Predictive Modelling ASSIGMENT

Submitted By

Vinit Sharma

Table of Contents

Figures of Contents

Problem 1: Linear Regression

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

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

This dataset is based on cubic zirconia details for company Gem Stones. This dataset has 26967 rows with 10 variables. Out of 10, three variables contain object data type and rest seven variables contain int and float datatype. In the below figure, we have shown the dataset of zirconia.

	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

FIGURE 1: ZIRCONIA DATASET

```
<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
```

FIGURE 2 DATATYPE INFORMATION OF VARIABLES

Brief summary of dataset has been described below. Dataset contains 9 independent variables and one dependent variables. Here, price is a predictive targeted value.

	carat	depth	table	x	y	z	price
count	26967.000000	26270.000000	26967.000000	26967.000000	26967.000000	26967.000000	26967.000000
mean	0.798375	61.745147	57.456080	5.729854	5.733569	3.538057	3939.518115
std	0.477745	1.412860	2.232068	1.128516	1.166058	0.720624	4024.864666
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	5.710000	3.520000	2375.000000
75%	1.050000	62.500000	59.000000	6.550000	6.540000	4.040000	5360.000000
max	4.500000	73.600000	79.000000	10.230000	58.900000	31.800000	18818.000000

FIGURE 3 DATASET DESCRIPTION

This dataset does contain null value for variable depth. Similarly, dataset does not have any duplicate rows.

```
> carat, Missing: 0 (0.0%)
> cut, Missing: 0 (0.0%)
> color, Missing: 0 (0.0%)
> clarity, Missing: 0 (0.0%)
> depth, Missing: 697 (2.6%)
> table, Missing: 0 (0.0%)
> x, Missing: 0 (0.0%)
> y, Missing: 0 (0.0%)
> z, Missing: 0 (0.0%)
> price, Missing: 0 (0.0%)
```

FIGURE 4 NULL VALUE IDENTIFICATION

Count plot has been drawn for Cut, Colour and Clarity variables. From the plot, Cut contains very high “Ideal” value. Similarly, “Colour” and “Clarity” have “G” and “S1” respectively.

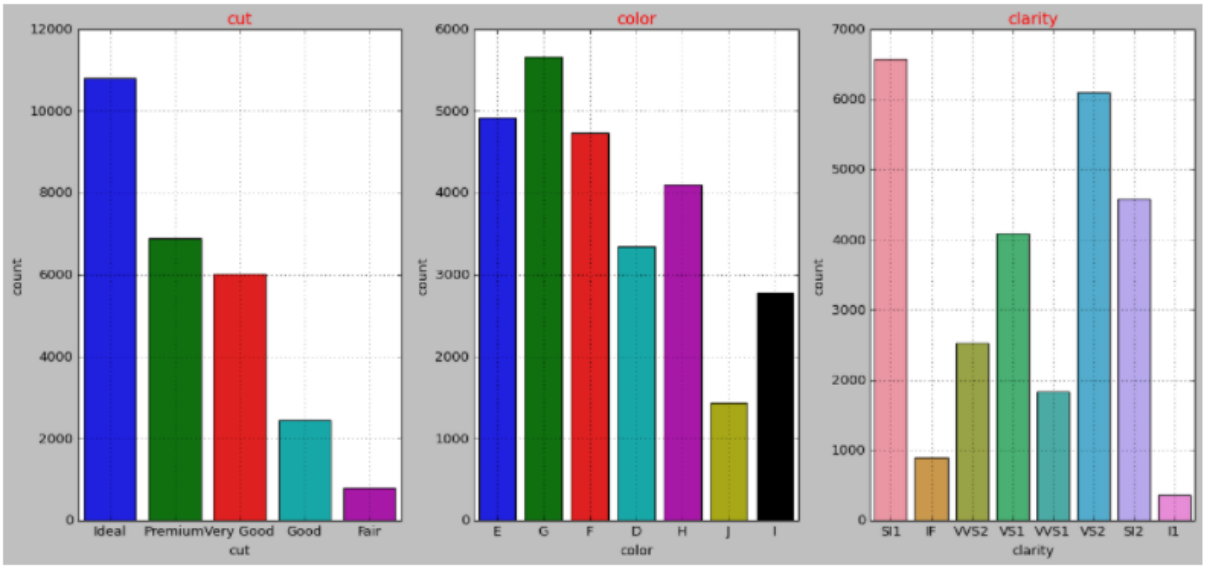


FIGURE 5 COUNTPLOT FOR DATASET

Boxplot is drawn to get the presence of outliers in the variables. “Carat”, “Depth” and “tables” are major outliers present variables.

