11/7/2021

Predictive Modeling Report
PGP -DSBA

Saloni Juwatkar
PGP – DATA SCIENCE AND BUSINESS ANALYTICS

Table of Contents

List of Figures

List of Tables

1 Problem Statement: 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.

Data Description

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 Worst to Best) 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, duplicate values). Perform Univariate and Bivariate Analysis.

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

Note: Both the above questions are answered together as exploratory data analysis is done in a sequence where null values and values with 0 in the columns are imputed before doing the Univariate and Bivariate Analysis.

The csv file was read and EDA was done and the following were the inferences drawn from the EDA.

Exploratory Data Analysis

- The dataset consists of 11 variables – **'Unnamed: 0, carat, cut, color, clarity, depth, table, x, y, z, price'**.

	Unnamed: 0	carat	cut	color	clarity	depth	table	x	y	z	price
0	1	0.30	Ideal	E	SI1	62.1	58.0	4.27	4.29	2.66	499
1	2	0.33	Premium	G	IF	60.8	58.0	4.42	4.46	2.70	984
2	3	0.90	Very Good	E	VVS2	62.2	60.0	6.04	6.12	3.78	6289
3	4	0.42	Ideal	F	VS1	61.6	56.0	4.82	4.80	2.96	1082
4	5	0.31	Ideal	F	VVS1	60.4	59.0	4.35	4.43	2.65	779

Figure 1: PS:1 Sample Dataset

	Unnamed: 0	carat	cut	color	clarity	depth	table	x	y	z	price
26962	26963	1.11	Premium	G	SI1	62.3	58.0	6.61	6.52	4.09	5408
26963	26964	0.33	Ideal	H	IF	61.9	55.0	4.44	4.42	2.74	1114
26964	26965	0.51	Premium	E	VS2	61.7	58.0	5.12	5.15	3.17	1656
26965	26966	0.27	Very Good	F	VVS2	61.8	56.0	4.19	4.20	2.60	682
26966	26967	1.25	Premium	J	SI1	62.0	58.0	6.90	6.88	4.27	5166

Figure 2: PS:1: Sample Tail Dataset

```
cz_df.shape
```

```
(26967, 11)
```

Figure 3: PS:1: Shape of Dataset

- The shape of the data is **(26967, 10).**

	count	mean	std	min	25%	50%	75%	max
Unnamed: 0	26967.0	13484.000000	7784.846691	1.0	6742.50	13484.00	20225.50	26967.00
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

Figure 4: PS:1: Data Description

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

Figure 5: PS:1: Data Info

```
CUT : 5
Fair          779
Good         2434
Very Good    6027
Premium      6880
Ideal       10805
Name: cut, dtype: int64


COLOR : 7
J    1440
I    2765
D    3341
H    4091
F    4722
E    4916
G    5650
Name: color, dtype: int64


CLARITY : 8
I1       362
IF       891
VVS1    1839
VVS2    2530
VS1     4086
SI2     4561
VS2     6092
SI1     6564
Name: clarity, dtype: int64
```

Figure 6: PS:1: Unique Data

- The data contains float, int and object datatypes.
- The variable 'Unnamed: 0' is not needed for exploratory data analysis or any further predictions. Hence, we choose to drop the column.
- There are three categorical variables **'cut, color and clarity'**. Cut is having a total of 5 unique values, color is having a total of 7 unique value and clarity is having a unique value of 8.
- **'Carat, depth, table, x, y, z, price'** are continuous variables.
- Price will be the target variable considered while building the Linear Regression model.

Duplicate Data Imputation:

- Number of duplicate rows found in the dataset were 34. These are dropped so as to getter a better prediction and can draw useful insights from the model.

```
Before (26967, 10)
After (26933, 10)
```

Figure 7: PS:1: Shape of Data before and after duplicate Data imputation

Missing/ Null Value Treatment: (Refer 1.2)

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

Figure 8: PS:1: Missing Data

- Here it is observed that there are null/Nan values in the depth column of the dataset.
- The values can be imputed using mean or median.
- Here mean is used to impute the null values in the dataset.

Zero Value Treatment (Refer 1.2)

	carat	cut	color	clarity	depth	table	x	y	z	price
5821	0.71	Good	F	SI2	64.1	60.0	0.00	0.00	0.0	2130
6034	2.02	Premium	H	VS2	62.7	53.0	8.02	7.95	0.0	18207
10827	2.20	Premium	H	SI1	61.2	59.0	8.42	8.37	0.0	17265
12498	2.18	Premium	H	SI2	59.4	61.0	8.49	8.45	0.0	12631
12689	1.10	Premium	G	SI2	63.0	59.0	6.50	6.47	0.0	3696
17506	1.14	Fair	G	VS1	57.5	67.0	0.00	0.00	0.0	6381
18194	1.01	Premium	H	I1	58.1	59.0	6.66	6.60	0.0	3167
23758	1.12	Premium	G	I1	60.4	59.0	6.71	6.67	0.0	2383

Figure 9: PS:1: Before Zero Value Treatment

- Here it is observed that x, y and z columns have 0 values in it.
- You can choose to drop these columns as it seems to be data entry issue and length cannot be 0.
- Hence, we drop these 8 records.

	count	mean	std	min	25%	50%	75%	max
carat	26925.0	0.797821	0.477085	0.20	0.40	0.70	1.05	4.50
depth	26925.0	61.745566	1.393430	50.80	61.10	61.80	62.50	73.60
table	26925.0	57.455305	2.231327	49.00	56.00	57.00	59.00	79.00
x	26925.0	5.729385	1.126081	3.73	4.71	5.69	6.55	10.23
y	26925.0	5.733152	1.163820	3.71	4.71	5.70	6.54	58.90
z	26925.0	3.538820	0.717483	1.07	2.90	3.52	4.04	31.80
price	26925.0	3936.249991	4020.983187	326.00	945.00	2373.00	5353.00	18818.00

Figure 10: PS:1: After Zero value treatment

- The new dataset has now shape as: (26925, 10) after duplicate, null values and zero value treatment.
- This dataset now can be using to do Univariate and Bivariate analysis.

Scaling (Refer 1.2)

- By looking at the data, we can see that the data is at different scaling. Here, scaling can be done but it would not have any effect on the regression models and their insights.
- Here it is also observed that the std deviation is not much.
- Hence, we choose to not to do scaling on the dataset as it won't have significant impact on any models.

