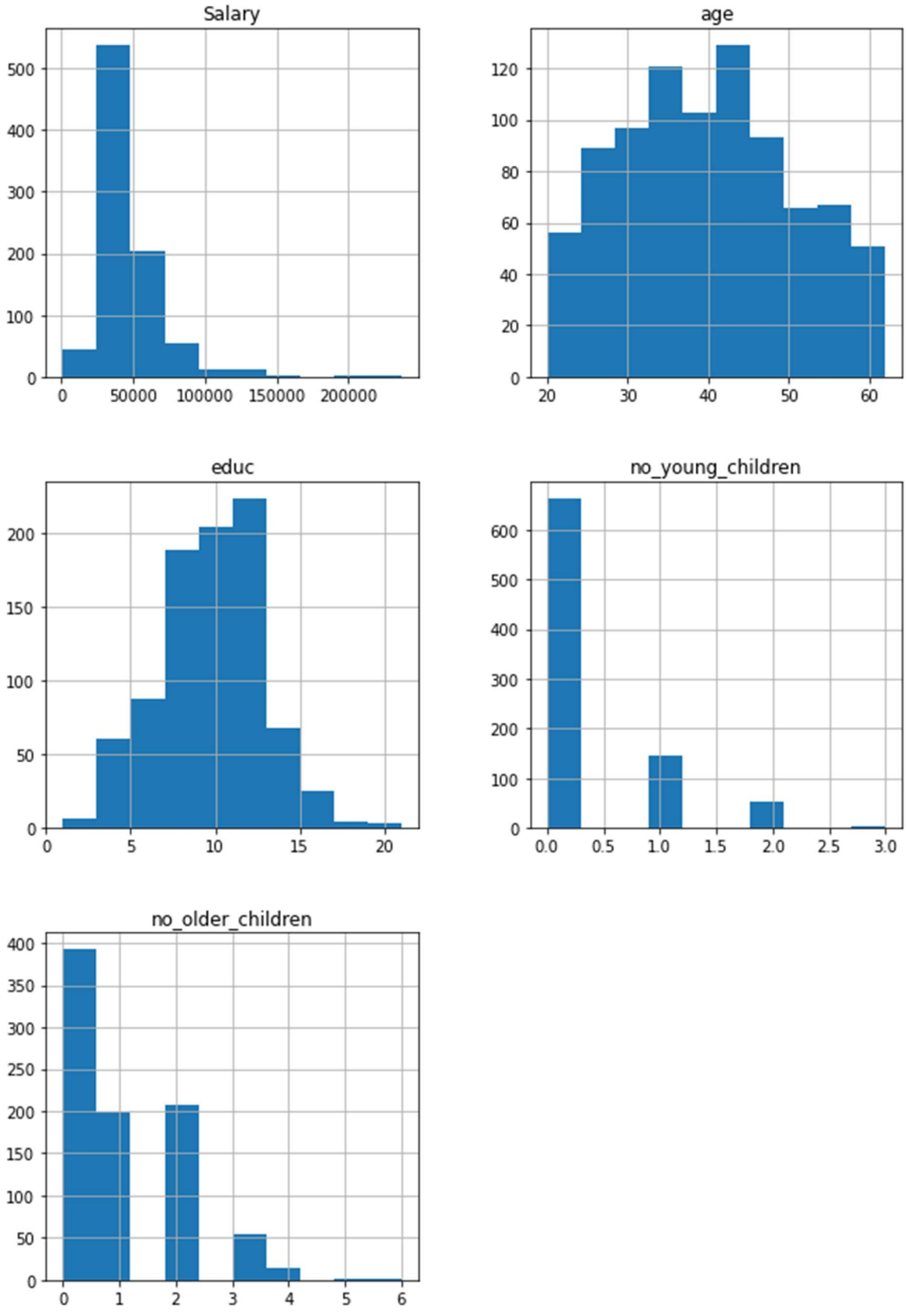


Figure 25: Histogram analysis of bivariate variables

MULTIVARIATE ANALYSIS :