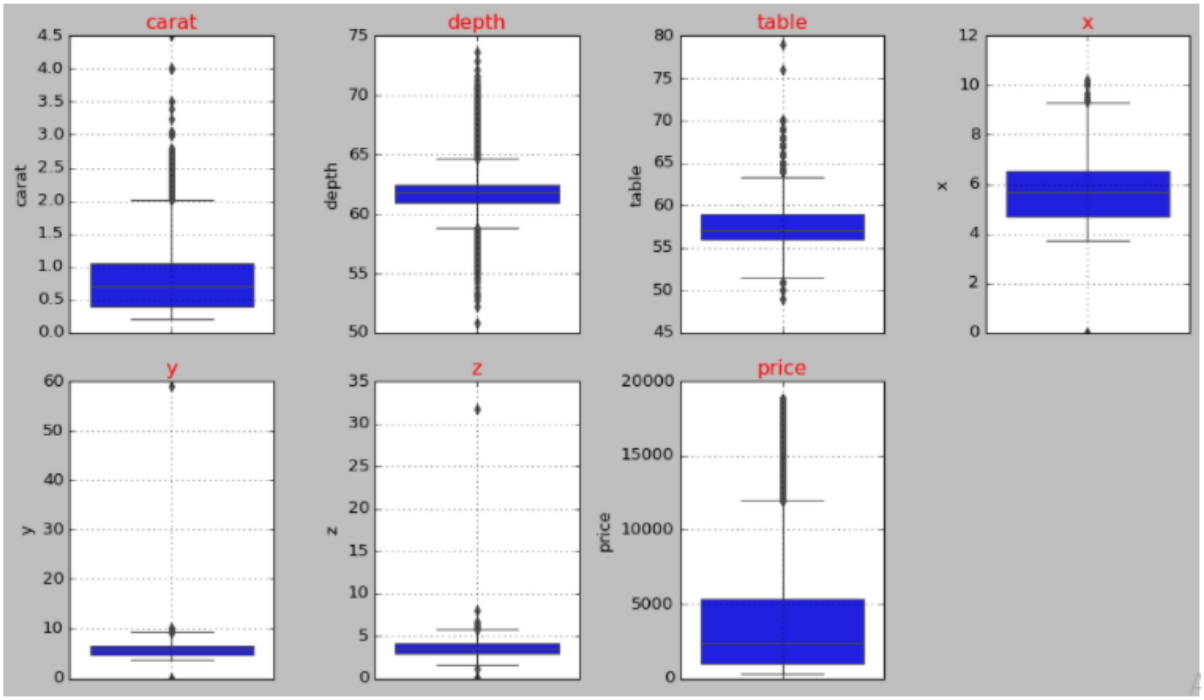


FIGURE 6 BOXPLOT FOR DATASET

Similarly, Displot is also plotted for the variables. As per plot, Depth and table are normally distributed. Other variables dist plots are bit skewed.

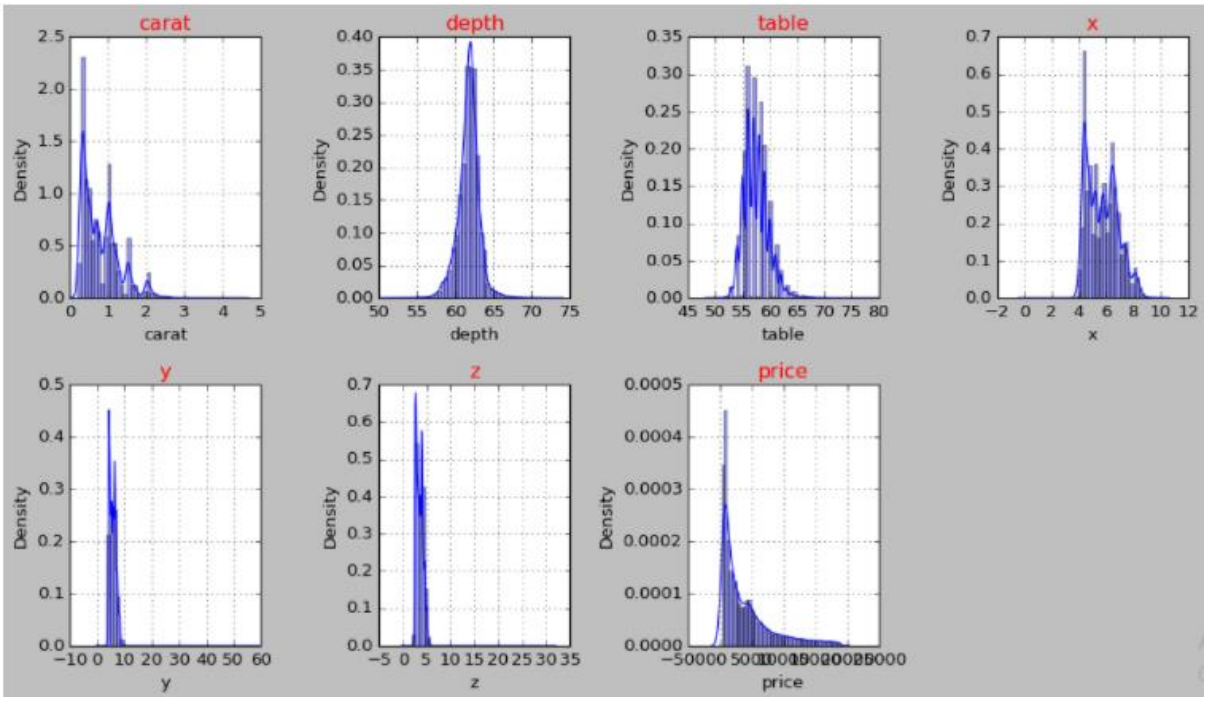


FIGURE 7 DISTPLOT FOR DATASET

Pair plot and correlation matrix is also plotted for the variables. As per plot, X,Y and Z are highly correlated with Carat variable.

