greatlearning

Project Report of

Predictive Modelling Module

<u>Project Report</u>

Submitted to

greatlearning

Submitted By

Group No. 5 Batch: 2021 Location: Pune

Group Members

1. **Niranjan Dhavan**
2. **Sagar Belagali**
3. **Nimit Kumar**
4. **Pravin Kumar**
5. **Takshay Sheetigar**
6. **Praveen Kulandia Arasu**
7. **Abhinav Pathak**

Problem 1: Linear Regression

Problem Statement –

You are hired by a company named Gemstone Co Ltd, which is a cubic zirconia manufacturer. You are provided with the dataset containing the prices and other attributes of approximately 27,000 pieces of cubic zirconia (which is an inexpensive synthesized diamond alternative with similar qualities of a diamond).

Objective

Objective is to help the agency in predicting whether a high school graduate will win a full scholarship on the basis of the information given in the data set. Also, find out the important factors which are instrumental in winning a full scholarship in colleges.

cubic_zirconia Data

The data dictionary is given below.

1. Carat - Carat weight of the cubic zirconia

2. Cut - Describes the cut quality of the cubic zirconia. Quality is in increasing order: Fair, Good, Very Good, Premium, Ideal.

3. Colour - Colour of the cubic zirconia.

4. Clarity - Cubic zirconia Clarity refers to the absence of the Inclusions and Blemishes. (In order from Best to Worst, FL = flawless, I3= level 3 inclusions) FL, IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1, I2, I3

5. Depth - The Height of a cubic zirconia piece, measured from the Culet to the table, divided by its average Girdle Diameter.

6. Table - The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter.

7. Price - Price of the cubic zirconia.

8. X - Length of the cubic zirconia in mm.

9. Y - Width of the cubic zirconia in mm.

10. Z - Height of the cubic zirconia in mm.

Performing exploratory data analysis on the dataset. Showcasing some charts & graphs.

1. Loading the data set- We will be loading the “cubic_zirconia.csv” file using pandas library in python. For this, we will be using read_csv file.

2. The head function will tell us the top head records in the data set. By default, python shows you only the top 5 record.

	Unnamed: 0	carat	cut	color	clarity	depth	table	x	y	z	price
0	1	0.3	Ideal	E	SI1	62.1	58	4.27	4.29	2.66	499
1	2	0.33	Premium	G	IF	60.8	58	4.42	4.46	2.7	984
2	3	0.9	Very Good	E	VVS2	62.2	60	6.04	6.12	3.78	6289
3	4	0.42	Ideal	F	VS1	61.6	56	4.82	4.8	2.96	1082
4	5	0.31	Ideal	F	VVS1	60.4	59	4.35	4.43	2.65	779

3. The tail function will tell us the last entries records in the data set. By default, python shows you only the last 5 records. Let's check tail for the totals/subtotals if any. The cubic zirconia data dataset doesn't contain any total/subtotals. Further basis inspection it was identified that Unnamed: 0 is useless column lets drop the Unnamed: 0 column.

	carat	cut	color	clarity	depth	table	x	y	z	price
26962	1.11	Premium	G	SI1	62.3	58	6.61	6.52	4.09	5408
26963	0.33	Ideal	H	IF	61.9	55	4.44	4.42	2.74	1114
26964	0.51	Premium	E	VS2	61.7	58	5.12	5.15	3.17	1656
26965	0.27	Very Good	F	VVS2	61.8	56	4.19	4.2	2.6	682
26966	1.25	Premium	J	SI1	62	58	6.9	6.88	4.27	5166

4. The shape attribute tells us a number of observations and variables we have in the data set. It is used to check the dimension of data. The data set has 26967 observations and 10 variables in the data set.

5. info() is used to check the Information about the data and the datatypes of each respective attribute.

```
<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)
```

Figure 1

- Looking at the data in the head function and in info, we come to know that the variables comprise of float, object and integer data types. sklearn in Python does not take the input of object data types while building Linear Regression models. So, we need to convert these variables into some numerical form.
- Basis Figure 1 we can see that there are three object type variables (cut, colour & clarity) which has the object data types which we need to convert into numerical form. Since the variable cut & clarity are in ordinal range, we are replacing the categorical variables with the numbers. Further we shall perform one hot encoding for colour variable. The conversion is done before running the linear regression model.
- Further basis above figure 1 we can see that there are 697 null values in the depth variable. Let's go ahead and drop the null values from the dataset. Post Dropping the null values we can see that we have lost the 2.58% of data loss. (i.e. Left with 26270 entries).
- Also, it was identified that there were 34 duplicated rows in the dataset. Will go ahead and drop the duplicated values from the dataset.

6. The described method will help to see how data has been spread for numerical values. We can clearly see the minimum value, mean values, different percentile values, and maximum values for the Income data set.

	count	mean	std	min	25%	50%	75%	max
carat	26236	0.79762	0.476691	0.2	0.4	0.7	1.05	4.5
depth	26236	61.745285	1.412243	50.8	61	61.8	62.5	73.6
table	26236	57.455877	2.230866	49	56	57	59	79
x	26236	5.728646	1.126332	0	4.71	5.69	6.54	10.23
y	26236	5.732487	1.165283	0	4.72	5.7	6.54	58.9
z	26236	3.536339	0.698608	0	2.9	3.52	4.04	8.06
price	26236	3935.926818	4019.809223	326	945	2374	5356	18818

Figure 2

- Basis Figure 2 We can see that min value as 0 in x, y & z. Where x, y, & z means the Length, width & height as 0. Where it cannot be zero.
- We can see there are 8 rows with Dimensions 'Zero'. We will Drop them as it seems better choice instead of filling them with any of Mean or Median.
- Post Cleaning the data we can see that we have lost the 2.74% of data. (i.e. Left with 26228 entries)

Exploratory Data Analysis

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations. We performed uni-variate and bi-variate analysis to get a better overview and to find outliers in our dataset. Outliers can occur due to some kind of errors while collecting the data and need to be removed so that it doesn't affect the performance of our model.

Uni-Variate Analysis.

Target / Predictor Variable Analysis - price.

