greatlearning
Learning for Life

Business Report

DSBA

Project - Predictive Modelling

Name: Sandeep Immadi

Date: 29/08/2021

CONTENTS:

Problem 1:

Problem 2:

List of Figures

Problem 1:

You are hired by a company Gem Stones co ltd, which is a cubic zirconia manufacturer. You are provided with the dataset containing the prices and other attributes of almost 27,000 cubic zirconia (which is an inexpensive diamond alternative with many of the same qualities as a diamond). The company is earning different profits on different prize slots. You have to help the company in predicting the price for the stone on the bases of the details given in the dataset so it can distinguish between higher profitable stones and lower profitable stones so as to have better profit share. Also, provide them with the best 5 attributes that are most important.

Variables:

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	cubic zirconia Clarity refers to the absence of the Inclusions and Blemishes. (In order from Best to Worst, IF = flawless, l1= level 1 inclusion) IF, VVS1, VVS2, VS1, VS2, Sl1, Sl2, l1
Depth	The Height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter.
Table	The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter.
Price	the Price of the cubic zirconia.
X	Length of the cubic zirconia in mm.
Y	Width of the cubic zirconia in mm.
Z	Height of the cubic zirconia in mm.

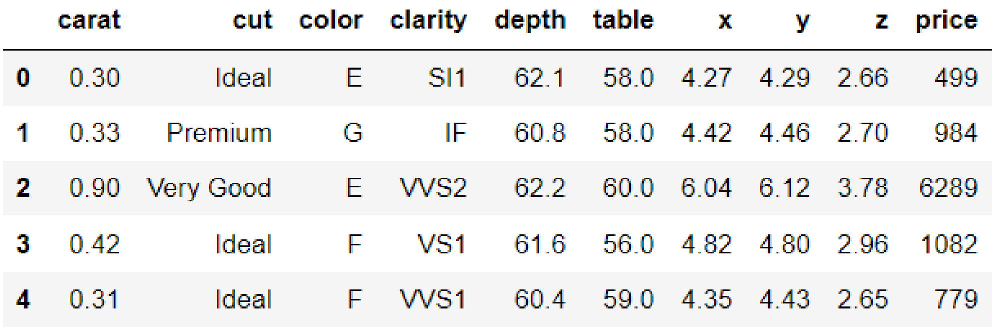
1.1 Read the data and do exploratory data analysis. Describe the data briefly. (Check the null values, Data types, shape, EDA). Perform Univariate and Bivariate Analysis.

<u>Exploratory Data Analysis:</u>

We will explore the Clustering Data set and perform the exploratory data analysis on the dataset. The major topics to be covered are below:

- Removing duplicates
- Missing value treatment
- Outlier Treatment
- Normalization and Scaling (Numerical Variables)
- Encoding Categorical variables (Dummy Variables)
- Univariate Analysis
- Bivariate Analysis

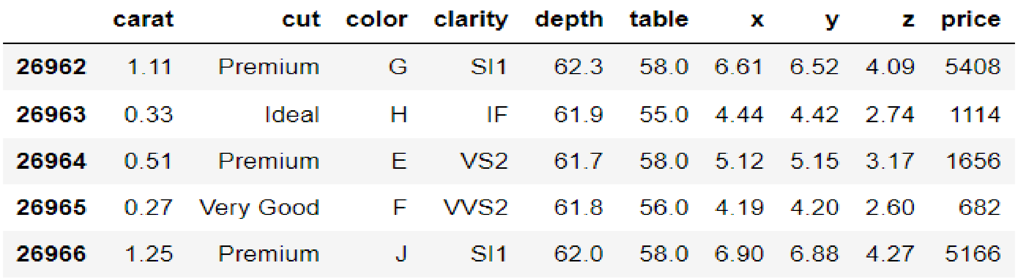
<u>Data Insights of top 5 records:</u>



	carat	cut	color	clarity	depth	table	x	y	z	price
0	0.30	Ideal	E	SI1	62.1	58.0	4.27	4.29	2.66	499
1	0.33	Premium	G	IF	60.8	58.0	4.42	4.46	2.70	984
2	0.90	Very Good	E	VVS2	62.2	60.0	6.04	6.12	3.78	6289
3	0.42	Ideal	F	VS1	61.6	56.0	4.82	4.80	2.96	1082
4	0.31	Ideal	F	VVS1	60.4	59.0	4.35	4.43	2.65	779

Fig.1

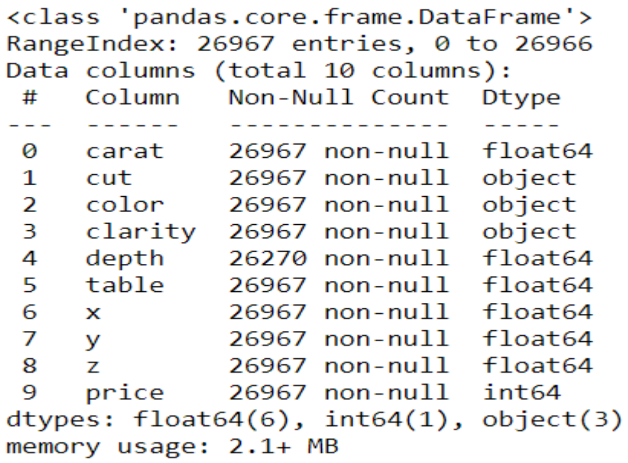
<u>Data Insights of bottom 5 records:</u>



	carat	cut	color	clarity	depth	table	x	y	z	price
26962	1.11	Premium	G	SI1	62.3	58.0	6.61	6.52	4.09	5408
26963	0.33	Ideal	H	IF	61.9	55.0	4.44	4.42	2.74	1114
26964	0.51	Premium	E	VS2	61.7	58.0	5.12	5.15	3.17	1656
26965	0.27	Very Good	F	VVS2	61.8	56.0	4.19	4.20	2.60	682
26966	1.25	Premium	J	SI1	62.0	58.0	6.90	6.88	4.27	5166

Fig.2

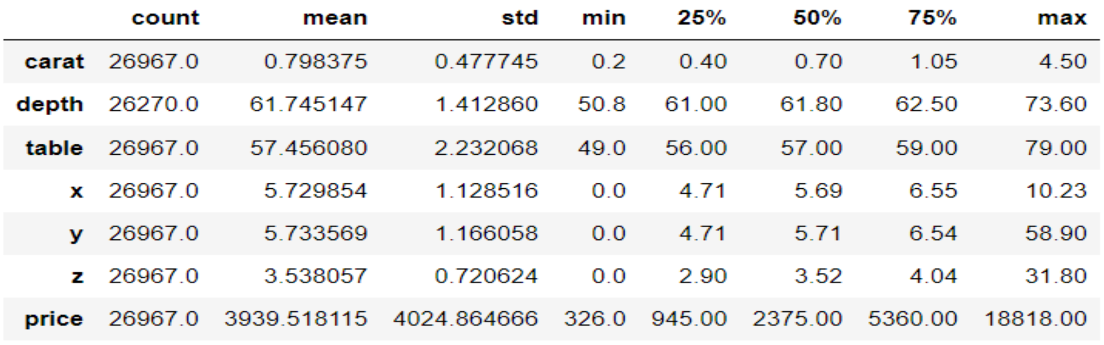
Data Info:



```
<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
```

Fig.3

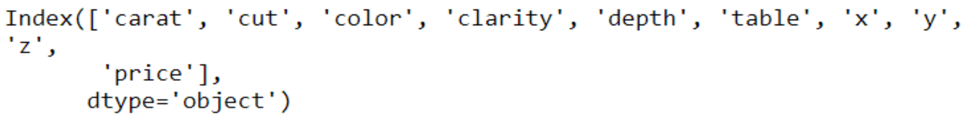
Data Summary:



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

Fig.4

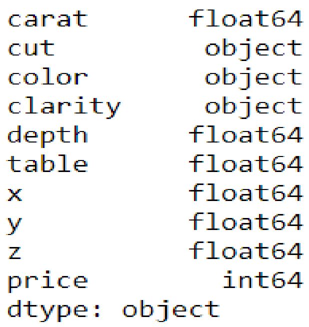
Total columns:



```
Index(['carat', 'cut', 'color', 'clarity', 'depth', 'table', 'x', 'y',
'z',
       'price'],
      dtype='object')
```

Fig.5

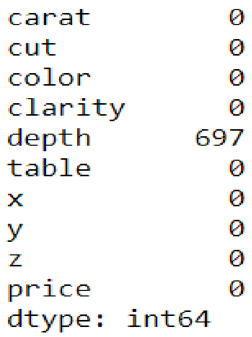
Data Types:



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

Fig.6

Missing Values:



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

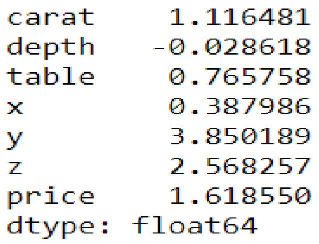
Fig.7

Data Dimensions: (26967,10)

Missing values: 697(depth)

Duplicates in the data: 34 duplicates were dropped

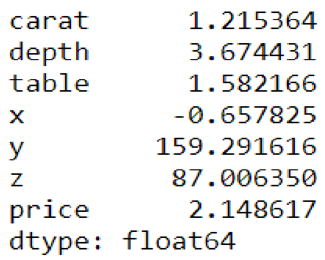
Skewness:



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

Fig.8

Kurtosis :



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

Fig.9

Outliers :

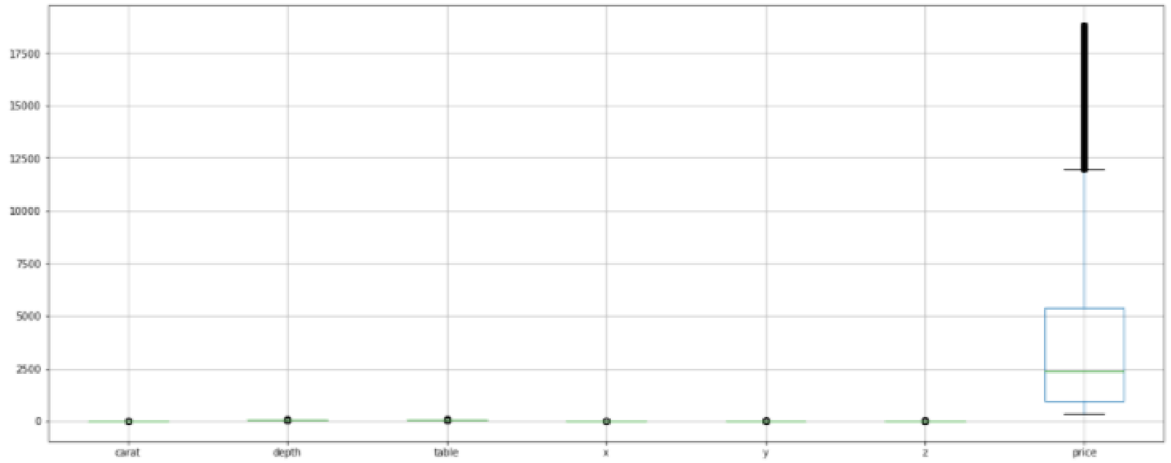


Fig.10

Histograms:

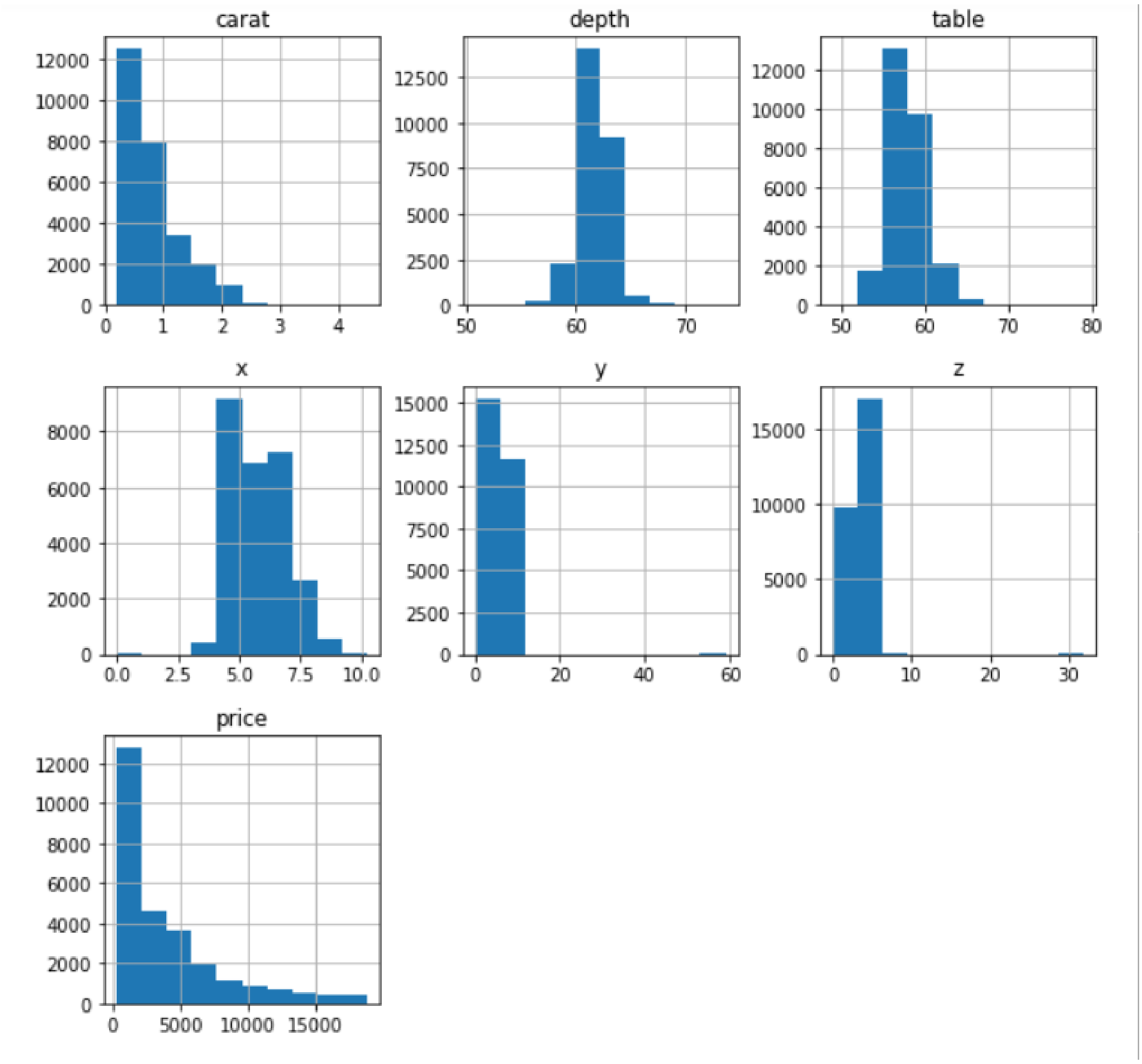


Fig.11

Inferences:

- The data consists of 26967 rows and 10 columns
- 697 missing values found in depth
- The present variables are both numeric and categorical types in nature i.e. float, int, and object data types are present
- 34 duplicates were dropped in the data.
- Mean and median for all variables are more or less equal
- The distribution of data in carat has positive skewness. The data range is 0 to 1, a major portion of data lies in this range.
- The distribution of depth is in the normal distribution and it ranges from 55 to 65.

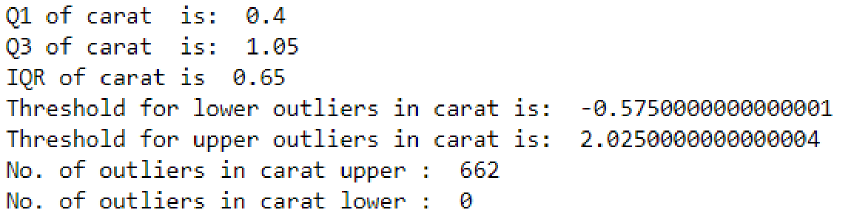
- The distribution of data in the table has positive skewness. The maximum distribution range is between 55 to 65.
- The distribution of x has positive skewness. The distribution range is 4 to 8.
- X represents the length of the cubic zirconia in mm.
- The distribution of Y is excessively positively skewed. Y represents the width of the cubic zirconia in mm.)
- The distribution of z is asymmetrically positively skewed. Z represents the height of the cubic zirconia in mm
- Price has positive skewness. The distribution range is between RS 100 to 8000
- Outliers' present in the data.