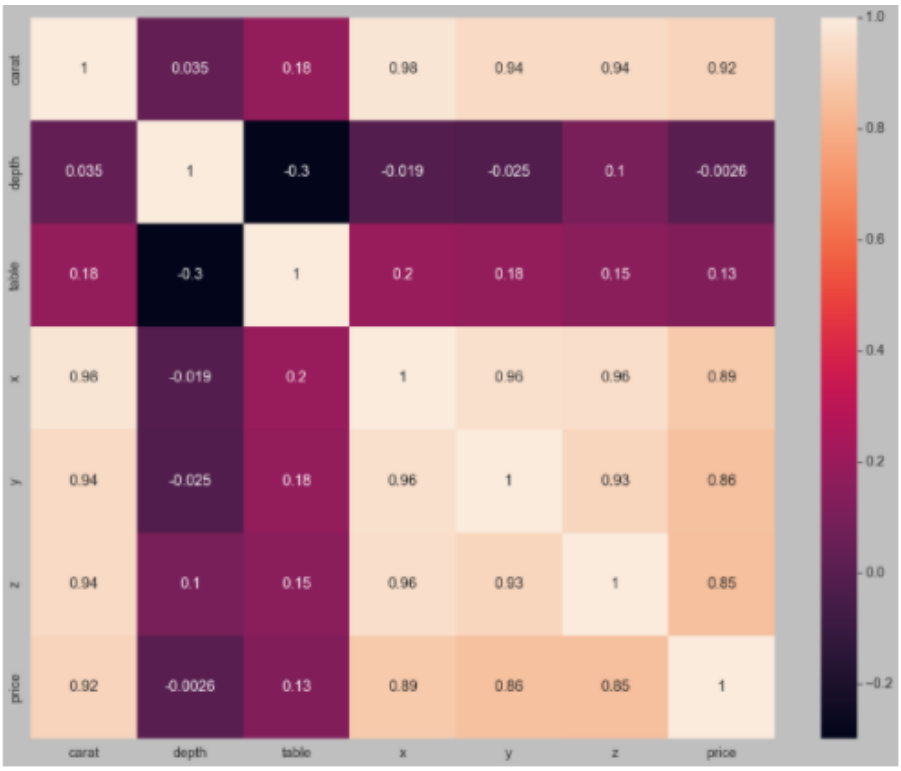


FIGURE 8 CORRELATION MATRIX OF DATASET

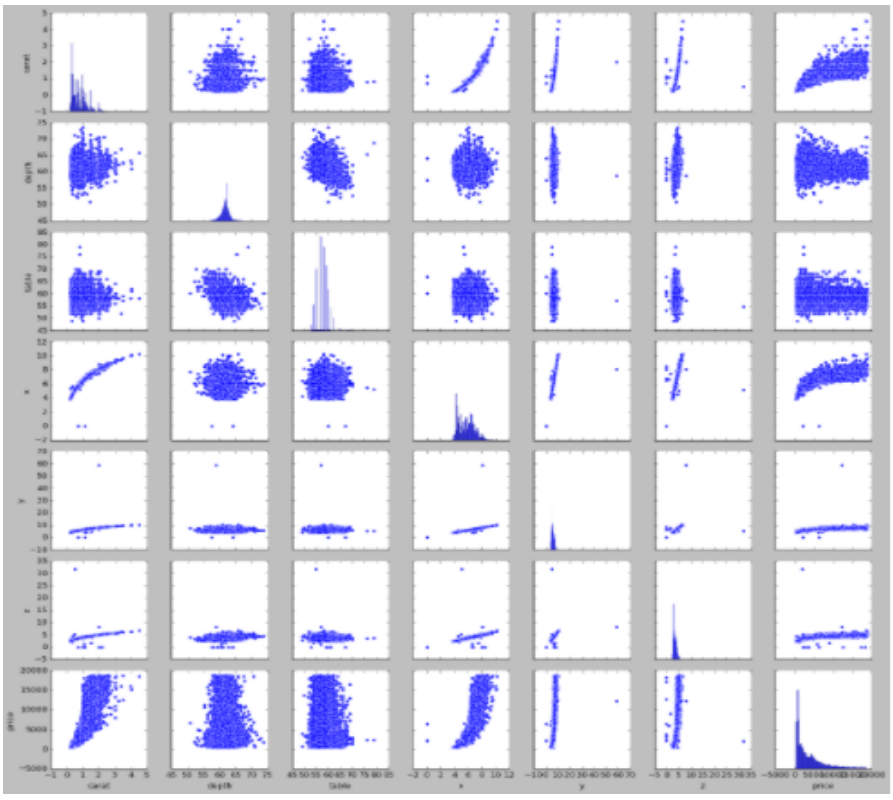


FIGURE 9 PAIRPLOT FOR DATASET

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

697 Null values are identified in the dataset for the "depth" variable. We have use the KNN imputer to rectified this null values. Complete process has coded in the code file. We have seen it percentage null value presence is less than 5%, we can also be dropped from the analysis. Scaling is not necessary for further analysis and variables without scaling does not affect the model performance in linear regression.