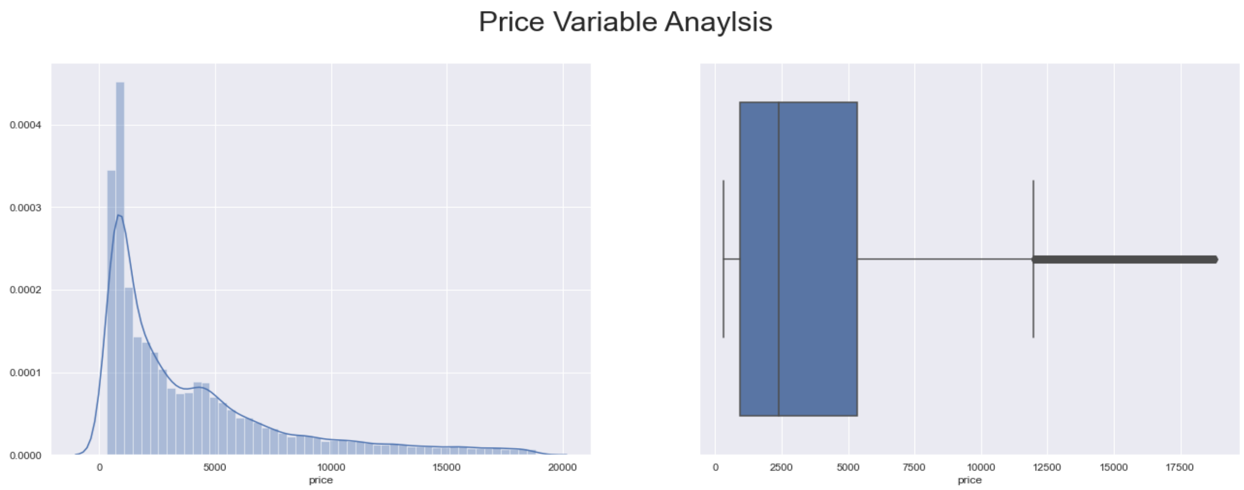


Figure 3

- The Price Variable distribution seems to be Highly Left-skewed.
- Basis skewness value we can see that distribution is highly skewed.
- Basis box plot we can see that there are outliers in the variable. Let's treat the outliers in the further process.

Response / Dependent Variable Analysis

*1). **Analyzing Feature: cut***

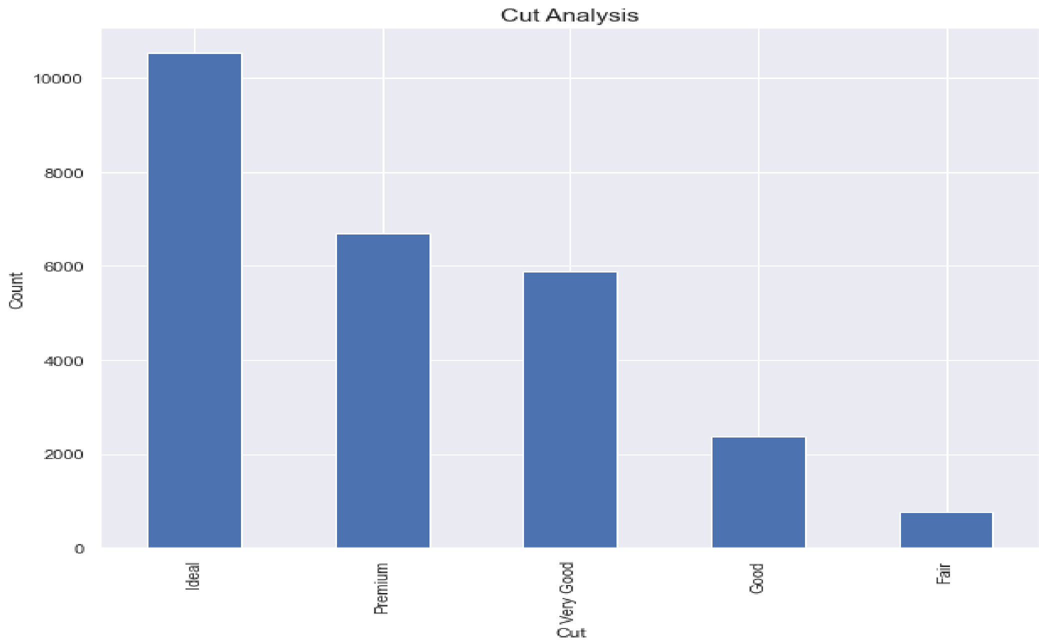


Figure 4

- Basis above figure 4 we can see that ideal cut diamond have comparably huge demand in the market followed by the premium cut.
- Basis above figure 4 we can also infer that quality of cut changes the demand for the product goes down.

*2). **Analyzing Feature: color***

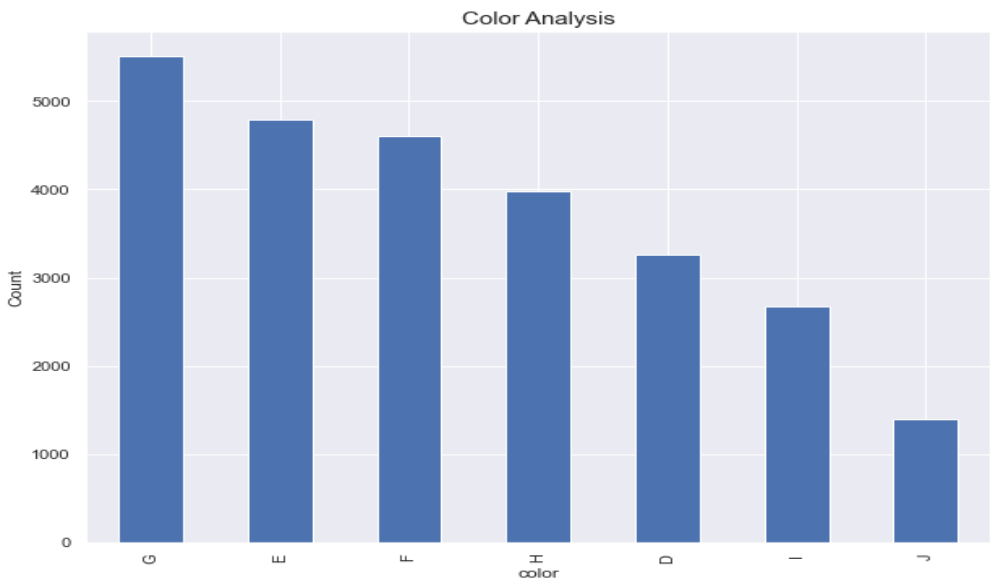


Figure 5

- Basis above figure we can see that G Colour diamond have comparably huge demand in the market as it might be lowly priced as compare to Colourless cibic zirconia.
- Basis above figure we can also infer that colourless cubic zirconia are high priced in market as compared to near colourless cubic zirconia.