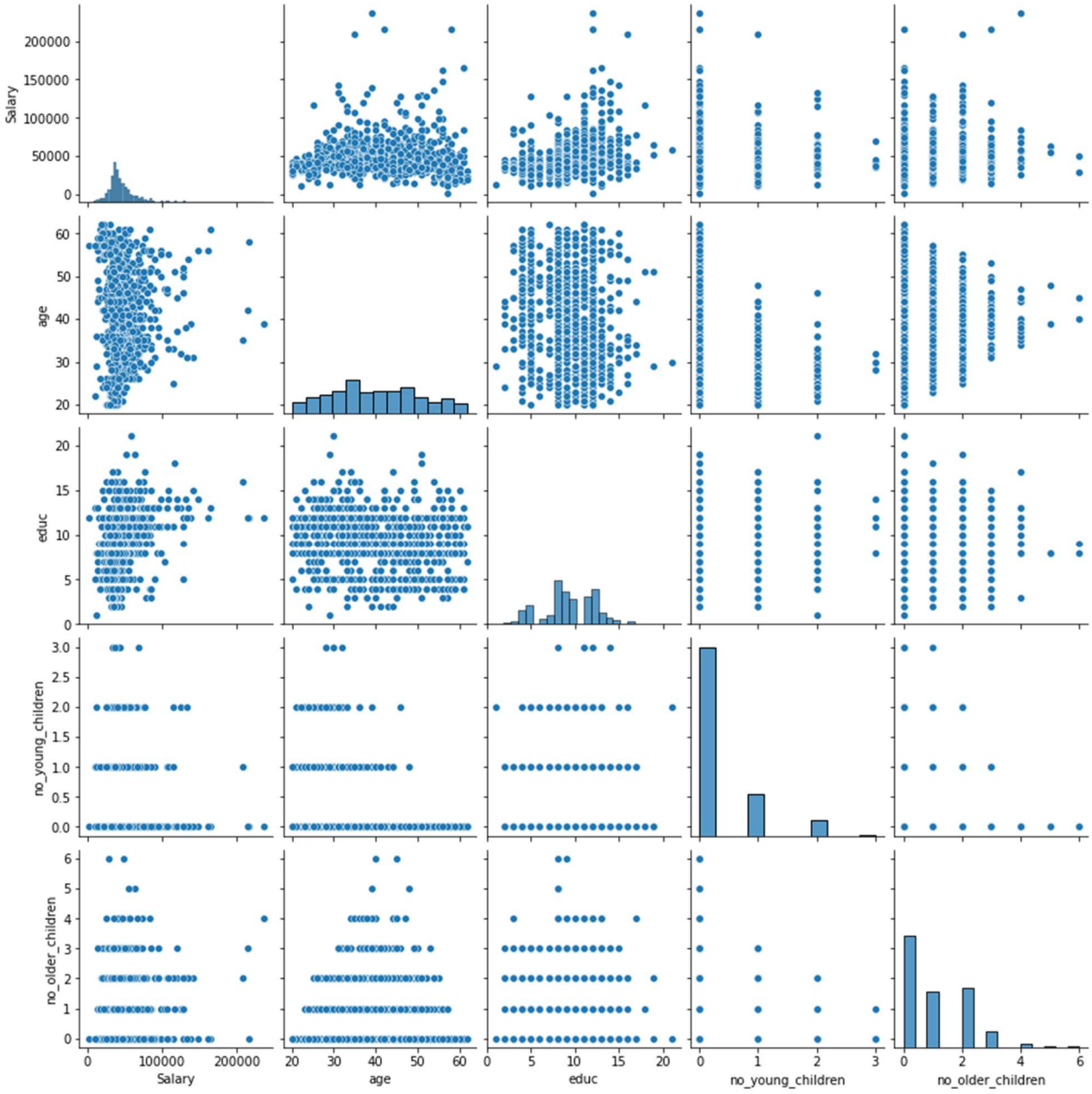


Figure 26: Multivariate analysis of all variables

Skewness values:

Salary	3.103216
no_young_children	1.946515
no_older_children	0.953951
age	0.146412
educ	-0.045501

Table 8: Skewness value between 5 independent variables

Correlation between variables of the dataset :

We will see correlation between independent variables to see which factors might influence choice of holiday package.

	Salary	age	educ	no_young_children	no_older_children
Salary	1.000	0.072	0.327	-0.030	0.114
age	0.072	1.000	-0.149	-0.519	-0.116
educ	0.327	-0.149	1.000	0.098	-0.036
no_young_children	-0.030	-0.519	0.098	1.000	-0.238
no_older_children	0.114	-0.116	-0.036	-0.238	1.000

CORRELATION HEATMAP :