1.3. **Encode the data (having string values) for Modelling. Split the data into train and test (70:30). Apply Linear regression using scikit learn.**

```
feature: cut
['Ideal', 'Premium', 'Very Good', 'Good', 'Fair']
Categories (5, object): ['Fair', 'Good', 'Ideal', 'Premium', 'Very Good']
[2 3 4 1 0]


feature: color
['E', 'G', 'F', 'D', 'H', 'J', 'I']
Categories (7, object): ['D', 'E', 'F', 'G', 'H', 'I', 'J']
[1 3 2 0 4 6 5]


feature: clarity
['SI1', 'IF', 'VVS2', 'VS1', 'VVS1', 'VS2', 'SI2', 'I1']
Categories (8, object): ['I1', 'IF', 'SI1', 'SI2', 'VS1', 'VS2', 'VVS1', 'VVS2']
[2 1 7 4 6 5 3 0]
```

FIGURE 10 ENCODING FOR OBJECT VARIABLES

Dataset split into test and train and further Linear model development is represented in below image.

```
# Copy all the predictor variables into X dataframe.
X = df.drop('price', axis=1)
y = df[['price']]
```

```
#Let us break the X and y dataframes into training set and test set. For this we will use
#Sklearn package's data splitting function which is based on random function

from sklearn.model_selection import train_test_split
```

```
# Split X and y into training and test set in 70:30 ratio

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30 , random_state=1)
```

```
# invoke the LinearRegression function and find the bestfit model on training data

regression_model = LinearRegression()
regression_model.fit(X_train, y_train)
```

```
LinearRegression()
```

FIGURE 11 TRAIN AND TEST SPLIT OF DATASET

```
The coefficient for carat is 11334.871925292742
The coefficient for cut is 61.13609057606334
The coefficient for color is -282.3279558988111
The coefficient for clarity is 290.80439964119887
The coefficient for depth is -153.6153582649673
The coefficient for table is -93.0158461315609
The coefficient for x is -1257.7208337611219
The coefficient for y is 4.417230255229015
The coefficient for z is -30.79709440939132
```

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

```
0.8866463978733947
```

```
# Model score - R2 or coeff of determinant
# R^2=1-RSS / TSS =  RegErr / TSS

regression_model.score(X_test, y_test)
```

```
0.8891401672088223
```

FIGURE 12 COEFFICENT RESULT FOR THE VARIABLES AND MODEL SCORE

From the stats model, we have performed the analysis and done the model development. R Square and Adj R square values are 0.887 in both cases. Other details such as Intercept and coefficient value are available in below detailed analysis from statsmodel.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.887
Model:                            OLS   Adj. R-squared:                  0.887
Method:                 Least Squares   F-statistic:                 1.638e+04
Date:                Sun, 27 Mar 2022   Prob (F-statistic):               0.00
Time:                        08:38:18   Log-Likelihood:            -1.6264e+05
No. Observations:               18853   AIC:                         3.253e+05
Df Residuals:                   18843   BIC:                         3.254e+05
Df Model:                           9
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept   1.648e+04    685.499     24.037      0.000    1.51e+04    1.78e+04
carat       1.133e+04    101.669    111.488      0.000    1.11e+04    1.15e+04
cut           61.1361      9.855      6.204      0.000      41.819      80.453
color       -282.3280      6.057    -46.611      0.000    -294.200    -270.456
clarity      290.8044      5.880     49.460      0.000     279.280     302.329
depth       -153.6154      8.213    -18.703      0.000    -169.714    -137.517
table        -93.0158      4.736    -19.642      0.000    -102.298     -83.734
x          -1257.7208     55.046    -22.848      0.000   -1365.617   -1149.825
y              4.4172     26.461      0.167      0.867     -47.448      56.282
z            -30.7971     44.070     -0.699      0.485    -117.178      55.584
==============================================================================
Omnibus:                     5052.816   Durbin-Watson:                   1.979
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           137214.249
Skew:                           0.698   Prob(JB):                         0.00
Kurtosis:                      16.143   Cond. No.                     5.94e+03
==============================================================================
```

FIGURE 13 RESULTS BASED ON STATSMODEL

```
# Model score - R2 or coeff of determinant
# R^2=1-RSS / TSS

regression_model.score(X_test, y_test)
```

```
0.8891401672088223
```

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

```
0.8866463978733947
```

FIGURE 14 MODEL SCORE BASED ON STATSMODEL

```
from statsmodels.stats.outliers_influence import variance_inflation_factor
```

```
vif = [variance_inflation_factor(X.values, ix) for ix in range(X.shape[1])]
```

```
i=0
for column in X.columns:
    if i < 11:
        print (column ,"--->",  vif[i])
        i = i+1
```

```
carat ---> 74.83266469964308
cut ---> 7.46395541152295
color ---> 3.6706974214198342
clarity ---> 6.20452772751817
depth ---> 513.0299528822784
table ---> 514.2263691304604
x ---> 1030.7428796303461
y ---> 347.56650628061817
z ---> 330.75765583065527
```

FIGURE 15 IVF RESULT BASED ON DATASET MODEL

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

Based on the analysis, Carat is main variable for the deciding price of diamond. Both are directly related. Carat measures on x,y,z values which is also make these variables essentials. Diamond price is also affected based on cut and clarity. Clarity VVS1, VVS2, VS1, VS2 and colour E, F, G also have shown positive result on diamond's price. It is suggested that cut with, Fair and Good can be avoid. Similarly, IF and L1 clarity is not advisable for the business. Based on these parameter, high price diamond category can be selected and reduce the lower price for better market strategy.

Problem 2: Logistic Regression and LDA

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

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.

This dataset has 872 rows with 7 variables. Out of 7, two variables contain object data type and rest five variables contain int datatype. In the below figure, we have shown the dataset of holiday.

	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

FIGURE 16 HOLIDAY PACKAGE 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
```

FIGURE 17 DATATYPE INFORMATION

This dataset does not contain null value. Similarly, dataset does not have any duplicate rows.

