PROJECT REPORT ON PREDICTIVE MODELING

Akshaya Parthasarathy

Batch: PGPDSBA_online_July E 2020

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.

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

Exploratory Data Analysis:

Head of the dataset: Verify whether the dataset is loaded correctly

	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

Dropping the column ‘Unnamed: 0’ since it’s an index column.

Shape of the dataset:

```
There are  26967  rows and  10  columns in the dataset.
```

Information of the dataset: There are ten variables in the dataset of which ‘cut, clarity and color’ are of object type and rest are of either float or int type.

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

There are some null values in depth column.

Null values check in the dataset:

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

There is a total of 697 records in depth which does not have a value. Other than that, there are no null/blanks in other columns.

Duplicate records check in the dataset:

Total number of duplicated records: 34

Since there is no unique identifier in the given dataset, we can consider these 34 records to be purely duplicates and remove from the dataset.

	carat	cut	color	clarity	depth	table	x	y	z	price
4756	0.35	Premium	J	VS1	62.4	58.0	5.67	5.64	3.53	949
6215	0.71	Good	F	SI2	64.1	60.0	0.00	0.00	0.00	2130
8144	0.33	Ideal	G	VS1	62.1	55.0	4.46	4.43	2.76	854
8919	1.52	Good	E	I1	57.3	58.0	7.53	7.42	4.28	3105
9818	0.35	Ideal	F	VS2	61.4	54.0	4.58	4.54	2.80	906

Shape of the dataset after removal of duplicates:

```
After removing duplicates, there are  26933  rows and  10  columns in the dataset.
```

Summary statistics of the dataset:

Numerical columns:

	carat	depth	table	x	y	z	price
count	26933.000000	26236.000000	26933.000000	26933.000000	26933.000000	26933.000000	26933.000000
mean	0.798010	61.745285	57.455950	5.729346	5.733102	3.537769	3937.526120
std	0.477237	1.412243	2.232156	1.127367	1.165037	0.719964	4022.551862
min	0.200000	50.800000	49.000000	0.000000	0.000000	0.000000	326.000000
25%	0.400000	61.000000	56.000000	4.710000	4.710000	2.900000	945.000000
50%	0.700000	61.800000	57.000000	5.690000	5.700000	3.520000	2375.000000
75%	1.050000	62.500000	59.000000	6.550000	6.540000	4.040000	5356.000000
max	4.500000	73.600000	79.000000	10.230000	58.900000	31.800000	18818.000000

Inference:

- Looking at the mean and median of the columns, except for 'price', other columns have almost similar mean and median which indicates less skewness in the dataset.
- 'Price' is right skewed.
- There are outliers in the dataset since the maximum value of all the columns is more than the upper limit of IQR.
- Variables x, y and z has zero values which should be checked upon.

Categorical columns:

	cut	color	clarity
count	26933	26933	26933
unique	5	7	8
top	Ideal	G	SI1
freq	10805	5653	6565

Looking at the unique values and the value counts for each:

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


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


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

From the problem statement, we can see that these categorical variables have some kind of order.

- **Cut:** Quality is in increasing order – Fair, Good, Very Good, Premium, Ideal

Given this order, the company is manufacturing more of 'Ideal' cut cubic zirconia compared to the rest.

- **Color:** D being the best and J is the worst.

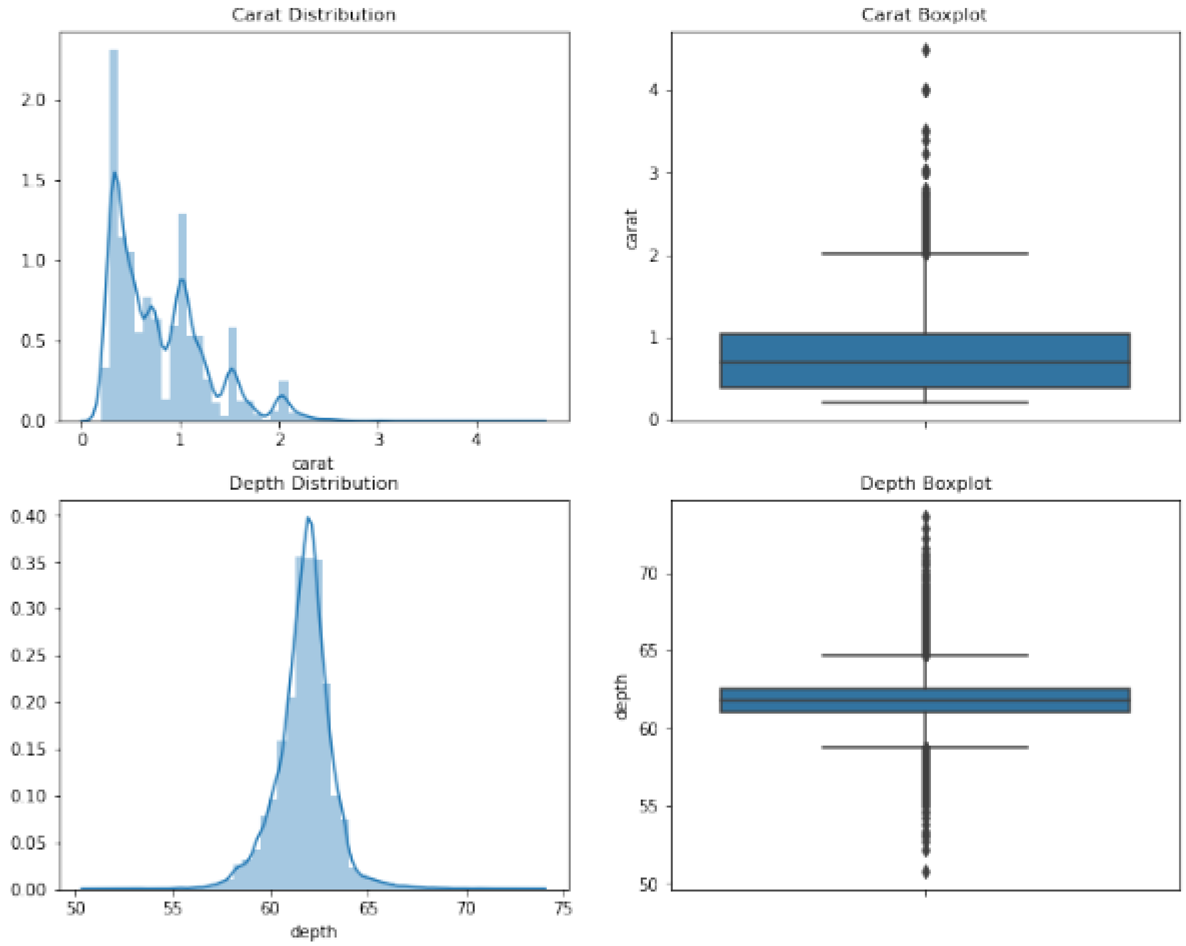
According to the dataset, ‘G’ color cubic zirconia is being manufactured comparatively more. And also, the company is least focused in manufacturing ‘J’ worst color in cubic zirconia.

- **Clarity:** Best to Worst, FL = flawless, I3= level 3 inclusions - FL, IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1, I2, I3

Lowest grade cubic zirconia (I2 and I3) is not at all manufactured by the company and also highest grade (FL). SI1 clarity are most commonly known as budget clarity cubic zirconia.

From this inference, we can say that the manufacturer is focused on cut of cubic zirconia by providing the best color and clarity for that cut. Upon further analysis we will be able to justify or find more about these categorical columns

Univariate Analysis of the dataset:



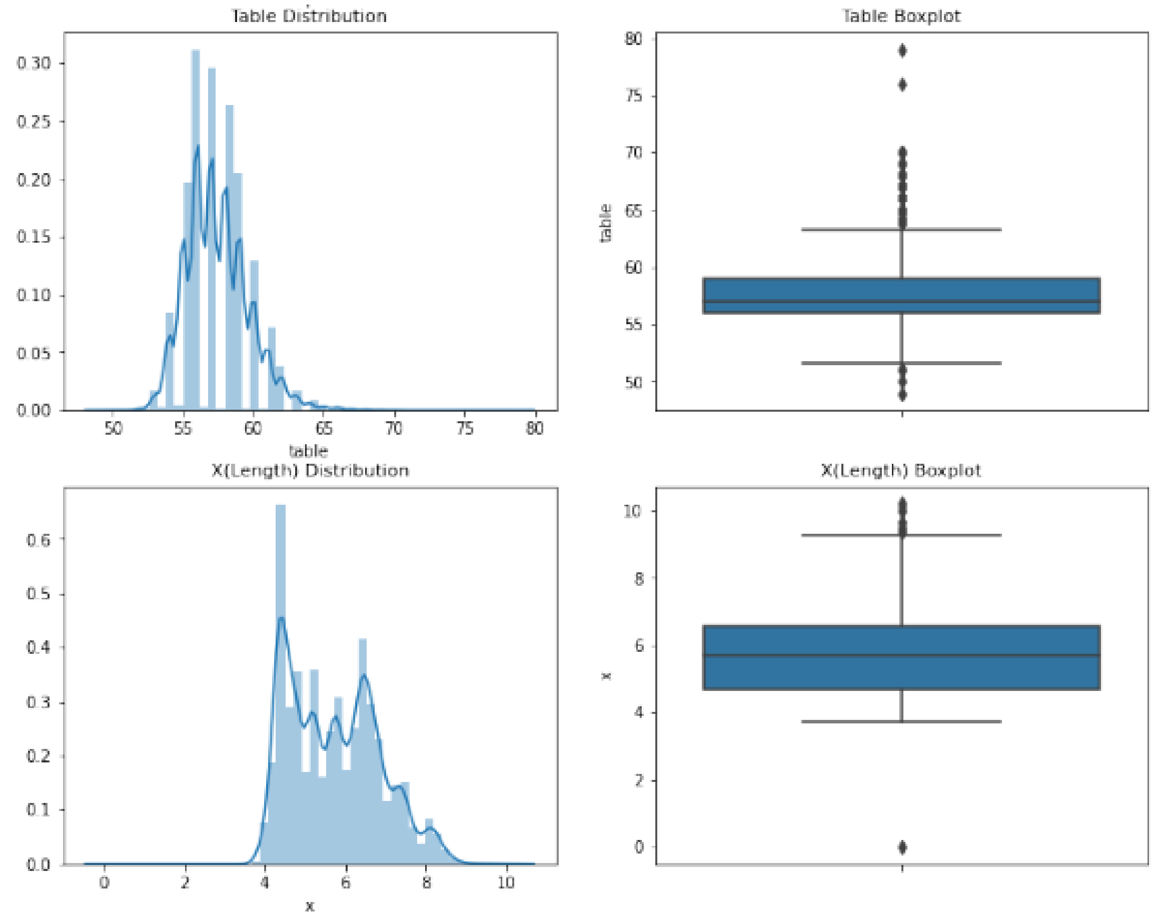
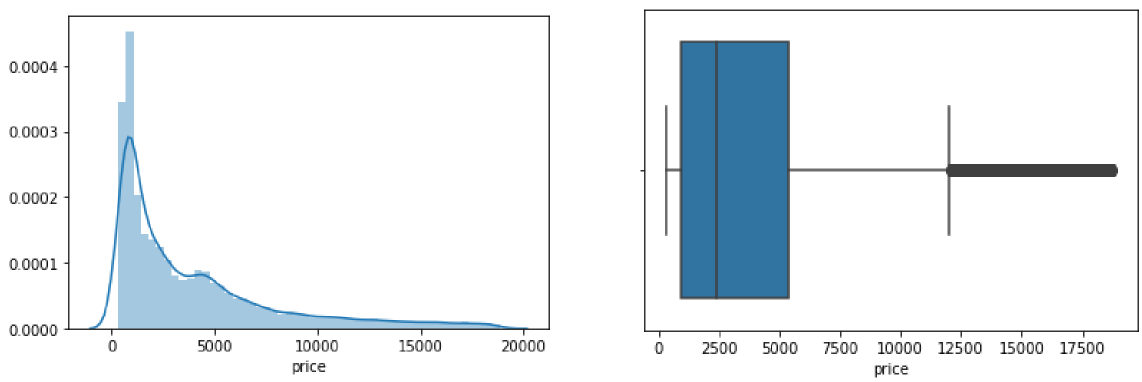
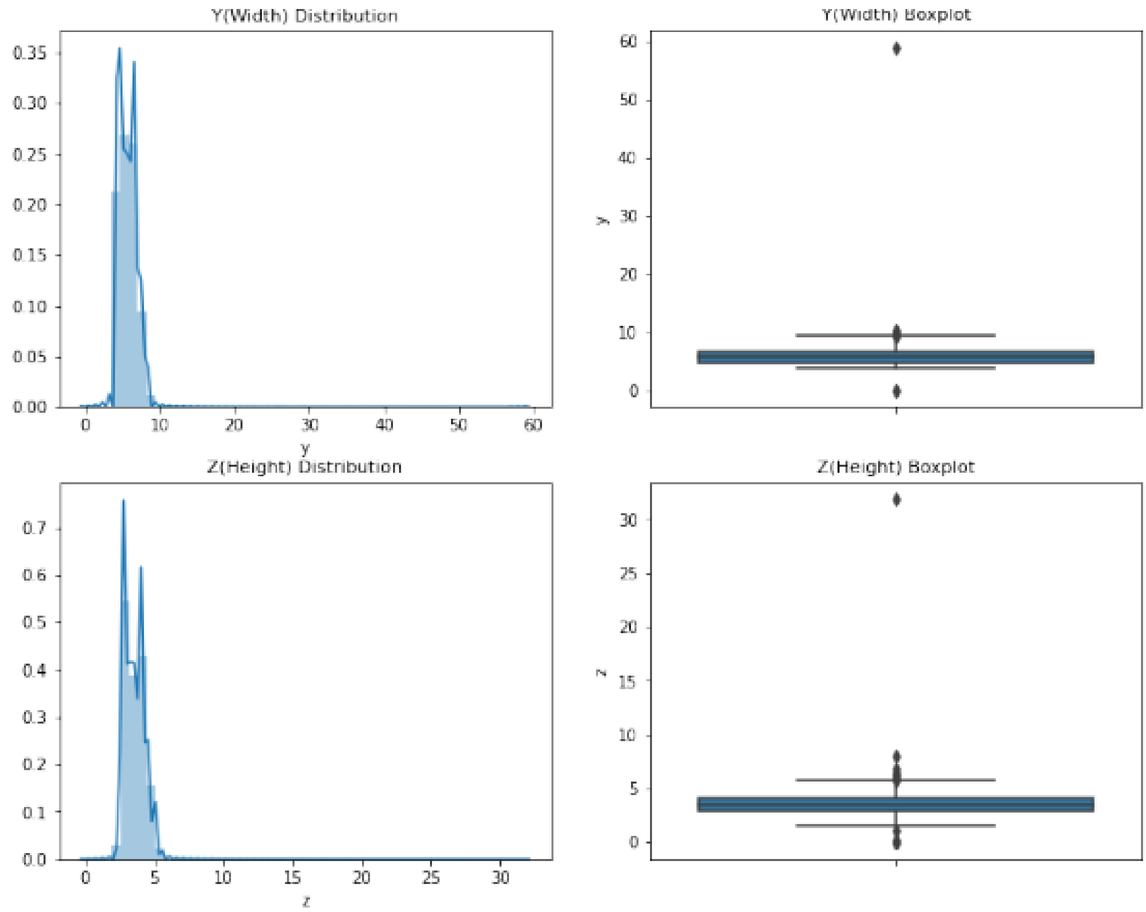


Table Distribution
table
Table Boxplot
table
X(Length) Distribution
x
X(Length) Boxplot
x


price
price

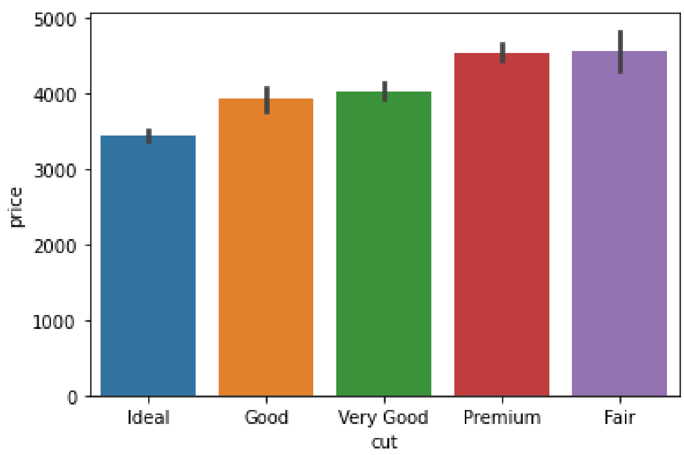


Inference: All variables have outliers present and also, we can observe that in depth, table, x, y and z there are several peaks which denotes clusters present in the dataset,

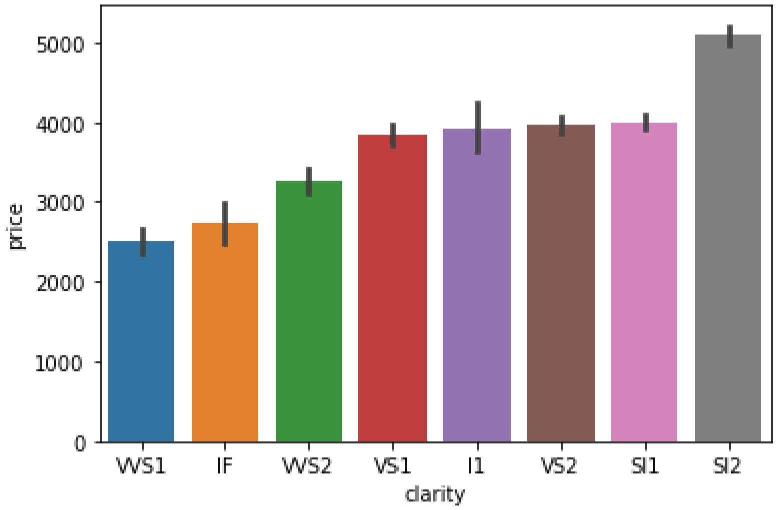
Outliers should be treated before building the model.

Bivariate analysis:

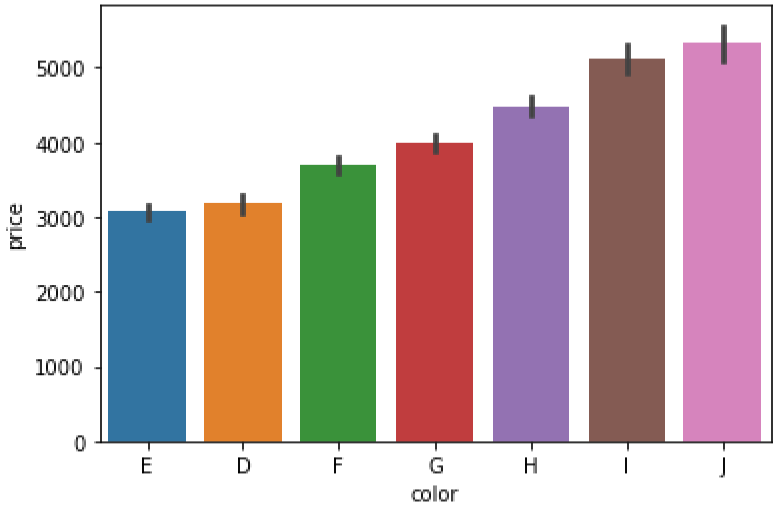
1. **Cut and Price:** Fair cut cubic zirconia are priced high whereas average price of ideal cut is less.



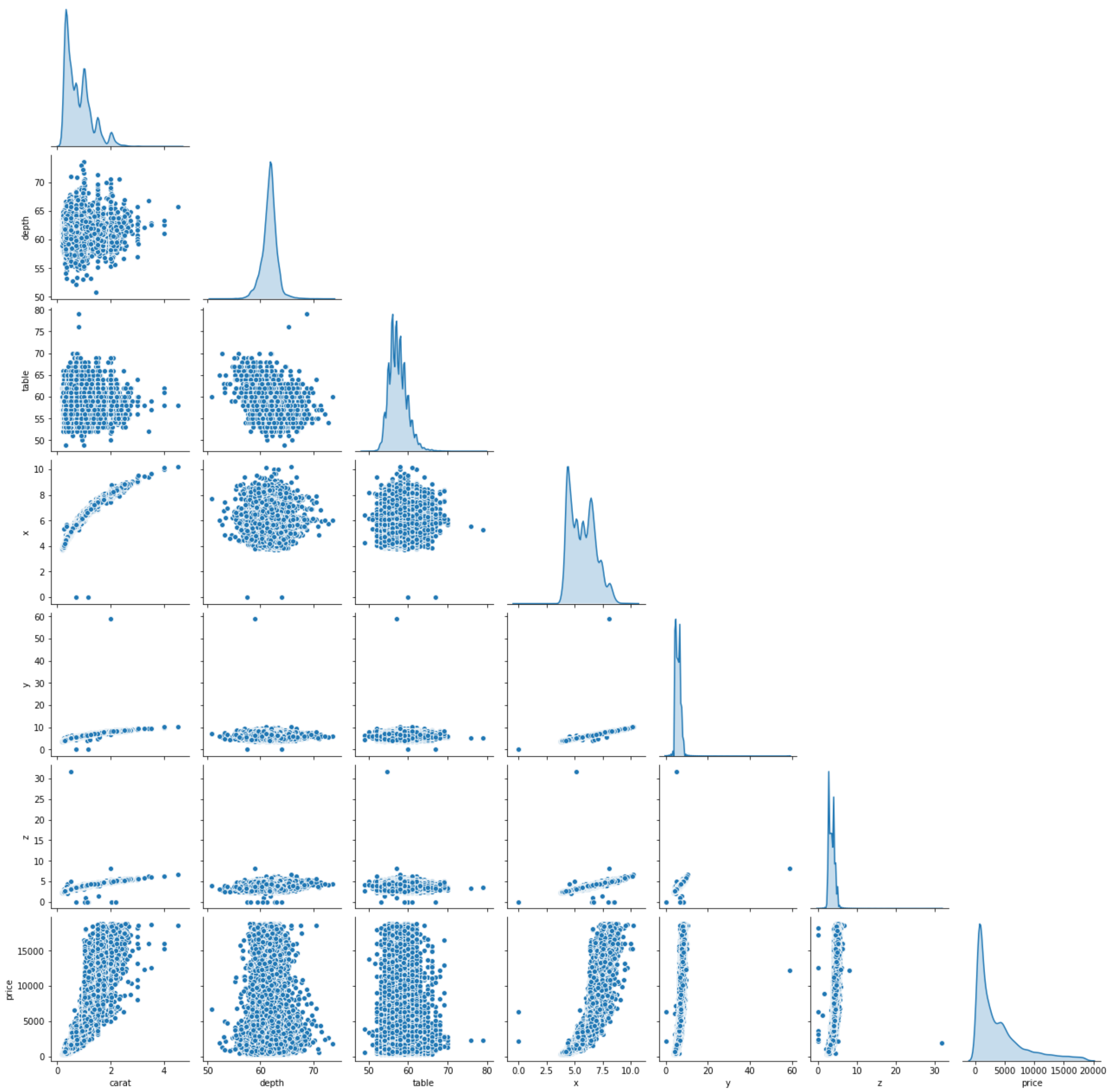
2. **Clarity and Price:** ‘SI2’ clarity cubic zirconia has high average price compared to others.