Univeriate Analysis:

carat:

- Std in carat (0.477745)
- The maximum and minimum carat is 4.5 and 0.2 respectively
- The mean and median values are 0.79 and 0.7 respectively
- No null values present in the data

Outliers proportion in carat:



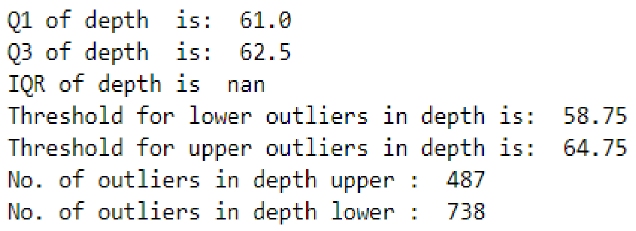
```
Q1 of carat  is:  0.4
Q3 of carat  is:  1.05
IQR of carat is  0.65
Threshold for lower outliers in carat is:  -0.5750000000000001
Threshold for upper outliers in carat is:  2.0250000000000004
No. of outliers in carat upper :  662
No. of outliers in carat lower :  0
```

Fig.12

depth:

- Std in depth (1.41)
- The maximum and minimum depth is 73 and 50 respectively
- The mean and median values are 61 and 61 respectively

Outliers proportion in depth:



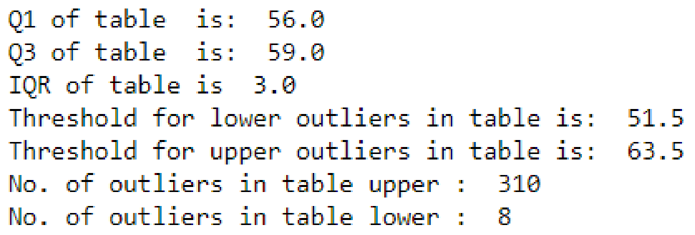
```
Q1 of depth  is:  61.0
Q3 of depth  is:  62.5
IQR of depth is  nan
Threshold for lower outliers in depth is:  58.75
Threshold for upper outliers in depth is:  64.75
No. of outliers in depth upper :  487
No. of outliers in depth lower :  738
```

Fig.13

table:

- Std in table (2.23)
- The maximum and minimum of table is 79 and 49 respectively
- The mean and median values are 57 and 57 respectively
- No null values present in the data

Outliers proportion in table:



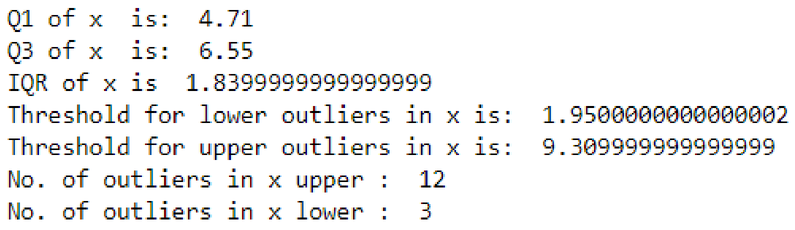
```
Q1 of table  is:  56.0
Q3 of table  is:  59.0
IQR of table is  3.0
Threshold for lower outliers in table is:  51.5
Threshold for upper outliers in table is:  63.5
No. of outliers in table upper :  310
No. of outliers in table lower :  8
```

Fig.14

x:

- Std in x (1.1)
- The maximum and minimum x is 10 and 0 respectively
- The mean and median values are 5 and 5 respectively
- No null values present in the data

Outliers proportion in x:



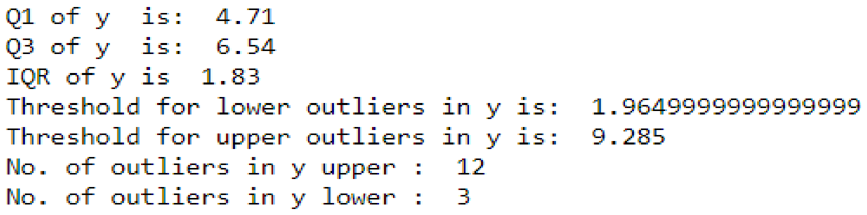
```
Q1 of x  is:  4.71
Q3 of x  is:  6.55
IQR of x is  1.8399999999999999
Threshold for lower outliers in x is:  1.9500000000000002
Threshold for upper outliers in x is:  9.309999999999999
No. of outliers in x upper :  12
No. of outliers in x lower :  3
```

Fig.15

y:

- Std in y (1.1)
- The maximum and minimum y is 58 and 0 respectively
- The mean and median values are 5 and 5 respectively
- No null values present in the data

Outliers proportion in y:



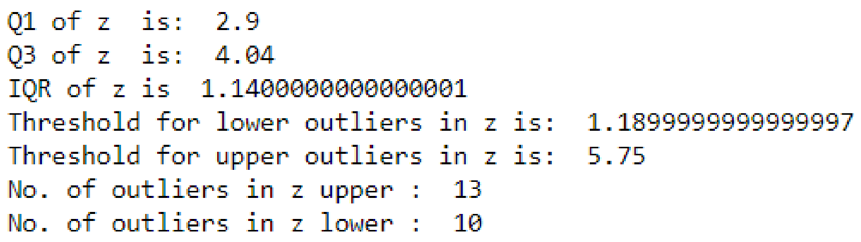
```
Q1 of y  is:  4.71
Q3 of y  is:  6.54
IQR of y is  1.83
Threshold for lower outliers in y is:  1.9649999999999999
Threshold for upper outliers in y is:  9.285
No. of outliers in y upper :  12
No. of outliers in y lower :  3
```

Fig.16

z:

- Std in z (0.7)
- The maximum and minimum z is 31 and 0 respectively
- The mean and median values are 3.5 and 3.5 respectively
- No null values present in the data

Outliers proportion in z:



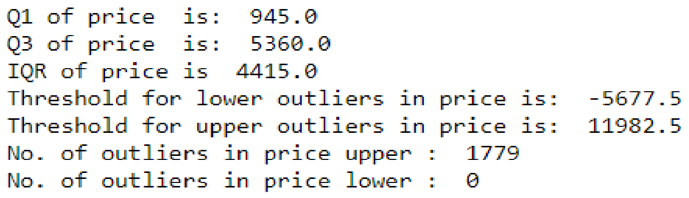
```
Q1 of z  is:  2.9
Q3 of z  is:  4.04
IQR of z is  1.1400000000000001
Threshold for lower outliers in z is:  1.1899999999999997
Threshold for upper outliers in z is:  5.75
No. of outliers in z upper :  13
No. of outliers in z lower :  10
```

Fig.17

price:

- Std in price (4024)
- The maximum and minimum price is 18818 and 326 respectively
- The mean and median values are 3939 and 2375 respectively
- No null values present in the data

Outliers proportion in price:



```
Q1 of price  is:  945.0
Q3 of price  is:  5360.0
IQR of price is  4415.0
Threshold for lower outliers in price is:  -5677.5
Threshold for upper outliers in price is:  11982.5
No. of outliers in price upper :  1779
No. of outliers in price lower :  0
```

Fig.18

<u>**Graphical comparison of categorical variables:**</u>

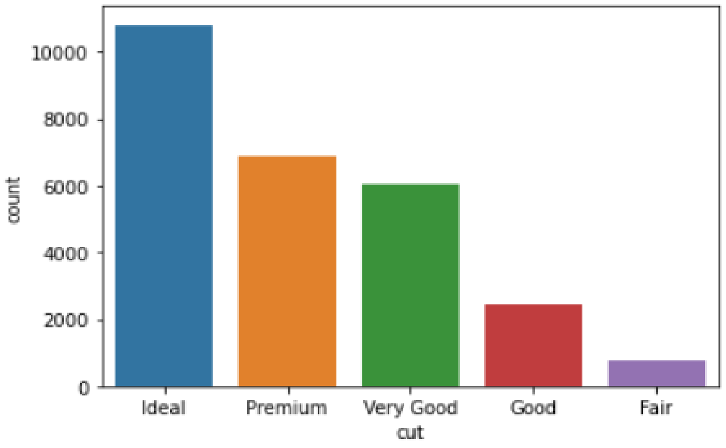


Fig.19

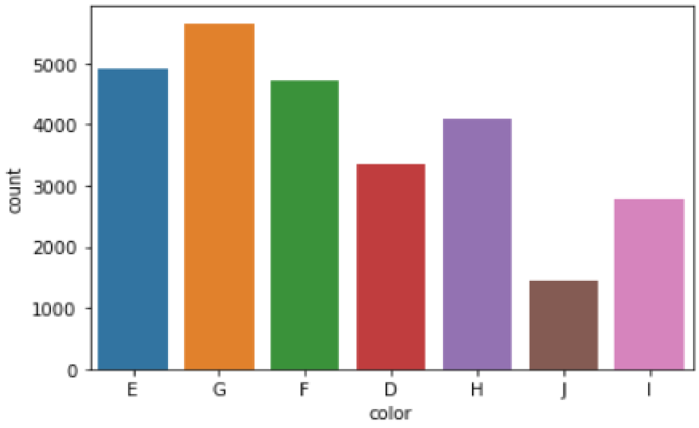


Fig.20

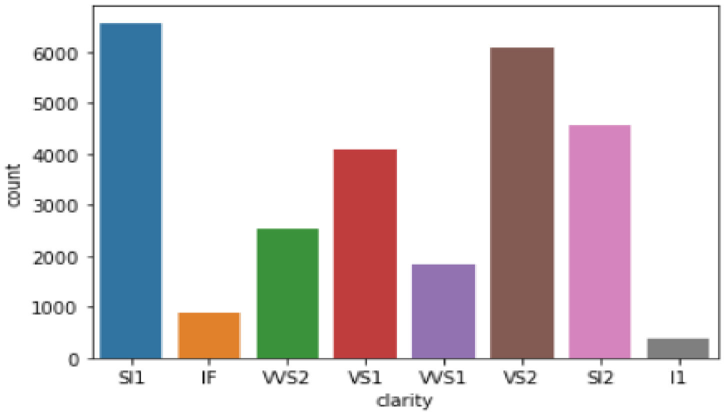


Fig.21

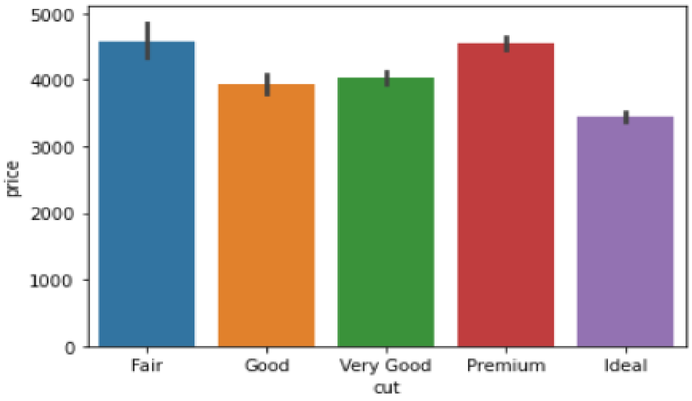


Fig.22

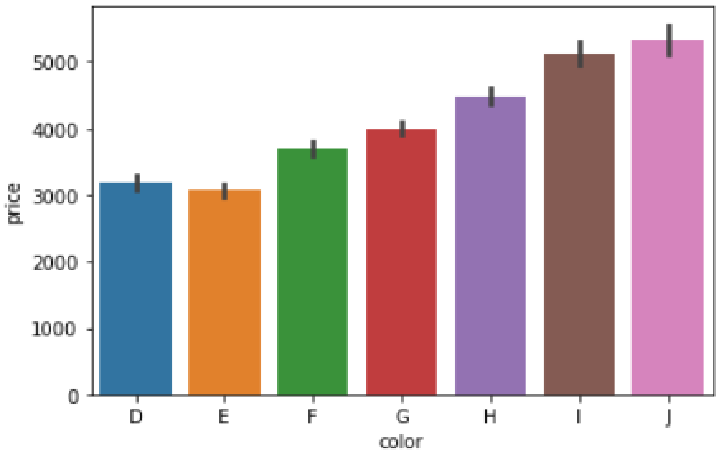


Fig.23

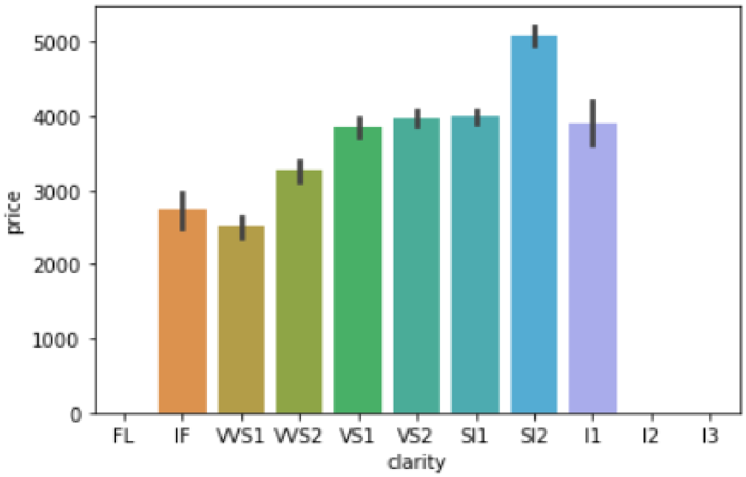
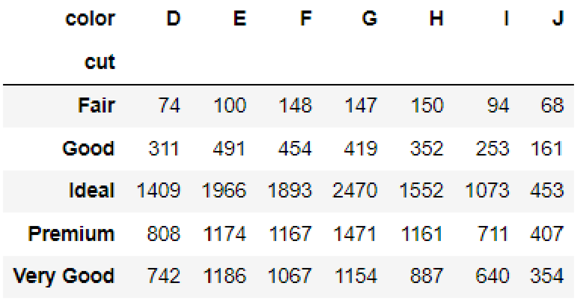


Fig.24



color	D	E	F	G	H	I	J
cut							
Fair	74	100	148	147	150	94	68
Good	311	491	454	419	352	253	161
Ideal	1409	1966	1893	2470	1552	1073	453
Premium	808	1174	1167	1471	1161	711	407
Very Good	742	1186	1067	1154	887	640	354

Fig.25

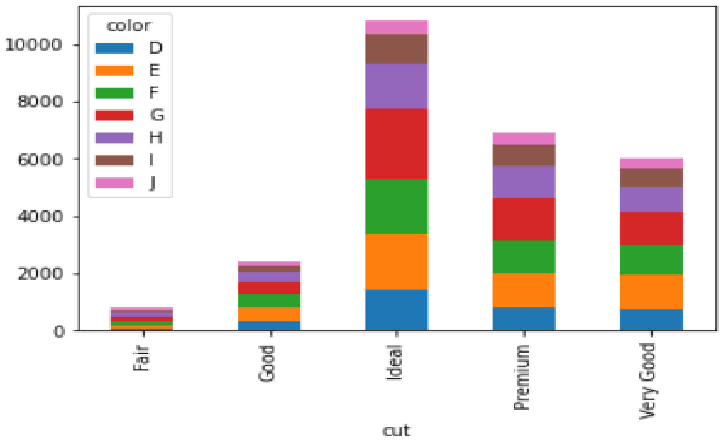
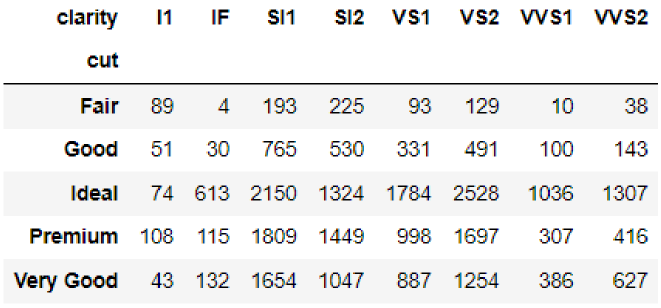


Fig.26



clarity	I1	IF	SI1	SI2	VS1	VS2	VVS1	VVS2
cut								
Fair	89	4	193	225	93	129	10	38
Good	51	30	765	530	331	491	100	143
Ideal	74	613	2150	1324	1784	2528	1036	1307
Premium	108	115	1809	1449	998	1697	307	416
Very Good	43	132	1654	1047	887	1254	386	627

Fig.27

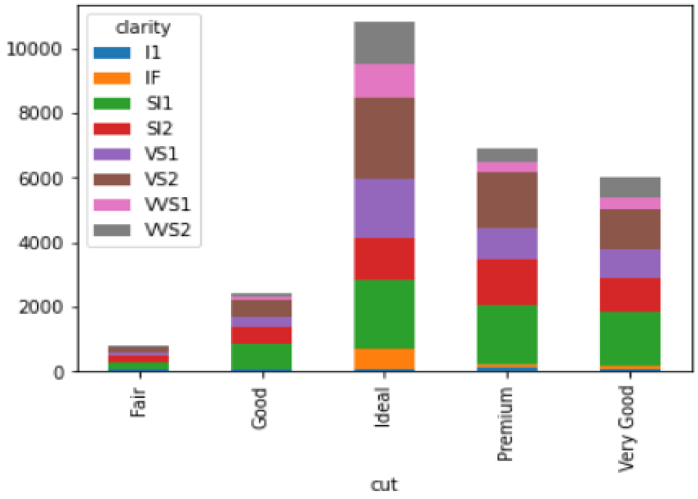


Fig.28

- Cut quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal.
- Colour of the cubic zirconia. With D being the worst and J the best.
- Cubic zirconia Clarity refers to the absence of the Inclusions and Blemishes. (In order from Best to Worst, IF = flawless, l1= level 1 inclusion) IF, VVS1, VVS2, VS1, VS2, Sl1, Sl2, l1
- Descending order of counts of each category looks:

- cut:
  - Ideal 10816
  - Premium 6899
  - Very Good 6030
  - Good 2441
  - Fair 781
  - Name: cut, dtype: int64

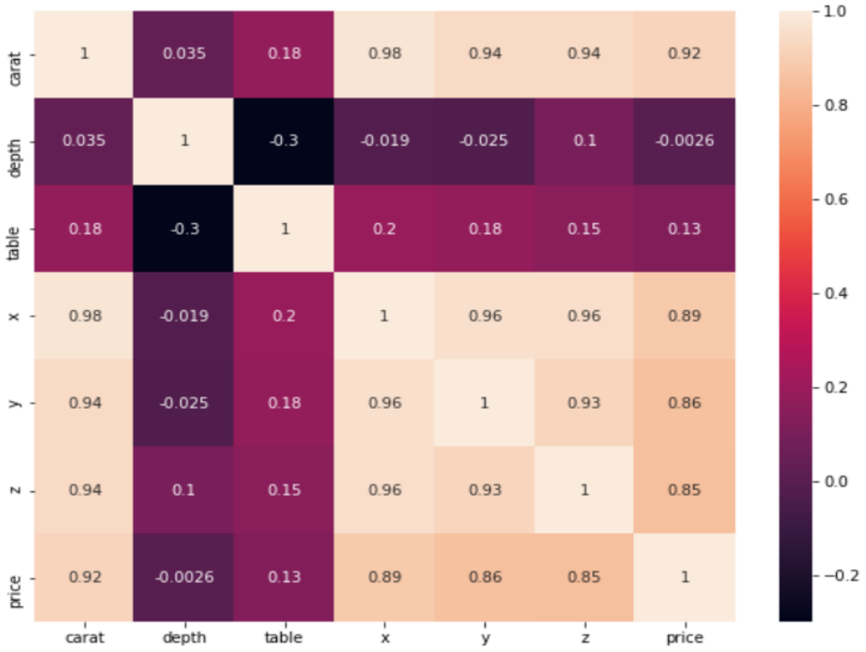
- color:
  - G 5661
  - E 4917
  - F 4729
  - H 4102
  - D 3344
  - I 2771
  - J 1443
  - Name: color, dtype: int64

- clarity:
  - SI1 6571
  - VS2 6099
  - SI2 4575
  - VS1 4093
  - VVS2 2531
  - VVS1 1839
  - IF 894
  - I1 365
  - Name: clarity, dtype: int64

- When we compare each category with price, we see that:
- cut vs price : Price is high for Fair,premium followed by very good cut,good,ideal respectively.
- color vs price : Price is high for J,I,H followed by G,F,D,E respectively.
- clarity vs price : Price is high for S12 followed by S11,VS2,VS1 respectively.
- cut vs color : ideal reaches maximum color distribution followed by premium and very good.
- cut vs clarity: ideal reaches maximum cut distribution followed by premium and very good.

Multivariate Analysis:

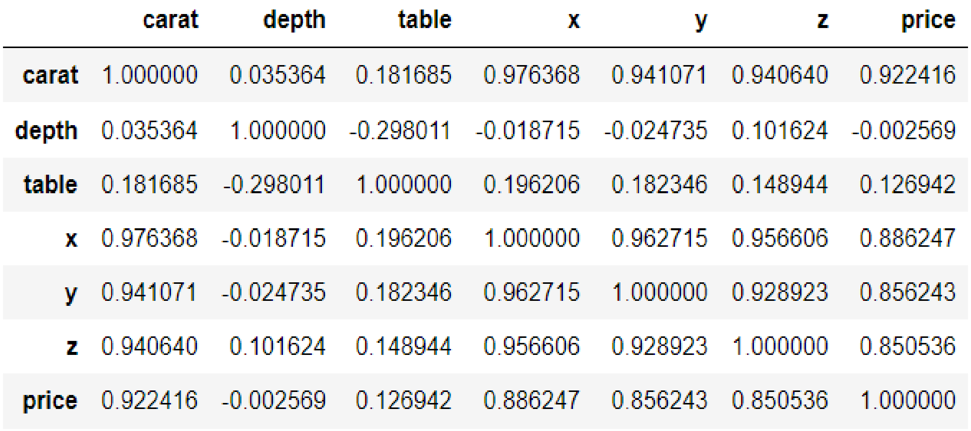
Correlation:



	carat	depth	table	x	y	z	price
carat	1	0.035	0.18	0.98	0.94	0.94	0.92
depth	0.035	1	-0.3	-0.019	-0.025	0.1	-0.0026
table	0.18	-0.3	1	0.2	0.18	0.15	0.13
x	0.98	-0.019	0.2	1	0.96	0.96	0.89
y	0.94	-0.025	0.18	0.96	1	0.93	0.86
z	0.94	0.1	0.15	0.96	0.93	1	0.85
price	0.92	-0.0026	0.13	0.89	0.86	0.85	1

Fig.29

Correlation Data frame:



	carat	depth	table	x	y	z	price
carat	1.000000	0.035364	0.181685	0.976368	0.941071	0.940640	0.922416
depth	0.035364	1.000000	-0.298011	-0.018715	-0.024735	0.101624	-0.002569
table	0.181685	-0.298011	1.000000	0.196206	0.182346	0.148944	0.126942
x	0.976368	-0.018715	0.196206	1.000000	0.962715	0.956606	0.886247
y	0.941071	-0.024735	0.182346	0.962715	1.000000	0.928923	0.856243
z	0.940640	0.101624	0.148944	0.956606	0.928923	1.000000	0.850536
price	0.922416	-0.002569	0.126942	0.886247	0.856243	0.850536	1.000000

Fig.30

Pair plot:

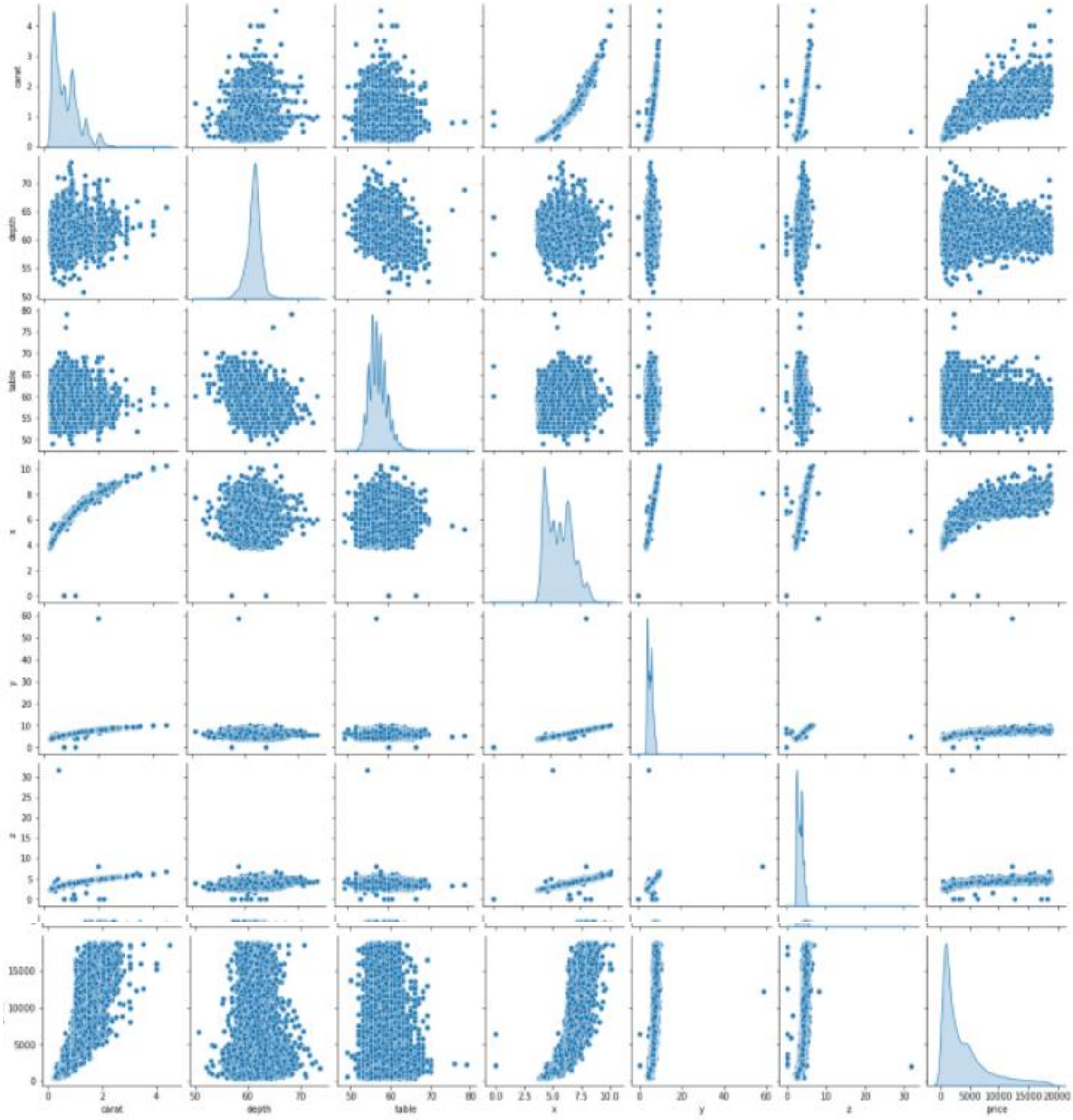


Fig.31

Inferences:

- Carat,length,width & height have strong correlation w.r.t the dependent variable price.
- The dataset contains high multicollinearity with Carat & length being the highest collinear with a value of 0.98.
- Cut, color and clarity have very low correlations with other variables while x, y, z and carat have high correlations.
- x & z : 0.99
- y & z : 0.99
- carat & x,y,z : 0.98
- price & carat : 0.94
- price & x,y,z: 0.91
- table & depth : -0.29
- x,y,price vs depth : negative correlation

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?

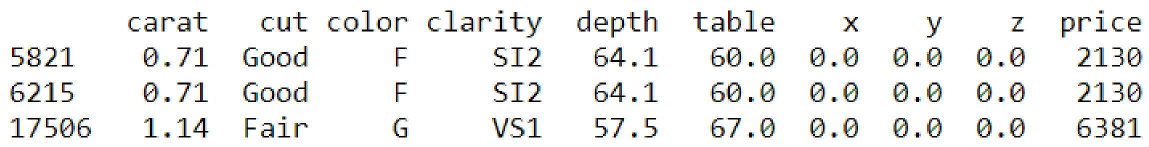
Variables with Null values:

- Depth = 697 values, Imputation performed with mean, because the data is normally distributed.

Check for values equal to zero:

- After checking all the variables, x,y and z variables have values equal to 0.

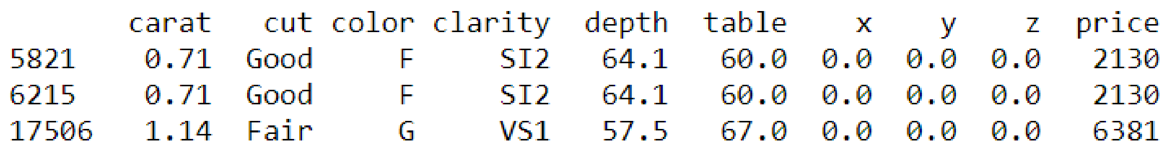
Variable: x



```
        carat   cut color clarity  depth  table    x    y    z  price
5821     0.71  Good     F     SI2   64.1   60.0  0.0  0.0  0.0   2130
6215     0.71  Good     F     SI2   64.1   60.0  0.0  0.0  0.0   2130
17506    1.14  Fair     G     VS1   57.5   67.0  0.0  0.0  0.0   6381
```

Fig.32

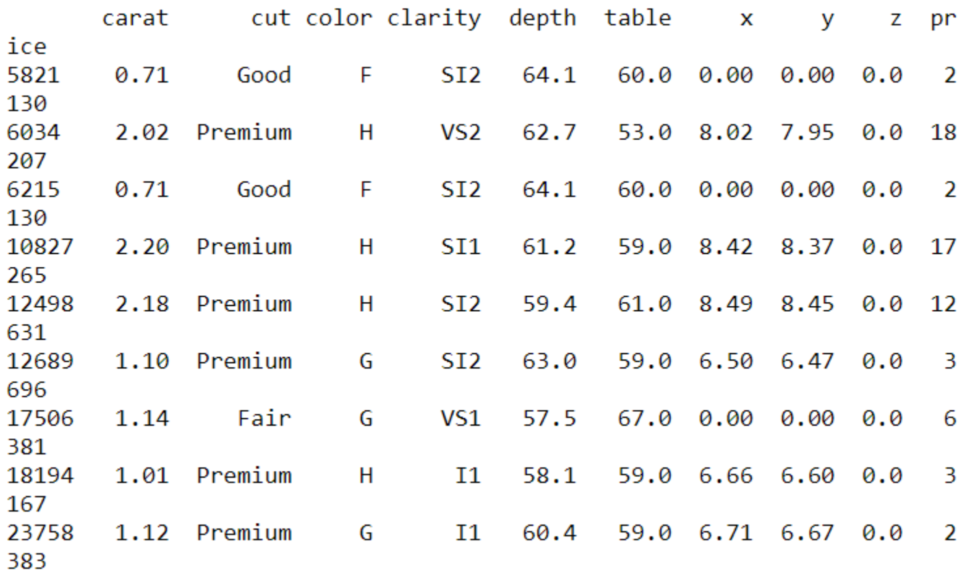
Variable: Y



```
        carat   cut color clarity  depth  table    x    y    z  price
5821     0.71  Good     F     SI2   64.1   60.0  0.0  0.0  0.0   2130
6215     0.71  Good     F     SI2   64.1   60.0  0.0  0.0  0.0   2130
17506    1.14  Fair     G     VS1   57.5   67.0  0.0  0.0  0.0   6381
```

Fig.33

Variable: z



```
       carat      cut color clarity  depth  table     x     y    z  pr
ice
5821    0.71     Good     F     SI2   64.1   60.0  0.00  0.00  0.0   2
130
6034    2.02  Premium     H     VS2   62.7   53.0  8.02  7.95  0.0  18
207
6215    0.71     Good     F     SI2   64.1   60.0  0.00  0.00  0.0   2
130
10827   2.20  Premium     H     SI1   61.2   59.0  8.42  8.37  0.0  17
265
12498   2.18  Premium     H     SI2   59.4   61.0  8.49  8.45  0.0  12
631
12689   1.10  Premium     G     SI2   63.0   59.0  6.50  6.47  0.0   3
696
17506   1.14     Fair     G     VS1   57.5   67.0  0.00  0.00  0.0   6
381
18194   1.01  Premium     H      I1   58.1   59.0  6.66  6.60  0.0   3
167
23758   1.12  Premium     G      I1   60.4   59.0  6.71  6.67  0.0   2
383
```

Fig.34

- Observations containing zeroes as the values of all variables, and which are not
- meaningful, are shown to be likely in certain regional data sets which may be subjected to multiple regression analysis.
- The biasing effects of such observations on regression statistics are shown and illustrated with a small data set.
- It is recommended that such zero
- observation be identified and removed from regional data sets prior to analysis.

Inference:

- Although in our case there are not many entries with 0.00s.
- x and y have 3 row entries and z have 8 entries.
- The entries can also be removed or can be imputed with the median values.
- Here the 0 values are considered as missing values and imputed with the median as their respective data are skewed.