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

FIGURE 18 NULL VALUE ANALYSIS

Brief summary of dataset has been described below. Dataset contains 6 independent variables and one dependent variables. Here, Holiday package is dependent variable.

	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

FIGURE 19 DATASET DESCRIPTION

Dataset contains outliers which shown in below boxplot. Salary variable shows maximum outlier.

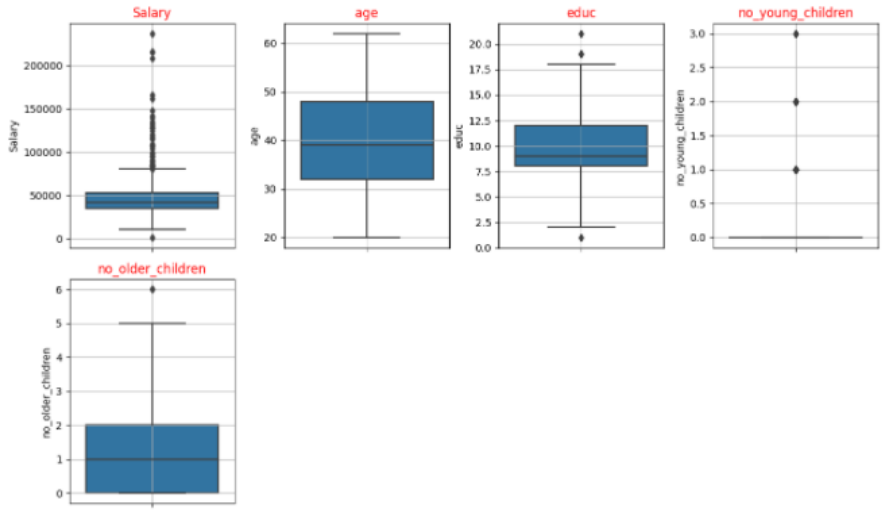
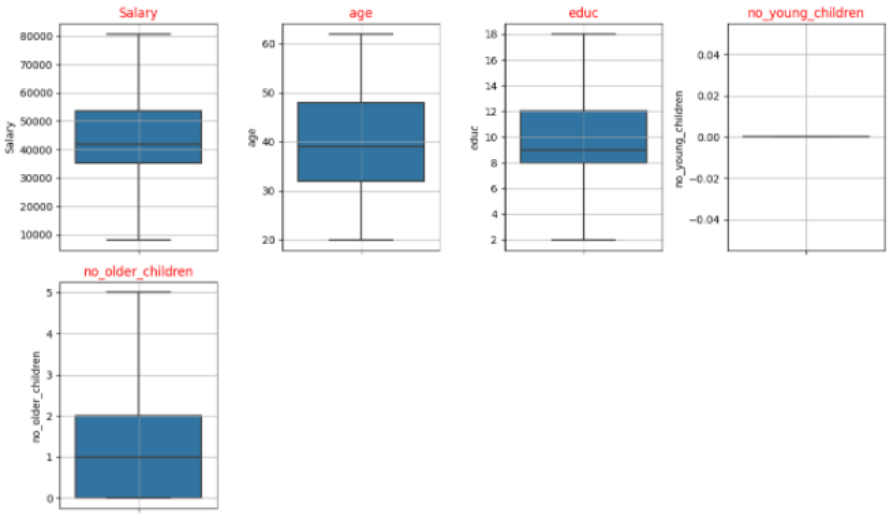
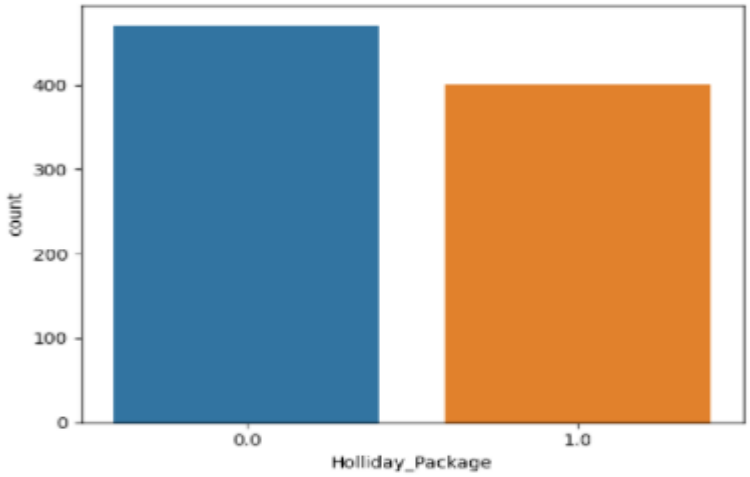


FIGURE 20 BOXPLOT FOR OUTLIER IDENTIFICATION

We have processed the outlier treatment on the dataset. In a below figure, we can see processed dataset without outliers.



Various count plot are drawn based on Holiday Package as a Hue. From plots, we inferred that count values of holiday package are almost same. In a similar view, count plot of no of young children and foreign with holiday packages as hue shows similar result.



Count plot of educ and no of older children have been plotted and shown in below image.