*3). **Analyzing Feature: clarity***

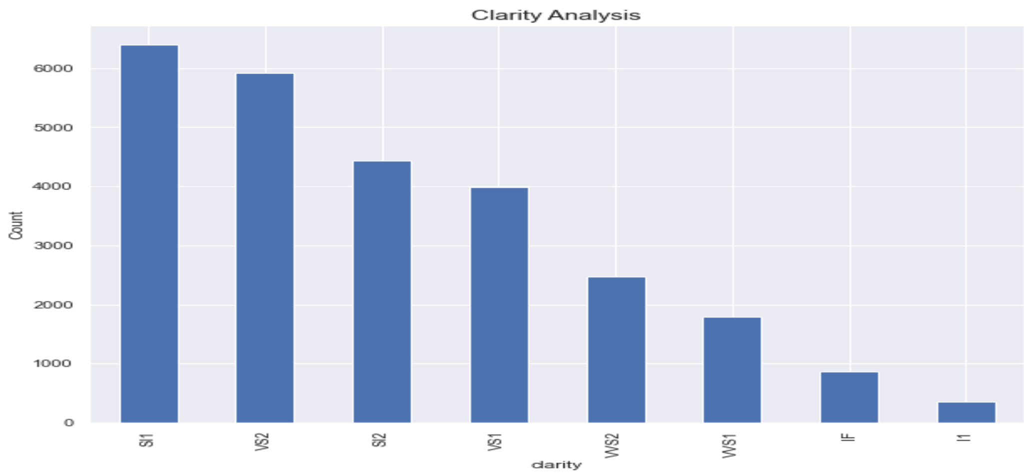


Figure 6

- Basis above figure we can see that SI1 Clarity cubic zirconia have comparably huge demand in the market followed by the VS2 as they might be prices lower as compare to flawless cubix zirconia.
- Basis above figure we can also infer that IF cubic zirconia Has lower demand in the market due to higher price bracket.

*4). **Analyzing Feature: depth***

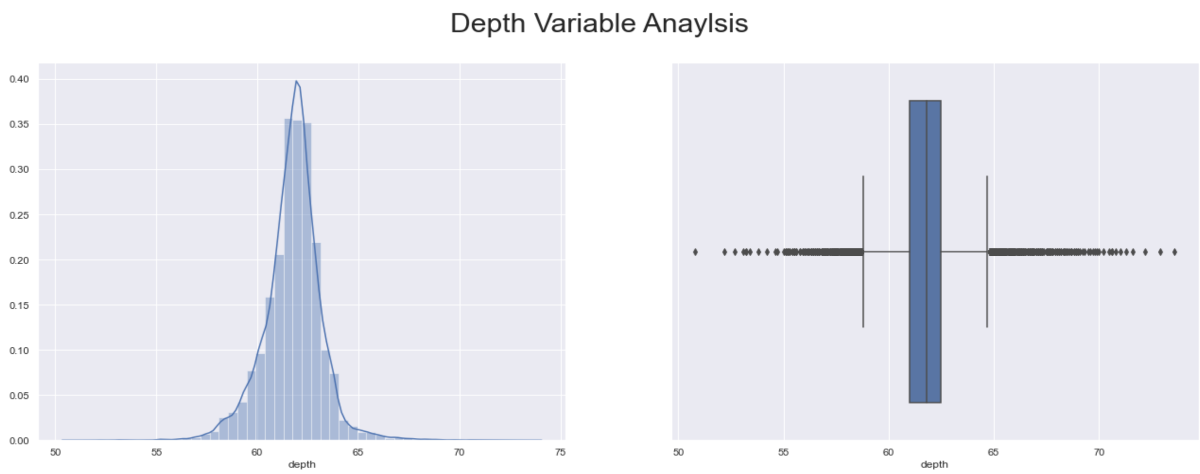


Figure 7

- Basis above figure we can see that Data distribution for 'depth' variable is slightly left-skewed.
- Basis skewness value we can see that distribution is approximately symmetric.
- Basis box plot we can see that there are outliers in the variable. Let's treat the outliers in the further process.

*5). **Analyzing Feature: x***

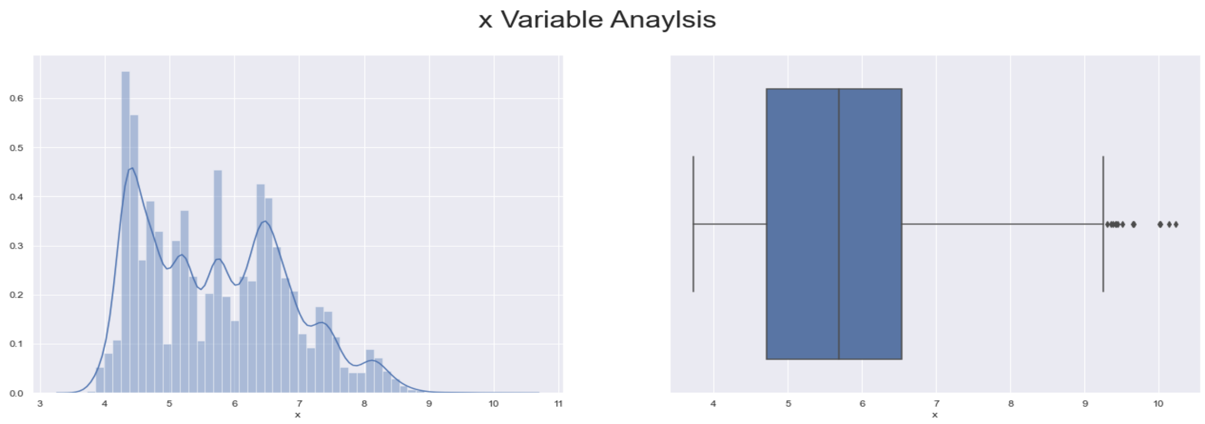


Figure 8

- Basis above figure we can see that Data distribution for 'x' variable is Right-skewed.
- Basis skewness value we can see that distribution is approximately symmetric.
- Basis box plot we can see that there are outliers in the variable. Let's treat the outliers in the further process.