Boxplot after treating outliers:

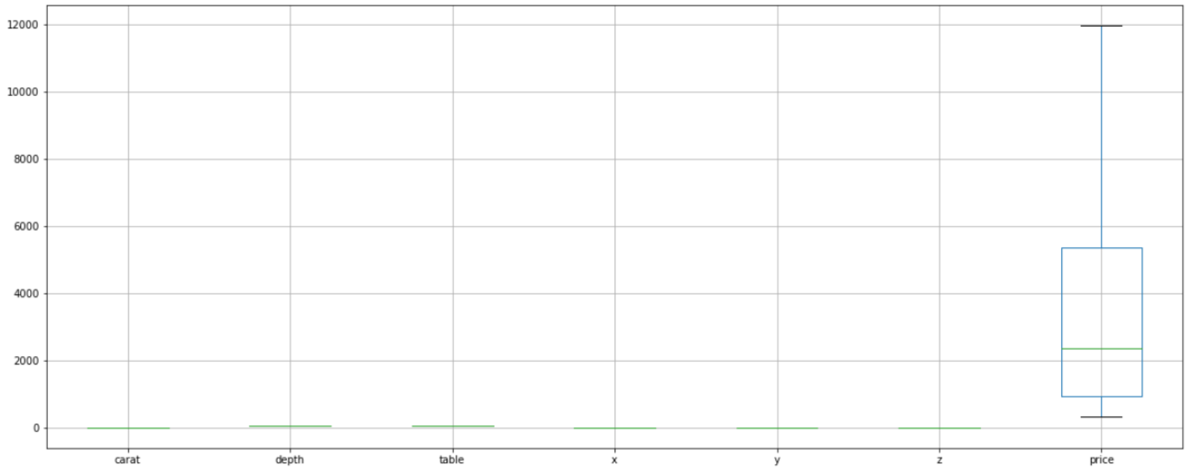


Fig.35

Is Scaling necessary:

- Scaling also called as centring is very important. It helps gradient descent to converge fast and reach the global minima.
- The box plot suggests the following:
- The values are already in scale.
- This is because carat, depth, table are functions of the length(x), width(y)and z(height)variables.
- They are in function with other variables such as density, culet height, girdle diameter etc.
- The Density is the major determinant of Carat weight of a stone.
- The price is the dependent variable. Hence, will heavily rely on the Carat.

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.

Data Encoding:

- we code using one hot encoding.
- The categorical variables are: Cut,color,clarity
- The cut and size determine the quality of a cubic zirconia.
- The four Cs used to measure the value of diamonds are also applied to this gemstone: clarity, cut, color and carat.
- All of these play a role in determining the price and value of a cubic zirconia.

Inference for Modelling:

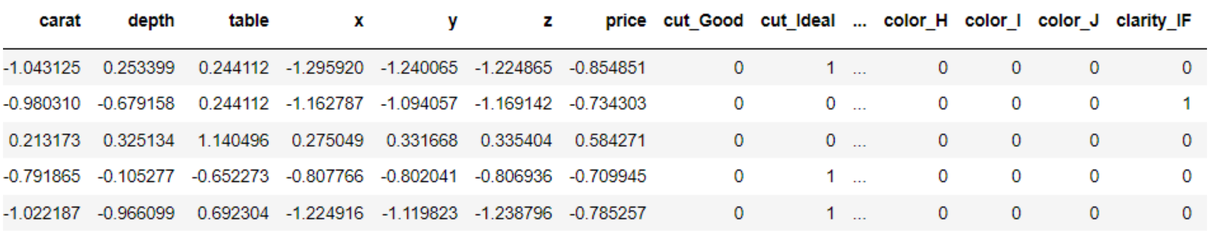
- We need to assess the CZs exact profitability by identifying them with respect to the four C’s.
- Therefore, every typological aspect of Cut, Color and Clarity need to be singled out for every piece.
- dummies from panda’s library are used.
- The Categorical data are ordinal in nature and exclusive for each and every piece of CZ.

Encoding:

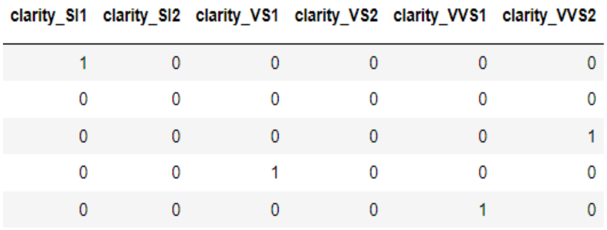
Creating a dummy encoding variable

- The number of dummy-coded variables needed is one less than the number of possible values, which is K-1.
- In statistics, this is called a dummy encoding variable, or dummy variable.
- The categorical variables are ordinal and hence need to be converted to binary.

After getting dummy variables:



carat	depth	table	x	y	z	price	cut_Good	cut_Ideal	...	color_H	color_I	color_J	clarity_IF
-1.043125	0.253399	0.244112	-1.295920	-1.240065	-1.224865	-0.854851	0	1	...	0	0	0	0
-0.980310	-0.679158	0.244112	-1.162787	-1.094057	-1.169142	-0.734303	0	0	...	0	0	0	1
0.213173	0.325134	1.140496	0.275049	0.331668	0.335404	0.584271	0	0	...	0	0	0	0
-0.791865	-0.105277	-0.652273	-0.807766	-0.802041	-0.806936	-0.709945	0	1	...	0	0	0	0
-1.022187	-0.966099	0.692304	-1.224916	-1.119823	-1.238796	-0.785257	0	1	...	0	0	0	0



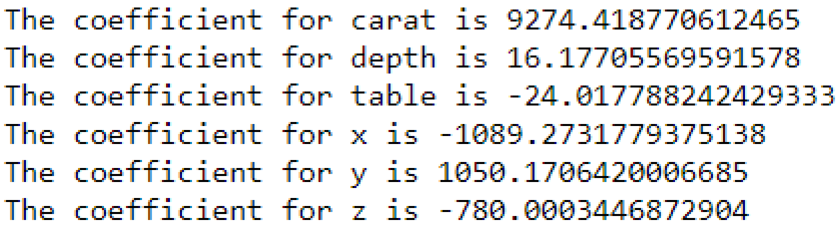
clarity_SI1	clarity_SI2	clarity_VS1	clarity_VS2	clarity_VVS1	clarity_VVS2
1	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	1
0	0	1	0	0	0
0	0	0	0	1	0

Fig.36

Data Split:

- X = independent variables
- Y = dependent variable: price
- From sklearn.model_selection is used
- Train_test_split is imported
- Test data: 30%
- Training data: 70%

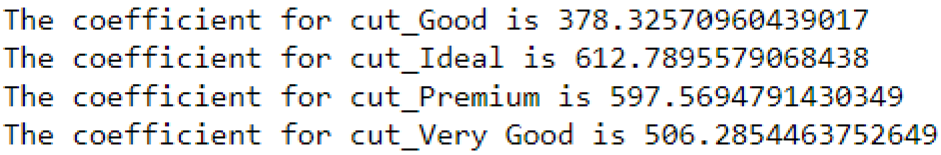
Applying linear regression, the coefficients of the determinants are as follows.



```
The coefficient for carat is 9274.418770612465
The coefficient for depth is 16.17705569591578
The coefficient for table is -24.017788242429333
The coefficient for x is -1089.2731779375138
The coefficient for y is 1050.1706420006685
The coefficient for z is -780.0003446872904
```

Fig.37

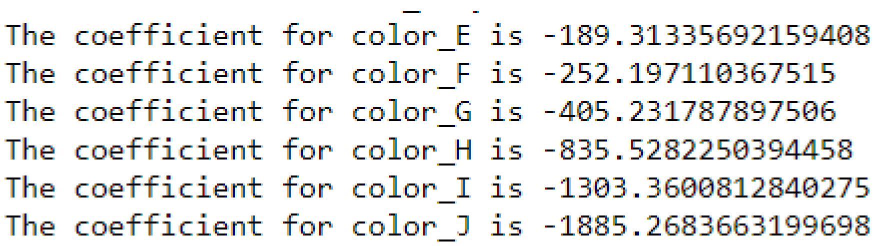
Cut:



```
The coefficient for cut_Good is 378.32570960439017
The coefficient for cut_Ideal is 612.7895579068438
The coefficient for cut_Premium is 597.5694791430349
The coefficient for cut_Very Good is 506.2854463752649
```

Fig.38

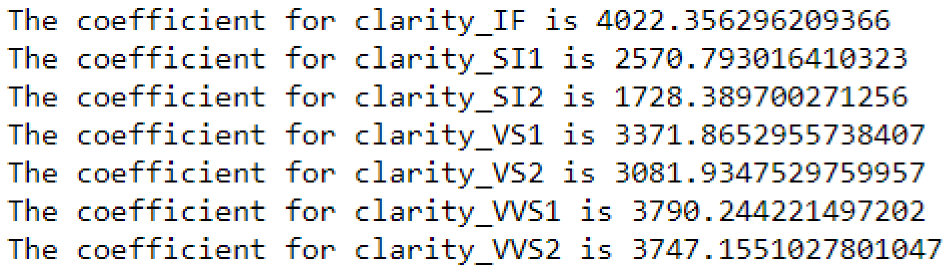
Color:



```
The coefficient for color_E is -189.31335692159408
The coefficient for color_F is -252.197110367515
The coefficient for color_G is -405.231787897506
The coefficient for color_H is -835.5282250394458
The coefficient for color_I is -1303.3600812840275
The coefficient for color_J is -1885.2683663199698
```

Fig.39

Clarity:



```
The coefficient for clarity_IF is 4022.356296209366
The coefficient for clarity_SI1 is 2570.793016410323
The coefficient for clarity_SI2 is 1728.389700271256
The coefficient for clarity_VS1 is 3371.8652955738407
The coefficient for clarity_VS2 is 3081.9347529759957
The coefficient for clarity_VVS1 is 3790.244221497202
The coefficient for clarity_VVS2 is 3747.1551027801047
```

Fig.40

- Intercept: -3147.283402498099

Performance Metrics:

R Squared:

- R-squared (coefficient of determination):
- R-squared is the proportion of the variance in the dependent variable that is expressed by the independent variables.
- The value for R-squared can range from 0 to 1.
- Values approaching 1 are a good indicator of the accuracy of the predictors.

  - R Squared Train: 0.9419557931252712
  - R Squared Test 0.9381643998102492

- At 94% R squared score the model is a good fit.

RMSE(root mean square error):

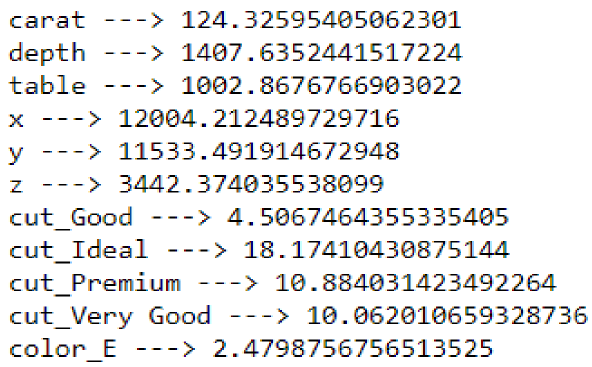
- RMSE is a measure of errors and its spread
- It explains the concentration of the data points around the best fit line.
- Max Value of dependent variable(price): 18818.000000
- Mean of dependent variable (price) : 3939.518115
- Min value of the dependent variable (price) : 326.000000

  - RMSE score for Training set: 832.4206056108892
  - RMSE score for Test set: 870.953429935481

- The RMSE score is closer to the minimum value in the Price variable, which means the Root mean square error shows the model is a good fit.

Variance inflation factor (VIF) and statsmodel:

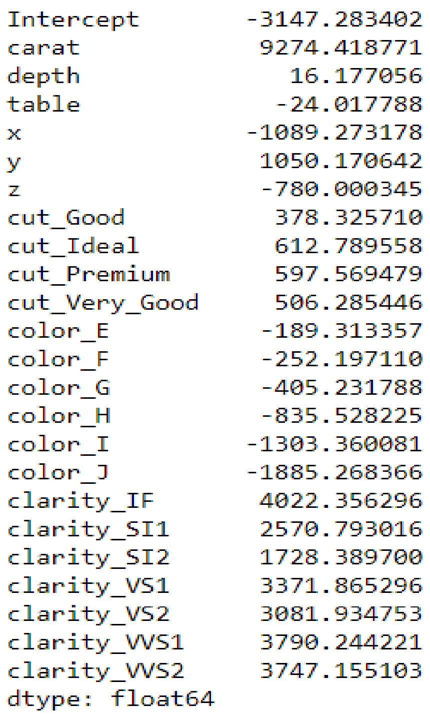
- The variance inflation factor (VIF) quantifies the extent of correlation between one predictor and the other predictors in a model.
- It is used for diagnosing collinearity/multicollinearity.
- Higher values signify that it is difficult to impossible to assess accurately the contribution of predictors to a model.



```
carat ---> 124.32595405062301
depth ---> 1407.6352441517224
table ---> 1002.8676766903022
x ---> 12004.212489729716
y ---> 11533.491914672948
z ---> 3442.374035538099
cut_Good ---> 4.5067464355335405
cut_Ideal ---> 18.17410430875144
cut_Premium ---> 10.884031423492264
cut_Very Good ---> 10.062010659328736
color_E ---> 2.4798756756513525
```

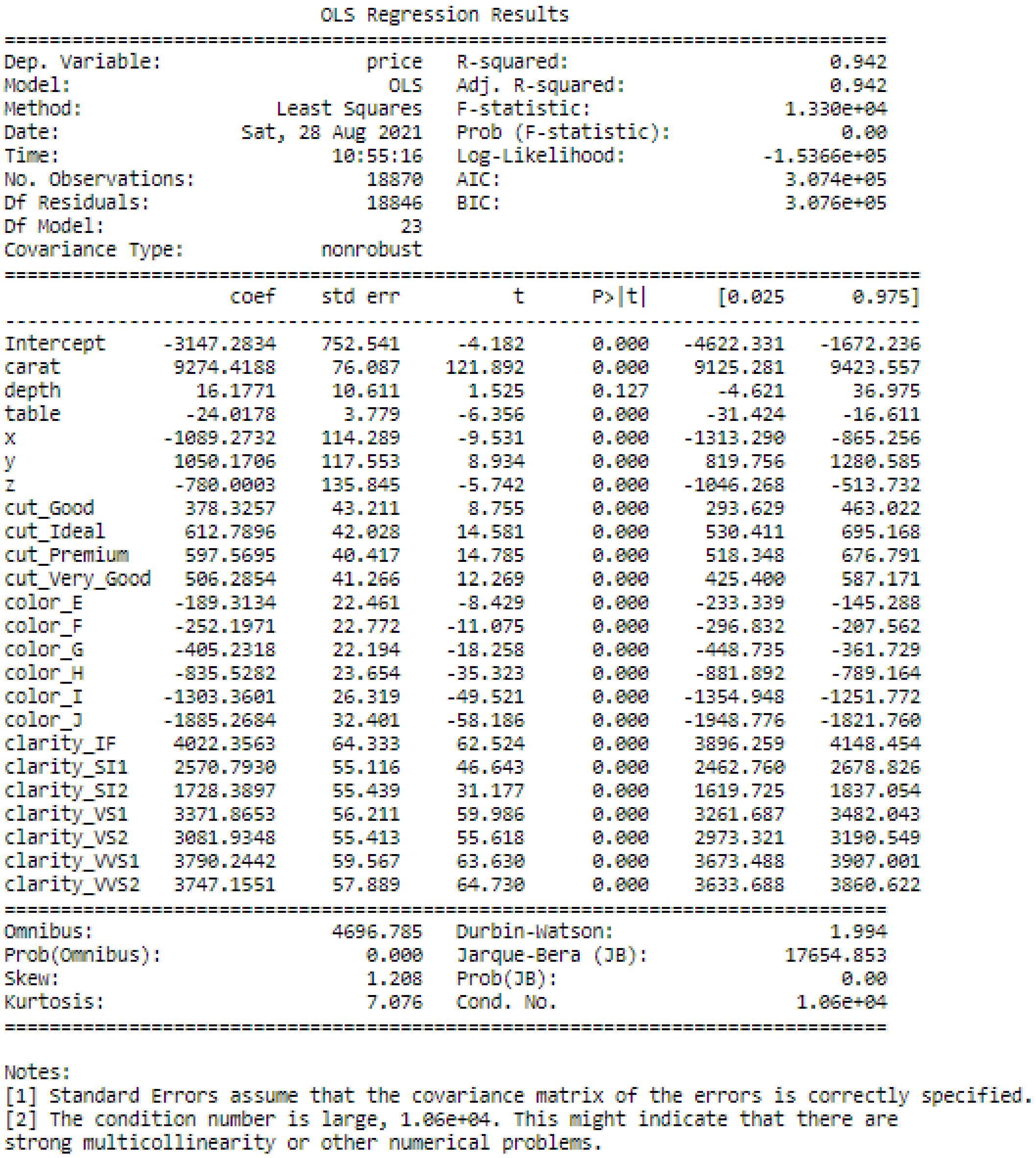
Fig.41

statsmodel:



```
Intercept       -3147.283402
carat            9274.418771
depth              16.177056
table             -24.017788
x               -1089.273178
y                1050.170642
z                -780.000345
cut_Good          378.325710
cut_Ideal         612.789558
cut_Premium       597.569479
cut_Very_Good     506.285446
color_E          -189.313357
color_F          -252.197110
color_G          -405.231788
color_H          -835.528225
color_I         -1303.360081
color_J         -1885.268366
clarity_IF       4022.356296
clarity_SI1      2570.793016
clarity_SI2      1728.389700
clarity_VS1      3371.865296
clarity_VS2      3081.934753
clarity_VVS1     3790.244221
clarity_VVS2     3747.155103
dtype: float64
```