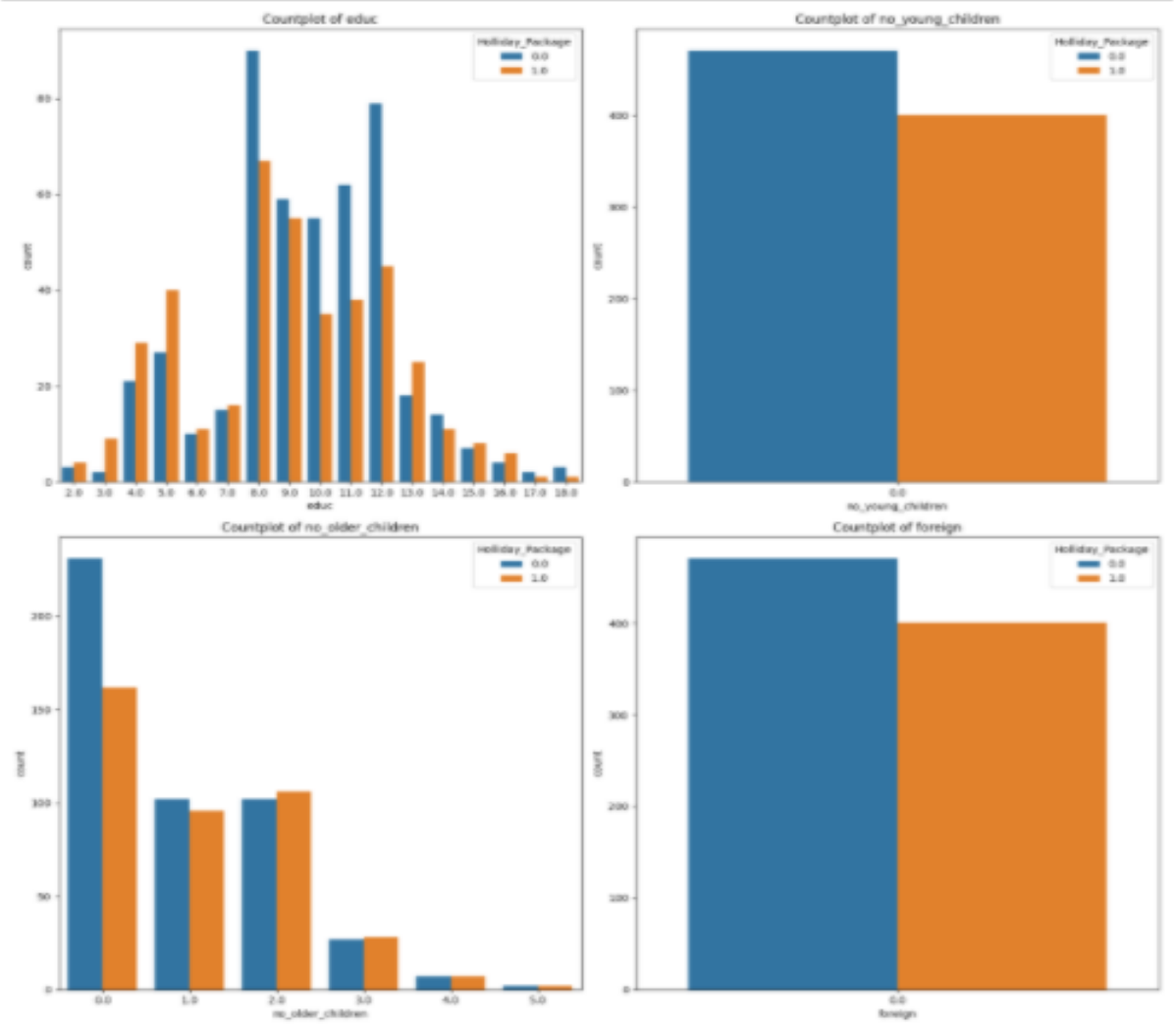


FIGURE 21 COUNTPLOT FOR DATASET

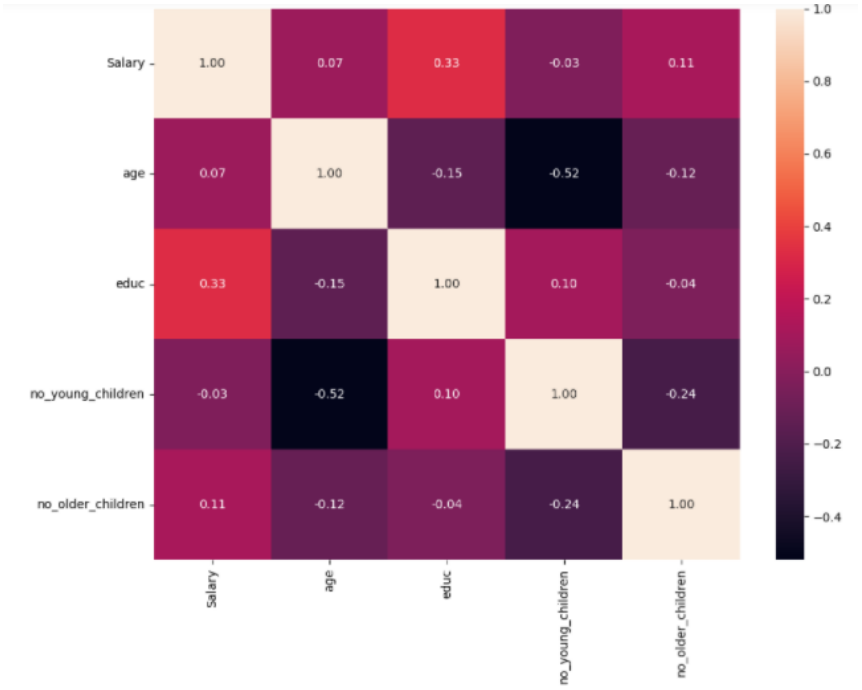


FIGURE 22 CORRELATION MATRIX

Correlation plot has been drawn to get the maximum correlated variables. As per the dataset, no_young_children are highly negatively corelated to age. In a below images, we can see the pair plot for the dataset.