Univariate Analysis

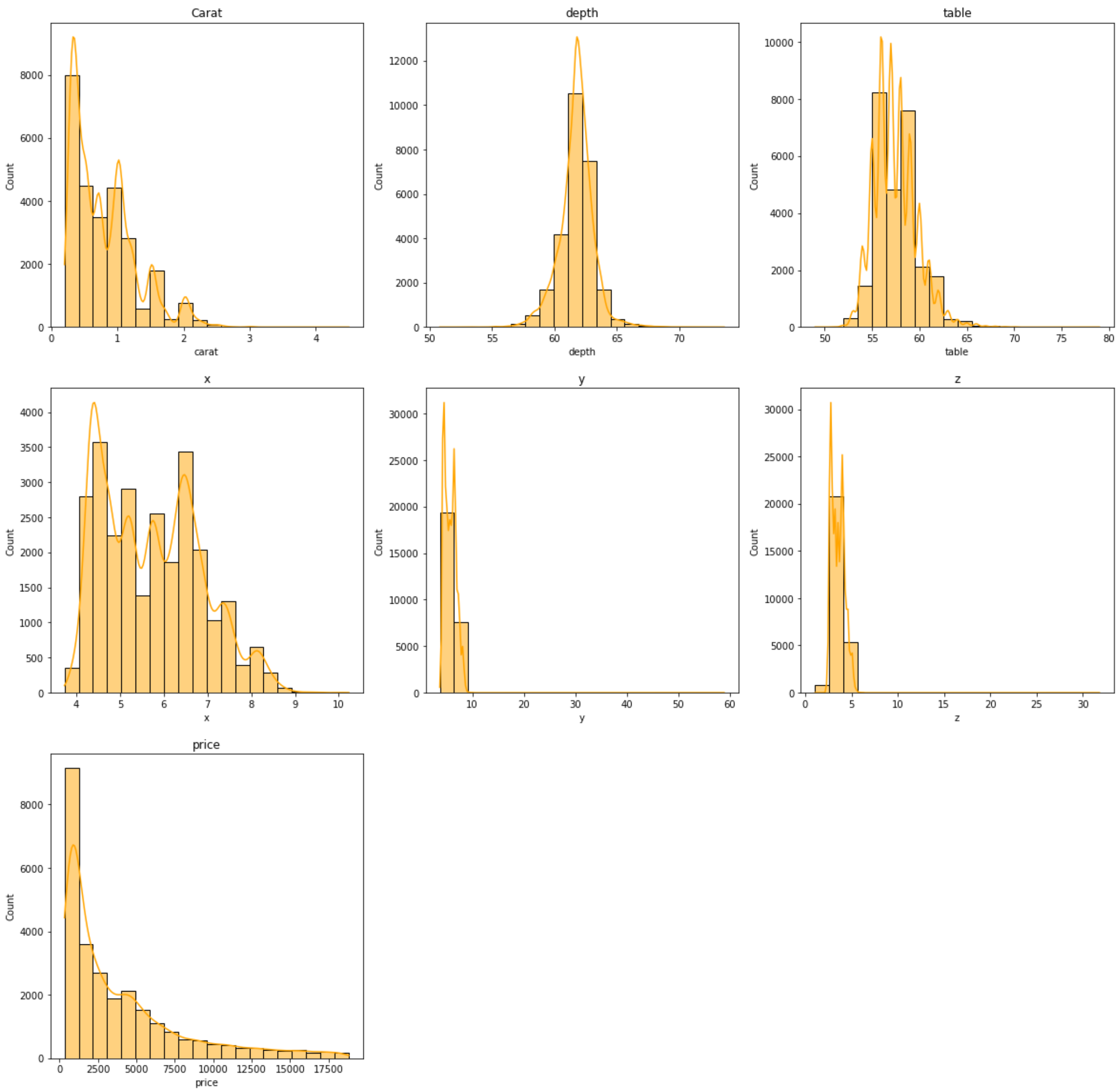

Figure 11: PS:1: Univariate Analysis

Inference:

- The plot shows that the Carat weight distribution of the cubic zirconia and it is positively skewed.
- Approximately 75% of the cubic zirconia stones have weight between 0.20 and 1.05 carats.
- Depth shows the height of a cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter. From the plot we can see that the data is almost normal distribution
- The depth of majority of cubic zirconia ranges between 60 and 62mm.

- The Width of the cubic zirconia's Table is expressed as a Percentage of its Average Diameter. The plot shows that the data is positively skewed
- Majority of the stones have Table value between 56 and 60.
- The plot shows the length of the cubic zirconia in mm. The distribution plot shows us that the data is positively skewed
- The average length of majority of zirconia stones lies between 4-7mm.
- The plot shows the Width of the cubic zirconia in mm. The distribution plot shows that the data is positively skewed
- The width of almost 75% of the stones ranges between 3-10mm with maximum value of 58mm.
- The plot shows the distribution of Height of the cubic zirconia in mm. The distribution of the data is positively skewed
- The average height ranges between 3-6mm.
- The above plot shows the price of the cubic zirconia. From the plot we can see that the data is positively skewed
- The price being our target variable displays a right skewed graph with approximately 75% of the stones costing within the range of 945 to 5360 with the remaining percentage to be the premium stones costing more than 10,000.

Boxplot For Outliers Treatment

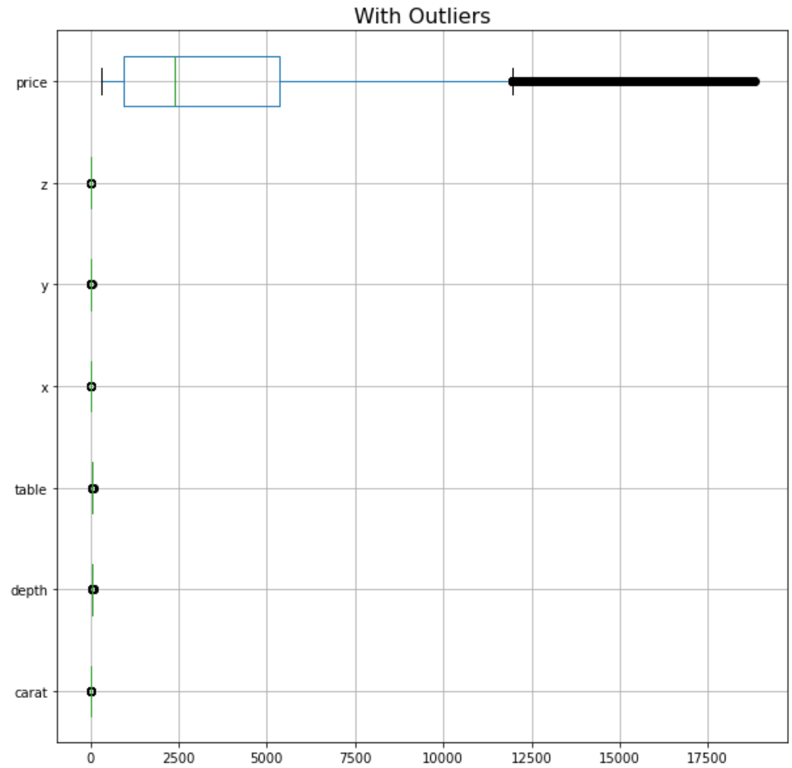

Figure 12: PS:1: Boxplot Before Outlier Treatment

The Boxplot shows a lot of outliers in the dataset which need to be treated.

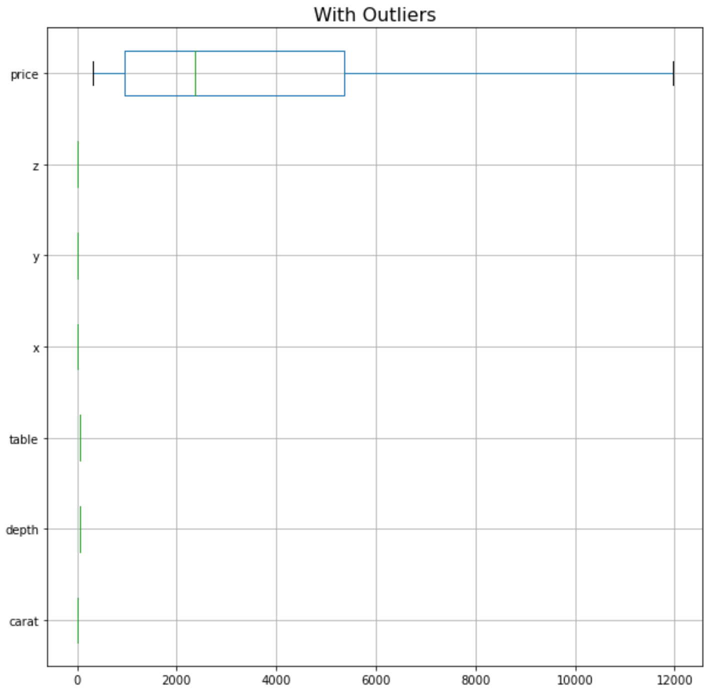

Figure 13: PS:1: Boxplot after Outlier Treatment

The outliers are treated with percentile method with which the dataset if ready to be used for building regression model.

Bivariate Analysis

The Heatmap and Pair plots shows the correlation between each variable.

- We can see that there is multicollinearity present in the data.
- The variables Carat with variables X, Y, Z and price are strongly correlated with each other.
- Similarly, x is strongly correlated with y and z.