Fig.42



```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                  price   R-squared:                       0.942
Model:                            OLS   Adj. R-squared:                  0.942
Method:                 Least Squares   F-statistic:                 1.330e+04
Date:                Sat, 28 Aug 2021   Prob (F-statistic):               0.00
Time:                        10:55:16   Log-Likelihood:            -1.5366e+05
No. Observations:               18870   AIC:                         3.074e+05
Df Residuals:                   18846   BIC:                         3.076e+05
Df Model:                          23                                         
Covariance Type:            nonrobust                                         
===================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept       -3147.2834    752.541     -4.182      0.000   -4622.331   -1672.236
carat            9274.4188     76.087    121.892      0.000    9125.281    9423.557
depth              16.1771     10.611      1.525      0.127      -4.621      36.975
table             -24.0178      3.779     -6.356      0.000     -31.424     -16.611
x               -1089.2732    114.289     -9.531      0.000   -1313.290    -865.256
y                1050.1706    117.553      8.934      0.000     819.756    1280.585
z                -780.0003    135.845     -5.742      0.000   -1046.268    -513.732
cut_Good          378.3257     43.211      8.755      0.000     293.629     463.022
cut_Ideal         612.7896     42.028     14.581      0.000     530.411     695.168
cut_Premium       597.5695     40.417     14.785      0.000     518.348     676.791
cut_Very_Good     506.2854     41.266     12.269      0.000     425.400     587.171
color_E          -189.3134     22.461     -8.429      0.000    -233.339    -145.288
color_F          -252.1971     22.772    -11.075      0.000    -296.832    -207.562
color_G          -405.2318     22.194    -18.258      0.000    -448.735    -361.729
color_H          -835.5282     23.654    -35.323      0.000    -881.892    -789.164
color_I         -1303.3601     26.319    -49.521      0.000   -1354.948   -1251.772
color_J         -1885.2684     32.401    -58.186      0.000   -1948.776   -1821.760
clarity_IF       4022.3563     64.333     62.524      0.000    3896.259    4148.454
clarity_SI1      2570.7930     55.116     46.643      0.000    2462.760    2678.826
clarity_SI2      1728.3897     55.439     31.177      0.000    1619.725    1837.054
clarity_VS1      3371.8653     56.211     59.986      0.000    3261.687    3482.043
clarity_VS2      3081.9348     55.413     55.618      0.000    2973.321    3190.549
clarity_VVS1     3790.2442     59.567     63.630      0.000    3673.488    3907.001
clarity_VVS2     3747.1551     57.889     64.730      0.000    3633.688    3860.622
==============================================================================
Omnibus:                     4696.785   Durbin-Watson:                   1.994
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17654.853
Skew:                           1.208   Prob(JB):                         0.00
Kurtosis:                       7.076   Cond. No.                     1.06e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.06e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Fig.43

Inference:

- Multicollinearity is still present in the dataset.
- The obtained stats model displays that its features do not add value to the
- model, hence those features can be removed and that'll ultimately lead to
- reducing the VIF value.
- This will build a better linear regression model.
- This means that the sklearn models is saying for a one unit increase in 'carat' price
- will increase by 9247
- Similarly for a one unit increase in table price will increase by -24
- For a one unit increase in depth price will increase by 16

- The coefficients of table, x and z being negative indicate multicollinearity in dataset.
- After all variables like table, depth and x,y,z all measure dimensions.
- The intercept of the model is: -3147.283402498099 a very low amount which collaborates with the
- fact that the coefficients of the variables being 0 the price will be 0.
- The intercept for our model is -3147.283402498099
- The Coefficient of determinant or model score for the training set is 94.2%(unscaled data)
- R2 for the test set is also 94.2% (unscaled data). This means that 94% of
- the variation in dependent variable price can be explained by the independent
- variables in this model.
- Further the Root Mean Square Error (RMSE) for training data is 832.4206056108892
- and RMSE for test data is 870.953429935481.
- While this does look like the model seems to be the right fit as scores are not dipping
- from training set to test set, we need to look at other statistical scores which we will
- get using stats model library, consider these statistical scores above

The scatter plot:

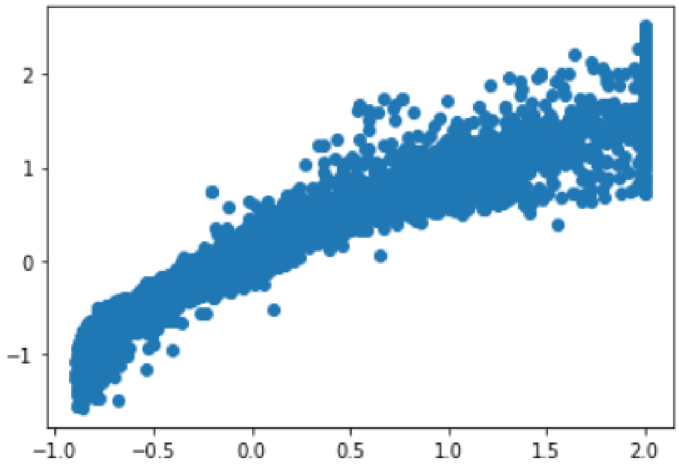
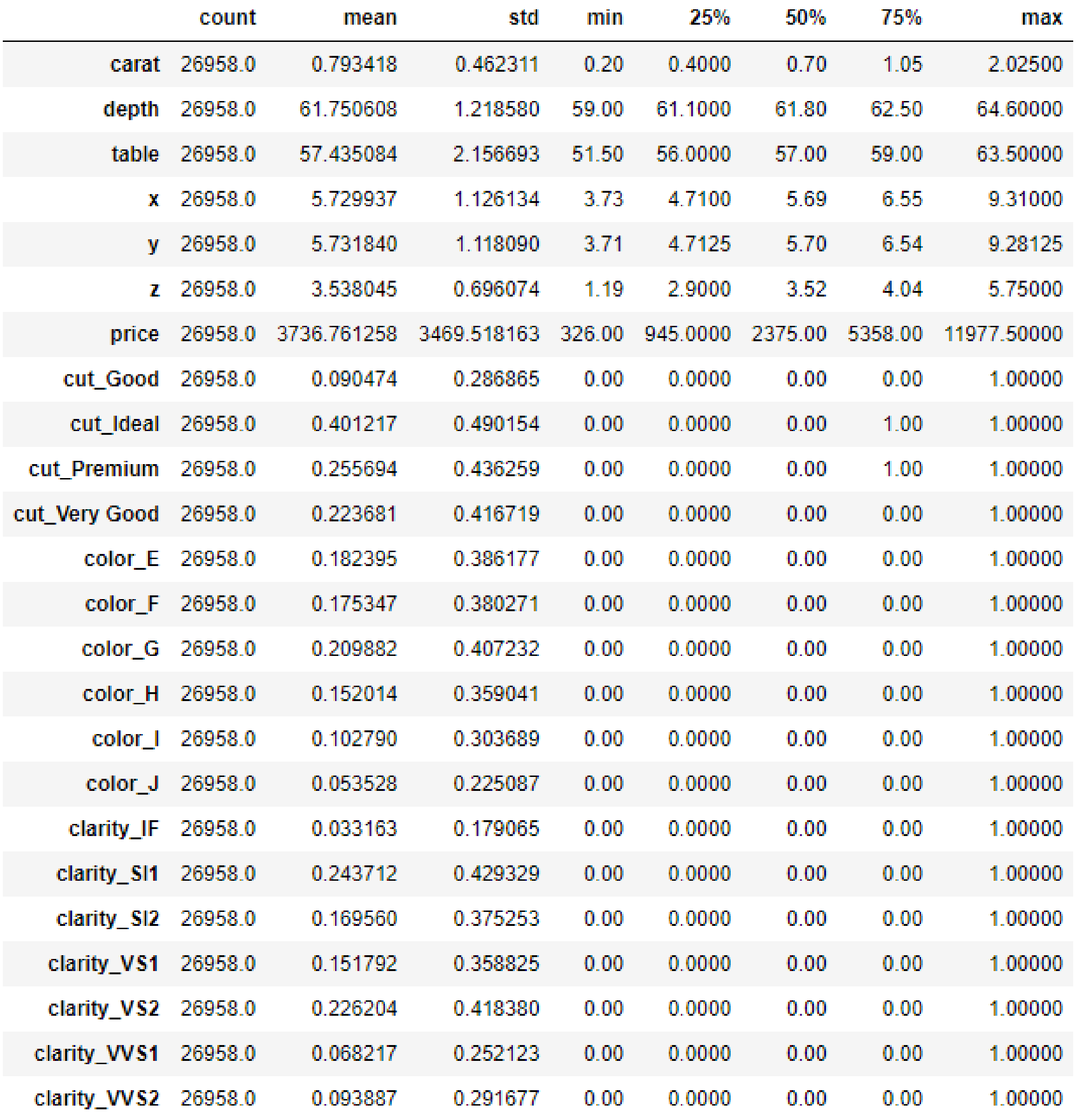


Fig.44

The Actual and predicted Price variable: The relationship is linear; hence the model is a good fit.

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

- From this model building exercise “carat” emerges as the key variable that determines price.
- Hence the manufacturer should focus on this aspect of stone, the appropriate carat value identification deserves further data analysis.
- one aspect worth noting is that 1 carat and 0.3 carat are both more present in the dataset.
- Further we had noticed during EDA that the stone fetching the highest price was not necessarily the one with highest color, or clarity or cut.



	count	mean	std	min	25%	50%	75%	max
carat	26958.0	0.793418	0.462311	0.20	0.4000	0.70	1.05	2.02500
depth	26958.0	61.750608	1.218580	59.00	61.1000	61.80	62.50	64.60000
table	26958.0	57.435084	2.156693	51.50	56.0000	57.00	59.00	63.50000
x	26958.0	5.729937	1.126134	3.73	4.7100	5.69	6.55	9.31000
y	26958.0	5.731840	1.118090	3.71	4.7125	5.70	6.54	9.28125
z	26958.0	3.538045	0.696074	1.19	2.9000	3.52	4.04	5.75000
price	26958.0	3736.761258	3469.518163	326.00	945.0000	2375.00	5358.00	11977.50000
cut_Good	26958.0	0.090474	0.286865	0.00	0.0000	0.00	0.00	1.00000
cut_Ideal	26958.0	0.401217	0.490154	0.00	0.0000	0.00	1.00	1.00000
cut_Premium	26958.0	0.255694	0.436259	0.00	0.0000	0.00	1.00	1.00000
cut_Very Good	26958.0	0.223681	0.416719	0.00	0.0000	0.00	0.00	1.00000
color_E	26958.0	0.182395	0.386177	0.00	0.0000	0.00	0.00	1.00000
color_F	26958.0	0.175347	0.380271	0.00	0.0000	0.00	0.00	1.00000
color_G	26958.0	0.209882	0.407232	0.00	0.0000	0.00	0.00	1.00000
color_H	26958.0	0.152014	0.359041	0.00	0.0000	0.00	0.00	1.00000
color_I	26958.0	0.102790	0.303689	0.00	0.0000	0.00	0.00	1.00000
color_J	26958.0	0.053528	0.225087	0.00	0.0000	0.00	0.00	1.00000
clarity_IF	26958.0	0.033163	0.179065	0.00	0.0000	0.00	0.00	1.00000
clarity_SI1	26958.0	0.243712	0.429329	0.00	0.0000	0.00	0.00	1.00000
clarity_SI2	26958.0	0.169560	0.375253	0.00	0.0000	0.00	0.00	1.00000
clarity_VS1	26958.0	0.151792	0.358825	0.00	0.0000	0.00	0.00	1.00000
clarity_VS2	26958.0	0.226204	0.418380	0.00	0.0000	0.00	0.00	1.00000
clarity_VVS1	26958.0	0.068217	0.252123	0.00	0.0000	0.00	0.00	1.00000
clarity_VVS2	26958.0	0.093887	0.291677	0.00	0.0000	0.00	0.00	1.00000

- in fact, this area needs some exploration; perhaps it will be worthwhile to cluster the dataset and then build models
- for each cluster for more accurate predictions.

- From this dataset we can also conclude that the market values mid value chain rather than premium stones in color and clarity.
- While E is the highest rated color it is not the most popular one, similarly the S1 clarity is preferred over IF clarity.

- The size of the stone also plays a key role in determining the price hence it will be
- worthwhile to invest in stones of a larger dimension, after doing an exploratory
- analysis of customer sentiment.

- We observe that the max price is for ‘Very Good’ cut followed by ‘Ideal’ this is
- interesting as literature identifies ‘Ideal’ as a more valuable cut than ‘Very Good’.
- This anomaly is consistent with the mean price of ‘Very Good’ zirconia also which is higher than the mean price of ‘Ideal’ zirconia.
- In fact the highest mean price is fetched by ‘Fair’ zirconia.
- Perhaps the stones of Fair, and very good cut were good on other attributes and hence scored a higher price.

- The max price is by SI1 type.
- The highest mean price is by SI2 type.
- The IF category gets the highest minimum price.
- G color has the max price Whereas J color(the poorest as per literature) has the highest average price

- The value of a future zirconia can be calculated using the equation below:
- PP =(-3147.28) * Intercept + (9274.42) * carat + (16.18) * depth + (-24.02) * table + (-1089.27) * x + (1050.17) * y + (-780.0) * z + (378.33) * cut_Good + (612.79) * cut_Ideal + (597.57) * cut_Premium + (506.29) * cut_Very_Good + (-189.31) * color_E + (-252.2) * color_F + (-405.23) * color_G + (-835.53) * color_H + (-1303.36) * color_I + (-1885.27) * color_J + (4022.36) * clarity_IF + (2570.79) * clarity_SI1 + (1728.39) * clarity_SI2 + (3371.87) * clarity_VS1 + (3081.93) * clarity_VS2 + (3790.24) * clarity_VVS1 + (3747.16) * clarity_VVS2 +
- Where PP= price predicted of zirconia
- And variables represent the respective value of the gem stone.
- In terms of importance carat,y,x,z and clarity emerge as the five most important variables(as per their coefficients).
- As y,x,z are width, length and height we interpret them as the size of the gem stone.

- Gem Stones co ltd should focus more in the dimensions of the cubic zirconia like weight, length, height & width.
- Should focus less in the depth of cubic zirconia.
- Should work towards keeping clarity of cubic zirconia in a range from VVS1 to SI1.

Problem 2:

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

Variables:

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

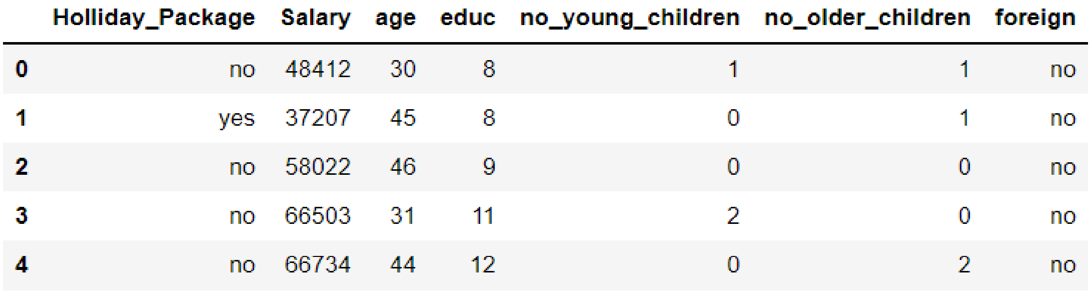
2.1 Data Ingestion: Read the dataset. Do the descriptive statistics and do null value condition check, write an inference on it. Perform Univariate and Bivariate Analysis. Do exploratory data analysis.

Exploratory Data Analysis:

We will explore the Clustering Data set and perform the exploratory data analysis on the dataset. The major topics to be covered are below:

- Removing duplicates
- Missing value treatment
- Outlier Treatment
- Normalization and Scaling (Numerical Variables)
- Encoding Categorical variables (Dummy Variables)
- Univariate Analysis
- Bivariate Analysis

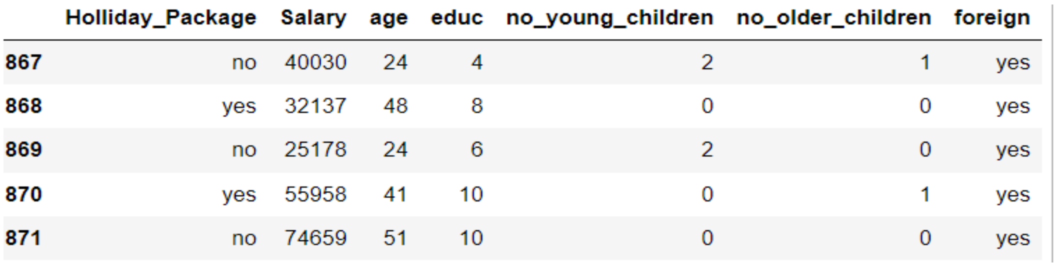
Data Insights of top 5 records:



	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

Fig.45

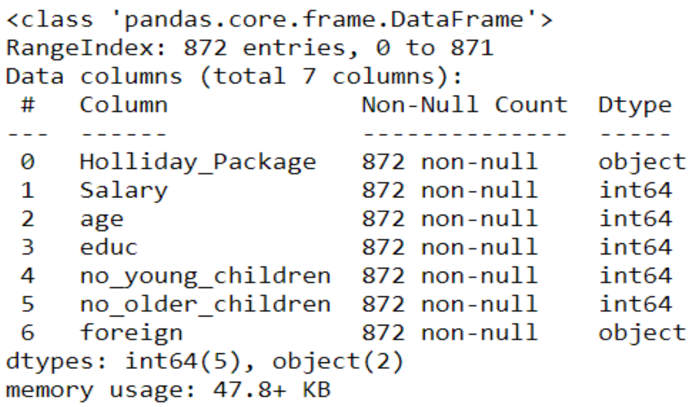
Data Insights of bottom 5 records:



	Holliday_Package	Salary	age	educ	no_young_children	no_older_children	foreign
867	no	40030	24	4	2	1	yes
868	yes	32137	48	8	0	0	yes
869	no	25178	24	6	2	0	yes
870	yes	55958	41	10	0	1	yes
871	no	74659	51	10	0	0	yes

Fig.46

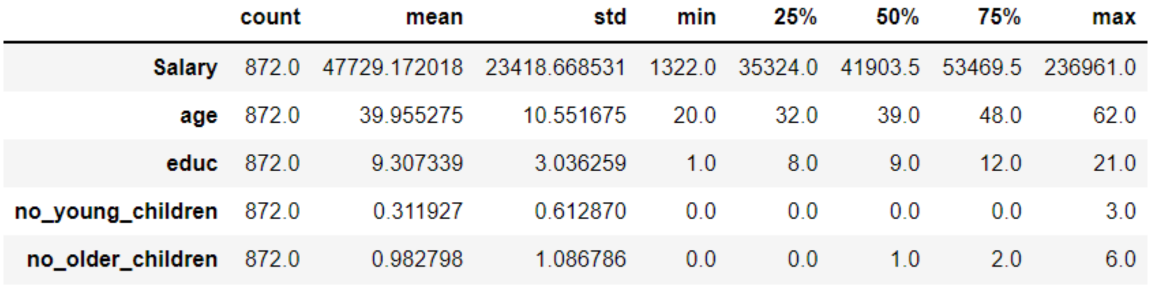
Data Info:



```
<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
```

Fig.47

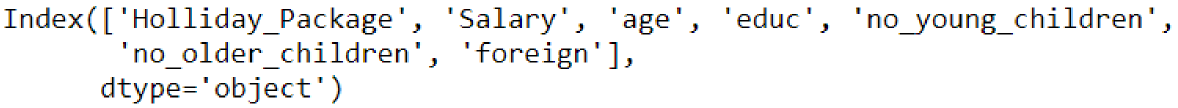
Data Summary:



	count	mean	std	min	25%	50%	75%	max
Salary	872.0	47729.172018	23418.668531	1322.0	35324.0	41903.5	53469.5	236961.0
age	872.0	39.955275	10.551675	20.0	32.0	39.0	48.0	62.0
educ	872.0	9.307339	3.036259	1.0	8.0	9.0	12.0	21.0
no_young_children	872.0	0.311927	0.612870	0.0	0.0	0.0	0.0	3.0
no_older_children	872.0	0.982798	1.086786	0.0	0.0	1.0	2.0	6.0

Fig.48

Total columns:



```
Index(['Holliday_Package', 'Salary', 'age', 'educ', 'no_young_children',
       'no_older_children', 'foreign'],
      dtype='object')
```

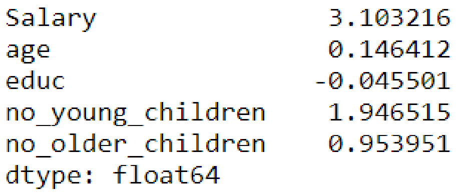
Fig.49

Data Dimensions: (872,7)

Missing values : 0

Duplicates in the data : 0

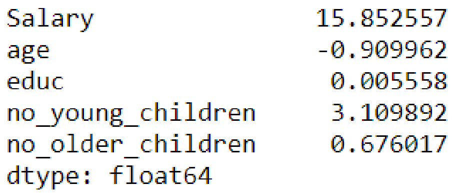
Skewness :



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

Fig.50

Kurtosis :



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

Fig.51

Outliers:

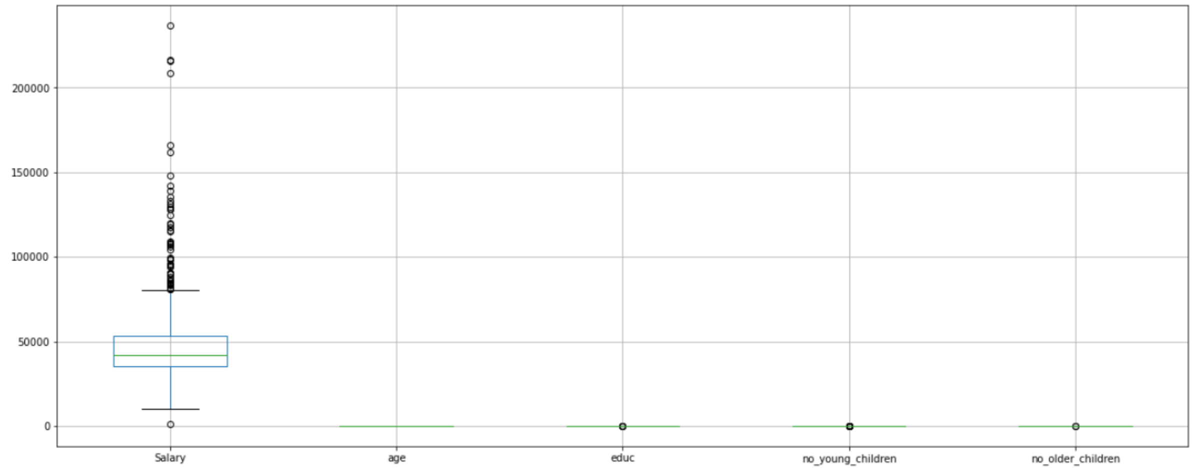


Fig.52

Histograms:

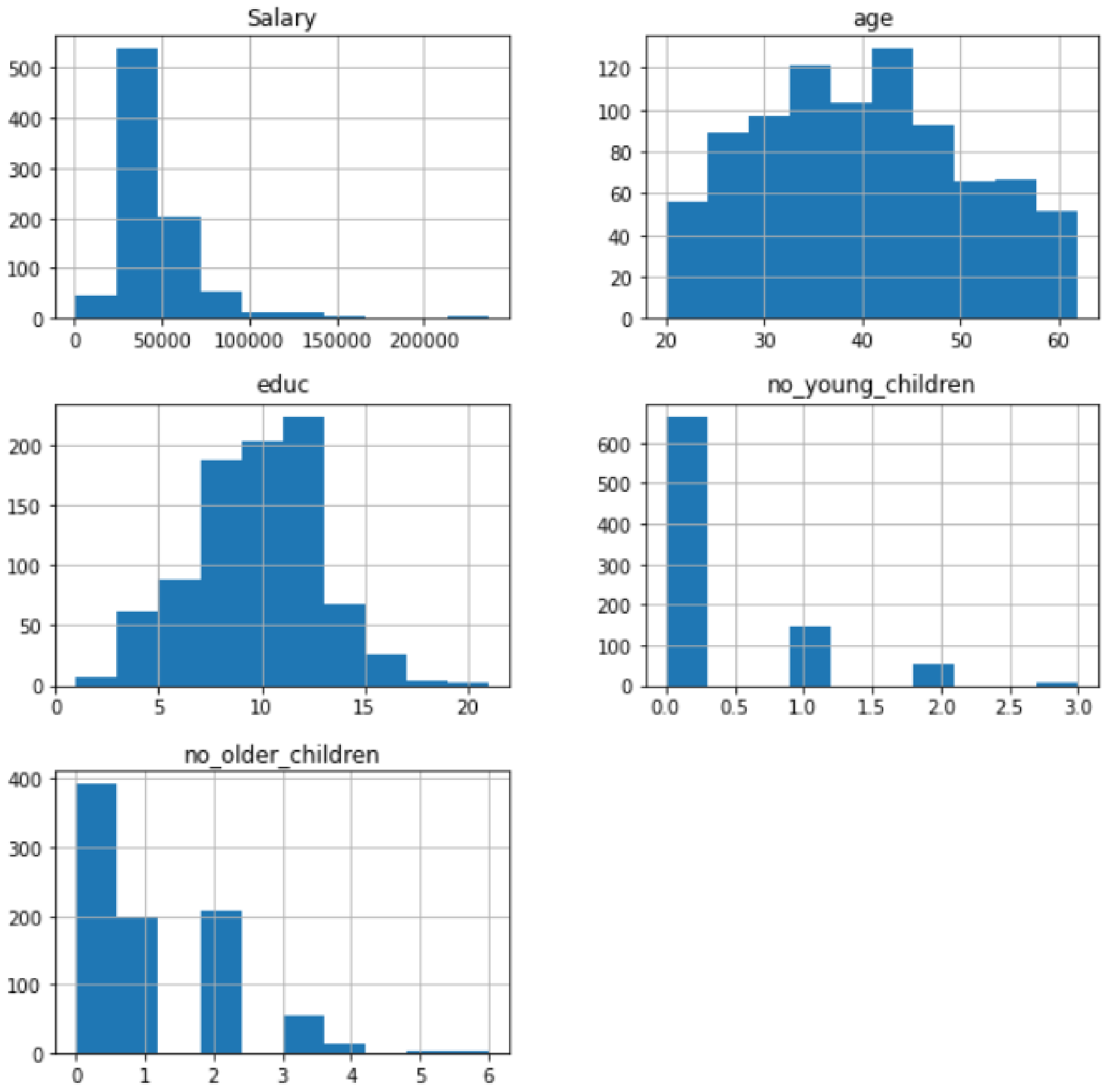


Fig.53

Univeriate Analysis:

Holiday_package

- No 471
- Yes 401

Countplot: Holiday Package

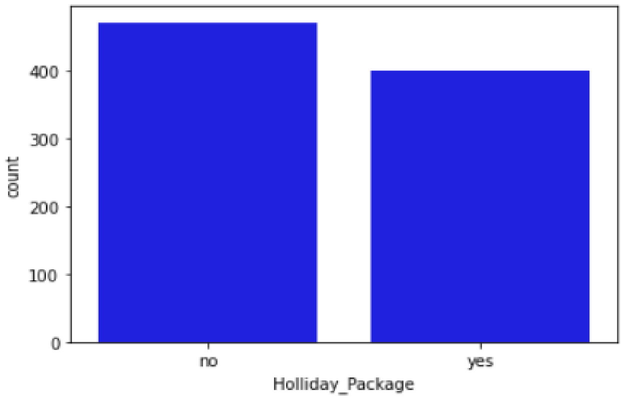


Fig.54

Foreign

- No 656
- Yes 216

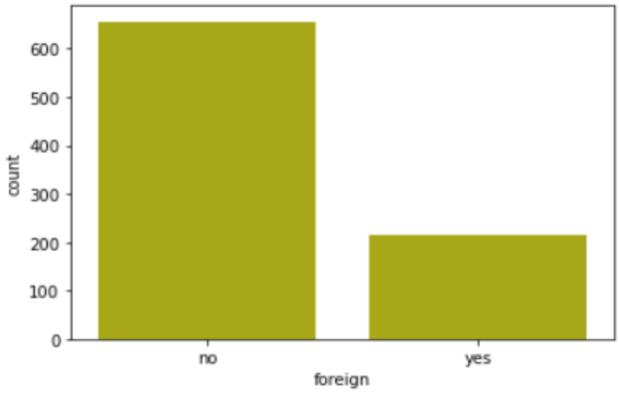


Fig.55

- Name: foreign, dtype: int64

Number of Young children of the employees

- 0 665
- 1 147
- 2 55
- 3 5

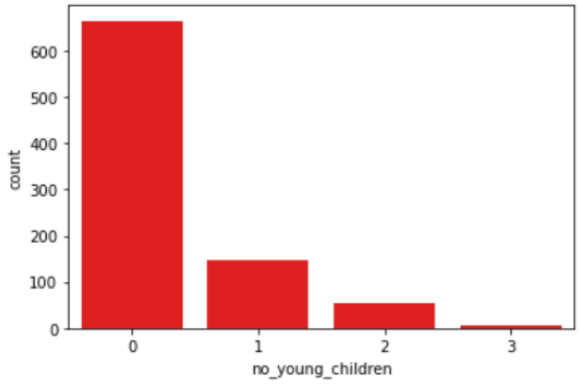


Fig.56

Number of older children of the employees

- 0 393
- 2 208
- 1 198
- 3 55
- 4 14
- 6 2

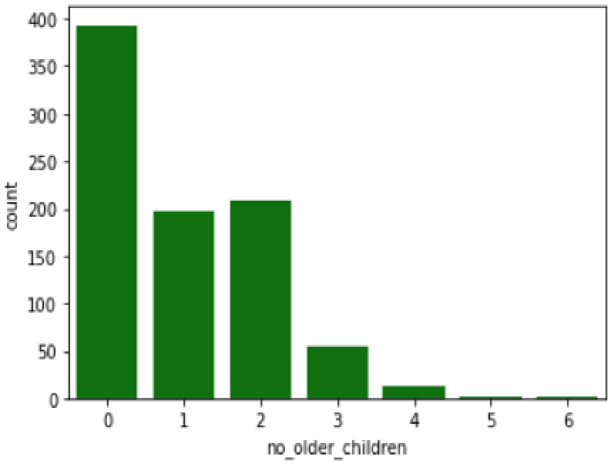


Fig.57

Salary:

- Mean: 47729.172018348625
- Median: 41903.5
- Mode: 32197
- Distribution: Positive skew

Age:

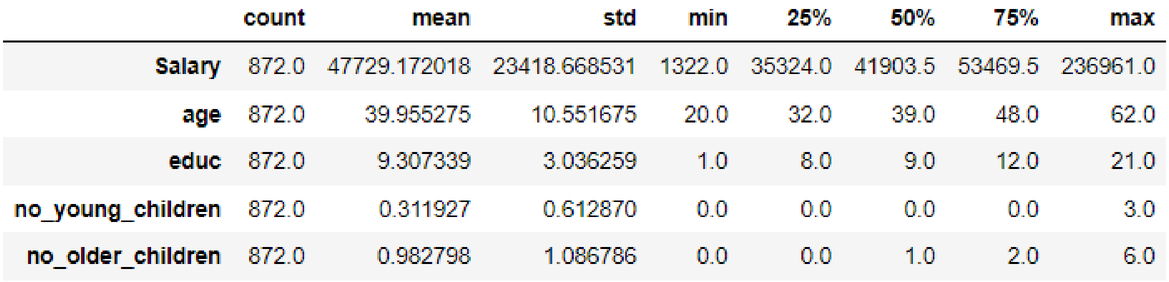
- Mean: 39.955275229357795
- Median: 39.0
- Mode: 44
- Distribution: Negative Skew

Number of years of Education:

- Mean: 9.307339449541285
- Median: 9.0
- Mode: 8
- Distribution: Positive Skew

Multivariate Analysis :

Correlation matrix :



	count	mean	std	min	25%	50%	75%	max
Salary	872.0	47729.172018	23418.668531	1322.0	35324.0	41903.5	53469.5	236961.0
age	872.0	39.955275	10.551675	20.0	32.0	39.0	48.0	62.0
educ	872.0	9.307339	3.036259	1.0	8.0	9.0	12.0	21.0
no_young_children	872.0	0.311927	0.612870	0.0	0.0	0.0	0.0	3.0
no_older_children	872.0	0.982798	1.086786	0.0	0.0	1.0	2.0	6.0

Fig.58

Heat Map:

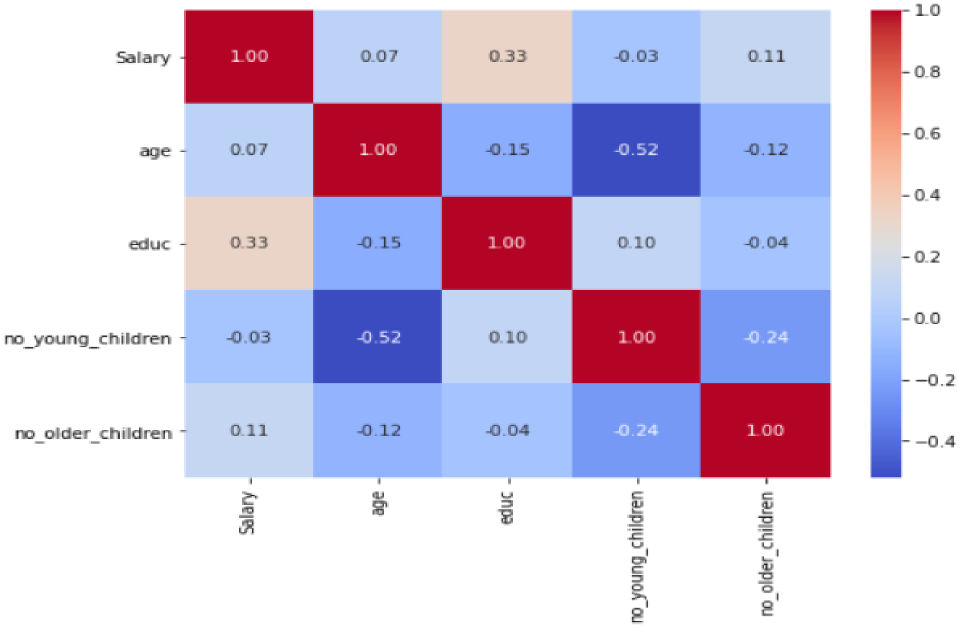


Fig.59

Pair Plot:

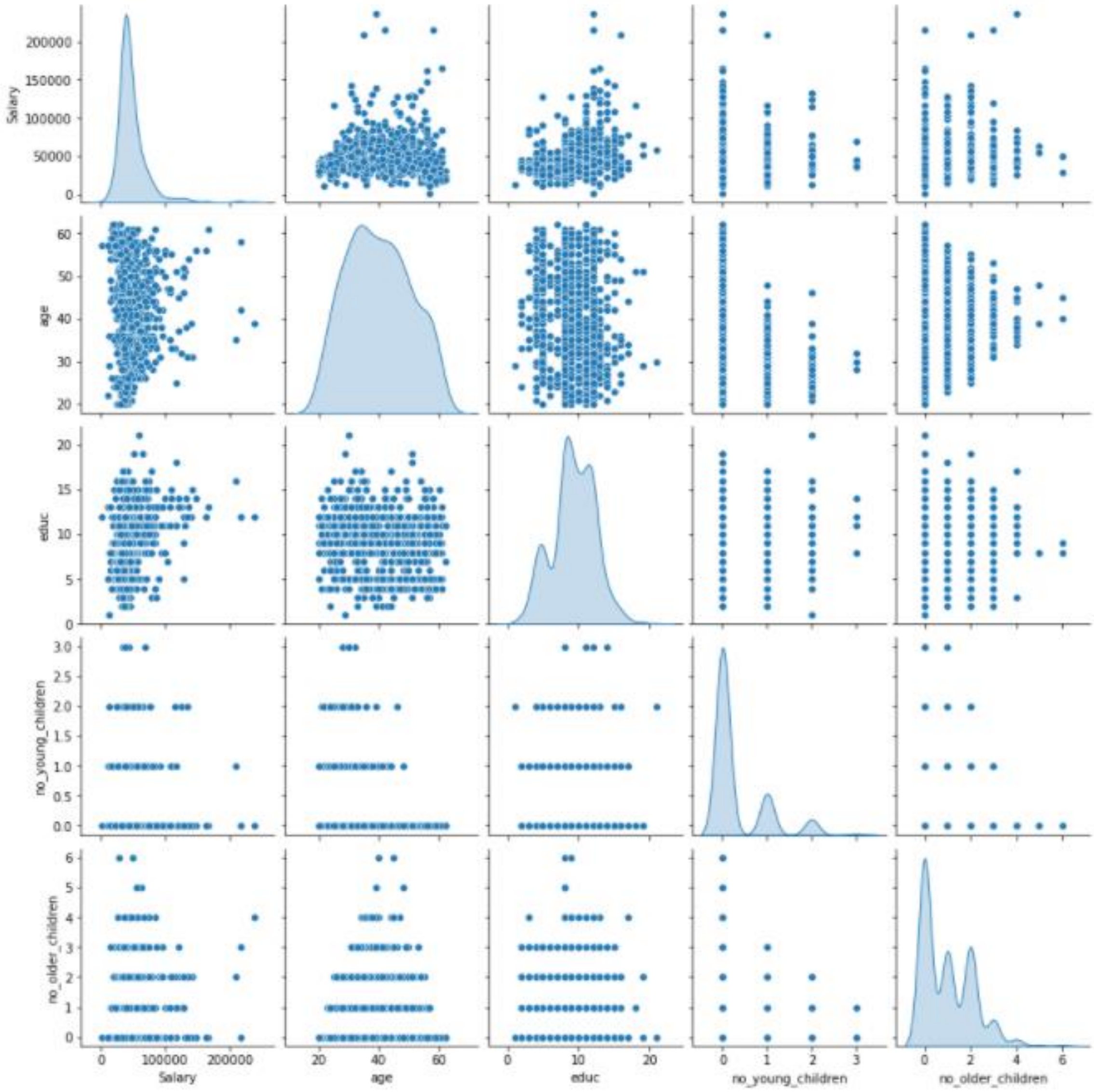


Fig.60

- There is not much of a correlation between the variables.
- No definite class differences
- No linear Corelation between the variables.
- Lack of Multicollinearity

Inferences:

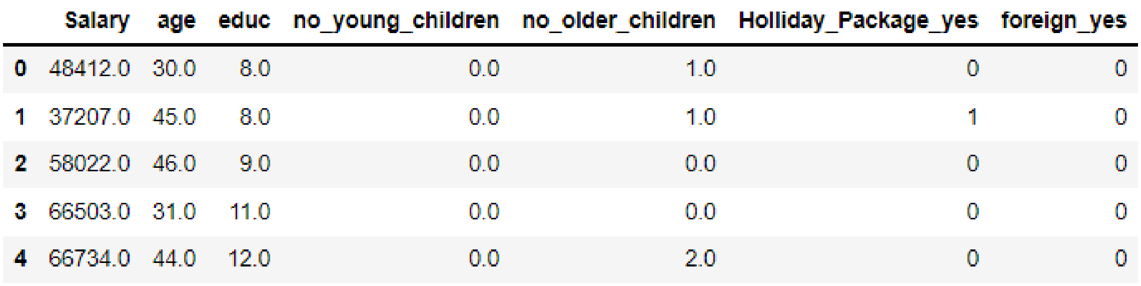
- After dropping the column “Unnanmed:0” the dataset has 7 variables and 872 observations.
- variables are of yes and no Boolean type(“Holiday_Package” and “foreign” and 5 are of numeric type (“Salary”, ”age”, ”edu”, ”no_young_children”, ”no_older_children”)
- There are no missing values in the dataset nor any duplicates.
- However, the variable “no_older_children” has 45.1% values as 0, also the variable “no_young_children” has 76% values as 0.
- The data is balanced in terms of dependent variable “Holliday_Package” with 54%
- values as False (No) and 46% as True (Yes)
- The salary is spread between 1322 to 236961, view spread of the salary below:
- The “age” variable is spread between 20 and 62
- The variable names “educ” that denotes years of formal education is spread between 1 and 21; with most values being between 8-12.
- The variable “no_young_children” is between 0-3 and most values are 0,
- The variable “no_older_children” is between 0-6 and most values are 0
- The last variable “foreign” has 656 False/No values and 216 True/Yes values.
- All the numeric variables except “age” have outliers
- Further we donot see a significant correlation between the variables, please see heat
- map below; only the no_young_children variable and age show a mid level negative correlation.
- The pairplot also reveals no significant correlations
- When we make a pairplot with a hue of dependent variable “Holliday_Package” we see that none of the variables is able to significantly identify between people who
- take the holiday package from those who don’t.
- With this we conclude the exploratory analysis the variables donot seem to be well chosen as none of them is able to significantly find the difference between those who opt for holiday package and those who don’t.
- Further the Depth versus Breadth analysis shows the number of observations to be limited and not sufficient to capture all the variations in the parameters.

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).

- The train-test split is a technique for evaluating the performance of a machine learning algorithm.
- It can be used for classification or regression problems and can be used for any supervised learning algorithm.
- The procedure involves taking a dataset and dividing it into two subsets.
- The first subset is used to fit the model and is referred to as the training dataset.
- The second subset is not used to train the model; instead, the input element of the dataset is provided to the model, then predictions are made and compared to the expected values.
- This second dataset is referred to as the test dataset.
- Train Dataset: Used to fit the machine learning model.
- Test Dataset: Used to evaluate the fit machine learning model.
- Hence, estimated the performance of the machine learning model on new data.
- The train-test procedure is appropriate when there is a sufficiently large dataset available.
- Splitting the data into test and train In the ratio of 30 and 70 for test and training respectively.

<u>Logistic Regression Model:</u>

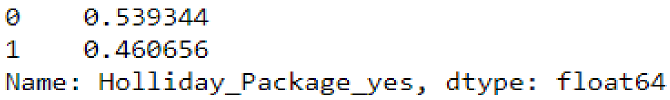
<u>Step 1: Convert categorical to dummy variables in data</u>



	Salary	age	educ	no_young_children	no_older_children	Holliday_Package_yes	foreign_yes
0	48412.0	30.0	8.0	0.0	1.0	0	0
1	37207.0	45.0	8.0	0.0	1.0	1	0
2	58022.0	46.0	9.0	0.0	0.0	0	0
3	66503.0	31.0	11.0	0.0	0.0	0	0
4	66734.0	44.0	12.0	0.0	2.0	0	0

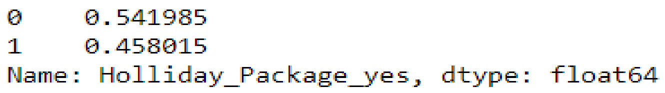
Fig.61

<u>Step 2: Data Split: Split the data into train and test (70:30)</u>



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

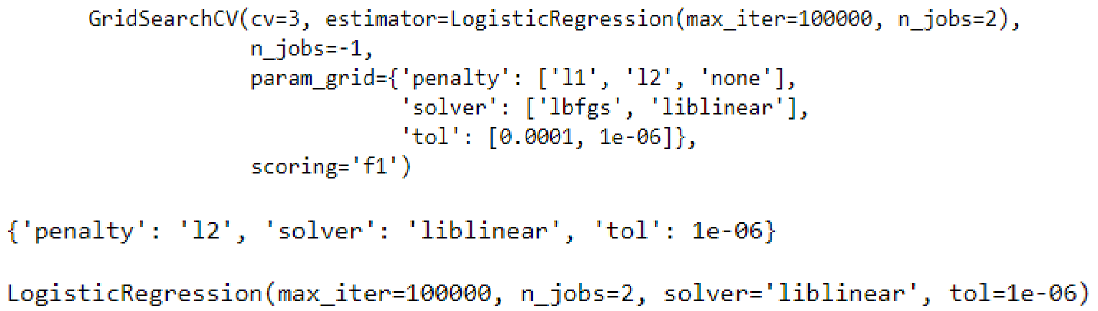
Fig.62



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

Fig.63

Step 3: Building Logistic Regression Model (Grid Search method):