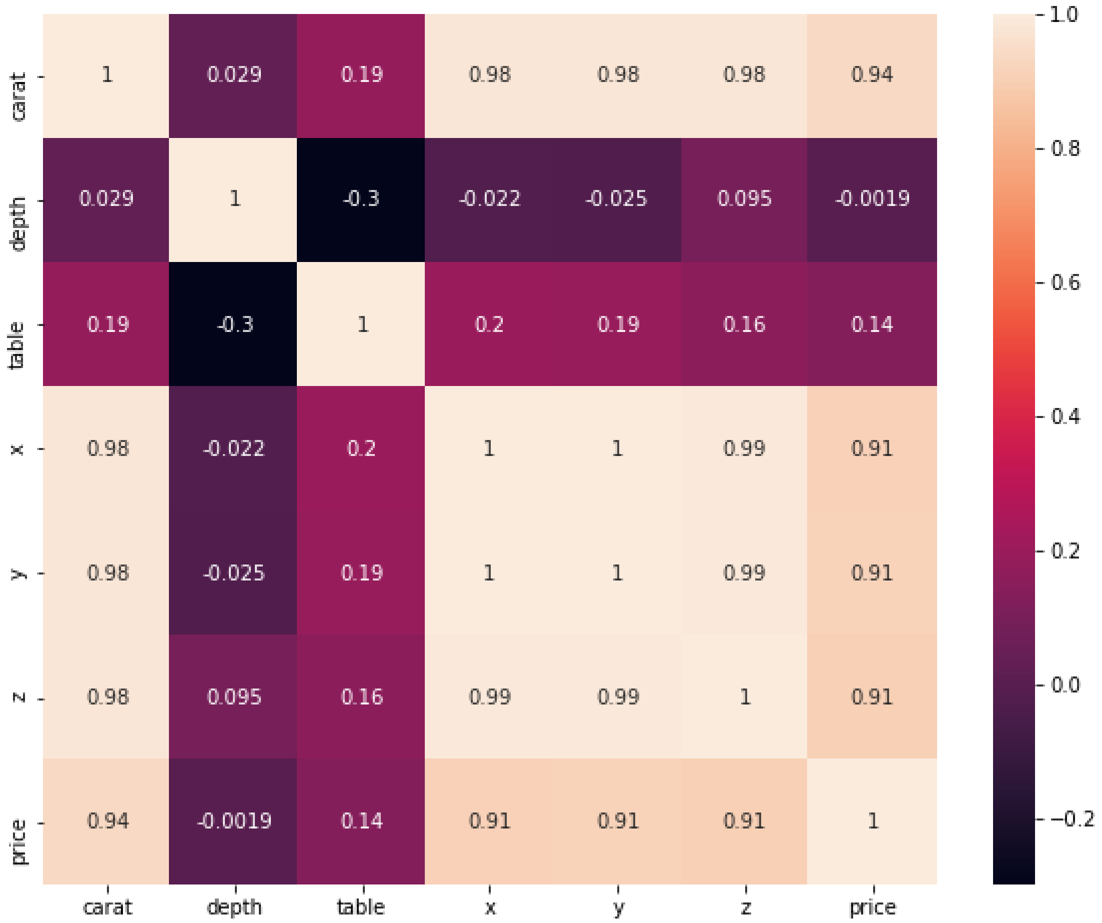
3. **Color and price:**



Multivariate Analysis using pair-plot:



Correlation plot:



Inference:

- There is heavy correlation between the variables x, y and z. These variables can affect the model performance due to this collinearity.
- Price has high correlation with carat weight.

1.2 Impute null values if present, also check for the values which are equal to zero. Do they have any meaning or do we need to change them or drop them? Do you think scaling is necessary in this case?

Imputation of Null values:

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

There are 697 null values in depth column.

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

697 rows × 10 columns

Depth is expressed as a percentage of cubic zirconia's height measured from the Culet to the table, divided by its average Girdle Diameter. Hence using the columns z(height) and y(width) to impute the missing NaN values in depth instead of going with the median.

After imputing,

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

Values that are equal to 0:

	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 observe that two records have 0 values in x, y and z. Instead of dropping the above rows, performing the below substitution for the corresponding columns.

Replacing the 0 values with median for column x and y.

Replacing the 0 values in z by using the same depth which was used before.

Z = (Depth * y(width)) / 100

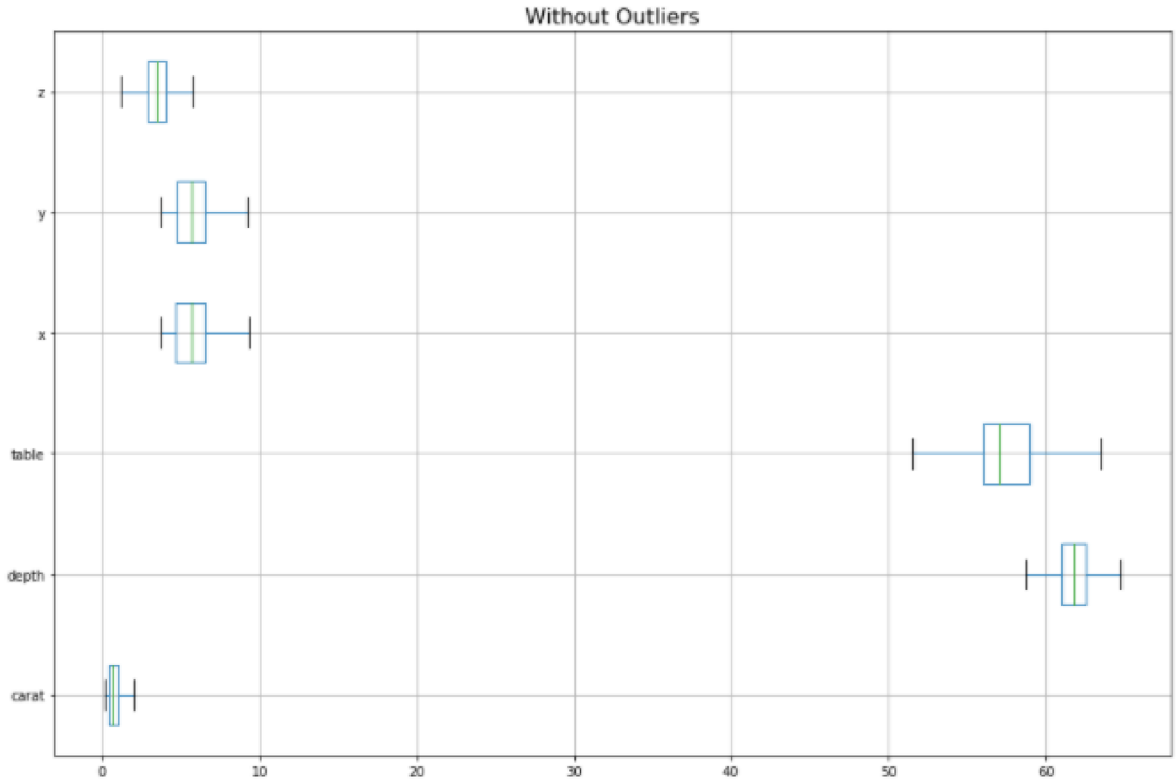
Verifying the index value [5821]:

```
carat        0.71
cut          Good
color           F
clarity       SI2
depth        64.1
table          60
x            5.69
y             5.7
z            3.65
price        2130
Name: 5821, dtype: object
```

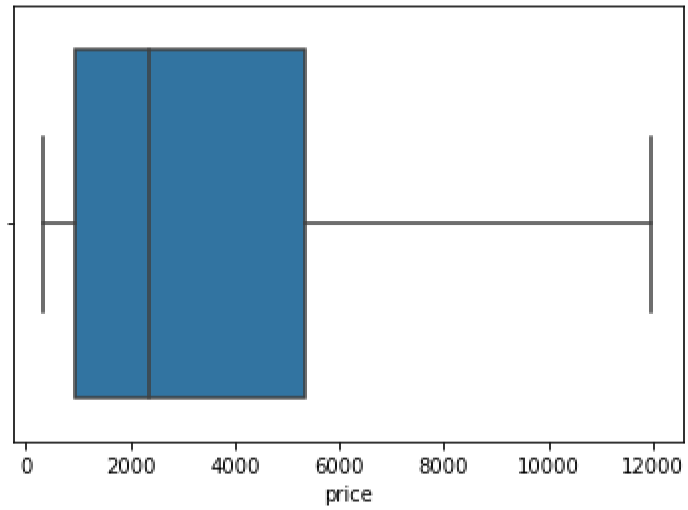
Outlier treatment: Before scaling the variables, since the dataset contains outliers, we have to treat the outliers in order for the output to be valid because if the scaling is done on the dataset with outliers then it would result in meaningless mean and standard deviation.

IQR treatment for outliers: Custom function is defined which takes column as input and returns two output for a particular column if the value is greater than maximum limit or less than minimum limit. Loop the function for all the variables such that it replaces the values greater than maximum limit by that limit and vice versa.

Boxplot of all variables after the outlier treatment:



Price variable after outlier treatment



Scaling the dataset:

Since the units of the independent variables are different, for example, x, y and z are represented in mm, carat is in weight, depth and table are represented as percentages. Also, the magnitude of each variables differs, depth and table are in 100s while rest of the numerical variables are within 10s range.