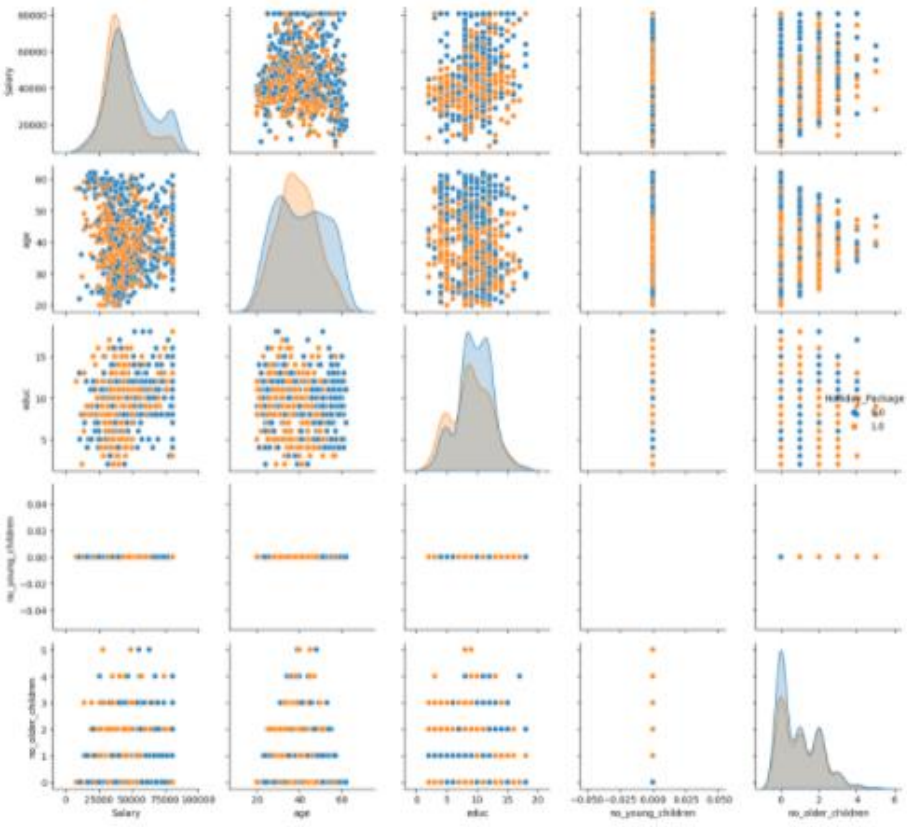


FIGURE 23 PAIRPLOT FOR DATASET

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 of the data has been done. The process is explained in the code file.

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


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

FIGURE 24 ENCODING FOR OBJECT VARIABLES

Data is splitted into train and test set with 70:30 portion. After split, we have processed for the logistic and LDR model development.

Logistic Regression:

```
X = df.drop('Holliday_Package',axis=1)
Y = df.Holliday_Package
```

```
test_size = 0.30
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=test_size)
```

Formulate a logistic regression model.

```
model = LogisticRegression()
model.fit(X_train, y_train)
ytrain_predict = model.predict(X_train)
y_predict = model.predict(X_test)
```

FIGURE 25 DATASET SPLIT AND LOGISTIC MODEL

Linear Discriminant Analysis:

```
clf = LinearDiscriminantAnalysis()
model=clf.fit(X_train, y_train)
model
```

```
LinearDiscriminantAnalysis()
```

```
ytrain_predict = model.predict(X_train)
y_predict = model.predict(X_test)
```

FIGURE 26 LDR MODEL FOR DATA

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.

Logistic Regression

Train Data: Accuracy, Confusion Matrix, Plot ROC curve and ROC_AUC score have been done on train data for logistic regression. Accuracy and AUC score are 53% and 0.580 respectively. Rest confusion matrix and ROC plot has been drawn below.

AUC: 0.580

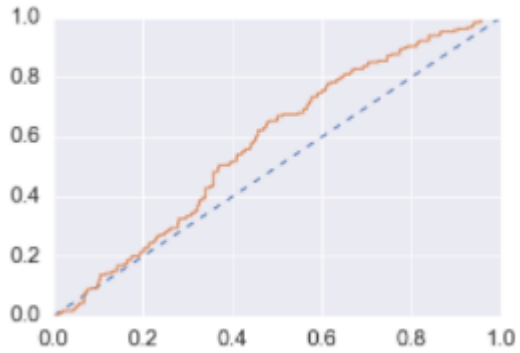


FIGURE 27 AUC SCORE AND CURVE

Confusion Matrix
[[256 67]
[222 65]]

Classification Report

	precision	recall	f1-score	support
0	0.54	0.79	0.64	323
1	0.49	0.23	0.31	287
accuracy			0.53	610
macro avg	0.51	0.51	0.47	610
weighted avg	0.52	0.53	0.48	610

FIGURE 28 CONFUSION MARTIX AND CLASSIFICATION REPORT FOR DATASET

Test Data: Accuracy, Confusion Matrix, Plot ROC curve and ROC_AUC score have been done on test data for logistic regression. Accuracy and AUC score are 52% and 0.575 respectively. Rest confusion matrix and ROC plot has been drawn below.