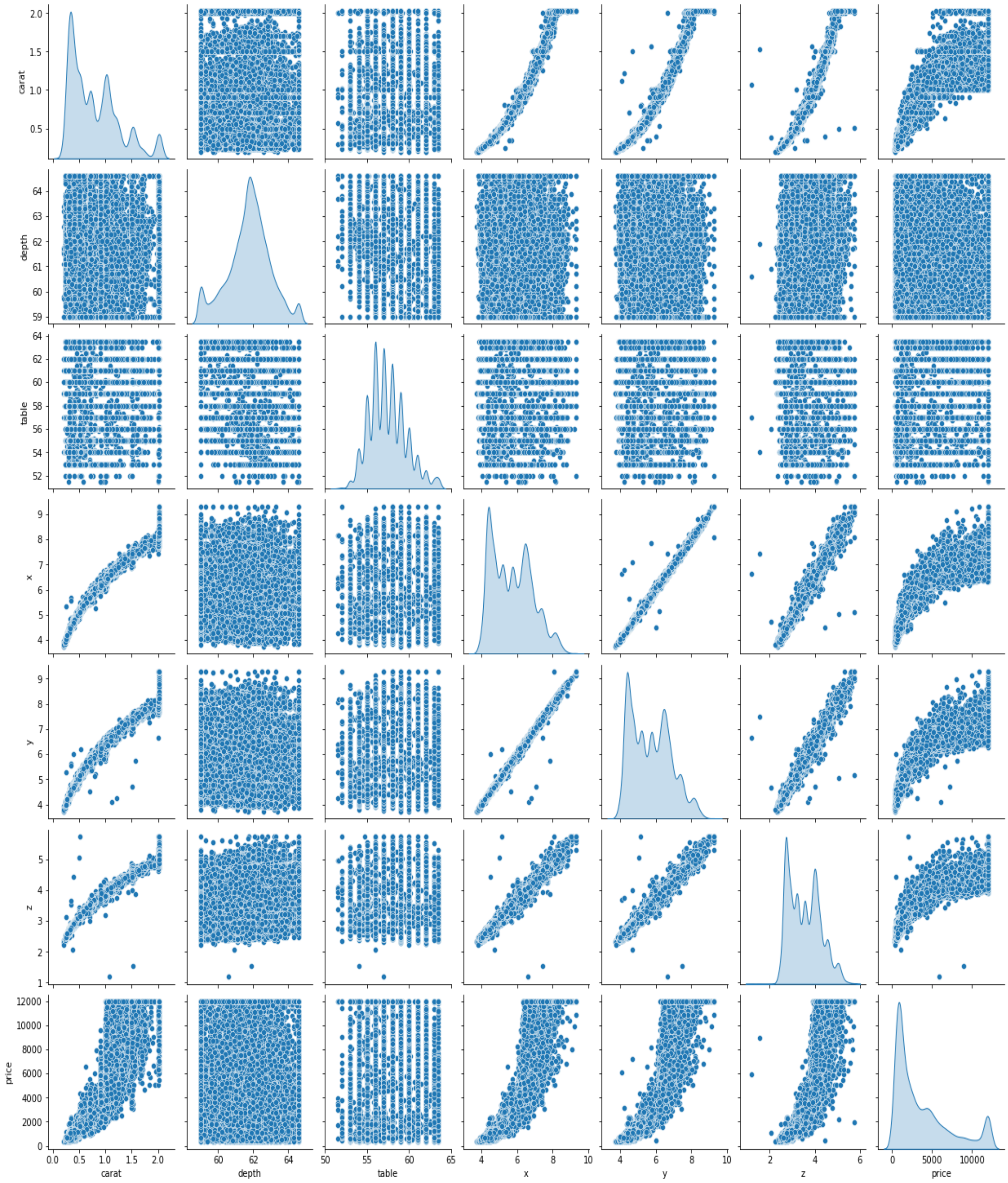

Figure 14: PS:1: Pair plot

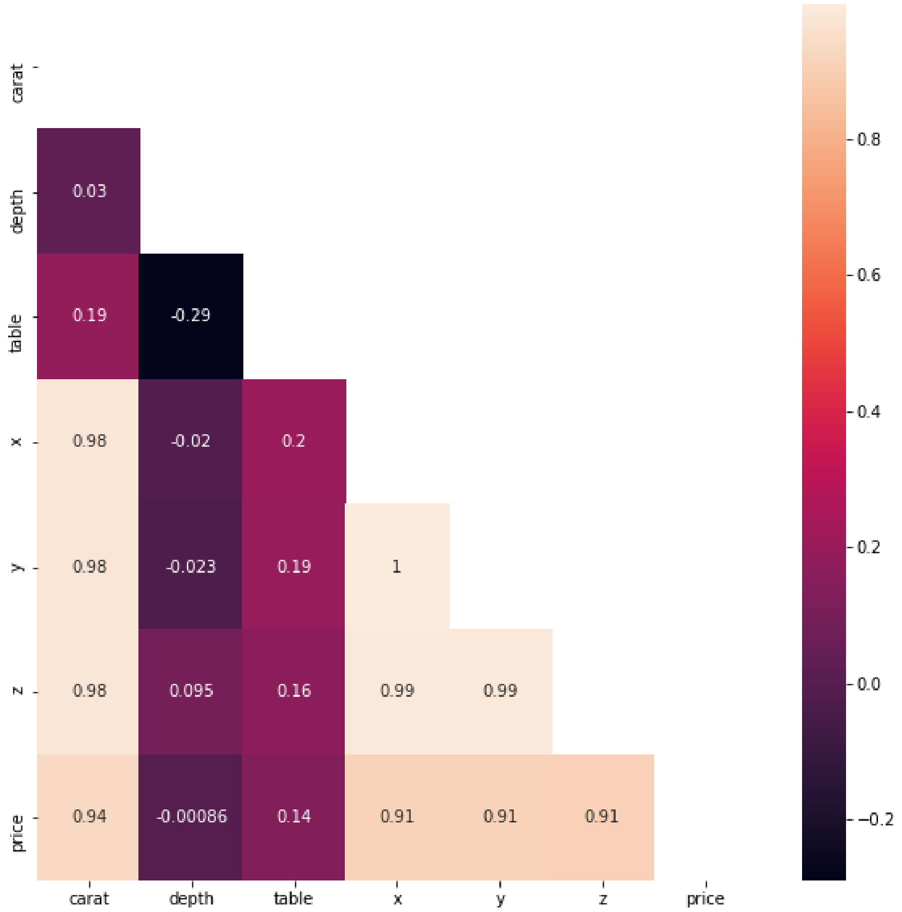

Figure 15: PS:1: Heatmap

1.3 Encode the data (having string values) for Modelling. Split the data into train and test (70:30). Apply Linear regression using scikit learn. Perform checks for significant variables using appropriate method from statsmodel. Create multiple models and check the performance of Predictions on Train and Test sets using Rsquare, RMSE & Adj Rsquare. Compare these models and select the best one with appropriate reasoning.

Encoding String Values

- We use get_dummies () function to encode the string values for modelling, i.e., converting the categorical variables to dummy or indicator variables.
- After converting the variables, the data looks as below:

```
cz_en= pd.get_dummies(cz_df, columns=['cut','color','clarity'],drop_first=True)
```

Figure 16: PS:1: Encoding

	carat	depth	table	x	y	z	price	cut_Good	cut_Ideal	cut_Premium	...	color_H	color_I	color_J	clarity_IF	clarity_SI1	clarity_SI2	clarity_
0	0.30	62.1	58.0	4.27	4.29	2.66	499.0	0	1	0	...	0	0	0	0	1	0	
1	0.33	60.8	58.0	4.42	4.46	2.70	984.0	0	0	1	...	0	0	0	1	0	0	
2	0.90	62.2	60.0	6.04	6.12	3.78	6289.0	0	0	0	...	0	0	0	0	0	0	
3	0.42	61.6	56.0	4.82	4.80	2.96	1082.0	0	1	0	...	0	0	0	0	0	0	
4	0.31	60.4	59.0	4.35	4.43	2.65	779.0	0	1	0	...	0	0	0	0	0	0	