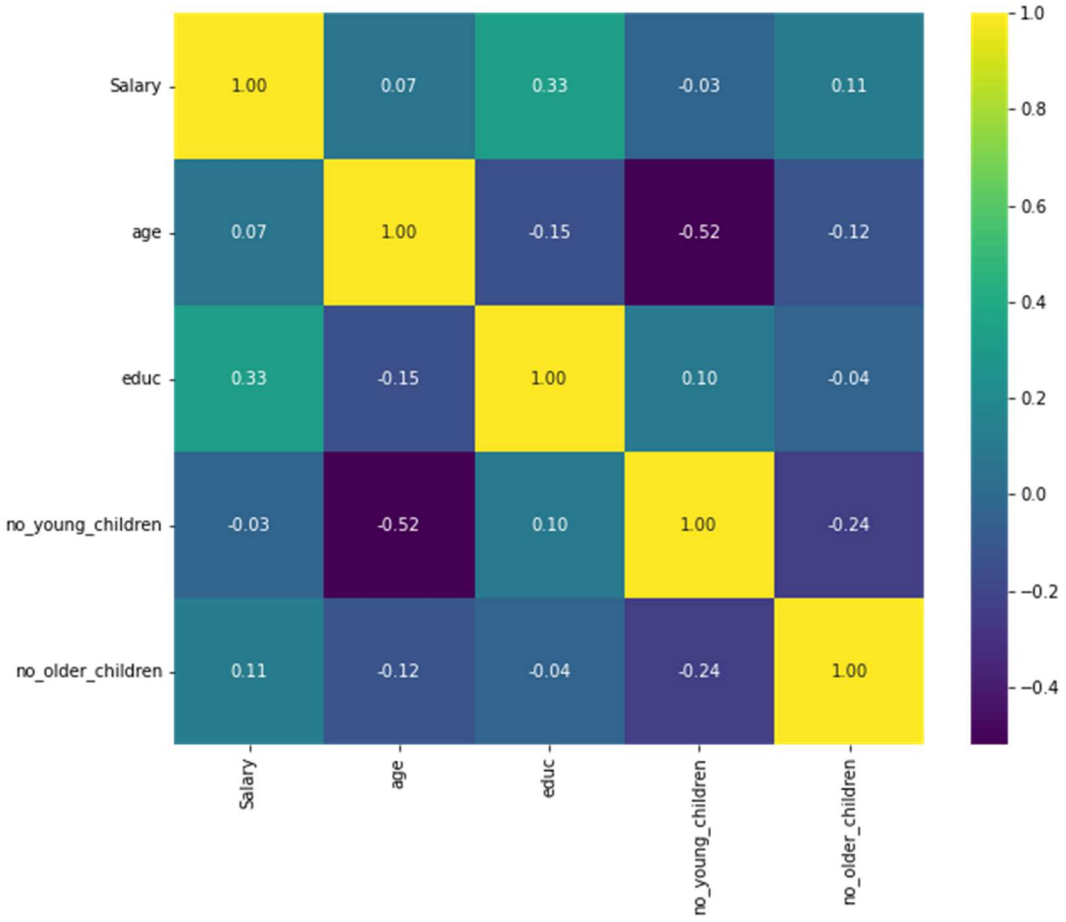


Figure 27: Correlation heatmap of all variables

Observations:

- We can relate that there isn't any strong correlation between any variables.
- Salary and education display moderate correlation and no_older_children are somewhat correlated with salary variable. However, there are no strong correlation in the data set.

Analysis 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).

Solution :

Logistic Regression is defined as a statistical approach, for calculating the probability outputs for the target labels. In its basic form, it is used to classify binary data. Logistic regression is very much similar to linear regression where the explanatory variables(X) are combined with weights to predict a target variable of binary class(y).

In the given dataset, the target variable – Holliday Package and an independent variable – Foreign are object variables. Let us study them one at a time.

Holliday_Package: The distribution seems to be fine, with 54% for no and 46% for yes.

Converting Categorical to Numerical Variable :

Head of the Data:

	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

Data set info :

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 872 entries, 0 to 871
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Holliday_Package   872 non-null    int8
 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    int8
dtypes: int64(5), int8(2)
memory usage: 74.9 KB
```

Train Test Split:

Split X and Y into training and test set in 70:30 ratio. This implies 70% of the total data will be used for training purposes and remaining 30% will be used for test purposes

Logistic Regression Model :

Formula 8: Logistic Regression

$$p = \frac{1}{1+e^{-y}} \text{ , Where } y = \beta_0 + \beta_1 x$$

Checking the data split for the dependent variable – Y in both train and test data. The percentage split between No and Yes seems to be almost same at 54% and 46%, respectively for both train and test data sets.

Data split for the target variable – Holliday_Package in training and test sets:

For Training Set:

```
0    0.539344
1    0.460656
Name: Holliday_Package, dtype: float64
```

For Testing Set:

```
0    0.541985
1    0.458015
Name: Holliday_Package, dtype: float64
```

Checking the data split for the dependent variable – Y in both train and test data. The percentage split between No and Yes seems to be almost same at 54% and 46%, respectively for both train and test data sets. The data proportion seems to be reasonable and we can continue with our model building as next steps.

As next steps, we will initiate the Logistic Regression function and will then fit the Logistic Regression model. There after we will predict on the training and test data set.

Fitting the Logistic Regression model :