AUC: 0.575

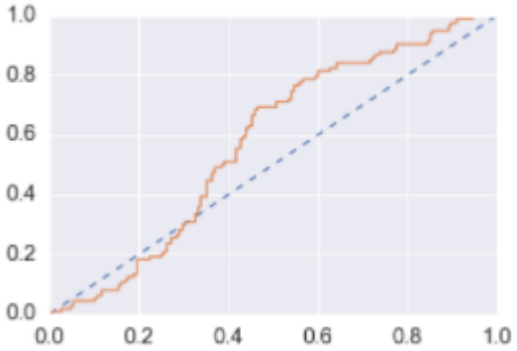


FIGURE 29 AUC SCORE AND CURVE

```
Confusion Matrix
 [[104  44]
 [ 82  32]]

Classification Report
              precision    recall  f1-score   support

           0       0.56      0.70      0.62       148
           1       0.42      0.28      0.34       114

    accuracy                           0.52       262
   macro avg       0.49      0.49      0.48       262
weighted avg       0.50      0.52      0.50       262
```

FIGURE 30 CONFUSION MARTIX AND CLASSIFICATION REPORT FOR DATASET

Linear Discrimant Regression

Train Data: Accuracy, Confusion Matrix, Plot ROC curve and ROC_AUC score have been done on train data for LDR. Accuracy and AUC score are 67% and 0.740 respectively. Rest confusion matrix and ROC plot has been drawn below.

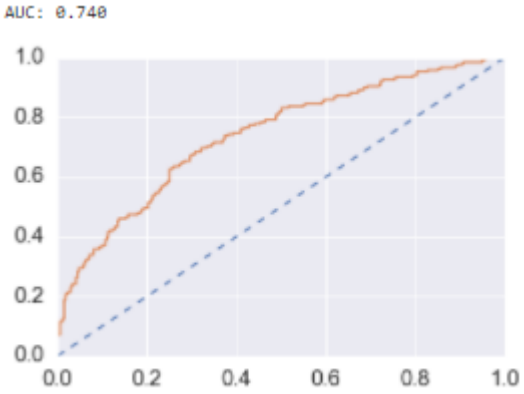


FIGURE 31 AUC SCORE AND CURVE

```
Confusion Matrix
 [[250  73]
 [128 159]]

Classification Report
              precision    recall  f1-score   support

           0       0.66      0.77      0.71       323
           1       0.69      0.55      0.61       287

    accuracy                           0.67       610
   macro avg       0.67      0.66      0.66       610
weighted avg       0.67      0.67      0.67       610
```

FIGURE 32 CONFUSION MARTIX AND CLASSIFICATION REPORT FOR DATASET

Test Data: Accuracy, Confusion Matrix, Plot ROC curve and ROC_AUC score have been done on test data for LDR. Accuracy and AUC score are 66% and 0.722 respectively. Rest confusion matrix and ROC plot has been drawn below.

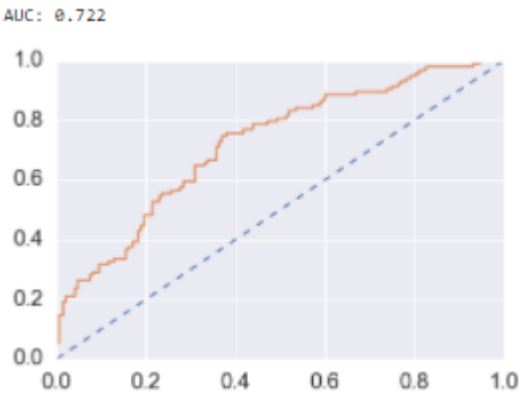


FIGURE 33 AUC SCORE AND CURVE

Confusion Matrix
[[107 41]
[48 66]]

Classification Report

	precision	recall	f1-score	support
0	0.69	0.72	0.71	148
1	0.62	0.58	0.60	114
accuracy			0.66	262
macro avg	0.65	0.65	0.65	262
weighted avg	0.66	0.66	0.66	262

FIGURE 34 CONFUSION MARTIX AND CLASSIFICATION REPORT FOR DATASET

Based on both regression result, LDR has better accuracy and AUC score compare to Logistic regression.

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

Both model (logistic and LDR) is created based on dataset. Some insights can be drawn from above models. On test dataset, precision values are 0.42 for logistic and 0.62 for LDR. LDR performs better compare to Logistic Regression.

On a recommendation view, Outliers needs to be treated before model development. Many of high earning employee does not purchase a package, company needs to permote the products.

Foreigner with no children could be more interested into purchasing a package, company should look for such options. People with older children can also be targeted for the packages with extra discount.

No_young_children are highly negatively corelated to age in respect to holiday package, we should keep in mind while permoting the products.