```
GridSearchCV(cv=3, estimator=LogisticRegression(max_iter=100000, n_jobs=2),
             n_jobs=-1,
             param_grid={'penalty': ['l1', 'l2', 'none'],
                         'solver': ['lbfgs', 'liblinear'],
                         'tol': [0.0001, 1e-06]},
             scoring='f1')

{'penalty': 'l2', 'solver': 'liblinear', 'tol': 1e-06}

LogisticRegression(max_iter=100000, n_jobs=2, solver='liblinear', tol=1e-06)
```

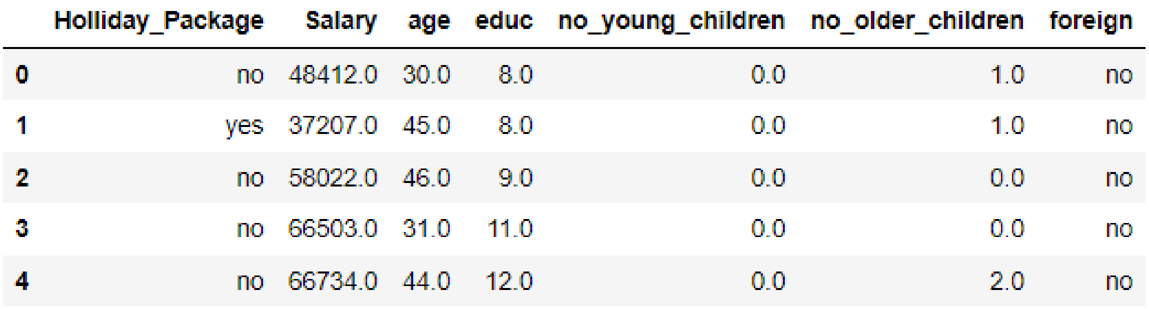
Fig.64

● Grid-search is used to find the optimal hyperparameters of a model which results in the most 'accurate' predictions.

● It's seen here that the Grid search method gave a liblinear solver. This liblinear solver is most suitable for small datasets.

● Penalty and tolerance have been found using this method

Linear Discriminant Analysis (LDA) Model:



	Holliday_Package	Salary	age	educ	no_young_children	no_older_children	foreign
0	no	48412.0	30.0	8.0	0.0	1.0	no
1	yes	37207.0	45.0	8.0	0.0	1.0	no
2	no	58022.0	46.0	9.0	0.0	0.0	no
3	no	66503.0	31.0	11.0	0.0	0.0	no
4	no	66734.0	44.0	12.0	0.0	2.0	no

Fig.65

- Data Split: Data Split
- X = Independent Variables
- Y = Dependent variable: Holiday Package
- Applying Logistic regression
- From sklearn.model_selection is used
- Train_test_split is imported
- Training Data: 70%
- Test Data: 30%
- Target: Holiday package Claim

- Training Set Value Count:
- 0.539344
- 0.460656
- Test Set Value Count:
- 0.541985
- 0.458015
- Parameters:
- max_iter=10000
- N_jobs=2
- Penalty='none'
- Solver='newton-cg'
- Verbose=True

- Predicting on Training and Test dataset
- Getting the Predicted Classes and respective Probabilities
- Applying Linear Discrimination Analysis
- Target: Holiday package Claim
- Applying the fit model on Linear Discriminant analysis
- Predicting on Training and Test dataset

- Getting the Predicted Classes and respective Probabilities

- We encode the dependent variable Holiday Package where the choice is “No” as 0 and 1 where the choice is “Yes”
- Similarly, the independent variable “foreign” we code using one hot coding.
- We remove outliers from the data by replacing the upper value outliers with the
- upper whisker value and the lower value outliers with the lower whisker values.

- Except for Holiday Package Claim and Foreign, the rest of the categorical data are ordinal.
- Holiday Package and Foreign have yes or no categories, using Panda’s feature of converting categorical objects to Codes.

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.

Confusion Matrix:

- The confusion matrix provides more insight into not only the performance of a predictive model, but also which classes are being predicted correctly, which incorrectly, and what type of errors are being made.
- The simplest confusion matrix is for a two-class classification problem, with negative (class 0) and positive (class 1) classes.

ROC Curves and ROC AUC:

- An ROC curve (or receiver operating characteristic curve) is a plot that summarizes the performance of a binary classification model on the positive class.
- Ideally, we want the fraction of correct positive class predictions to be 1 (top of the plot) and the fraction of incorrect negative class predictions to be 0 (left of the plot). This highlights that the best possible classifier that achieves perfect skill is the top-left of the plot (coordinate 0,1)
- The curve provides a convenient diagnostic tool to investigate one classifier with different threshold values and the effect on the TruePositiveRate and FalsePositiveRate.
- One might choose a threshold in order to bias the predictive behavior of a classification model.

ROC Area Under Curve (AUC) Score:

- Instead, the area under the curve can be calculated to give a single score for a classifier model across all threshold values. This is called the ROC area under curve or ROC AUC or sometimes ROCAUC.
- The score is a value between 0.0 and 1.0 for a perfect classifier.

Precision-Recall Curves and AUC:

- Precision is a metric that quantifies the number of correct positive predictions made.
- It is calculated as the number of true positives divided by the total number of true positives and false positives.

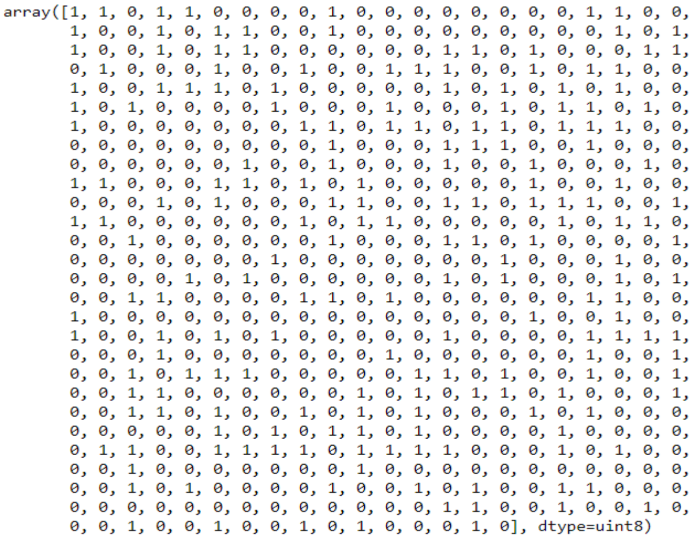
Precision-Recall Area Under Curve (AUC) Score:

- The Precision-Recall AUC is just like the ROC AUC, in that it summarizes the curve with a range of threshold values as a single score.

- The score can then be used as a point of comparison between different models on a binary classification problem where a score of 1.0 represents a model with perfect skill.

Logistic Regression Model:

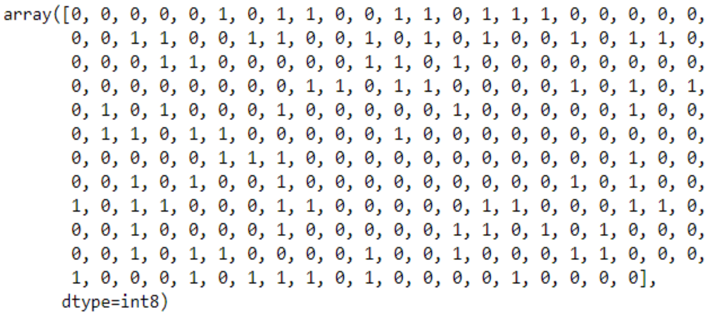
Prediction on the training set:



```
array([1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0,
       1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1,
       1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1,
       0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0,
       1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0,
       1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0,
       1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0,
       1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0,
       0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1,
       1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0,
       0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1,
       0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,
       0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1,
       0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0,
       1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0,
       1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1,
       0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1,
       0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 1,
       0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1,
       0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0,
       0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0,
       0, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0,
       0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0,
       0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0], dtype=uint8)
```

Fig.66

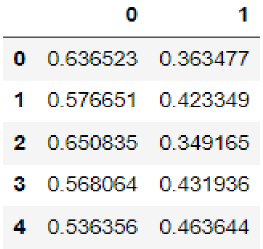
Prediction on the test set:



```
array([0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0,
       0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0,
       0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1,
       0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0,
       0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
       0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
       1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0,
       0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0,
       0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0,
       1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0],
      dtype=int8)
```

Fig.67

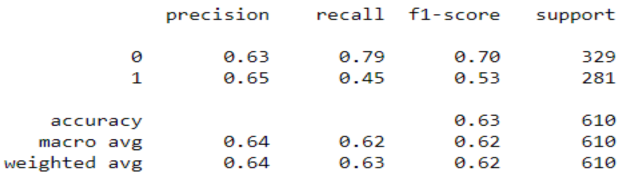
Probabilities on the test set:



	0	1
0	0.636523	0.363477
1	0.576651	0.423349
2	0.650835	0.349165
3	0.568064	0.431936
4	0.536356	0.463644

Fig.68

Confusion matrix on the training data:



```
              precision    recall  f1-score   support

           0       0.63      0.79      0.70       329
           1       0.65      0.45      0.53       281

    accuracy                           0.63       610
   macro avg       0.64      0.62      0.62       610
weighted avg       0.64      0.63      0.62       610
```

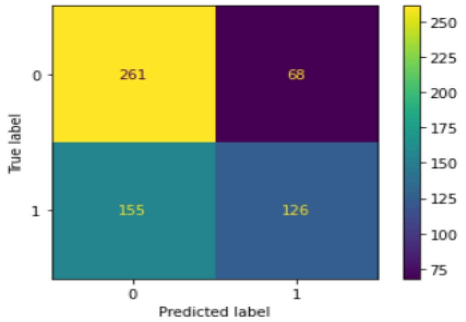
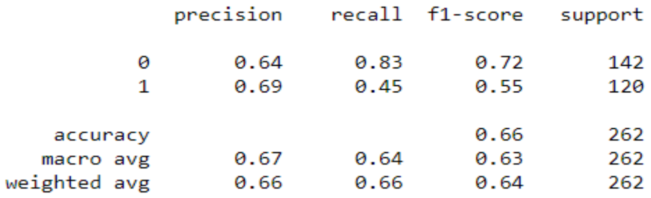


Fig.69

Confusion matrix on the test data:



```
              precision    recall  f1-score   support

           0       0.64      0.83      0.72       142
           1       0.69      0.45      0.55       120

    accuracy                           0.66       262
   macro avg       0.67      0.64      0.63       262
weighted avg       0.66      0.66      0.64       262
```

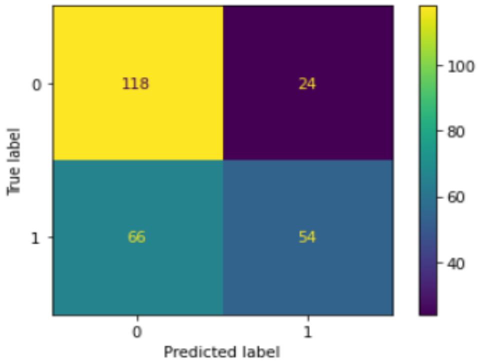
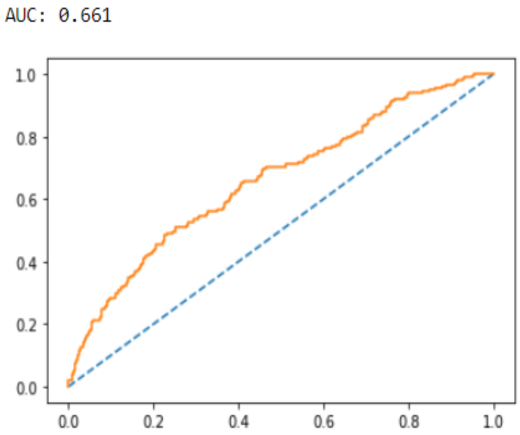


Fig.70

Accuracy of the training Data: 0.6344262295081967

AUC and ROC for the training data:

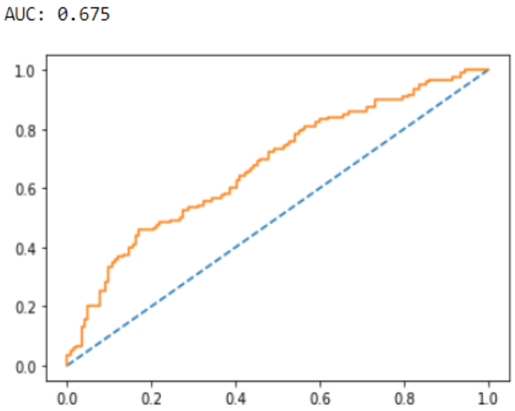


```
AUC: 0.661
```

Fig.71

Accuracy of the test Data: 0.6564885496183206

AUC and ROC for the testing data:



```
AUC: 0.675
```

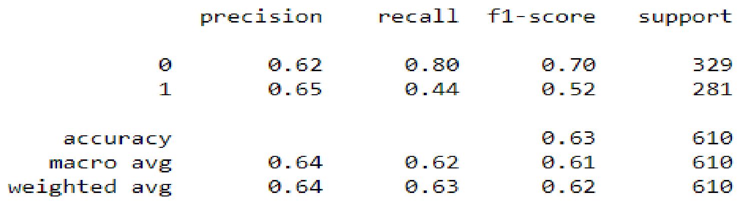
Fig.72

Linear Discriminant Analysis (LDA) Model:

Training data probability prediction: 0.6327868852459017

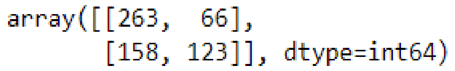
Test data probability prediction: 0.6564885496183206

Classification report and confusion matrix of training data:



```
              precision    recall  f1-score   support

           0       0.62      0.80      0.70       329
           1       0.65      0.44      0.52       281

    accuracy                           0.63       610
   macro avg       0.64      0.62      0.61       610
weighted avg       0.64      0.63      0.62       610
```

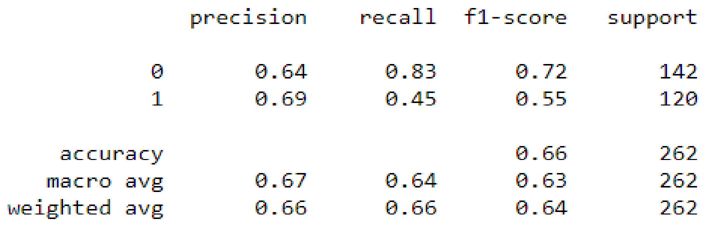
Fig.73



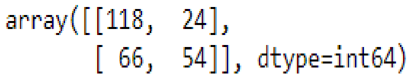
```
array([[263,  66],
       [158, 123]], dtype=int64)
```

Fig.74

Classification report and confusion matrix of test data:



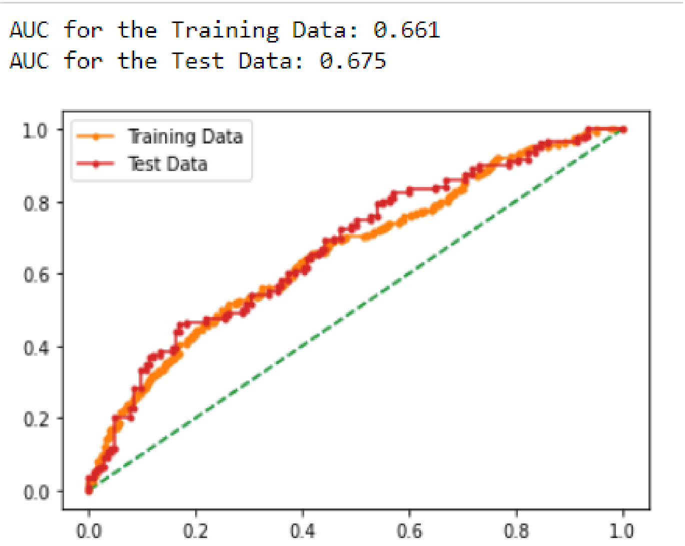
```
              precision    recall  f1-score   support

           0       0.64      0.83      0.72       142
           1       0.69      0.45      0.55       120

    accuracy                           0.66       262
   macro avg       0.67      0.64      0.63       262
weighted avg       0.66      0.66      0.64       262
```



```
array([[118,  24],
       [ 66,  54]], dtype=int64)
```

Fig.75

AUC and ROC for the training and test data:



```
AUC for the Training Data: 0.661
AUC for the Test Data: 0.675
```

Fig.76

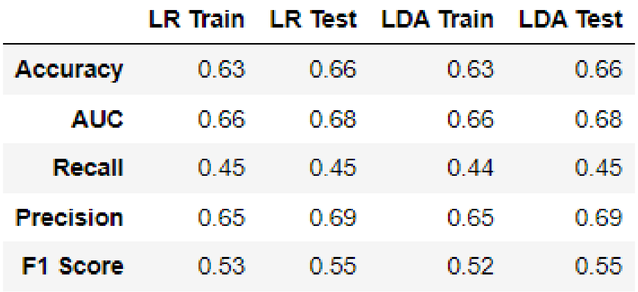
Comparing the Linear Regression and Linear Discriminant Analysis (LDA) Models:

Precision, Recall and F1 Scores of Train and Test data of Linear Regression Model

- lr_train_precision 0.65
- lr_train_recall 0.45
- lr_train_f1 0.53
- lr_test_precision 0.69
- lr_test_recall 0.45
- lr_test_f1 0.55

- lda_train_precision 0.65
- lda_train_recall 0.44
- lda_train_f1 0.53
- lda_test_precision 0.69
- lda_test_recall 0.45
- lda_test_f1 0.55

Comparing the Linear Regression and Linear Discriminant Analysis (LDA) Models - based on their respective Precision, Recall and F1 Scores of Train and Test data :



	LR Train	LR Test	LDA Train	LDA Test
Accuracy	0.63	0.66	0.63	0.66
AUC	0.66	0.68	0.66	0.68
Recall	0.45	0.45	0.44	0.45
Precision	0.65	0.69	0.65	0.69
F1 Score	0.53	0.55	0.52	0.55

Fig.77

Accuracy:

- Accuracy can be defined as the percentage of correct predictions made by our classification model.
- The formula is:
- Accuracy = Number of Correct predictions/number of rows in data
- Which can also be written as:
- Accuracy = (TP+TN)/number of rows in data

Precision:

- Precision indicates out of all positive predictions; how many are actually positive. It is defined as a ratio of correct positive predictions to overall positive predictions.
- Precision = Predictions actually positive/Total predicted positive.
- Precision = TP/TP+FP

Recall:

- Recall indicates out of all actually positive values; how many are predicted positive. It is a ratio of correct positive predictions to the overall number of positive instances in the dataset.
- Recall = Predictions actually positive/Actual positive values in the dataset.
- Recall = TP/TP+FN

F1 score:

- An f1 score is defined as the harmonic mean of precision and recall.
- Used to predict if a particular employee has to be promoted or not and promotion is the positive outcome.
- In this case, promoting an incompetent employee(false positive) and not promoting a deserving candidate(false negative) can both be equally risky for the company.
- When avoiding both false positives and false negatives are equally important. we need a trade-off between precision and recall. This is when we use the f1 score as a metric.

Threshold:

- Any machine learning algorithm for classification gives output in the probability format,
- In order to assign a class to an instance for binary classification, we compare the probability value to the threshold, i.e if the value is greater than or less than the threshold.

AUC-ROC:

- We use the receiver operating curve to check model performance.
- Wikipedia defines ROC as: “A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied”.

- We encode the dependent variable Holiday Package where the choice is “No” as 0 and 1 where the choice is “Yes”
- Similarly, the independent variable “foreign” we code using one hot coding.
- We remove outliers from the data by replacing the upper value outliers with the upper whisker value and the lower value outliers with the lower whisker values.

- Pd.Categorical: it converts the object values like Yes and No in to numeric values in the dataset for Holliday_Package & Foreign

- Using sklearn.model_selection import train_test_split

- And split the data into train and test (70:30), to scaled data and from the correlation matrix we saw that cut, color and clarity are not important so we drop them in second dataset.

- Applying Logistic Regression and Linear Discriminant Analysis and fit on the dataset and check the score for both training and test dataset, confusion matrix, classification report, accuracy and roc
- auc curve and find the test scores for the dataset

- By comparing both the models we find that:
- Logistic Regression: the model score is less than Linear Discriminant Analysis in every way

- Linear Discriminant Analysis: the model score is more than Logistic Regression in every way

- It is not that Logistic Regression is bad it is because Logistic Regression works best on large data sets while it is inverse in case of Linear Discriminant Analysis

- Consider the following table comparing the models; as the Accuracy, Recall, Precision & F1 is higher for LDA we choose the LDA model.

2.4 Inference: Basis on these predictions, what are the insights and recommendations. Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present.

- With the dataset given and the variables chosen we were able to identify as age, salary and education as three variables that help decide whether a person chooses a holiday package or not.
- People of a lower age even from a lower salary bracket choose holiday more than people from higher age group.
- In this business case, a travel agency is looking for statistical evident predictions on employees likely to prefer to opt for the holiday package or not.
- The given dataset has information about the economic, behavioural, and age group. Based on the exploratory data analysis and models built, below are the business insights.

Insights:

- The comparison between the employee age range, salary, holiday package preference shows that the holiday package is preferred by employee’s salary range below 50,000.

- Employees' salaries ranging below 50,000 are of age range 30 -50. Hence, this age
- group has actively opted for the holiday package
- Employees aged over 50 to 60 have shown a tendency of not opting for the holiday package.
- On the other hand, employees with a salary of more than 150,000 are also not opting for the holiday package. This salary range has employees from both 30 - 50 and 50 - 60 age groups
- The holiday package is also not preferred by the employees having young children.
- Employees with older children are opting for the package normally.
- We can see three major groups where strategy implementation is needed to grow the holiday package subscription:
- Old age group - Employees aged between 50-60
- Elite group - Salary range more than 150,000

Business Recommendations:

- The business recommendation can be given based on the targeting segments created.
- Old age group - Employees aged between 50-60
- The holiday package of the holy-places (religious places) can be pitched to these people
- A survey asking their preferences, beliefs can help in creating a personalized package, which will have a high probability of acceptance
- Elite group - Salary range more than 150,000
- This group would majorly prefer to have an experience-based holiday package.
- Looking at their salary range, they can be offered for abroad trip for the age group between 30 - 50.
- This package can have some personalized adventurer stay experience deal included For the age group between 50 - 60, this package can have a family trip experience both domestic and international.
- This package can also have a whole trip experience with a personal local guide at a holy place.
- Employees with small children
- These people can be offered a trip to parks such as Disneyland (if higher salary range) or domestic parks such as Wonderla, Queens Land, etc. (if low salary range)
- Kids' special experiences such as live cartoon shows will be an add-on advantage in the package and increase the chance of opting for the package.
- Apart from these groups, families with older children can be offered a family holiday
- plan to keep up/continue their tendency of opting for the holiday packages.
- Perhaps older people are not finding holidays that appeal to their interest, the

company should design packages for an older audience as older people often have higher salaries and more disposable income.

- Also, we need to fine tune our variable selection and identify variables that help distinguish between those who chose holiday package or not as current variable selection did not help build a very robust model.