5 rows × 24 columns

Figure 17: PS:1: Data Encoded Dataframe

After encoding the dataset is as below.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 26925 entries, 0 to 26966
Data columns (total 24 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   carat           26925 non-null  float64
 1   depth           26925 non-null  float64
 2   table           26925 non-null  float64
 3   x               26925 non-null  float64
 4   y               26925 non-null  float64
 5   z               26925 non-null  float64
 6   price           26925 non-null  float64
 7   cut_Good        26925 non-null  uint8
 8   cut_Ideal       26925 non-null  uint8
 9   cut_Premium     26925 non-null  uint8
 10  cut_Very Good   26925 non-null  uint8
 11  color_E         26925 non-null  uint8
 12  color_F         26925 non-null  uint8
 13  color_G         26925 non-null  uint8
 14  color_H         26925 non-null  uint8
 15  color_I         26925 non-null  uint8
 16  color_J         26925 non-null  uint8
 17  clarity_IF      26925 non-null  uint8
 18  clarity_SI1     26925 non-null  uint8
 19  clarity_SI2     26925 non-null  uint8
 20  clarity_VS1     26925 non-null  uint8
 21  clarity_VS2     26925 non-null  uint8
 22  clarity_VVS1    26925 non-null  uint8
 23  clarity_VVS2    26925 non-null  uint8
dtypes: float64(7), uint8(17)
memory usage: 2.1 MB
```

Figure 18: PS:1: Data Info

Test and Train Split

- We split the train and test data as 70% and 30%. We copy all the predictor variable i.e., Price in to X data frame and copy the target into y data frame.
- X frame looks like below.

	carat	depth	table	x	y	z	cut_Good	cut_Ideal	cut_Premium	cut_Very Good	...	color_H	color_I	color_J	clarity_IF	clarity_SI1	clarity_SI2	clarit
0	0.30	62.1	58.0	4.27	4.29	2.66	0	1	0	0	...	0	0	0	0	1	0	
1	0.33	60.8	58.0	4.42	4.46	2.70	0	0	1	0	...	0	0	0	1	0	0	
2	0.90	62.2	60.0	6.04	6.12	3.78	0	0	0	1	...	0	0	0	0	0	0	
3	0.42	61.6	56.0	4.82	4.80	2.96	0	1	0	0	...	0	0	0	0	0	0	
4	0.31	60.4	59.0	4.35	4.43	2.65	0	1	0	0	...	0	0	0	0	0	0	

5 rows × 23 columns

Figure 19: PS:1: X- Frame

```
X_train (18847, 23)
X_test (8078, 23)
y_train (18847, 1)
y_test (8078, 1)
```

Figure 20: PS:1: Train and Test Dataset

Linear Regression

Model 1

- The coefficient for independent attributes posts running linear regression on the train set is
- The intercept for our model is: - 3079.9408597176566
- R square on training data: 0.9404
- R square on training data: 0.941
- RMSE on Training data: 843.75
- RMSE on Testing data: 842.088

```
The coefficient for carat is 9200.336626821463
The coefficient for depth is 12.387011169526165
The coefficient for table is -23.08429213992879
The coefficient for x is -1177.3863159028929
The coefficient for y is 1082.3347702739356
The coefficient for z is -640.4608264842769
The coefficient for cut_Good is 387.29874760284565
The coefficient for cut_Ideal is 629.8858957652236
The coefficient for cut_Premium is 598.672582564512
The coefficient for cut_Very Good is 502.39269944588887
The coefficient for color_E is -188.8757658538538
The coefficient for color_F is -231.23337497096648
The coefficient for color_G is -411.0818661557577
The coefficient for color_H is -831.5176210701098
The coefficient for color_I is -1330.1184456500287
The coefficient for color_J is -1861.610644676013
The coefficient for clarity_IF is 3995.2161849354998
The coefficient for clarity_SI1 is 2535.9074240164423
The coefficient for clarity_SI2 is 1712.1729307119385
The coefficient for clarity_VS1 is 3355.1185668722756
The coefficient for clarity_VS2 is 3072.161615713959
The coefficient for clarity_VVS1 is 3776.8961134977267
The coefficient for clarity_VVS2 is 3766.786946331175
```

Figure 21: PS:1: Coefficients of attributes

94.1% of the variation in the price is explained by the predictors in the model for train data set. Hence the model works good for both test and train dataset

Model 2 - Using Stats Model

- We will use statsmodels.formula.api package to build the Stats model. For this we need to combine the train and test dataset using pd.concat () function. The new train and test datasets are given as data_trainand data_test.
- We will now formulate an expression where dependent variable is a function of all the independent variables:

expr = 'price ~ carat + depth + table + x + y + z + cut_Good + cut_Ideal + cut_Premium + cut_Very_Good + color_E + color_F + color_G + color_H + color_I+ color_J + clarity_IF + clarity_SI1 + clarity_SI2 + clarity_VS1 + clarity_VS2 + clarity_VVS1 + clarity_VVS2'

- Model Summary is shown below

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.940
Model:                            OLS   Adj. R-squared:                  0.940
Method:                 Least Squares   F-statistic:                 1.293e+04
Date:                Sat, 06 Nov 2021   Prob (F-statistic):               0.00
Time:                        21:10:26   Log-Likelihood:            -1.5373e+05
No. Observations:               18847   AIC:                         3.075e+05
Df Residuals:                   18823   BIC:                         3.077e+05
Df Model:                          23
Covariance Type:            nonrobust
=================================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------
Intercept     -3079.9409    749.396     -4.110      0.000   -4548.825   -1611.057
carat          9200.3366     77.388    118.886      0.000    9048.649    9352.024
depth            12.3870     10.465      1.184      0.237      -8.124      32.899
table           -23.0843      3.834     -6.021      0.000     -30.599     -15.570
x             -1177.3863    136.486     -8.626      0.000   -1444.912    -909.861
y              1082.3348    138.149      7.835      0.000     811.550    1353.120
z              -640.4608    131.083     -4.886      0.000    -897.395    -383.527
cut_Good        387.2987     44.032      8.796      0.000     300.992     473.605
cut_Ideal       629.8859     42.845     14.701      0.000     545.905     713.867
cut_Premium     598.6726     41.056     14.582      0.000     518.200     679.146
cut_Very_Good   502.3927     42.226     11.898      0.000     419.626     585.160
color_E        -188.8758     22.706     -8.318      0.000    -233.382    -144.369
color_F        -231.2334     23.018    -10.046      0.000    -276.350    -186.117
color_G        -411.0819     22.504    -18.267      0.000    -455.193    -366.971
color_H        -831.5176     24.000    -34.647      0.000    -878.560    -784.475
color_I       -1330.1184     26.755    -49.715      0.000   -1382.561   -1277.676
color_J       -1861.6106     32.764    -56.819      0.000   -1925.831   -1797.390
clarity_IF     3995.2162     64.905     61.555      0.000    3867.997    4122.436
clarity_SI1    2535.9074     55.575     45.631      0.000    2426.976    2644.839
clarity_SI2    1712.1729     55.883     30.639      0.000    1602.638    1821.708
clarity_VS1    3355.1186     56.782     59.087      0.000    3243.820    3466.417
clarity_VS2    3072.1616     55.924     54.934      0.000    2962.545    3181.778
clarity_VVS1   3776.8961     60.182     62.758      0.000    3658.934    3894.859
clarity_VVS2   3766.7869     58.502     64.387      0.000    3652.117    3881.457
==============================================================================
Omnibus:                     4642.691   Durbin-Watson:                   2.002
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17342.712
Skew:                           1.197   Prob(JB):                         0.00