Even though all the variables are in numerical forms, it's not easy to compare them because of the units and range. Scaling will allow for all our data to be transformed to a more normal distribution. For this case study, standard scaler from sklearn has been used to transform the variables.

Scaling does not affect the model score or r^2 or coefficient of determinant. Trend of the predictor and predicted variables would remain the same. Intercept and coefficients of the features will change. This would remove any effects that would be present from one variable from having an incorrect magnitude of influence on our predictor variable.

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

Encoding the data having string values:

Linear regression requires the dependent and independent variables to be of numerical datatype.

Cut, Clarity and color have order within the values. Hence for this case study, we are encoding the variables with respect to the given order in the problem statement.

- Cut – Fair: 0, Good: 1, Very Good: 2, Premium: 3, Ideal: 4
- Color – J: 0, I: 1, H: 2, G: 3, F: 4, E: 5, D: 6
- Clarity – FL: 10, IF: 9, VVS1: 8, VVS2: 7, VS1: 6, VS2: 5, SI1: 4, SI2: 3, I1: 2, I2: 1, I3: 0

After encoding,

	carat	cut	color	clarity	depth	table	x	y	z	price
0	0.30	4	5	4	62.1	58.0	4.27	4.29	2.66	499.0
1	0.33	3	3	9	60.8	58.0	4.42	4.46	2.70	984.0
2	0.90	2	5	7	62.2	60.0	6.04	6.12	3.78	6289.0
3	0.42	4	4	6	61.6	56.0	4.82	4.80	2.96	1082.0
4	0.31	4	4	8	60.4	59.0	4.35	4.43	2.65	779.0

Linear Regression is a supervised learning technique which is linear combination of the explanatory variables in order to predict a dependent variable. Linear Regression model was performed for both scaled dataset and normal dataset after splitting the dataset into training and testing.

Model without scaling:

Coefficients of the features:

```
The coefficient for carat is 8898.078080443403
The coefficient for cut is 111.1965740535257
The coefficient for color is 278.50334439216533
The coefficient for clarity is 440.2993797749575
The coefficient for depth is 59.85821572547661
The coefficient for table is -12.350386424476934
The coefficient for x is -1109.702804699163
The coefficient for y is 1562.6820064690473
The coefficient for z is -1359.0775106853498
```

Intercept

```
The intercept for our model is -7589.609520945982
```

Model with scaling:

Coefficients of the features:

```
The coefficient for carat is 1.185638455357753
The coefficient for cut is 0.03568993324947086
The coefficient for color is 0.13698553437075517
The coefficient for clarity is 0.2090597038409563
The coefficient for depth is 0.02175784449911442
The coefficient for table is -0.007681564646 2652385
The coefficient for x is -0.36018521463020514
The coefficient for y is 0.5035891704677921
The coefficient for z is -0.2726522844705763
```

Intercept

```
The intercept for our model is 0.001197154302511145
```

We can infer from above that, after scaling the dataset the coefficients are interpretable and the intercept became close to 0 since the data is centred.

Table, x(length) and z(height) are having a negative relationship with the target variable.

Performance metrics: Having a look at R^2, RMSE values of the model.

R^2 statistical measure: It is to determine how close the data is to our fitted regression line (best fit line). Higher the value of r-squared the better the model.

Performance metrics	**Training data**	**Testing data**
R-squared	0.931	0.931
RMSE	0.262	0.262

RMSE: Relative distance between the predicted and actual values. Lower the RMSE better the model. Since in this case study we are just using one Linear Regression model we don't have other model results to compare the RMSE.

Looking at our metrics measures, we can say that our model is able to capture 93.1% of variability around the mean projection.

In order to find the best features out of the lot, we will be using statsmodels OLS method for getting the P-values of each variable and by comparing with the adjusted r-squared metric.

Adjusted r-squared metric: This metric is useful when we have more than one variable. As we add more independent variable r-square will go up irrespective of the goodness of the variable. Whereas adjusted r-square will penalise the variable if it's not a good predictor, value comes down.

In the given dataset, there are a total of 9 independent variables, for finding out the best out of this we will be using p-value derived from the statsmodels.

Upon trying different combinations to find the relevant variables, there are five variables which contributes to the total adjusted r-squared value, which means those five variables will be enough to predict the target 'price' variable.

Null hypothesis claims that there is no relationship between the target and independent variables

- When p-value for a variable is greater than 0.05, that means the variable is useless and the correlation is by chance.
- When p-value is less than 0.05, We can reject the null hypothesis and say that there is relationship between the target and independent variables.

OLS Regression Results

Dep. Variable:	price	R-squared:	0.931
Model:	OLS	Adj. R-squared:	0.931
Method:	Least Squares	F-statistic:	5.057e+04
Date:	Tue, 12 Jan 2021	Prob (F-statistic):	0.00
Time:	18:22:15	Log-Likelihood:	-1598.1
No. Observations:	18853	AIC:	3208.
Df Residuals:	18847	BIC:	3255.
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1.475e-17	0.002	-7.69e-15	1.000	-0.004	0.004
carat	1.1699	0.011	110.872	0.000	1.149	1.191
cut	0.0402	0.002	20.473	0.000	0.036	0.044
clarity	0.2125	0.002	100.640	0.000	0.208	0.217
color	0.1372	0.002	67.706	0.000	0.133	0.141
x	-0.1159	0.011	-10.968	0.000	-0.137	-0.095

Omnibus:	2664.875	Durbin-Watson:	1.988
Prob(Omnibus):	0.000	Jarque-Bera (JB):	9260.165
Skew:	0.704	Prob(JB):	0.00
Kurtosis:	6.131	Cond. No.	11.9

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

The p-value is 0 is variables: carat, cut, clarity, color, x (length)

Moreover, the above variables are itself enough to explain the maximum variability. Adding extra variables like y, z, depth and table does not increase/decrease the r-squared value.

Final linear regression model:

The final Linear Regression equation is

price = b0 + b1 * carat + b2 * cut + b3 * clarity + b4 * color + b5 * x

price = Intercept (0.00) + (1.17) * carat + (0.04) * cut + (0.21) * clarity + (0.14) * color +

(-0.12) * x

When carat increases by 1 unit, price increases by 1.17 standard deviation units, keeping all other predictors constant.

There are also some negative co-efficient values, for instance, x (length) has its corresponding co-efficient as -0.12. This implies, when the x increases by 1 unit, the price decreases by -0.12 standard deviation units, keeping all other predictors constant.

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

Business objective: The company is earning different profits on different prize slots. By predicting the price for the stone on the basis 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.

Important features that we got from model building are carat, cut, clarity, color and x.

Business insights from the dataset:

Carat: Generally, price of the cubic zirconia increases with carat weight. Binning the carat weight into four groups and comparing the mean values of price of each group.

1: 0.2 – 0.8, 2: 0.8 – 1.4, 3: 1.4 – 2.0, 4: 2.0 – 2.7 carat weight units respectfully.

carat_bin	price
1	1362.824217
2	5536.455286
3	10062.361163
4	11701.545000

Color: From the initial analysis, we observe that worst color cubic zirconia's are priced higher than the best ones.

Cut: Fair cut diamonds are priced higher than Ideal cut.

Clarity: Medium clarity diamonds 'SI2' are priced higher than the rest comparatively.

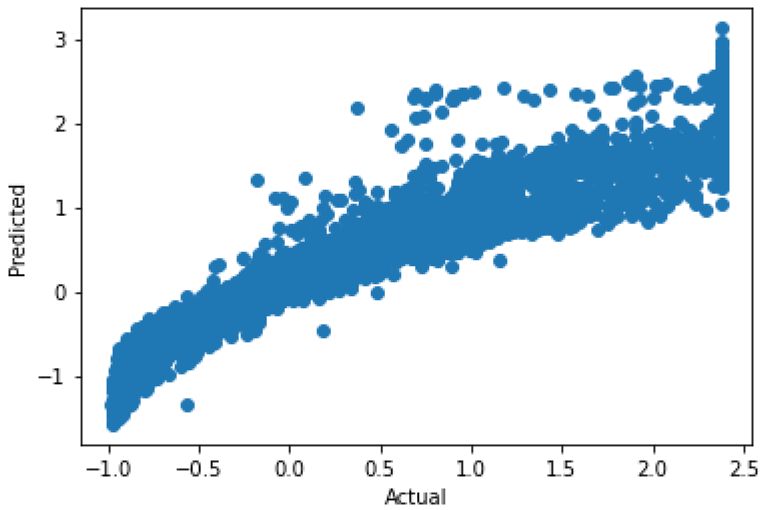
From the regression model built, we can say that the price is not significantly affected by the increase in cut and x(length).

The model built has a predictive power of 93.1%, but the assumptions of linear regression model are not all satisfied. Because of the multicollinearity within the dataset, there are certain features like x, y and z which are highly correlated.

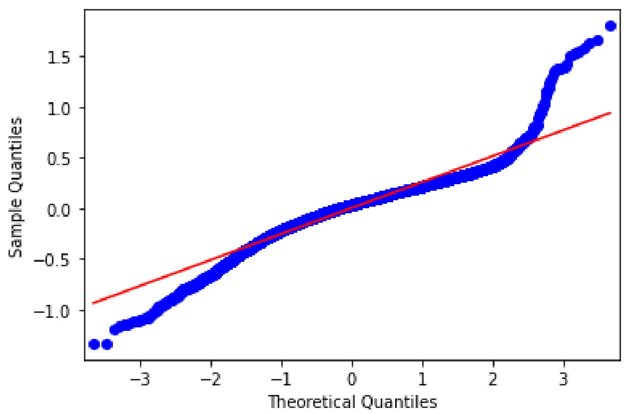
Variation inflation factor to test multicollinearity:

```
carat ---> 32.88401661704267
cut ---> 1.5098921180400915
color ---> 1.120143093473736
clarity ---> 1.241949390809762
depth ---> 5.173833861687716
table ---> 1.6313990083249286
x ---> 423.07323589567164
y ---> 412.65427848166877
z ---> 269.9228023449004
```

A scatter plot between the actual and predicted values of ‘y’:



Non-normality in the data:



Residuals plot is used in order to determine the normality. There are some curves at the starting and ending of the line which indicates non-normal distribution of the dataset which could be caused due to the clusters which was observed.

Business recommendations:

Our goal is to predict price so that we can improve profit share by distinguishing between higher profitable stones and lower profitable stones.

In order to increase the profit margin,

- Manufacturer can focus entirely on the carat weight of the stone irrespective of the cut, clarity and color. Since the price of the stone increases with carat weight. This trend could generate more profit.
- Nearly 40% of the stones manufactured by the company have 'Ideal' cut but the price of these cut stones is comparatively less than 'Fair' cut. Moderating this price range with respect to the cut quality can contribute more to profit.
- With respect to the color feature, looks like the company has priced the least color quality highly. If the carat weight, clarity and cut dominates, then this over-pricing can be neglected. Since the color of the stone is mostly not visible to the naked eye, this feature is lightly monitored by the company. And also, consumers will be mainly focused on the carat weight rather than color of the stone.
- Since high clarity stones are not that easy to manufacture, company's focus is mainly on 'SI1' clarity grade but the price of SI1 clarity stone is less than the average price of SI2 clarity. Hence if the manufacturer can increase the production of SI2 clarity stones that can yield a higher profitable stone.