```
LogisticRegression(max_iter=10000, n_jobs=2, penalty='none', solver='newton-cg',
                   verbose=True)
```

Predicted Classes and Probs :

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

Model Accuracy Scores :

Formula 9: Accuracy

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{True Negatives} + \text{False Positive} + \text{False Negatives}}$$

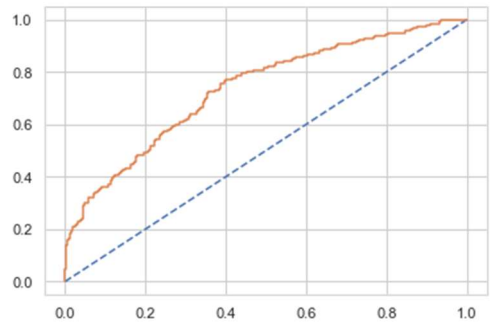
Training Data:

The Model Score for training data is 0.6672

Testing Data:

The Model Score for testing data is 0.6527

AUC and ROC for the training data:



AUC and ROC for the testing data:

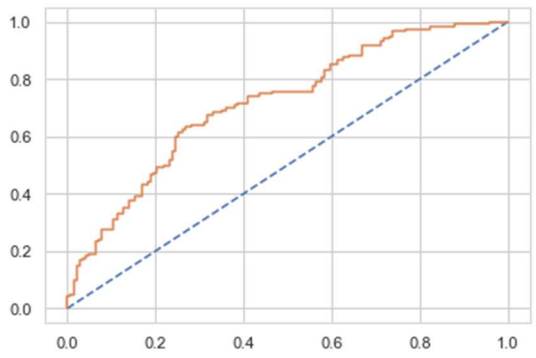
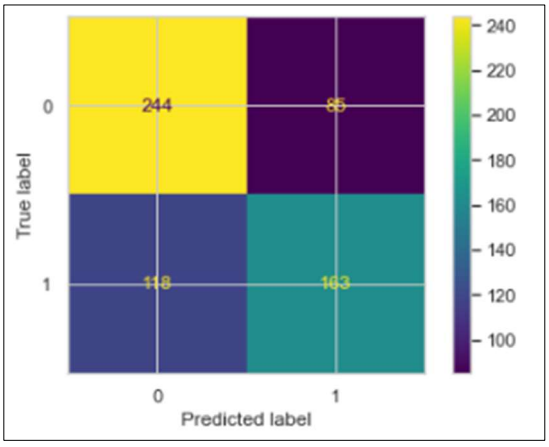


Figure 28: *AUC and ROC for the Train and Test data*

Confusion Matrix for the training data:

```
array([[244,  85],
       [118, 163]], dtype=int64)
```

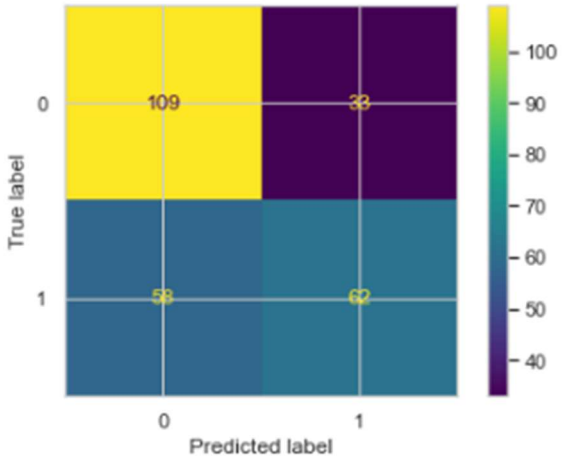


Classification Report :

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

Confusion Matrix for test data:

```
array([[109,  33],
       [ 58,  62]], dtype=int64)
```



Classification Report :

	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

The accuracy scores aren't too different and can be considered as right fit models avoiding the scenarios of underfit and overfit models. We can apply for GridSearchCV here to finetune the model further to see if helps improve the results. GridSearchCV is an iterative method of obtaining the best model based on a scoring metric provided by us and the parameters provided.

Applying GridSearchCV for Logistic Regression