*5). **Analyzing Feature: y***

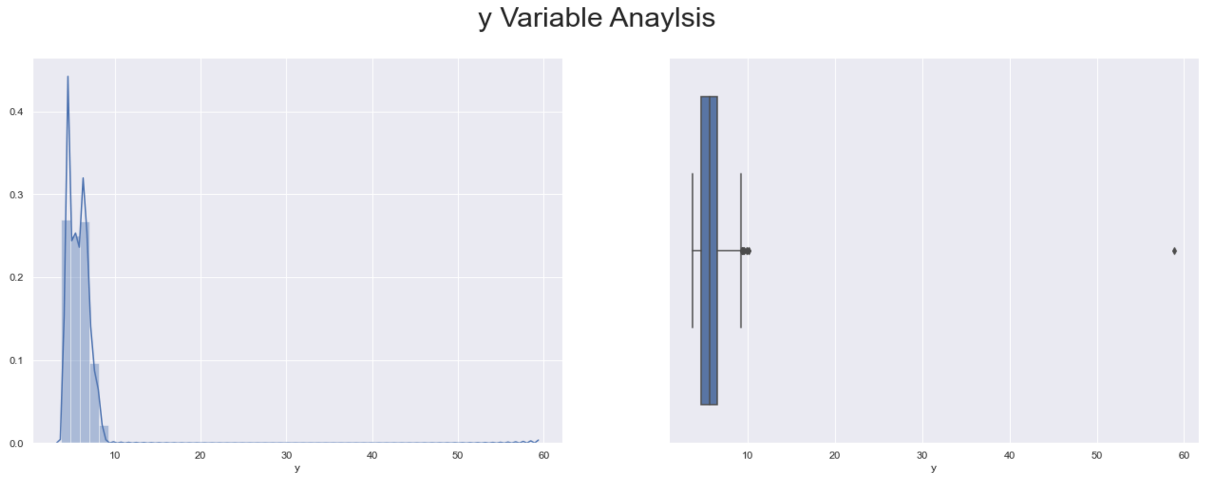


Figure 9

- Basis above figure we can see that Data distribution for 'y' variable is highly left-skewed.
- Basis skewness value we can see that distribution is highly skewed.

- Basis box plot we can see that there are couple of outliers in the variable. Let's treat the outliers in the further process.

*6). **Analyzing Feature: z***

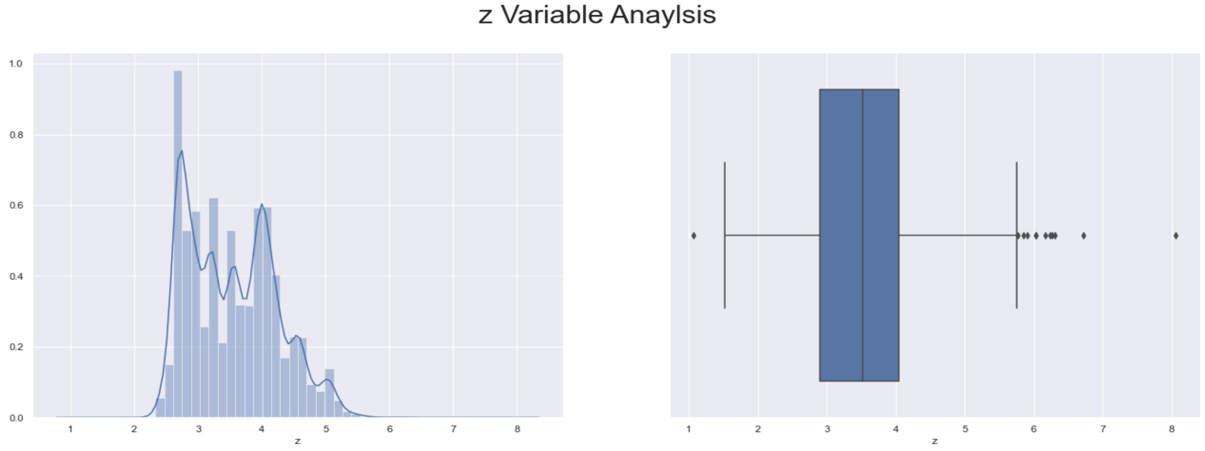


Figure 10

- Basis above figure we can see that Data distribution for 'z' variable is Right-skewed.
- Basis skewness value we can see that distribution is approximately symmetric.
- Basis box plot we can see that there are couple of outliers in the variable. Let's treat the outliers in the further process.

Bi-Variate Analysis.

Factors influencing price of the property.

Analyzing Feature: Price Vs carat

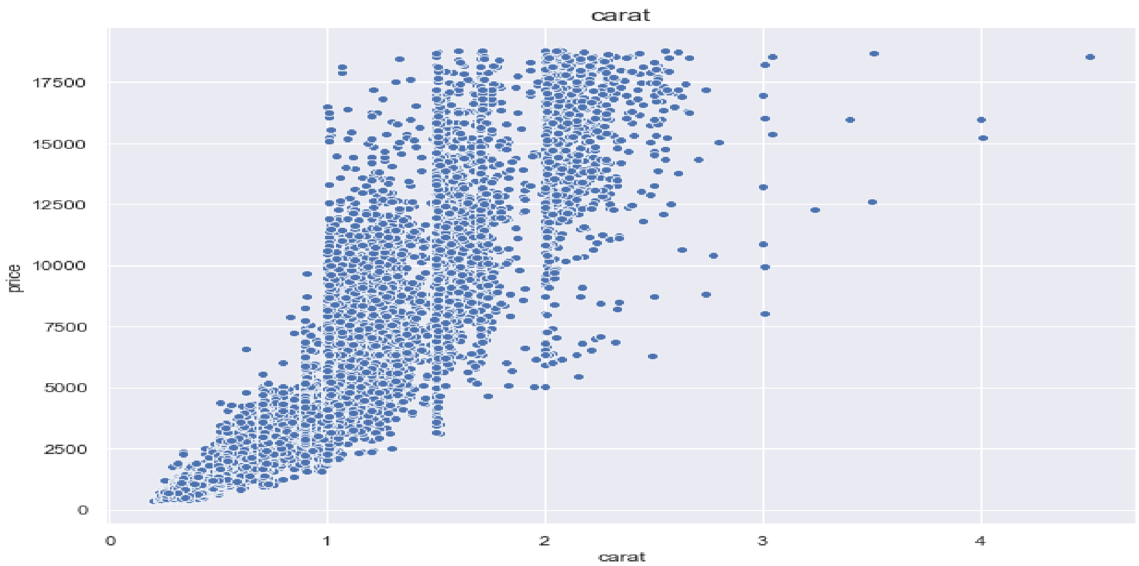


Figure 11

- *Basis above figure we can see that as the carat cubic zirconia increases the prices increases.*