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.

1.1 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:

Head of the dataset: Verify whether the dataset is loaded correctly

	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

Information of the dataset:

```
<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
```

Two object datatypes (Holliday_Package and foreign) needs to converted to numerical datatype. There are no NaN values in the dataset.

Shape of the dataset:

```
There are  872  rows and  7  columns in the dataset
```

Summary statistics of numerical datatypes:

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

In the given dataset, we cannot consider any of the columns maximum value or minimum value as outlier. Since they look legitimate under practical circumstances.

Summary statistics of categorical datatypes:

	Holliday_Package	foreign
count	872	872
unique	2	2
top	no	no
freq	471	656

'Holliday_Package' is the target variable which has nearly 54% of the observations under 'No' category. Hence the models will be accurate enough to predict these observations.

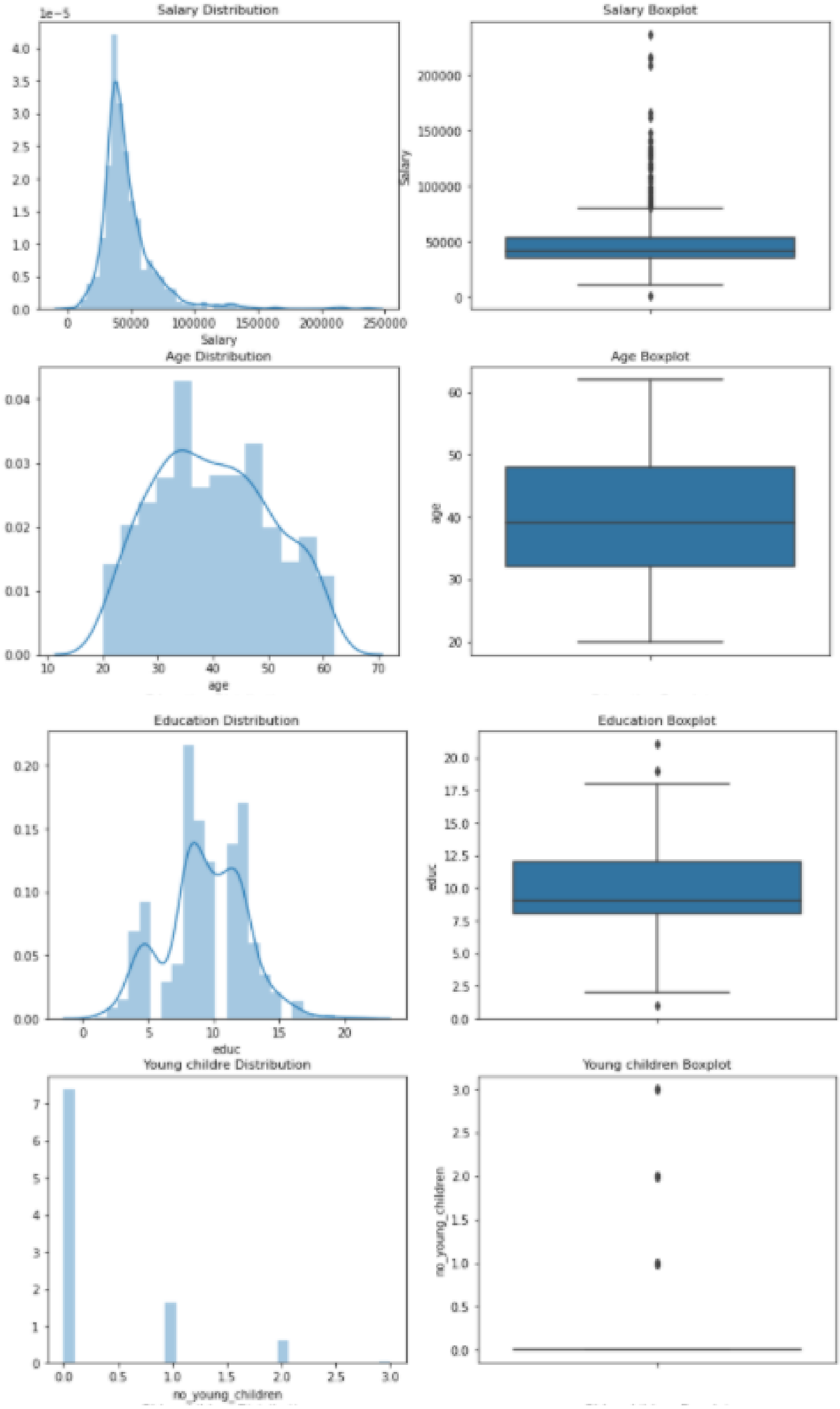
75% of the records are having 'No' in the foreign column.

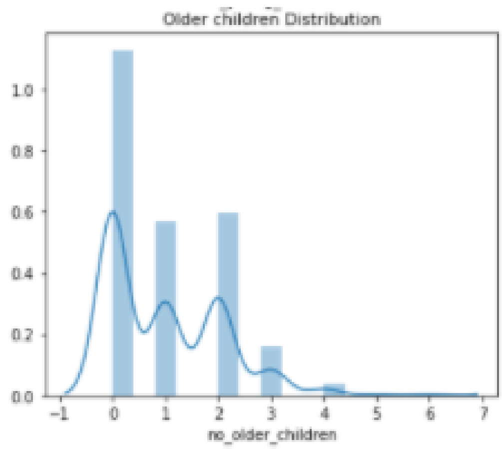
Missing values/Duplicate records check.

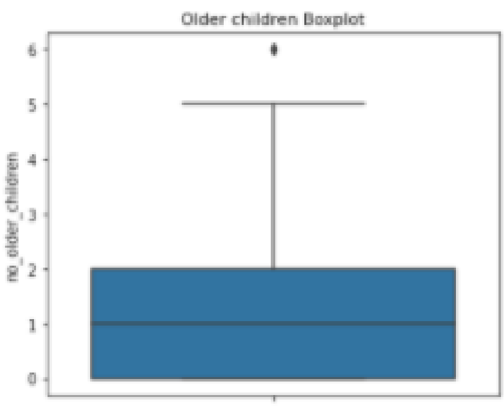
```
Holliday_Package     0
Salary               0
age                  0
educ                 0
no_young_children    0
no_older_children    0
foreign              0
dtype: int64
```

There are no missing values or duplicate records in the dataset.

Univariate analysis of the variables:





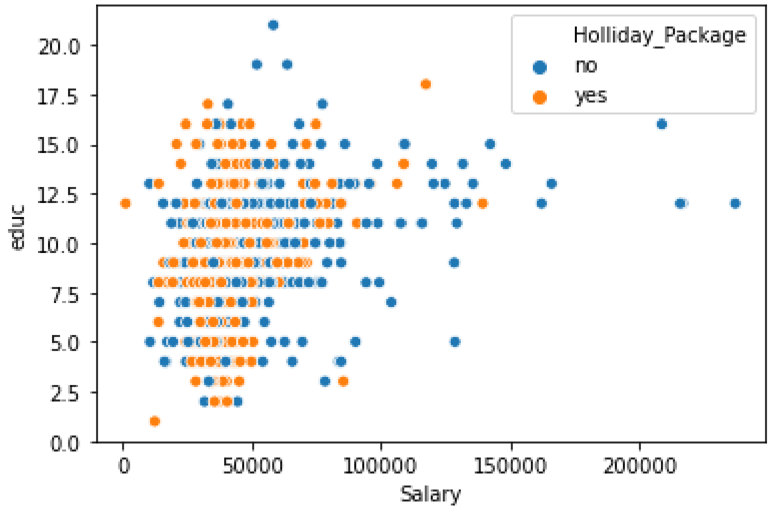


Inferences:

- Salary is right skewed whereas Age follows a normal distribution with no outliers.
- Education, Young children and older children distribution are having outliers. But for this case study outlier treatment is not being performed since these are some valid scenarios.
- There are clusters in education, young children and older children distribution.

Bivariate analysis with few variables:

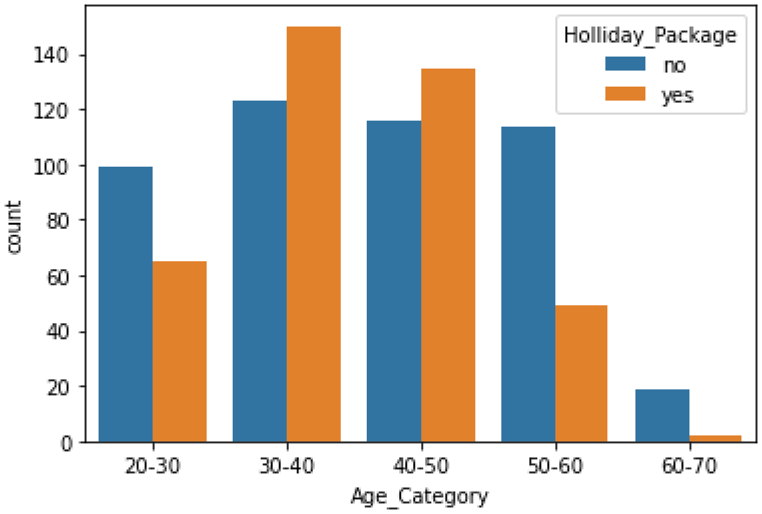
- **Scatter plot between Salary and education with hue as Holliday_Package:**



We could infer that people whose salary is high or if they have more years of experience, they are not opting for holiday package. These can be outliers, but in practical terms, situations seem possible.