Kurtosis:                       7.043   Cond. No.                     1.04e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.04e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Figure 22: Model 2 Summary

94 % of the variation in the price is explained by the predictors in the model for train data set. Hence the model works good for both test and train dataset

Equation:

(-3079.94) * Intercept + (9200.34) * carat + (12.39) * depth + (-23.08) * table + (-1177.39) * x + (1082.33) * y + (-640.46) * z + (387.3) * cut_Good + (629.89) * cut_Ideal + (598.67) * cut_Premium + (502.39) * cut_Very_Good + (-188.88) * color_E + (-231.23) * color_F + (-411.08) * color_G + (-831.52) * color_H + (-1330.12) * color_I + (-1861.61) * color_J + (3995.22) * clarity_IF + (2535.91) * clarity_SI1 + (1712.17) * clarity_SI2 + (3355.12) * clarity_VS1 + (3072.16) * clarity_VS2 + (3776.9) * clarity_VVS1 + (3766.79) * clarity_VVS2 +

Model 3 – 2nd Iteration (No Depth Attribute)

- Depth attribute dropped to reduce high multicollinearity.
- We will now formulate an expression where dependent variable is a function of all the independent variables:

  expr1 = 'price ~ carat + table + x + y + z + cut_Good + cut_Ideal + cut_Premium + cut_Very_Good + color_E + color_F + color_G + color_H + color_I+ color_J + clarity_IF + clarity_SI1 + clarity_SI2 + clarity_VS1 + clarity_VS2 + clarity_VVS1 + clarity_VVS2'

94 % of the variation in the price is explained by the predictors in the model for train data set. Hence the model works good for both test and train dataset

Equation:

(-2250.1) * Intercept + (9208.47) * carat + (-23.99) * table + (-1204.8) * x + (1028.71) * y + (-514.55) * z + (391.04) * cut_Good + (626.22) * cut_Ideal + (594.94) * cut_Premium + (501.32) * cut_Very_Good + (-188.91) * color_E + (-230.98) * color_F + (-410.66) * color_G + (-831.04) * color_H + (-1329.2) * color_I + (-1861.14) * color_J + (3995.55) * clarity_IF + (2538.31) * clarity_SI1 + (1714.06) * clarity_SI2 + (3356.47) * clarity_VS1 + (3074.03) * clarity_VS2 + (3777.96) * clarity_VVS1 + (3768.19) * clarity_VVS2 +

Inference: The overall P value is less than alpha, so rejecting H0 and accepting Ha that at least 1 regression co-efficient is not 0. Here all regression coefficients are not 0. Also, R square value is 94% as was seen from the previous model as well which concludes that this is fairly good model for our predictions and hence to increase the profits for the company.

Since all the models give 94% variations, we can choose any model.

Model Summary:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.940
Model:                            OLS   Adj. R-squared:                  0.940
Method:                 Least Squares   F-statistic:                 1.352e+04
Date:                Sat, 06 Nov 2021   Prob (F-statistic):               0.00
Time:                        21:10:26   Log-Likelihood:            -1.5373e+05
No. Observations:               18847   AIC:                         3.075e+05
Df Residuals:                   18824   BIC:                         3.077e+05
Df Model:                          22
Covariance Type:            nonrobust
=================================================================================
                    coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------
Intercept     -2250.0974    264.803     -8.497      0.000   -2769.135   -1731.060
carat          9208.4662     77.083    119.461      0.000    9057.376    9359.556
table           -23.9918      3.756     -6.387      0.000     -31.355     -16.629
x             -1204.7959    134.509     -8.957      0.000   -1468.446    -941.146
y              1028.7057    130.510      7.882      0.000     772.893    1284.518
z              -514.5515     76.606     -6.717      0.000    -664.707    -364.396
cut_Good        391.0366     43.919      8.904      0.000     304.951     477.122
cut_Ideal       626.2229     42.734     14.654      0.000     542.460     709.985
cut_Premium     594.9358     40.935     14.534      0.000     514.700     675.171
cut_Very_Good   501.3213     42.217     11.875      0.000     418.572     584.070
color_E        -188.9134     22.707     -8.320      0.000    -233.420    -144.406
color_F        -230.9787     23.017    -10.035      0.000    -276.094    -185.864
color_G        -410.6553     22.502    -18.250      0.000    -454.761    -366.550
color_H        -831.0396     23.997    -34.631      0.000    -878.076    -784.004
color_I       -1329.2023     26.744    -49.701      0.000   -1381.623   -1276.781
color_J       -1861.1360     32.762    -56.808      0.000   -1925.352   -1796.920
clarity_IF     3995.5484     64.905     61.560      0.000    3868.329    4122.768
clarity_SI1    2538.3103     55.538     45.704      0.000    2429.451    2647.170
clarity_SI2    1714.0572     55.861     30.685      0.000    1604.565    1823.549
clarity_VS1    3356.4692     56.772     59.122      0.000    3245.192    3467.746
clarity_VS2    3074.0258     55.903     54.989      0.000    2964.451    3183.600
clarity_VVS1   3777.9601     60.176     62.782      0.000    3660.009    3895.911
clarity_VVS2   3768.1874     58.491     64.423      0.000    3653.540    3882.835
==============================================================================
Omnibus:                     4646.699   Durbin-Watson:                   2.002
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            17384.030
Skew:                           1.198   Prob(JB):                         0.00
Kurtosis:                       7.049   Cond. No.                     2.59e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.59e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Figure 23: Model 3 Summary

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

Final Equation:

(-2250.1) * Intercept + (9208.47) * carat + (-23.99) * table + (-1204.8) * x + (1028.71) * y + (-514.55) * z + (391.04) * cut_Good + (626.22) * cut_Ideal + (594.94) * cut_Premium + (501.32) * cut_Very_Good + (-188.91) * color_E + (-230.98) * color_F + (-410.66) * color_G + (-831.04) * color_H + (-1329.2) * color_I + (-1861.14) * color_J + (3995.55) * clarity_IF + (2538.31) * clarity_SI1 + (1714.06) * clarity_SI2 + (3356.47) * clarity_VS1 + (3074.03) * clarity_VS2 + (3777.96) * clarity_VVS1 + (3768.19) * clarity_VVS2 +

- The exploratory analysis clearly showed us that diamonds with cuts in ideal, premium and very good cuts brought in more profits to the company. Hence, we can recommend to bring in more marketing strategies to promote these cuts. For e.g., advertising or inviting any social media influencers.
- Similarly, for the color H, I, J are bringing in more profits, so we need to maintain the same and use these colours to bring in more profits to the company. While looking at the other colours that is not bringing any profits, we can either decrease their price or promote them, so they sell out.
- Since diamonds are most sold when their clarity is much higher, the jeweller should make sure that they are of the finest quality hence bringing in more customers.

2 Problem Statement: 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.

Data Description

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

	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

Figure 24: PS 2: Sample Dataset

- The dataset consists of 8 variables: '**Unnamed: 0, Holliday_Package, Salary, age, educ, no_young_children, no_older_children, foreign'**
- Since we do not need the variable Unnamed for prediction or model building, we can drop the column.