```
GridSearchCV(cv=3, estimator=LogisticRegression(max_iter=10000, n_jobs=2),
             n_jobs=-1,
             param_grid={'penalty': ['l2', 'none'], 'solver': ['sag', 'lbfgs'],
                         'tol': [0.0001, 1e-05]},
             scoring='f1')
```

Post running the GridSearchCV, we can check on the accuracy scores again to identify if the model performance has improved or not.

Getting the probabilities on the test set:

	0	1
0	0.534650	0.465350
1	0.556290	0.443710
2	0.539223	0.460777
3	0.549676	0.450324
4	0.572353	0.427647

Confusion matrix on the training data:

	precision	recall	f1-score	support
0	0.54	1.00	0.70	329
1	0.00	0.00	0.00	281
accuracy			0.54	610
macro avg	0.27	0.50	0.35	610
weighted avg	0.29	0.54	0.38	610

Confusion matrix on the test data:

	precision	recall	f1-score	support
0	0.54	1.00	0.70	142
1	0.00	0.00	0.00	120
accuracy			0.54	262
macro avg	0.27	0.50	0.35	262
weighted avg	0.29	0.54	0.38	262

The accuracy scores for:

Train data is at 54% Test data is at 54%

There is a reduction in accuracy score for the both train and test data.

Post creating this model, we will then move on to creating the Linear Discriminant Analysis (LDA).

LDA takes the help of prior probabilities to predict the corresponding target probabilities. Prior probabilities are the probability of y (say equal to 1) without taking into account any other data or variables. The corresponding updated probabilities when the covariates (Xs) are available is called the posterior probabilities. We want to find P(Y=1|X). Thus, a Linear Discriminant Analysis (LDA) discriminates between the two classes by looking at the features (Xs)

LDA Model :

Formula 10 : LDA model

The LDA model gives linear combinations of the predictor variables as follows:

$$DS = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots .. + \beta_k X_k$$

Where: DS = Discriminant Score

β's = Discriminant weight (coefficients)

X's = Explanatory (Predictor or independent) variables

Training Data and Test Data Confusion Matrix Comparison:

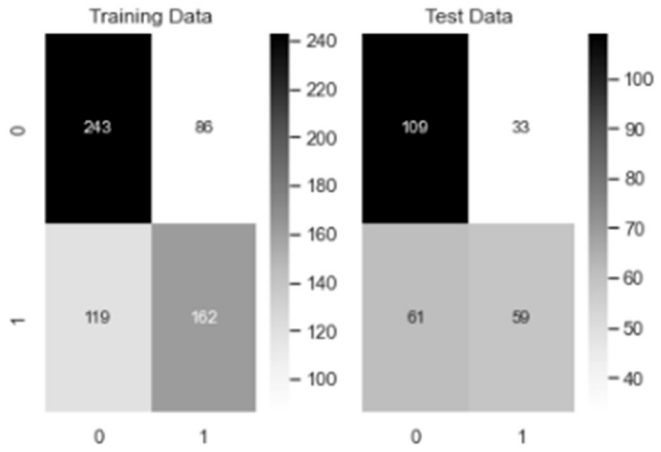


Figure 29: LDA model - *Training Data and Test Data Confusion Matrix Comparison*

Training Data and Test Data Classification Report Comparison:

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

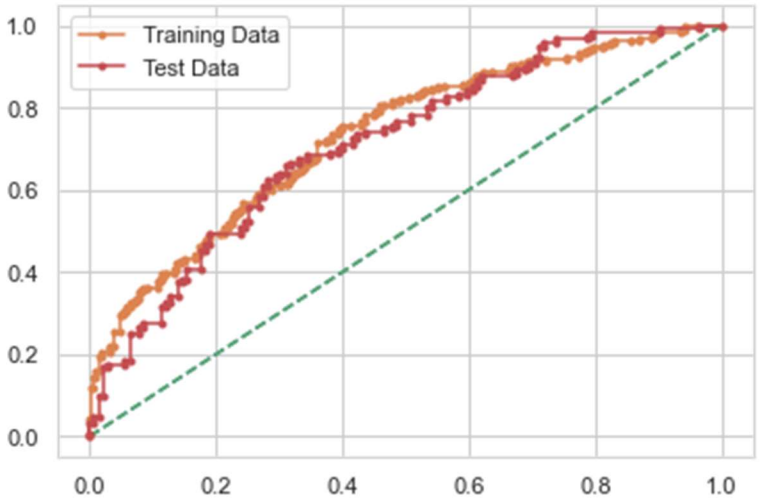
Conclusion:

The model accuracy on the training as well as the test set is about 54%, which is roughly little different proportion as the class 0 observations in the dataset. This model is affected by a class imbalance problem. Since we only have 862 observations, if re-build the same LDA model with a greater number of data points, an even better model could be built.

AUC and ROC for the training data:

AUC for the Training Data: 0.733

AUC for the Test Data: 0.714

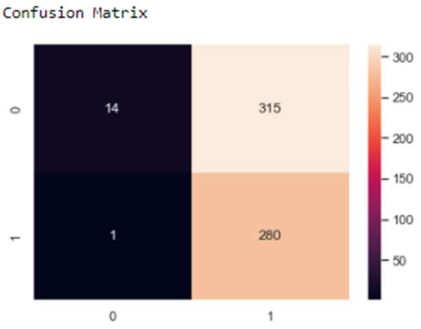


Changing the cut-off values for maximum accuracy:

	Salary	age	educ	no_young_children	no_older_children	foreign
821	38974	47	12	0	2	1
805	40270	33	8	2	0	1
322	32573	30	11	1	0	0
701	43839	43	11	0	1	1
773	33060	40	5	1	1	1

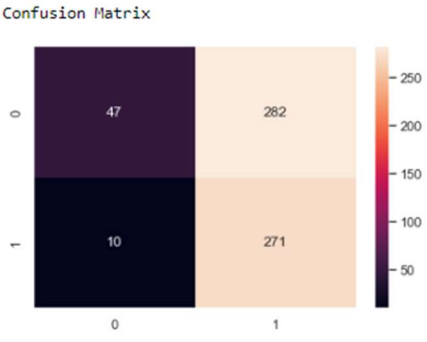
```
0.1

Accuracy Score 0.482
Recall Score 0.9964
F1 Score 0.6393
```



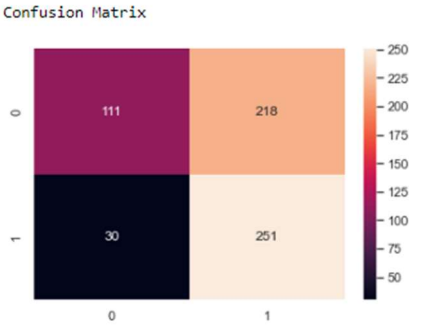