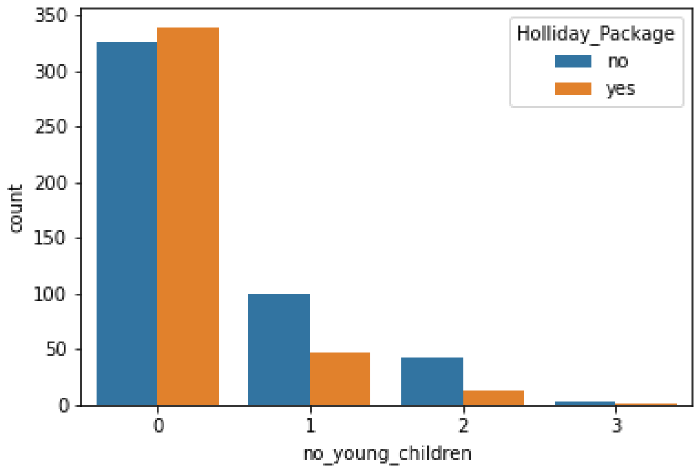
- **Count plot for Age and Holliday_Package:**

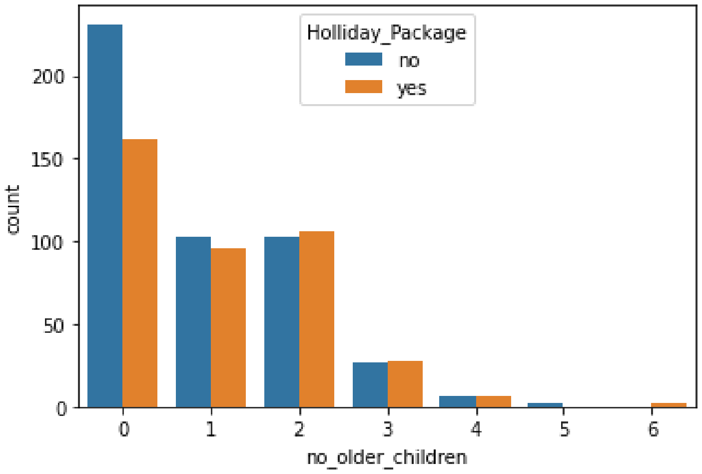
Creating bins for Age column and comparing it with the target variable.



Looking at the age category, people between the ages 30-50 are opting for holiday package.

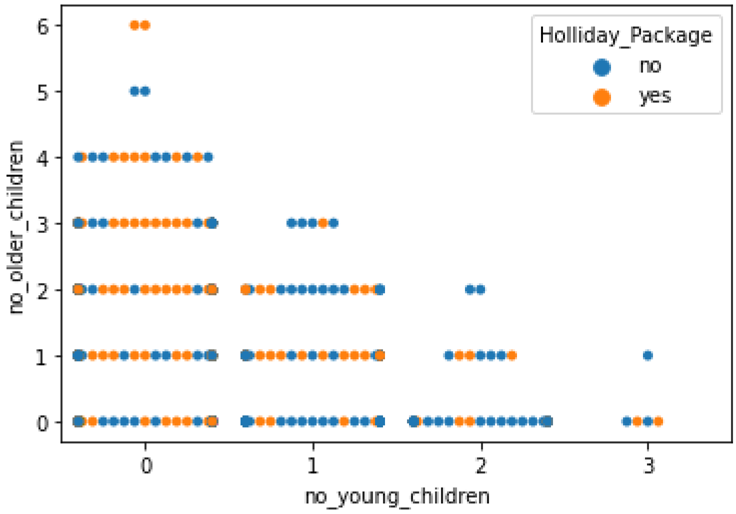
- **Count plot of No. of young children and no. of older children with target:**





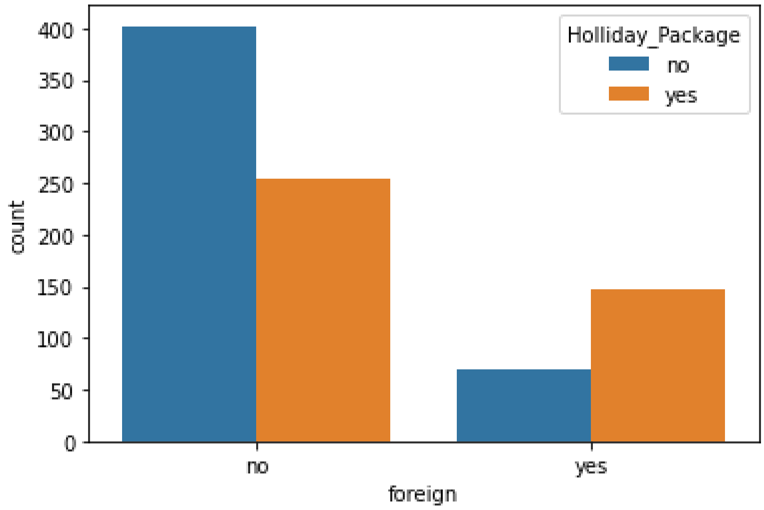
Families not having young and older children are more. Those with no young children are highly likely to opt for holiday package and families with 2 and 3 older children are opting for the package.

Combining the two features with holiday package as hue.



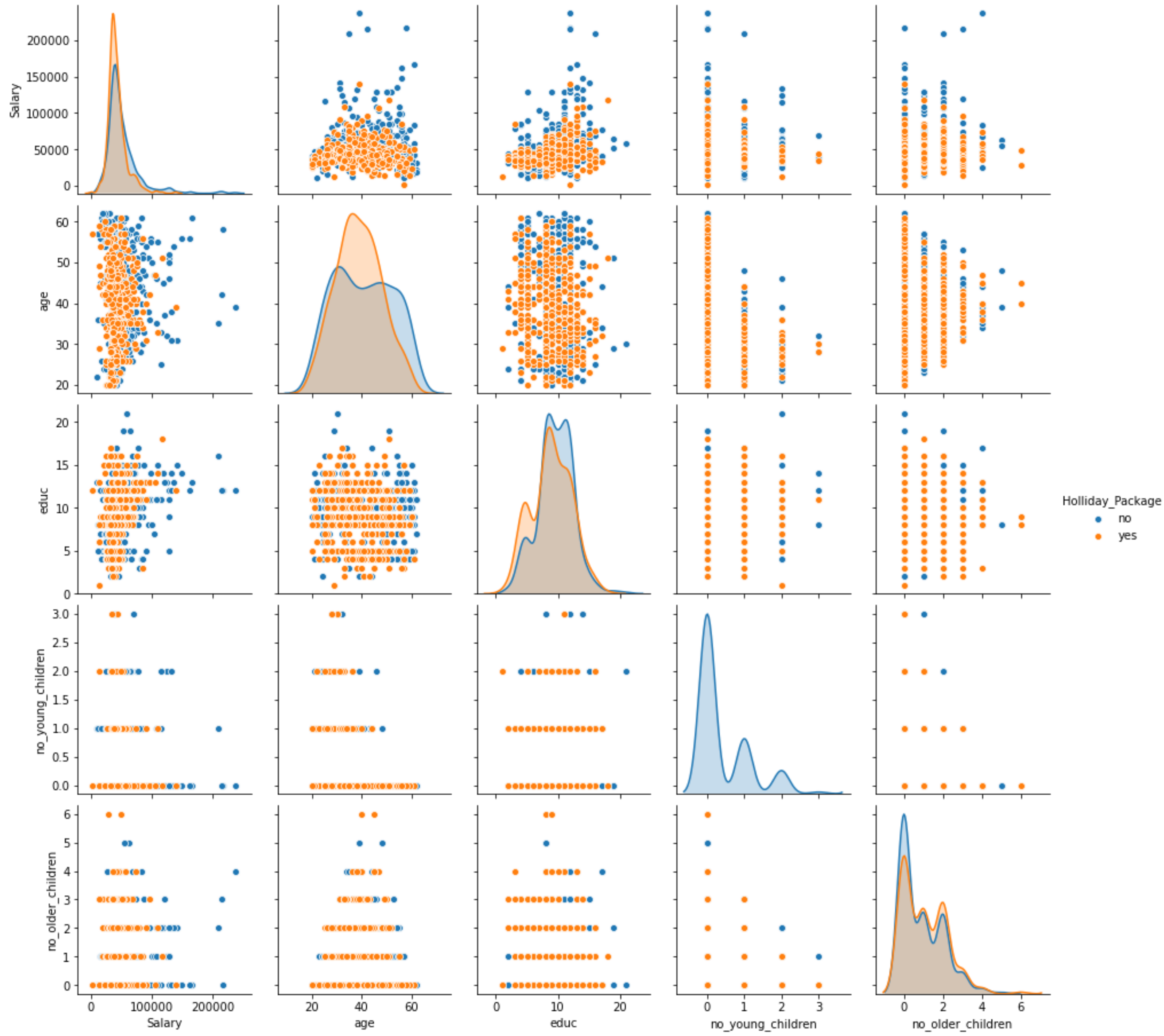
The trend shows that as the number of young children increases number of older children decreases which seems obvious. Families with no young children are opting for holiday package than the rest.

- **Count plot of foreign and holiday package**



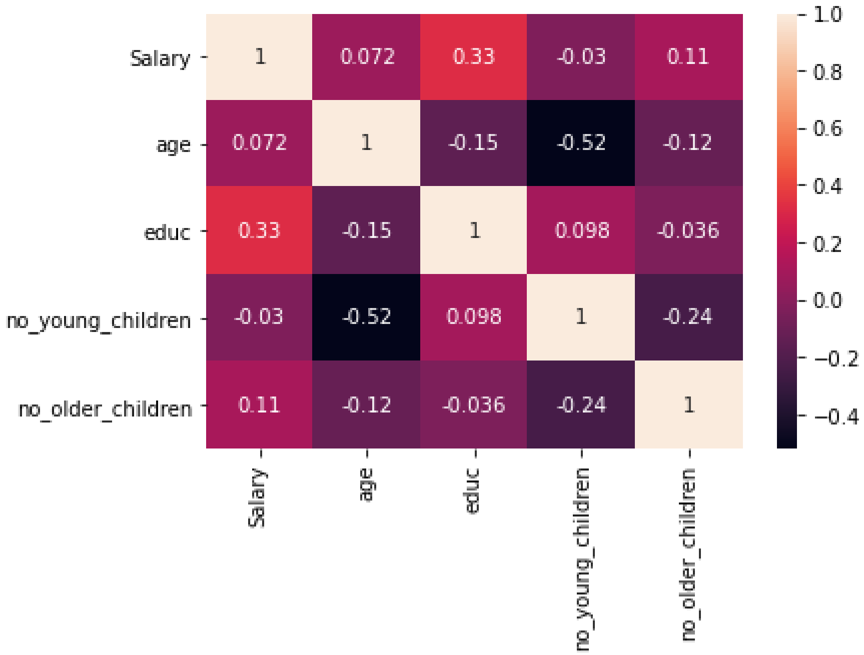
Native families are less likely to opt for holiday package than foreign.

Multivariate analysis:



There is overlap between 'yes' and 'no' category of holiday package in almost all variables which says that by looking at the pair plot alone we cannot decide the strong predictors for target variable. No. of young children column has only distribution of 'no' category since there might be very few observations who have opted for holiday package with a greater number of young children.

Correlation plot:



There is no correlation between the variables in the dataset.

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).