```
hp_df.shape
```

```
(872, 7)
```

Figure 25: PS 2: Dataset Shape

- The shape of the data (872, 7).

	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 26: PS 2: Data Description

- There are two categorical variables – Holliday_Package and foreign.
- The minimum value for age is 20 and maximum is 62.

```
<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 27: PS 2: Data Info

Null Data Analysis

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

Figure 28: PS 2: Null Data Analysis

- There are no null values present in the dataset.

Duplicate Rows

```
Number of duplicate rows = 0
```

Figure 29: PS 2: Duplicate Rows

- There are no duplicates values in the data.

Unique Values of Categorical Variables

- Holliday_Package has two values: no and yes. No has a total of 471 values whereas yes has 401 values.

- Foreign has two values: no and yes. No has 656 values and yes has 216 values.

```
Holliday_Package
no     471
yes    401
Name: Holliday_Package, dtype: int64


foreign
no     656
yes    216
Name: foreign, dtype: int64
```

Figure 30: PS 2: Unique attributes

Univariate Analysis

As evident in the plots, we understand that salary distribution, no_young_children, and no_older_children are positively skewed while age and educ are normally distributed.

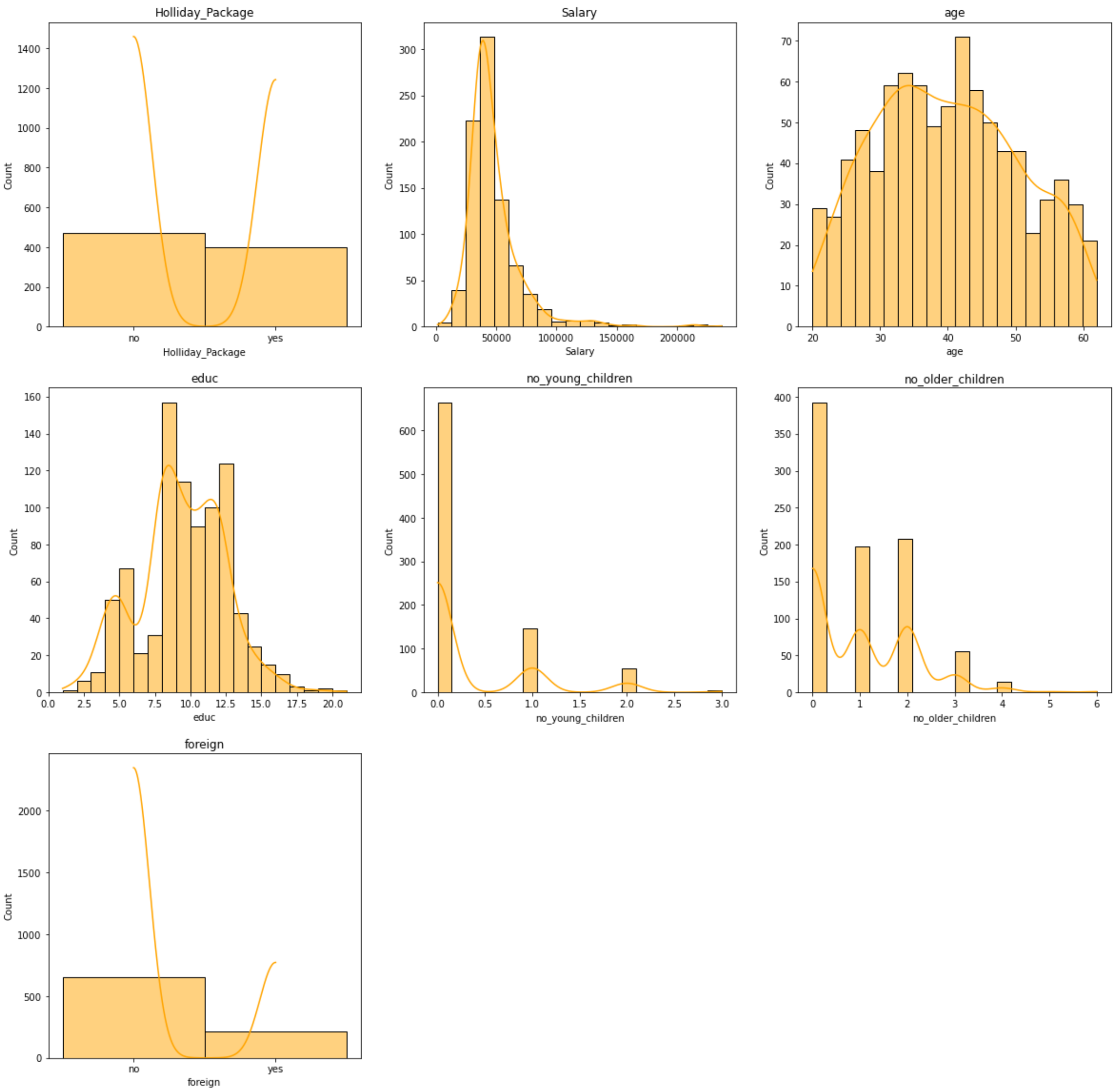

Figure 31: PS 2: Univariate Analysis

Bivariate Analysis

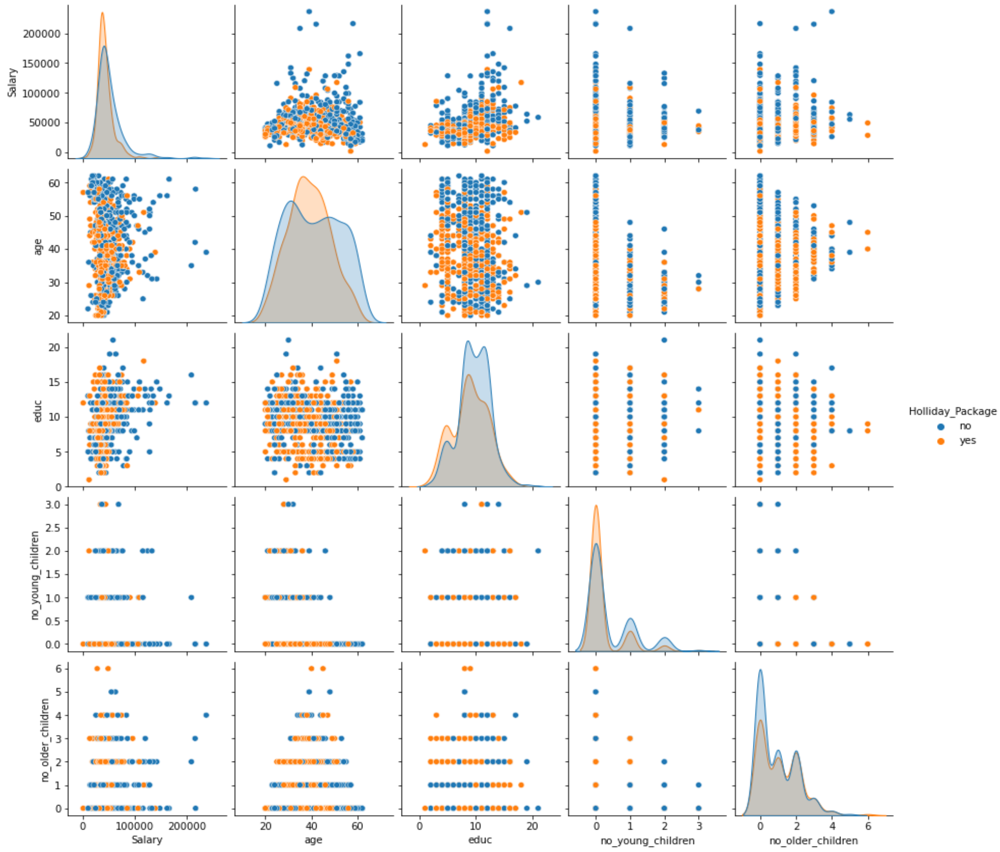

Figure 32: PS 2: Pair plot

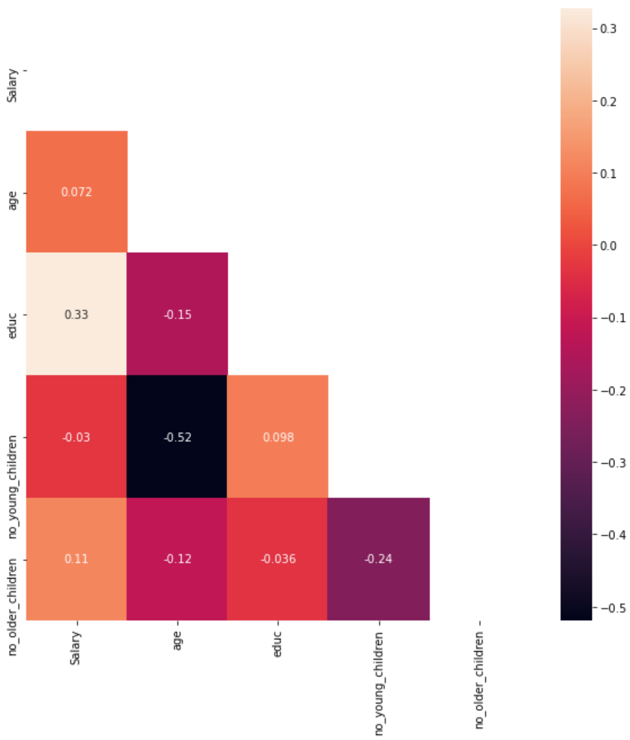

Figure 33: PS 2: Heatmap

There is no correlation between the data, the data seems to be normal. There is no huge difference in the data distribution among the holiday package. No multi collinearity in the data.

REMOVING OUTLIERS

- From univariate analysis we could find that, there are many outliers present in the data. For Logistic Regression and LDA, it is better to treat the outliers in order to get the best results.

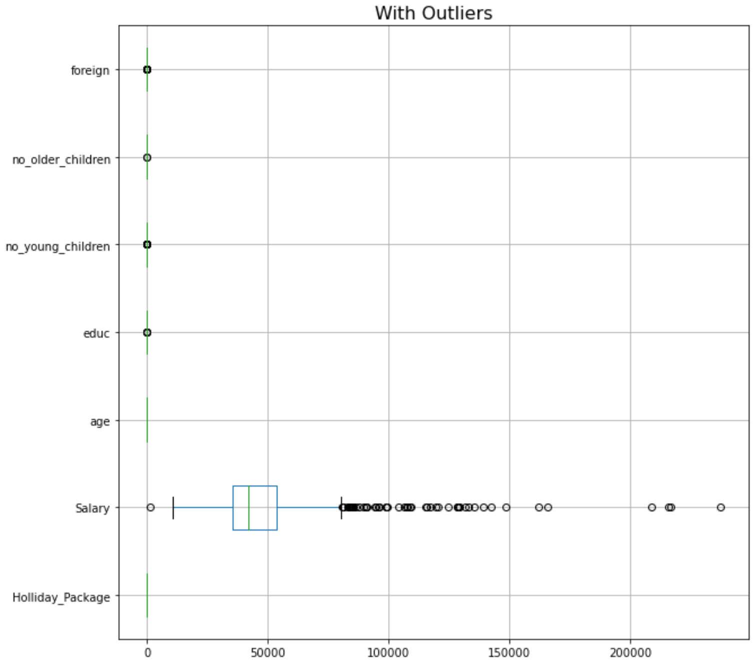

Figure 34: PS 2: Boxplot Before Outlier Treatment

- After treating the outliers, the data looks as below. There are no outliers present in the data after treating it.