Analyzing Feature: Price Vs x (Length)

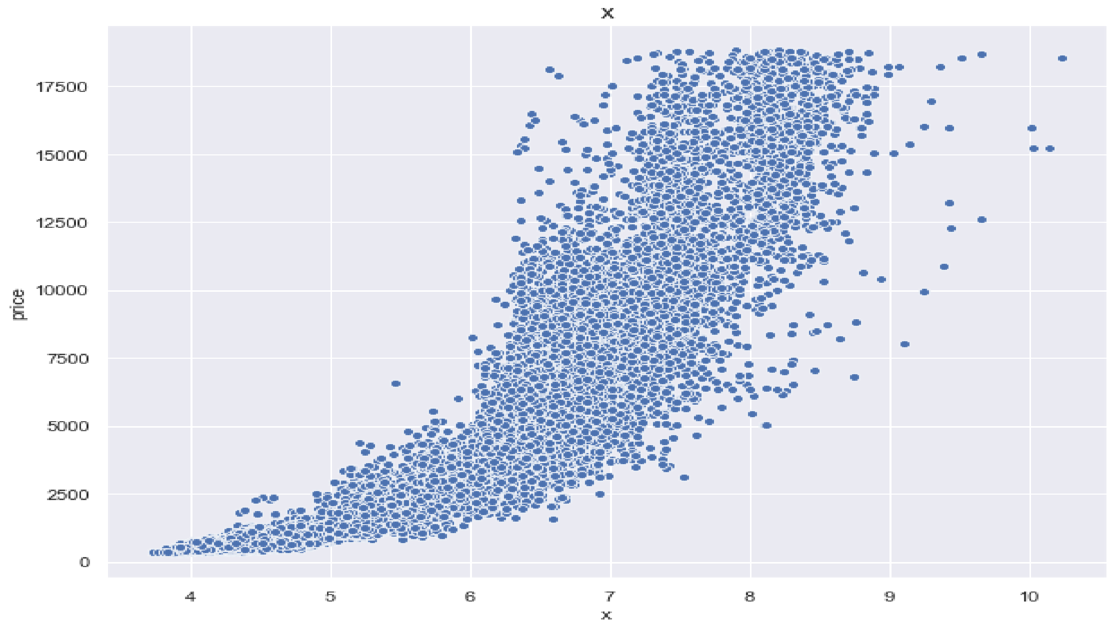


Figure 12

- Basis above figure we can see that as the cubic zirconia Length of the cubic zirconia in mm increases the prices increases.

Analyzing Feature: Price Vs z (Height)

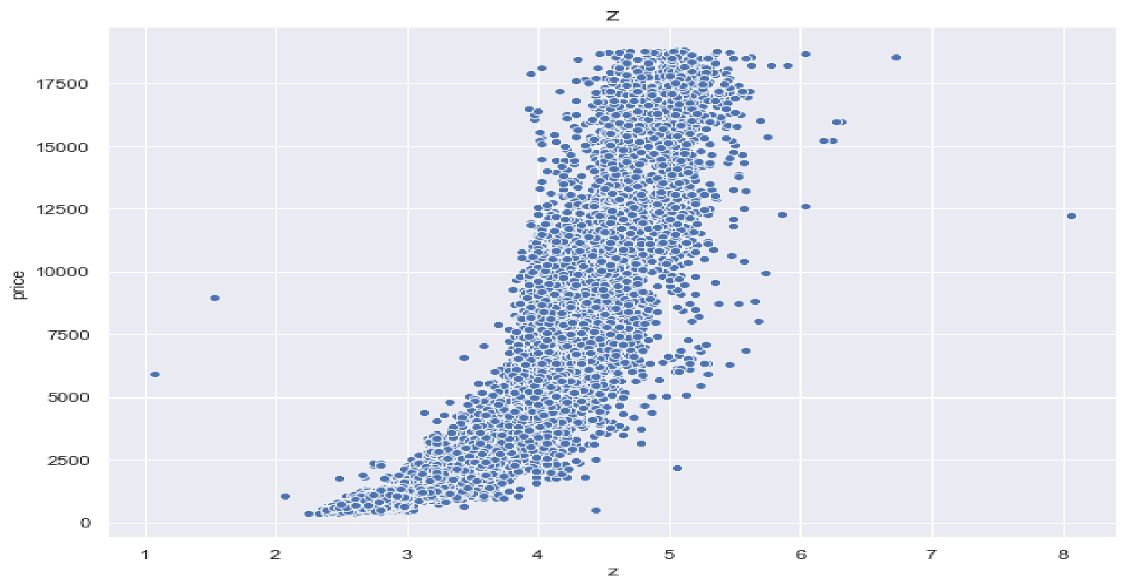


Figure 13

- Basis above figure we can see that as the cubic zirconia Height of the cubic zirconia in mm. increases the prices increases

- ***Analyzing Feature: Price Vs cut***

Figure 14

- Basis above figure we can see that fair cut is being highly priced in the market.
- Basis above figure we can see that premium cut has second highly price cubic zirconia in the market.