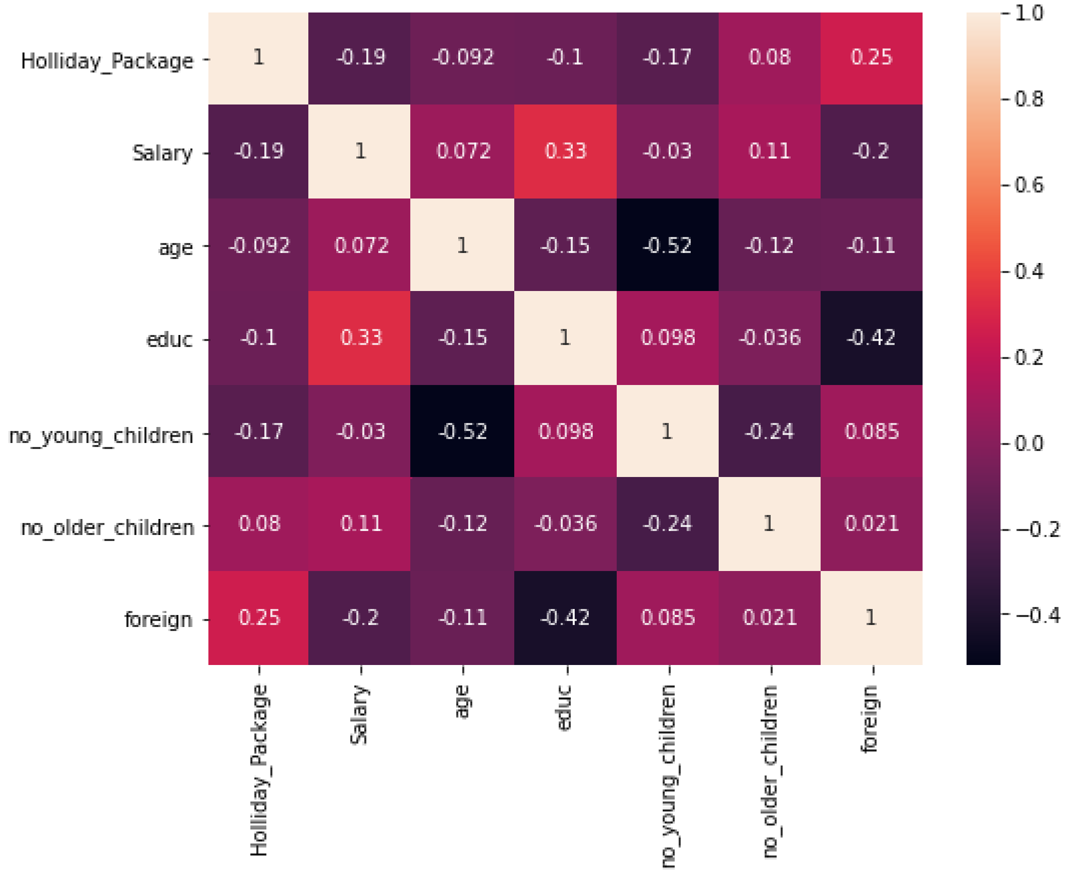
Encoding the columns with string value: Using Pd.Categorical and getting the codes of each value since there are only two categories.

```
feature: Holliday_Package
['no', 'yes']
Categories (2, object): ['no', 'yes']
[0 1]


feature: foreign
['no', 'yes']
Categories (2, object): ['no', 'yes']
[0 1]
```

	Holliday_Package	Salary	age	educ	no_young_children	no_older_children	foreign	Age_Category
0	0	48412	30	8	1	1	0	30-40
1	1	37207	45	8	0	1	0	40-50
2	0	58022	46	9	0	0	0	40-50
3	0	66503	31	11	2	0	0	30-40
4	0	66734	44	12	0	2	0	40-50

Correlation between the target (Holliday_Package) and other variables after encoding the dataset:



There is hardly any correlation within the variables.

- **First step** of building a model is to **Separate the dataset into X and y variable.**

For the given business problem of tour and travel agency, 'Holliday_Package' is the target variable since the problem is to come up with a model to predict whether an employee will opt for package or not.

X – Independent variable (Removing 'Holliday_Package' variable)

Y – Dependent/ Target variable (Having only 'Holliday' variable)

- **Second step** is to **Split the data into training and testing test.**

Splitting the data as 70% training and 30% testing.

Output of this step will be: Training independent variable (X_train), Testing independent variable (X-test), Training dependent variable (train_labels) and testing dependent variable (test_labels).

- **Third step is to build model for each LDA and Logistic Regression and fourth step is to predict on training and testing set**

Logistic Regression is a supervised learning method for classification. It establishes relationship between dependent class variables and independent class variables using regression. Logistic regression assign probabilities to different classes to which a data point is likely to belong. In order to do this, the classifier takes the weighted sum of the features and bias to represent the class of interest of a particular data point, this linear output is passed through a sigmoid function in order to get the values between the range (0,1).

Using **Logit function from statsmodels** in order to determine the p-value of variables and to determine if it's a good predictor.

```
Optimization terminated successfully.
         Current function value: 0.612003
         Iterations 5
                         Results: Logit
=================================================================
Model:              Logit            Pseudo R-squared: 0.113
Dependent Variable: Holliday_Package AIC:              1079.3335
Date:               2021-01-14 10:20 BIC:              1107.9582
No. Observations:   872              Log-Likelihood:   -533.67
Df Model:           5                LL-Null:          -601.61
Df Residuals:       866              LLR p-value:      1.3367e-27
Converged:          1.0000           Scale:            1.0000
No. Iterations:     5.0000
-----------------------------------------------------------------
                   Coef.  Std.Err.    z    P>|z|  [0.025  0.975]
-----------------------------------------------------------------
Salary            -0.0000   0.0000 -3.8962 0.0001 -0.0000 -0.0000
age               -0.0173   0.0051 -3.3697 0.0008 -0.0273 -0.0072
educ               0.1105   0.0241  4.5779 0.0000  0.0632  0.1579
no_young_children -0.9674   0.1518 -6.3732 0.0000 -1.2649 -0.6699
no_older_children  0.0924   0.0678  1.3623 0.1731 -0.0405  0.2252
foreign            1.6075   0.1891  8.5020 0.0000  1.2369  1.9781
=================================================================
```

p-values for all variables are less than 0.05 except ‘no_older_children’. Hence removing it for further model building.

```
Optimization terminated successfully.
         Current function value: 0.613067
         Iterations 5
                           Logit Regression Results                           
==============================================================================
Dep. Variable:       Holliday_Package   No. Observations:                  872
Model:                          Logit   Df Residuals:                      867
Method:                           MLE   Df Model:                            4
Date:                Sat, 16 Jan 2021   Pseudo R-squ.:                  0.1114
Time:                        20:07:45   Log-Likelihood:                -534.59
converged:                       True   LL-Null:                       -601.61
Covariance Type:            nonrobust   LLR p-value:                 5.338e-28
=====================================================================================
                        coef    std err          z      P>|z|      [0.025      0.975]
-------------------------------------------------------------------------------------
Salary            -1.472e-05   3.93e-06     -3.742      0.000   -2.24e-05   -7.01e-06
age                  -0.0175      0.005     -3.417      0.001      -0.027      -0.007
educ                  0.1156      0.024      4.846      0.000       0.069       0.162
no_young_children    -1.0098      0.149     -6.774      0.000      -1.302      -0.718
foreign               1.6473      0.187      8.801      0.000       1.280       2.014
=====================================================================================
```

Grid search CV parameters used:

```
GridSearchCV(cv=3, estimator=LogisticRegression(max_iter=10000, verbose=True),
             n_jobs=-1,
             param_grid={'penalty': ['l1', 'l2', 'elastic-net', 'none'],
                         'solver': ['sag', 'lbfgs', 'liblinear', 'newton-dg',
                                    'saga'],
                         'tol': [0.0001, 1e-06]},
             scoring='f1')
```

Best parameters obtained:

```
{'penalty': 'l1', 'solver': 'liblinear', 'tol': 1e-06}

LogisticRegression(max_iter=10000, penalty='l1', solver='liblinear', tol=1e-06,
                   verbose=True)
```

Fit the model and predict the probabilities:

	0	1
0	0.677952	0.322048
1	0.568767	0.431233
2	0.689577	0.310423
3	0.516185	0.483815
4	0.541628	0.458372

Linear Discriminant Analysis is a linear classification machine learning algorithm.

The algorithm involves developing a probabilistic model per class based on the specific distribution of observations for each input variable. A new data point is then classified by calculating the conditional probability of it belonging to each class and selecting the class with the highest probability.

This model is useful when we have independent variables are a clear distinguishers of target variable.

Grid search CV parameters used:

```
GridSearchCV(cv=3, estimator=LinearDiscriminantAnalysis(), n_jobs=-1,
             param_grid={'solver': ['svd', 'lsqr', 'eigen'],
                         'tol': [0.0001, 1e-05]},
             scoring='accuracy')
```

Best parameters obtained:

```
{'solver': 'svd', 'tol': 0.0001}

LinearDiscriminantAnalysis()
```

Fit the model and predict the probabilities:

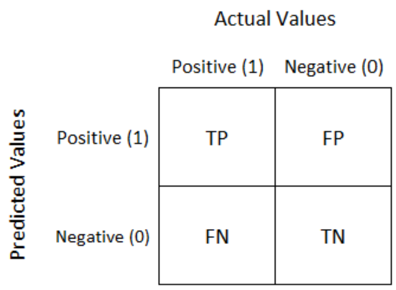
	0	1
0	0.711940	0.288060
1	0.544269	0.455731
2	0.721473	0.278527
3	0.500140	0.499860
4	0.539389	0.460611

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.

Fifth step of the model is to evaluate it and see how good it will perform for future records.

Some of the model evaluation techniques are:

- Accuracy – how precisely the model classifies the data points.
- Confusion Matrix – 2 * 2 tabular structure reflecting the model performance in four blocks



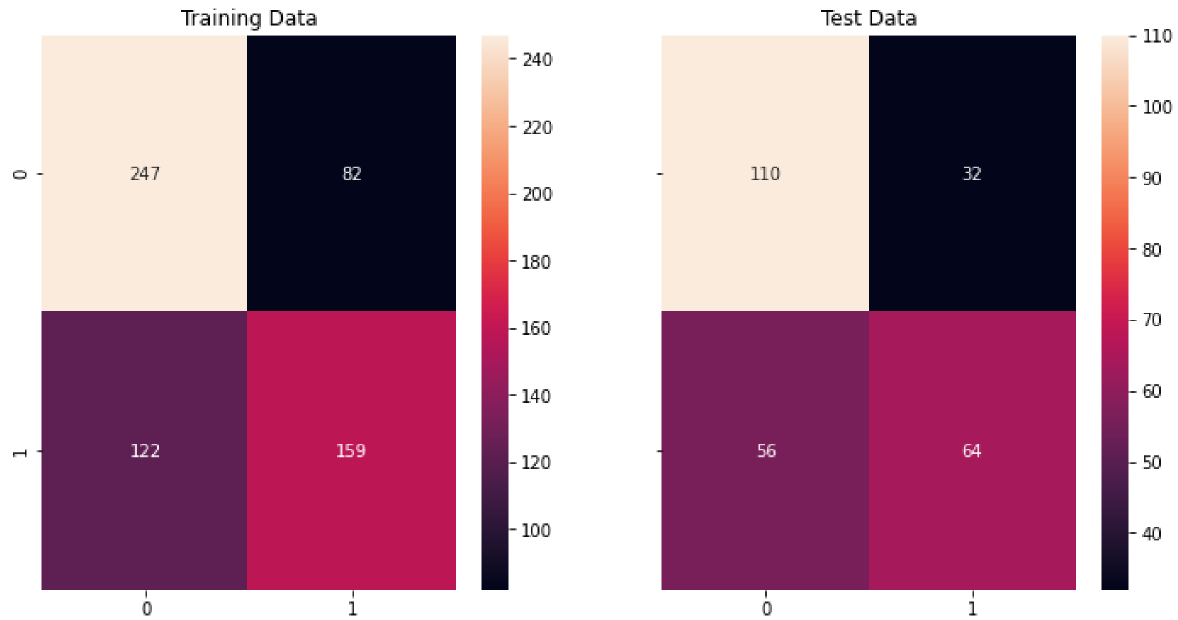
- Receiver operating characteristics (ROC) curve – A technique to visualize classifier performance
- ROC_AUC score – Area under curve, which is by calculating the percentage area below the curve.