```
0.2

Accuracy Score 0.5213
Recall Score 0.9644
F1 Score 0.6499
```



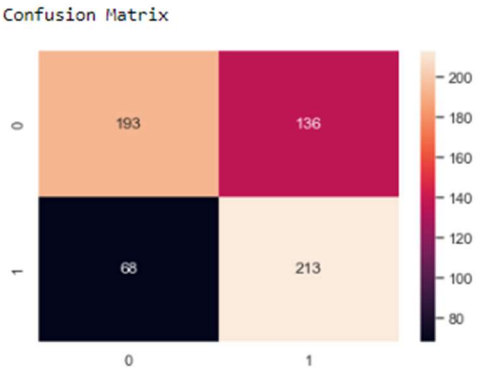

```
0.3

Accuracy Score 0.5934
Recall Score 0.8932
F1 Score 0.6693
```



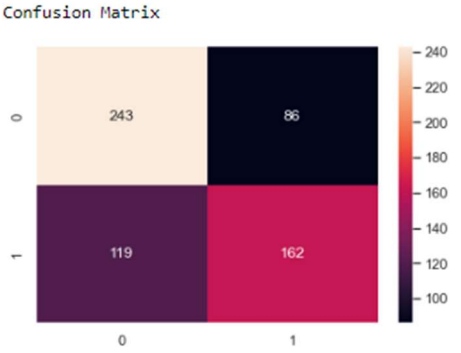
0.4

Accuracy Score 0.6656
Recall Score 0.758
F1 Score 0.6762



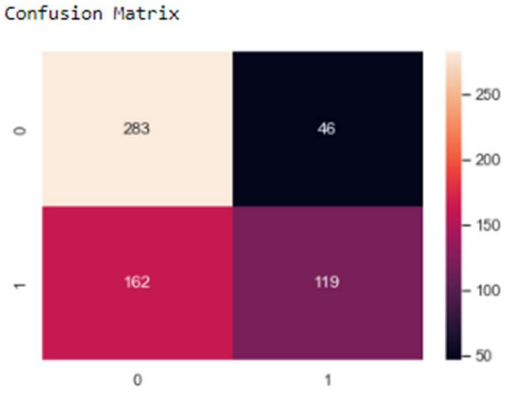
0.5

Accuracy Score 0.6639
Recall Score 0.5765
F1 Score 0.6125



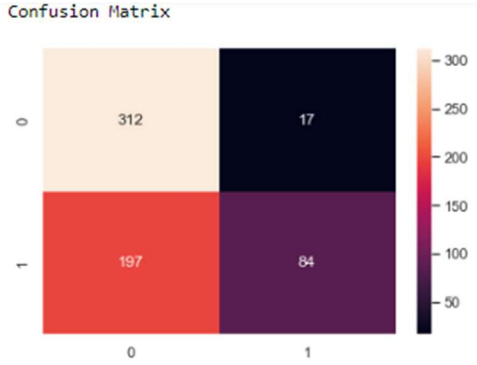
0.6

Accuracy Score 0.659
Recall Score 0.4235
F1 Score 0.5336



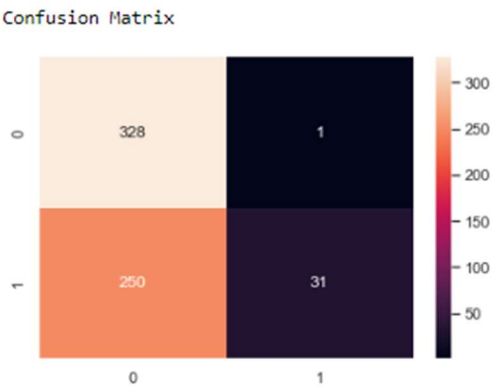
0.7

Accuracy Score 0.6492
Recall Score 0.2989
F1 Score 0.4398



0.8

Accuracy Score 0.5885
Recall Score 0.1103
F1 Score 0.1981



0.9

Accuracy Score 0.5426
Recall Score 0.0071
F1 Score 0.0141

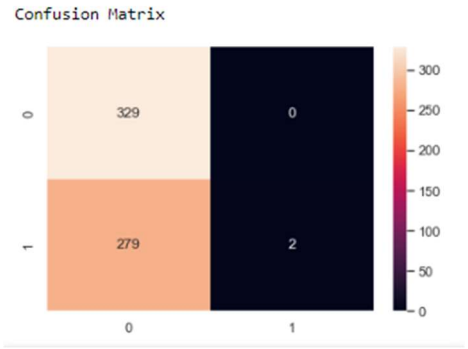
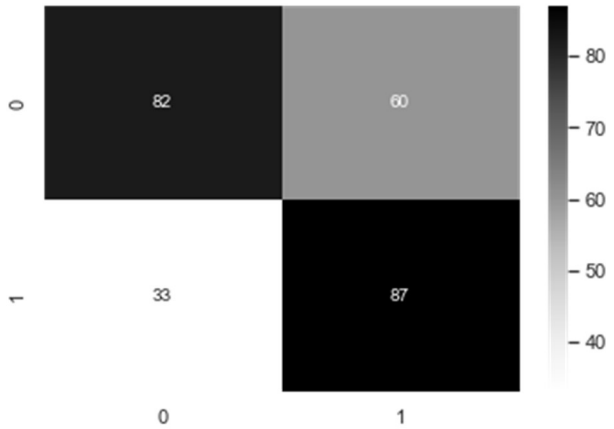


Figure 30: Probability matrix

We see that 0.4 and 0.5 gives better accuracy than the rest of the custom cut-off values. But 0.1 cut-off gives us the best 'f1-score'. Here, we will take the cut-off as 0.4 to get the optimum 'f1' score. Let us evaluate the predictions of the test data using these cut-off values.

Confusion matrix:



Classification Report of the default cut-off 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

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

Inferences using custom cut-off test data:

For {Customer who did not go for holiday (Label 0)}:

Precision (71%) – 71% of Customers who did not go for holiday are correctly predicted, out of all Customers who did not go for holiday that are predicted.

Recall (58%) – Out of all the Customers who actually did not go for holiday, 58% of Customers who did not go for holiday have been predicted correctly.

For {Customer who went for holiday (Label 1)}:

Precision (59%) – 59% of Customers who went for holiday are correctly predicted, out of all Customers who went for holiday that are predicted.

Recall (72%) – Out of all the Customers who actually went for holiday, 72% of Customers who went for holiday have been predicted correctly.

Overall accuracy of the model – 64 % of total predictions are correct.

Accuracy, AUC, Precision and Recall for test data is almost in line with training data. This proves no overfitting or underfitting has happened, and overall, the model is a good model for classification.

Implementing the model:

```
Optimization terminated successfully.
         Current function value: 0.612003
         Iterations 5
                           Logit Regression Results
==============================================================================
Dep. Variable:       Holliday_Package   No. Observations:                  872
Model:                         Logit   Df Residuals:                      866
Method:                          MLE   Df Model:                            5
Date:                Tue, 17 May 2022   Pseudo R-squ.:                  0.1129
Time:                        22:19:05   Log-Likelihood:                -533.67
converged:                       True   LL-Null:                       -601.61
Covariance Type:            nonrobust   LLR p-value:                 1.337e-27
==============================================================================
                     coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Salary        -1.584e-05   4.07e-06     -3.896      0.000   -2.38e-05   -7.87e-06
age              -0.0173      0.005     -3.370      0.001      -0.027      -0.007
educ              0.1105      0.024      4.578      0.000       0.063       0.158
no_young_children -0.9674      0.152     -6.373      0.000      -1.265      -0.670
no_older_children  0.0924      0.068      1.362      0.173      -0.041       0.225
foreign            1.6075      0.189      8.502      0.000       1.237       1.978
==============================================================================
```

Analysis 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.

Solution:

Confusion matrix cells has the terms: True Positive (TP)- The values which are predicted as True and are actually True. True Negative (TN)- The values which are predicted as False and are actually False. False Positive (FP)- The values which are predicted as True but are actually False. False Negative (FN)- The values which are predicted as False but are actually True.

ROC Curve:

Receiver Operating Characteristic (ROC) measures the performance of models by evaluating the trade-offs between sensitivity (true positive rate) and false (1- specificity) or false positive rate.

AUC Curve:

The area under curve (AUC) is another measure for classification models is based on ROC. It is the measure of accuracy judged by the area under the curve for ROC.

Predicted Classes and Probs :

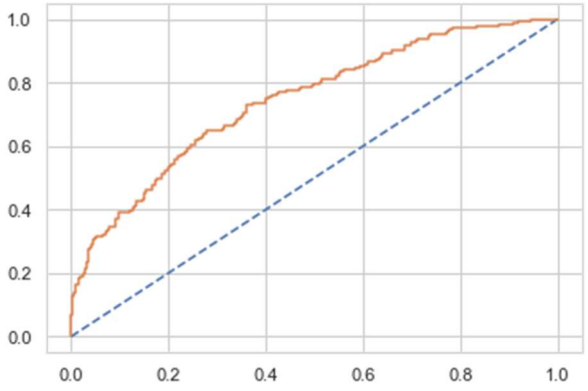
	0	1
0	0.753599	0.246401
1	0.287308	0.712692
2	0.888743	0.111257
3	0.974783	0.025217
4	0.499096	0.500904

The accuracy scores for:

Train data is at 68% Test data is at 64.5%

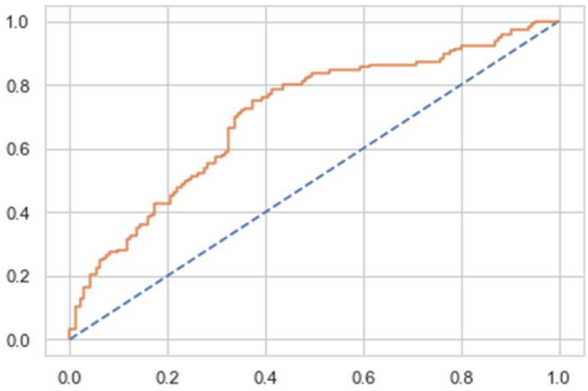
Train Model Roc_AUC Score:

AUC score is 0.743



Test Model Roc_AUC Score:

AUC score is 0.743



Confusion matrix on train data:

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

The result is being telling us that there are 252+163 correct predictions and 121+74 wrong predictions.

Classification Report (For training data):

	precision	recall	f1-score	support
0	0.68	0.77	0.72	326
1	0.69	0.57	0.63	284
accuracy			0.68	610
macro avg	0.68	0.67	0.67	610
weighted avg	0.68	0.68	0.68	610

Confusion Matrix for Test Data:

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

The result is being telling us that we have 102+67 correct predictions and 50+43 incorrect predictions.

Classification Report (For testing data):

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

Implementing the model:

```
Optimization terminated successfully.
         Current function value: 0.612003
         Iterations 5
                           Logit Regression Results
==============================================================================
Dep. Variable:       Holliday_Package   No. Observations:                  872
Model:                         Logit   Df Residuals:                      866
Method:                          MLE   Df Model:                            5
Date:                Fri, 20 May 2022   Pseudo R-squ.:                 0.1129
Time:                        21:05:49   Log-Likelihood:                -533.67
converged:                       True   LL-Null:                       -601.61
Covariance Type:            nonrobust   LLR p-value:                 1.337e-27
==============================================================================
                     coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Salary         -1.584e-05   4.07e-06     -3.896      0.000   -2.38e-05   -7.87e-06
age               -0.0173      0.005     -3.370      0.001      -0.027      -0.007
educ               0.1105      0.024      4.578      0.000       0.063       0.158
no_young_children -0.9674      0.152     -6.373      0.000      -1.265      -0.670
no_older_children  0.0924      0.068      1.362      0.173      -0.041       0.225
foreign            1.6075      0.189      8.502      0.000       1.237       1.978
==============================================================================
```

Linear Discriminate Analysis:

Classification Report (For training data):

	precision	recall	f1-score	support
0	0.67	0.77	0.72	326
1	0.68	0.56	0.61	284
accuracy			0.67	610
macro avg	0.67	0.66	0.66	610
weighted avg	0.67	0.67	0.67	610

Confusion matrix on train data:

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

Classification Report (For testing data):

	precision	recall	f1-score	support
0	0.68	0.79	0.73	145
1	0.67	0.55	0.60	117
accuracy			0.68	262
macro avg	0.68	0.67	0.67	262
weighted avg	0.68	0.68	0.67	262

Confusion matrix on test data:

```
array([[114,  31],
       [ 53,  64]], dtype=int64)
```

Observations:

While looking the metrices for both training and the test data, it seems the accuracy scores are same on both models at 66%. Our model is close enough to be treated as a right fit model. The current model is not struggling with being an over fit model or an under fit model.

The AUC scores for both the training and test data are also same at 74.3%.

The model performance is good on F1 score as well with training data performing better at 62% while the test data gave a F1 score of 57%.

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

Inferences:

- Here Salary and education seems to be important parameters which is an important predictor.
- While performing the bivariate analysis we observe that Salary for employees opting for holiday package and for not opting for holiday package is similar in nature. However, the distribution is fairly spread out for people not opting for holiday packages.
- The distribution of data for age variable with holiday package is also similar in nature. The range of age for people not opting for holliday package is more spread out when compared with people opting for yes.
- We can observe that employees in middle range (34 to 45 years) are going for holiday package as compared to older and younger employees
- 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.
- Employees with older children has almost similar distribution for opting and not opting for holiday packages across the number of children levels and hence it is not an important predictor for this model.

Interpretations:

- There is no effect of salary, age, and education on the prediction for Holliday_Package. These variables don't seem to impact the decision to opt for holiday packages as we couldn't establish a strong relation of these variables with the target variable.

- Foreign has emerged as a strong predictor with a positive coefficient value. The likelihood of a foreigner opting for a holiday package is high.

- no_young_children variable is negating the probability for opting for holiday packages, especially for couple with number of young children at 2.

Recommendations:

- The company should really focus on foreigners to drive the sales of their holiday packages as that's where the majority of conversions are going to come in.

- The company can try to direct their marketing efforts or offers toward foreigners for a better conversion opting for holiday packages.

- The company should not target parents with younger children.

- The chances of selling to parents with 2 younger children is probably the lowest. This gives the fact that parents try and avoid visiting with younger children.

- If the firm wants to target parents with older children, that still might end up giving favourable return for their marketing efforts then spent on couples with younger children.