Analyzing Feature: Price Vs color

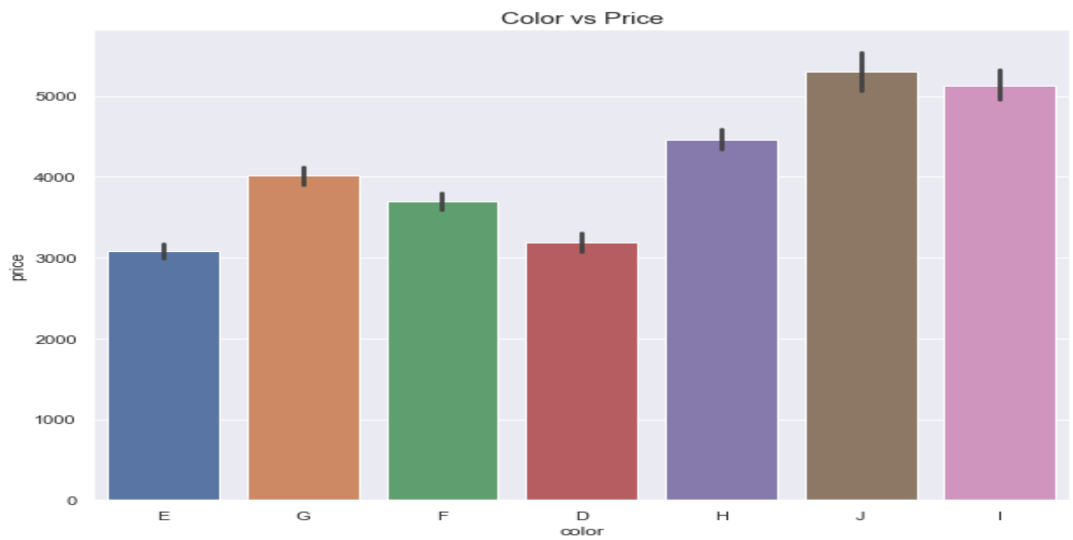


Figure 15

- Basis above figure we can see that J Colour is being highly priced in the market.
- Basis above figure we can see that Near colourless cubic zirconia has high prices in the market.

Analyzing Feature: Price Vs clarity

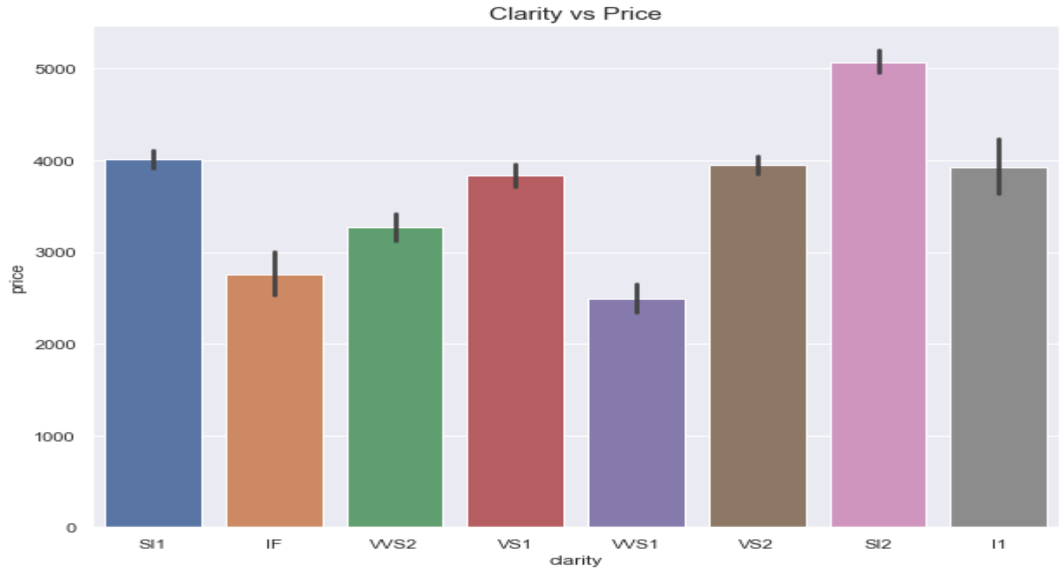


Figure 16

- Basis above figure we can see that SI1 Colour is being highly priced in the market. Followed by SI1
- Basis above figure we can see that Small inclusions cubic zirconia, very small inclusions have higher prices as compared to the Flawless cubic zirconia.

Multi-Variate Analysis.

Analyzing Feature: Price Vs Carat

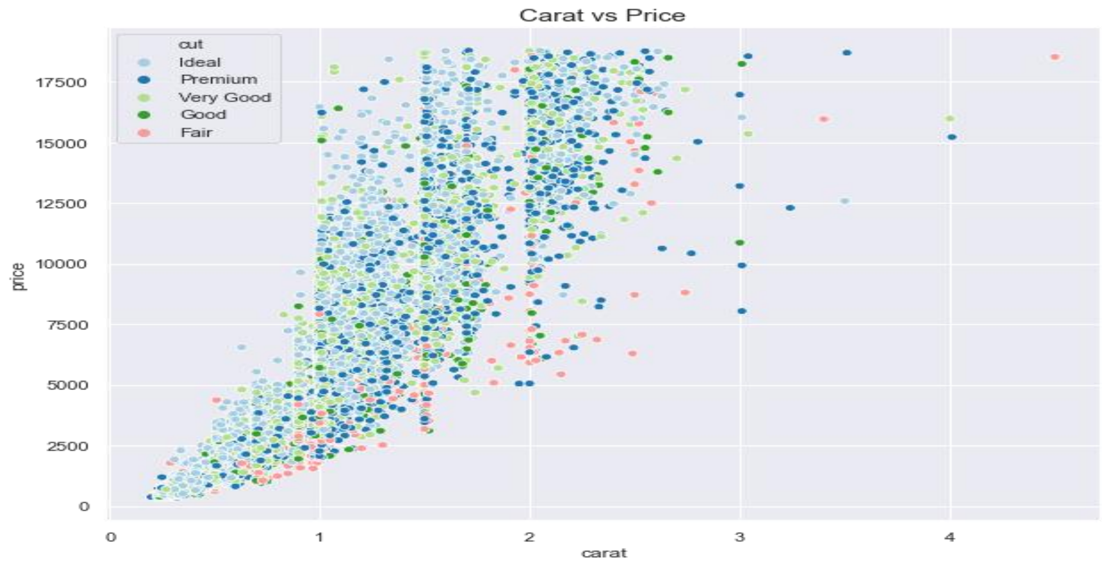


Figure 17

- Basis above figure we can see that ideal cut cubic zirconia are Highly priced in the market irrespective of the carat.
- Basis above figure we can see that Fair cut cubic zirconia are lowly priced in the market as compared to the other cuts.
- Basis above figure we can see that premium & very good cut cubic zirconia are averagely priced in the market.