Logistic Regression performance metrics:

Accuracy score:

```
Accuracy of training data:  0.6655737704918033
Accuracy of testing data:  0.6641221374045801
```

Confusion matrix:



Classification report:

Classification Report of the training data:

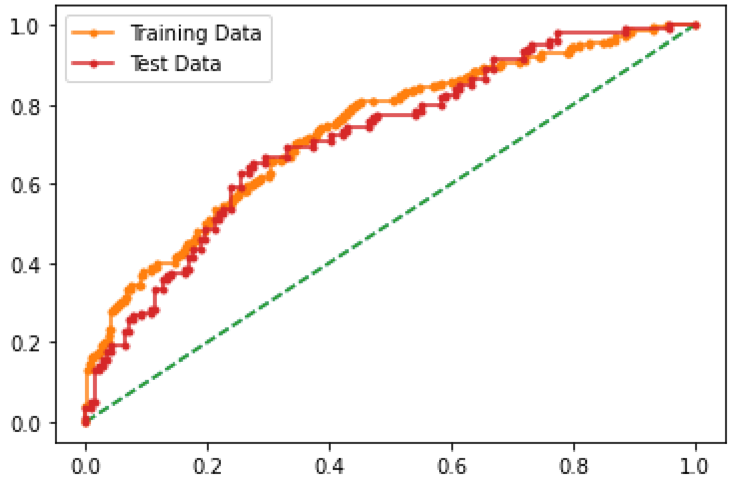
	precision	recall	f1-score	support
0	0.67	0.75	0.71	329
1	0.66	0.57	0.61	281
accuracy			0.67	610
macro avg	0.66	0.66	0.66	610
weighted avg	0.66	0.67	0.66	610

Classification Report of the test data:

	precision	recall	f1-score	support
0	0.66	0.77	0.71	142
1	0.67	0.53	0.59	120
accuracy			0.66	262
macro avg	0.66	0.65	0.65	262
weighted avg	0.66	0.66	0.66	262

AUC-ROC curve for training and testing data:

AUC for the Training Data: 0.734
AUC for the Test Data: 0.718



Inference:

Metrics	**Training set**	**Testing set**
Accuracy	0.66	0.66
Precision	0.66	0.67
Recall	0.57	0.53
F1 score	0.61	0.59

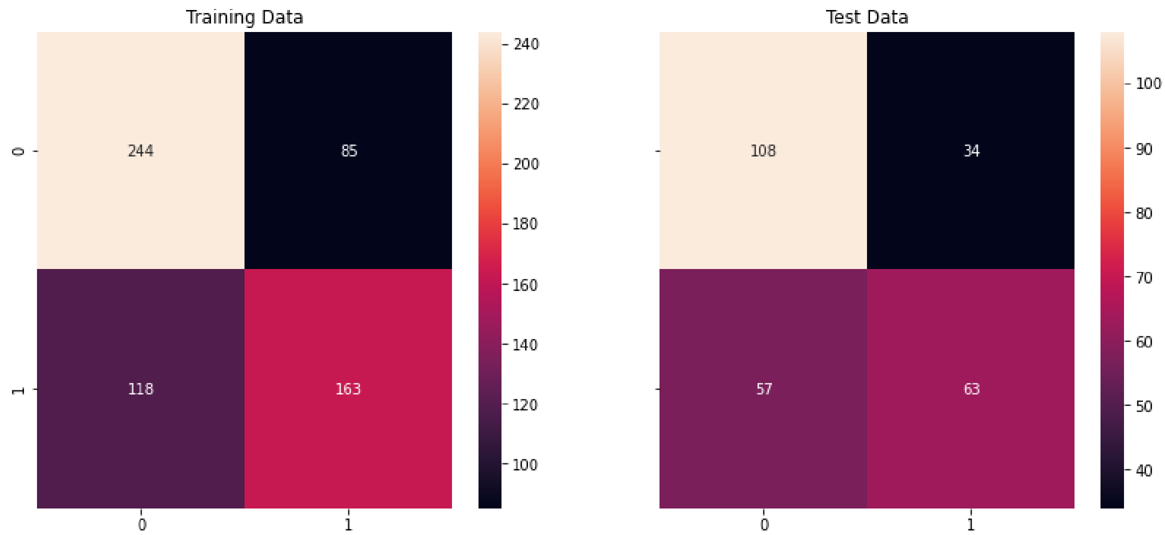
The metrics accuracy is the same for both training and test set. Since the proportion of the classes (1,0) are more or less equal, accuracy score can be reliable to check the performance of the model. There is no evidence of over fitting or under fitting in the model. There is a decrease in recall and F1 score in testing set than training set.

Linear Discriminate Analysis performance metrics:

Accuracy scores:

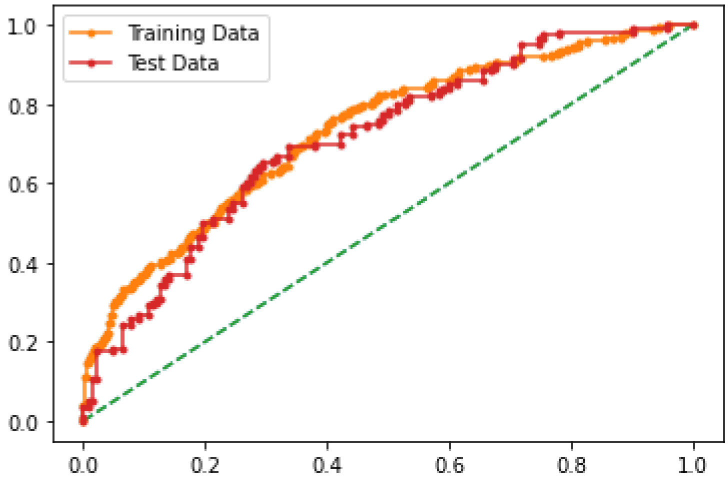
```
Accuracy of the training set:  0.6672131147540984
Accuracy of the training set:  0.6526717557251909
```

Confusion matrix:



AUC-ROC curve for both training and testing set:

```
AUC for the Training Data: 0.733
AUC for the Test Data: 0.715
```



Classification report:

Classification Report of the training data:

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

Classification Report of the test data:

	precision	recall	f1-score	support
0	0.65	0.76	0.70	142
1	0.65	0.53	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

Inference:

Metrics	**Training set**	**Testing set**
Accuracy	0.66	0.65
Precision	0.66	0.65
Recall	0.58	0.53
F1 score	0.62	0.58

LDA performs decreases in training set than testing set. Accuracy score has reduced. There is a significant drop in both recall and F1 score.

Best model for the given case study:

From the above results, we can say that **Logistic Regression** seem to be the optimized model for the given tour and travel agency. Since in this case study, classes are not well separated hence LDA lacks the accuracy in discriminating between the class.

Moreover, the training and test set results are in line for logistic regression rather than LDA. Because the small amount of data and just two classes in the predictor variable, choosing logistic regression as the final optimised model.

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

The goal of the business problem is to predict whether an employee of a company will opt for a holiday package or not. Knowing whether the holiday package will be opted/not will help the tour agency in selling relevant holiday packages to employees who are opting and for those who are opting, agency can come up with a better offer to encourage employees to opt in. This would help them to change their plans accordingly and sell the packages wisely.

Using the given dataset provided by the company, models like LDA and logistic regression were built and evaluated the performance metrics.

Here the target variable is 'Holliday_Package' (Yes – Opted in/No – Not opted). Factors that are of significant importance in classifying the target variable are salary, age, education, number of young children and foreign.

Notable inferences from the analysis:

- Employee whose salary is high or if they have more years of experience, they are not opting for holiday package.
- Employees with no young children are highly likely to opt in.
- Age group between 30-50 have more chances of opting for the family package.
- Employees from foreign are inclined to opt in for the package.
- Employees with less years of education have opted in for the package.

Insights from the model:

- 54% of the data points are in 'No' category of holiday package and 46% of the data points are in 'Yes' category. Since the data points are balanced between the categories, model accuracy score is a reliable performance mease.
- Out of the two models, Logistic regression has performed consistently with both training and testing data.

Our business problem is to identify the holiday package, we are focused on determining the false positives and false negatives.

- False positive (FP) - Datapoints that are actually false but predicted as true. This is also known as type 1 error. In order to reduce the type 1 error, we have to increase the precision of the model (among the points identified as positives by the model how many are actually positive).

    Type 1 error in this case study means model has classified the data point as 1 instead of 0. The tour agency will most likely miss out on these employees who are not going to opt in for the package. This type of error is of priority in this business context, since tagging the actual negative as positive affects the agencies scope of expanding business by selling the package.

- False negative (FN) – Datapoints that are actually true but predicted as false. This is known as type 2 error. In order to reduce to type 2 error. We have to increase recall (how many actual true data points are identified as true by the model)

    Type 2 error is might not be of priority for our case study, since predicting the actual true data points are false will not cause any damages to the tour agency because the employees will anyway opt for the package irrespective of any incentives from the tour agency.

Recommendations for the business:

If the holiday package comes out as Yes:

- Targeting those employees who are likely to opt in for the package by giving them extra benefits like discounted stays, exclusive offers, free tour planners and guides would help the agency to attract the employees and retain the business.
- For foreign employees, language translator can be allotted as an add-on travel package service.
- Suggesting a travel itinerary as a complimentary gift for the employees who opt-in will help the agency to grab those employees who are not sure how to plan.

If the holiday package comes out as No:

- Offering insurance coverage or highlighting the benefits of the agency's services will help the agency to engage the employees for opting in.
- Suggesting additional products on top of the standard package, like offering free breakfast for the entire stay, complimentary services related to the destination will attract the employees.
- Making the travel booking hassle free.
- Engaging employees who are below the age of 30 and above 50 by suggesting places suitable for these age groups.
- Arranging care taker for employees having young/older children during their holiday would definitely attract employees to opt in for the holiday package.

Here we have built model with 5 independent variables for predicting the 'Holliday_Package' dependent variable. If we had some more factors like consumer behavioural patterns, social conditions and individual needs those of which can affect the holiday package status predominantly could make the model better in predicting 'Yes' stances.