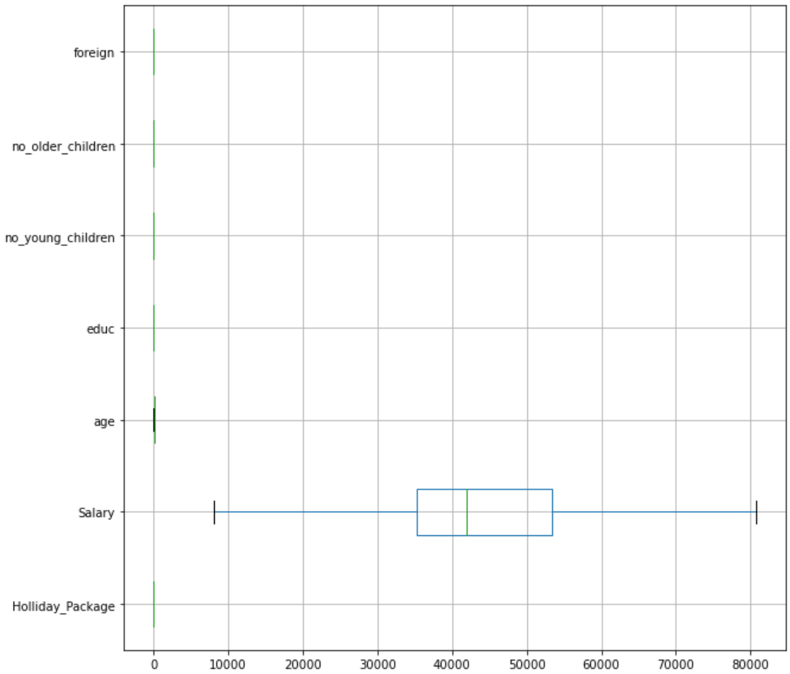

Figure 35: PS 2: Boxplot after Treatment

2.2 Do not scale the data. Encode the data (having string values) for Modelling. Data Split: Split the data into train and test (70:30). Apply Logistic Regression and LDA (linear discriminant analysis).

Encoding the categorical variables

- We use get_dummies () function to encode the string values for modelling, i.e., converting the categorical variables to dummy or indicator variables.
- After converting the variables, the data looks as below:

	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

Figure 36: Encoded Dataset

Train and Test Split

- Copying the predictor variable into an X data frame and target variable into Y data frame.

- Then we split the Train and test data as 70% and 30%.

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

Figure 37: Train and Test Set

- Y_train value counts:

```
0    0.534426
1    0.465574
```

Figure 38: Y- Train

- Y_test value counts:

```
0    0.553435
1    0.446565
```

Figure 39: Y Test

Logistic Regression Model

- Fitting the train and test data into logistic regression model:

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

- Predicting Probablities on the test data:

	0	1
0	0.696807	0.303193
1	0.332213	0.667787
2	0.620128	0.379872
3	0.686886	0.313114
4	0.354964	0.645036

Figure 40: Probability of Test Data

Logistic Regression using Grid Search

- Fitting the train and test data into logistic regression model(Grid Search):

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

- Predicting Probability on Test Data

	0	1
0	0.591060	0.408940
1	0.540422	0.459578
2	0.548785	0.451215
3	0.598272	0.401728
4	0.530048	0.469952

Figure 41: Probability of Test Data : Grid search

Linear Discriminant Analysis

- For LDA, you need to encode the data type and convert categorical target variable to integer (0 or 1).
- Then we copy the target and predictor variable into X and Y data frame and split the data into Test and train in 70% and 30%.
- We fit the data into train and test using lineardiscriminantanalysis()
- We fit the model into that and predict the test and Train Probabilities.

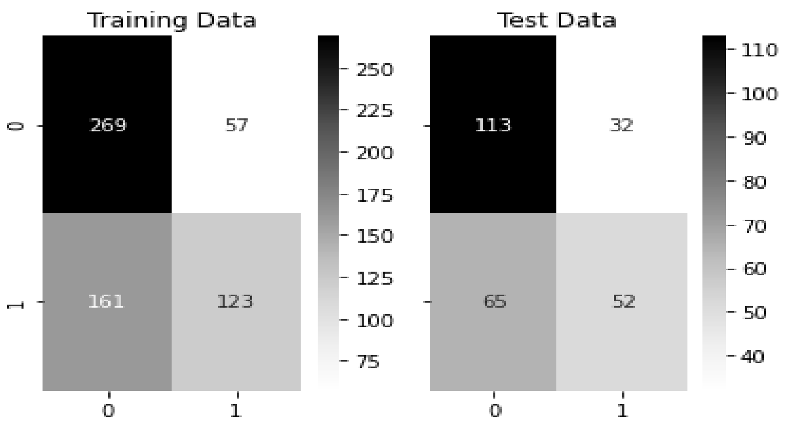

Figure 42: LDA confusion Matrix

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 Performance

Test Data

1. Classification Report

	precision	recall	f1-score	support
0	0.63	0.78	0.70	145
1	0.62	0.44	0.52	117
accuracy			0.63	262
macro avg	0.63	0.61	0.61	262
weighted avg	0.63	0.63	0.62	262

Figure 43 : Logit - Classification Report - Test Data

2. Confusion Matrix

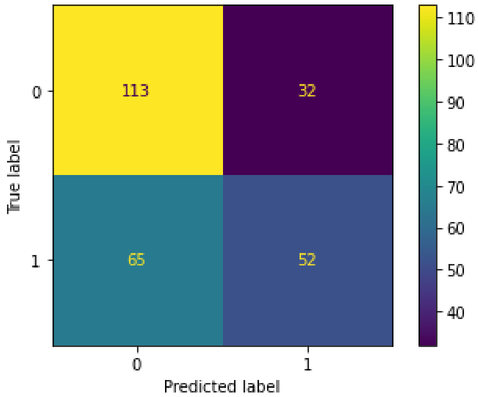

Figure 44: Logit -Confusion Matrix - Test Data

3. AUC

AUC is 0.667

4. Accuracy

The Accuracy of the Data is 62.9%

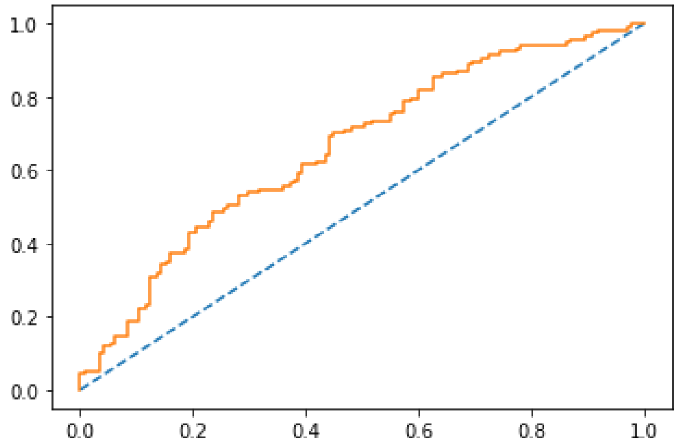

Figure 45: Logit - AUC - Test Data

Train Data

1. Classification Report

	precision	recall	f1-score	support
0	0.63	0.81	0.71	326
1	0.67	0.44	0.54	284
accuracy			0.64	610
macro avg	0.65	0.63	0.62	610
weighted avg	0.65	0.64	0.63	610