Analyzing Feature: Price Vs Carat

Figure 18

- Basis above figure we can see that I1 clarity cubic zirconia are lower priced in the market irrespective of the carat.
- Basis above figure we can see that SI2 clarity cubic zirconia are priced Above I1 clarity cubic zirconia in the market irrespective of the carat.
- Basis above figure we can see that IF cut cubic zirconia are Highly priced in the market as compared to the other cuts despite of being the lower carats (i.e. Below 2carat).

__Analyzing Feature: Price Vs Carat__

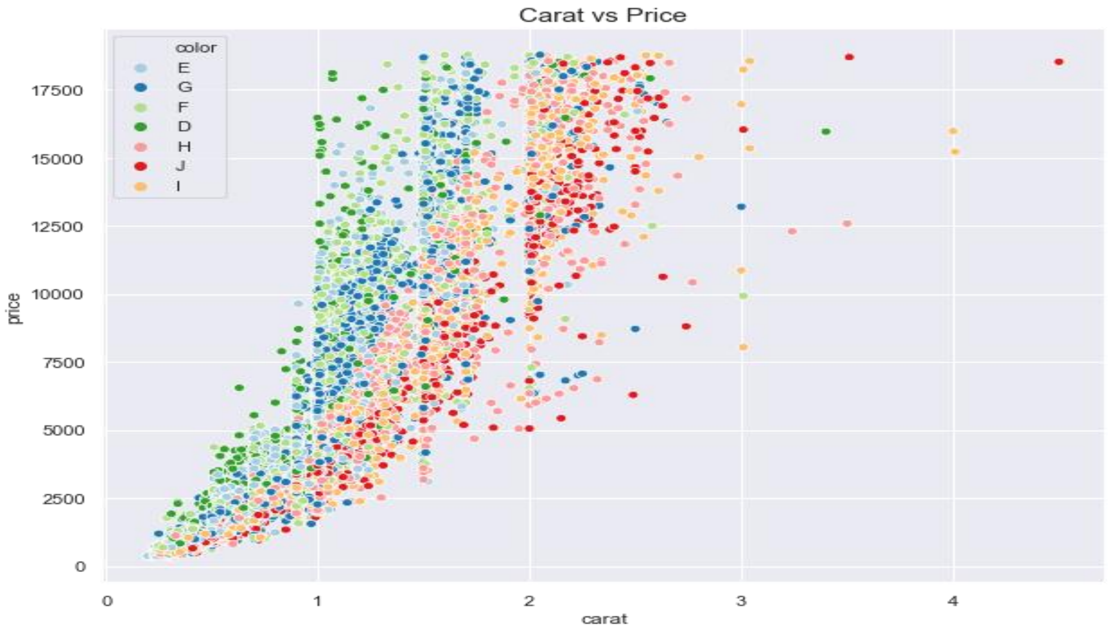


Figure 19

- Basis above figure we can see that H & I Colour cubic zirconia are lower priced in the market irrespective of the carat.
- Basis above figure we can see that D, E & F Colour cubic zirconia are Highley priced in the market.

Data Pre-Processing

- Basis analysis of above we can see that there are outliers in the variables. Let's go ahead and treat the outliers.
- As mentioned above couple of variables are categorical variables, we shall Replace the categorical variables with the numbers as its ordinal range. (i.e. For Variable cut & clarity).
- sklearn in Python does not take the input of object data types when building linear regression model. So, we need to convert these variables into some numerical form. We shall perform one hot encoding for them (i.e. colour variable). Post Data Pre-processing the head of data looks like.

	carat	cut	clarity	depth	table	x	y	z	price	color_E	color_F	color_G	color_H	color_I	color_J
0	0.3	4	4	62.1	58	4.27	4.29	2.66	499	1	0	0	0	0	0
1	0.33	3	9	60.8	58	4.42	4.46	2.7	984	0	0	1	0	0	0
2	0.9	2	7	62.2	60	6.04	6.12	3.78	6289	1	0	0	0	0	0
3	0.42	4	6	61.6	56	4.82	4.8	2.96	1082	0	1	0	0	0	0
4	0.31	4	8	60.4	59	4.35	4.43	2.65	779	0	1	0	0	0	0

- We are not scaling the dataset. And proceeded with dataset as it is.

Linear Regression

Linear regression is a way to identify a relationship between two or more variables. We use this relationship to predict the values for one variable for a given set of value(s) of the other variable(s). The variable, which is used in prediction is termed as independent/explanatory/regressor variable where the predicted variable is termed as dependent/target/response variable. Linear regression assumes that the dependent variable is linearly related to the estimated parameter(s).

$$y = c+mx$$

In machine learning and regression literature the above equation is used in the form:

$$y = w0+w1x$$

Where w0 is intercept on y-axis, w1 is slope of line, x is an explanatory variable and y is the response variable.

1. Descriptive Linear Regression

2. Predictive Linear Regression

Descriptive Linear Regression

- Descriptive Linear Regression – Main Objective of Descriptive Linear Regression is to understand the relation between Features / Variables.
- In Descriptive Linear Regression we need to look after assumptions i.e. VIF Values (for Multicollinearity). And we select variables basis significance of p value. Whereas for descriptive type assumptions and metric both stand important.
- For Descriptive type we don't divide the dataset into train and test. Also, We Use statmodel for as python library for Descriptive Linear Regression.

Firstly, we will use descriptive linear regression to understand which all variables are significant variables that impact the price variable.

Descriptive linear regression Models

Model 1 (Using All the variables)

- Firstly, we will import statsmodels.formula.api as SM model
- In first model will run the model using all the variables (i.e. price~carat+cut+clarity+depth+table+x+y+z+color_E+color_F+color_G+color_H+color_I+color_J).