Figure 46: Logit - Classification Report - Train Data

2. Confusion Matrix

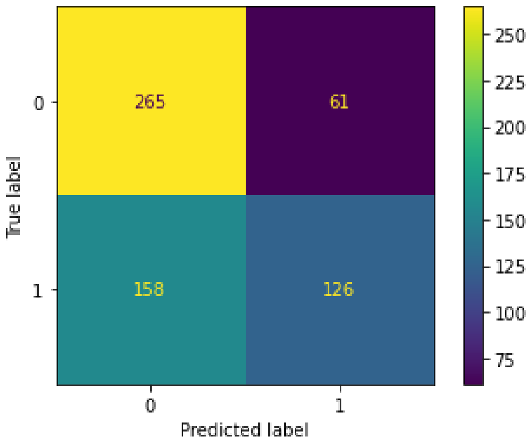

Figure 47: Logit -Confusion Matrix - Train Data

3. AUC
   AUC is 0.667

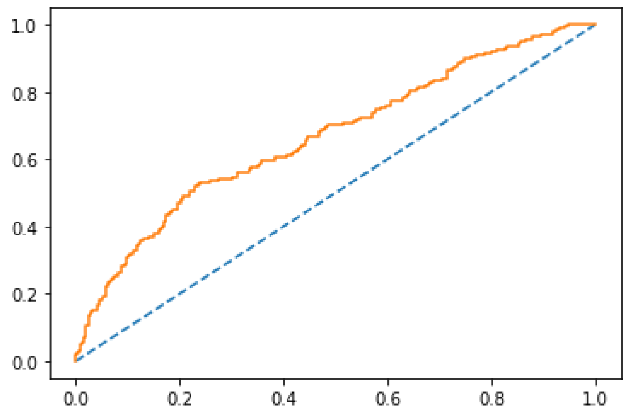

Figure 48: Logit - AUC - Train Data

4. Accuracy
   Accuracy of the train Data is 64.1%.

LDA Performance

Test Data

1. Classification Report

Classification Report of the test data:

	precision	recall	f1-score	support
0	0.63	0.78	0.70	145
1	0.62	0.44	0.52	117
accuracy			0.63	262
macro avg	0.63	0.61	0.61	262
weighted avg	0.63	0.63	0.62	262

Figure 49: LDA - Classification Report - Test Data

2. Confusion Matrix

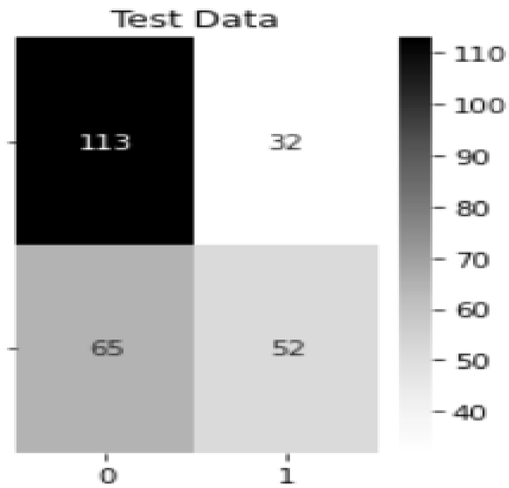

Figure 50: LDA - Confusion Matrix - Test Data

3. AUC
   AUC is 0.662

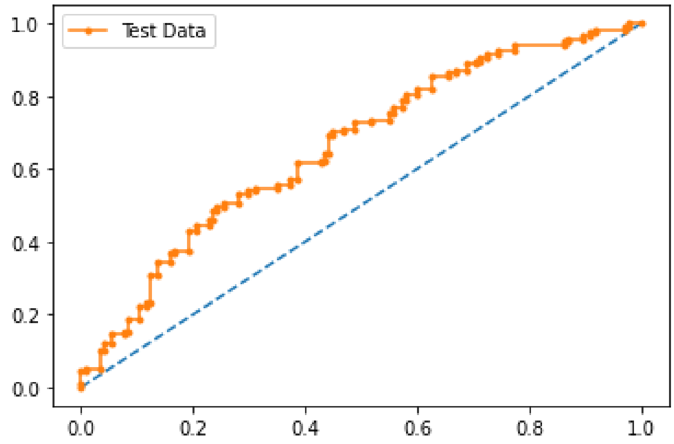

Figure 51: LDA - AUC - Test Data

4. Accuracy

   The score of LDA model on test data is: 0.63

Train Data

1. Classification Report

Classification Report of the training data:

	precision	recall	f1-score	support
0	0.63	0.83	0.71	326
1	0.68	0.43	0.53	284
accuracy			0.64	610
macro avg	0.65	0.63	0.62	610
weighted avg	0.65	0.64	0.63	610

Figure 52: LDA - Classification Report - Train Data

2. Confusion Matrix

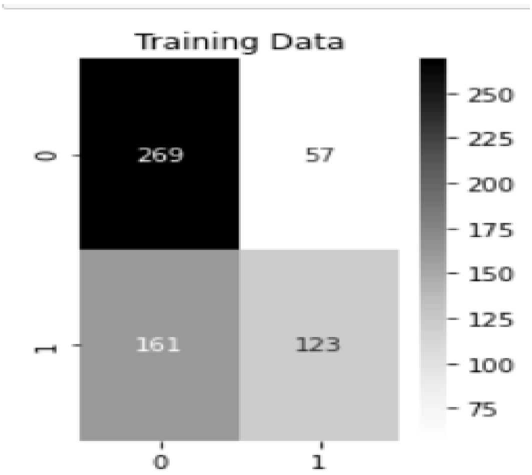

Figure 53: LDA - Confusion Matrix - Train Data

3. AUC
   AUC is 0.667

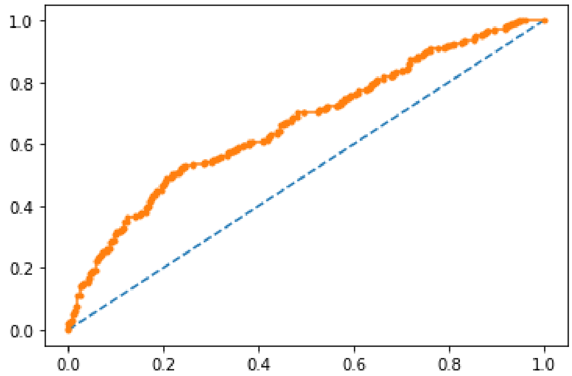

Figure 54: LDA - AUC - Train Data

4. Accuracy

   The score of LDA model on train data is: 0.643

Inference: Comparison of the Models

Table 1: Comparison of Models

	Logistic Regression		LDA	
	Train Set	Test Set	Train Set	Test Set
Accuracy	0.641	0.629	0.643	0.63
AUC	0.667	0.667	0.667	0.662
Recall	0.44	0.44	0.43	0.44
Precision	0.67	0.62	0.68	0.62
F-1 Score	0.54	0.52	0.53	0.52

- As, both the models are performing almost similar in terms of accuracy, fitness, false positive rate, false negative rate, true positive rate, true negative rate, precision and recall either will not give very different results.
- As both the values of the dependent variable are almost equally explained by each variable, the model’s performance is poor.
- In spite of these, we will recommend to use Logistic Regression (logit) model over LDA as it can be helpful to get additional information like the coefficients of each variable which in terms helps to get more insights about the effect of independent variables on the dependent variables.
- As stated above, both the models- Logistics and LDA offers almost similar results.

- While LDA offers flexibility to control or change the important metrices such as precision, recall and F1 score by changing the custom cut-off.
- Like in this case study, the moment we changed the cut-off to 40%, we were able to improve our precision, recall and F1 scores considerably.
- Further, this is up to the business if they would allow the play with the custom cut off values or no.
- Though for this case study, I have chosen to proceed with logistics regression as it is easier to implement, interpret and very efficient to train.
- Also, our dependent variable is following a binary classification of classes and hence it is ideal for us to rely on the logistic regression model to study the test case at hand.

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

Inference

- If employee is foreigner and employee not having young children, chances of opting for Holiday Package is good. Special offer can be designed to domestic employees to opt for Holiday Package.
- Many high salary employees are not opting for Holiday Package, company can focus on high salary employees to sell Holiday
- Package. Employees having older children are not opting for Holiday Package. Age of the employee is not a material in opting for holiday package.
- It can be observed from coefficient arrived from both models that opting for Holiday package has strong negative relation with number of young children. Holiday packages can be modified to make infant and young children friendly to attract more employees having young children.
- The most important factors for a user to opt for a Holiday Package are the person being a foreign national, and the number of young children.
- The chances of a user opting for a Holiday Package increases when he/she is a foreign national and reduces People with young children don’t prefer to go for Holiday Packages
- People at a mid-age level (25-45) is the age group who opt for holiday packages the most, as people grow old there is a decline in the interest on holiday packages
- People completing higher education seems to be more inclined towards holiday packages People with very low salary don’t prefer holiday packages.

Recommendation:

- The company should really focus on Foreigners to drive sales of their holiday packages as that's where the majority of conversions are going to come in.
- The company can try to direct their marketing efforts or offers towards foreigners for a better conversion opting for holiday packages.
- The company should also stay away from targeting parents with younger children. The chances of selling to parents with 2 younger children is probably the lowest. This also gels with the fact that parents try and avoid visiting with younger children.
- If the firm wants to target parents with older children, that still might end up giving favourable return for them marketing efforts then spent on couples with younger children.

The End!