OLS Regression Results

Dep. Variable:	price	R-squared:	0.933			
Model:	OLS	Adj. R-squared:	0.933			
Method:	Least Squares	F-statistic:	2.62E+04			
Date:	Mon, 19 Apr 2021	Prob (F-statistic):	0			
Time:	00:08:07	Log-Likelihood:	-2.15E+05			
No. Observations:	26228	AIC:	4.31E+05			
Df Residuals:	26213	BIC:	4.31E+05			
Df Model:	14					
Covariance Type:	nonrobust					
	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-6347.3872	733.729	-8.651	0	-7785.537	-4909.238
carat	8968.8233	68.975	130.031	0	8833.629	9104.017
cut	107.6421	6.115	17.602	0	95.656	119.628
clarity	423.8668	3.772	112.379	0	416.474	431.26
depth	66.3467	10.394	6.383	0	45.973	86.72
table	-10.9503	3.277	-3.342	0.001	-17.373	-4.528
x	-1164.1699	101.95	-11.419	0	-1363.997	-964.342
y	1663.5257	100.834	16.498	0	1465.885	1861.166
z	-1487.4303	138.225	-10.761	0	-1758.359	-1216.502
color_E	-211.978	20.354	-10.415	0	-251.873	-172.083
color_F	-284.8784	20.618	-13.817	0	-325.29	-244.467
color_G	-457.9352	20.113	-22.768	0	-497.358	-418.512
color_H	-888.053	21.49	-41.324	0	-930.174	-845.932
color_I	-1332.4456	23.992	-55.537	0	-1379.471	-1285.42
color_J	-1879.0297	29.388	-63.939	0	-1936.632	-1821.428
Omnibus:	3904.258	Durbin-Watson:	2.005			
Prob(Omnibus):	0	Jarque-Bera (JB):	18556.862			
Skew:	0.649	Prob(JB):	0			
Kurtosis:	6.911	Cond. No.	1.14E+04			

- Basis above model we can that adjusted r square is 93.3%. Which seems to be good. Also Checking the p values as the p values seems to significant variables. But the condition number seems to higher side which in turn shows that there is multi-collinearity.
- Let's check the VIF values for the all the variables.

	variables	VIF
0	carat	124.670505
1	cut	10.447436
2	clarity	13.033655
3	depth	1275.586253
4	table	891.050097
5	x	10703.9252
6	y	9415.45591
7	z	3639.445516
8	color_E	2.475554
9	color_F	2.441419
10	color_G	2.784461
11	color_H	2.292532
12	color_I	1.916987
13	color_J	1.506095

Basis above figure we can see that there are many variables having high VIF which shows that there is multicollinearity in the independent variables. Basis VIF scores above as there is multicollinearity Lets consider the threshold of 5 and start dropping the variables of VIF above 5 one by one basis the variable which are less co-related with price variable.

Model 2 (Model with Dropping high infinity VIF values (i.e. clarity Variable).

- In second model will run the model using all the variables (i.e. price~carat+cut+depth+table+x+y+z+color_E+color_F+color_G+color_H+color_I+color_J).

OLS Regression Results

Dep. Variable:	price	R-squared:	0.901
Model:	OLS	Adj. R-squared:	0.901
Method:	Least Squares	F-statistic:	1.84E+04
Date:	Mon, 19 Apr 2021	Prob (F-statistic):	0
Time:	00:08:09	Log-Likelihood:	-2.21E+05
No. Observations:	26228	AIC:	4.41E+05
Df Residuals:	26214	BIC:	4.41E+05
Df Model:	13		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	1796.3002	888.774	2.021	0.043	54.255	3538.345
carat	9753.8466	83.528	116.773	0	9590.126	9917.567
cut	163.1454	7.42	21.988	0	148.603	177.688
depth	22.0613	12.644	1.745	0.081	-2.721	46.844
table	-31.0174	3.983	-7.788	0	-38.823	-23.211
x	-2431.6106	123.338	-19.715	0	-2673.36	-2189.861
y	2487.4534	122.417	20.32	0	2247.51	2727.397
z	-1652.8745	168.247	-9.824	0	-1982.647	-1323.102
color_E	-87.4234	24.739	-3.534	0	-135.914	-38.933
color_F	-46.3618	24.964	-1.857	0.063	-95.292	2.569
color_G	-78.0962	24.135	-3.236	0.001	-125.402	-30.79
color_H	-645.5286	26.026	-24.803	0	-696.542	-594.515
color_I	-1037.5961	29.029	-35.743	0	-1094.495	-980.697
color_J	-1589.7551	35.635	-44.612	0	-1659.602	-1519.908

Omnibus:	6528.028	Durbin-Watson:	2
Prob(Omnibus):	0	Jarque-Bera (JB):	42237.238
Skew:	1.039	Prob(JB):	0
Kurtosis:	8.859	Cond. No.	1.14E+04

- Basis above model we can that adjusted r square is 90.1%. Which seems to be good but dropping compared to above model 1. Also Checking the p values al the p values seems to significant variables. Except variable depth & color_F. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to higher side which in turn shows that there is multi-collinearity.
- Let's check the VIF values.

	variables	VIF
0	carat	123.831937
1	cut	10.28869
2	depth	1232.791717
3	table	890.994017
4	x	10622.65636
5	y	9268.255154
6	z	3581.528405
7	color_E	2.467635
8	color_F	2.414616
9	color_G	2.704453
10	color_H	2.268737
11	color_I	1.894213
12	color_J	1.494441

- Still we see that there are variables with VIF value above 5. Let's keep on dropping the variable above VIF 5.

Model 3 (Model with Dropping high infinity VIF values (i.e. clarity & cut)).

- In third model will run the model using all the variables (i.e. price~carat+depth+table+x+y+z+color_E+color_F+color_G+color_H+color_I+color_J).

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.899
Model:	OLS	**Adj. R-squared:**	0.899
Method:	Least Squares	**F-statistic:**	1.95E+04
Date:	Mon, 19 Apr 2021	**Prob (F-statistic):**	0
Time:	00:08:11	**Log-Likelihood:**	-2.21E+05
No. Observations:	26228	**AIC:**	4.42E+05
Df Residuals:	26215	**BIC:**	4.42E+05
Df Model:	12		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	8791.4438	837.49	10.497	0	7149.918	1.04E+04
carat	9799.1081	84.268	116.285	0	9633.938	9964.278
depth	-38.3009	12.455	-3.075	0.002	-62.714	-13.888
table	-77.7239	3.4	-22.862	0	-84.387	-71.06
x	-2240.5675	124.159	-18.046	0	-2483.925	-1997.21
y	2244.2626	123.033	18.241	0	2003.111	2485.414
z	-1609.9902	169.776	-9.483	0	-1942.761	-1277.219
color_E	-91.2106	24.965	-3.654	0	-140.144	-42.277
color_F	-53.2113	25.191	-2.112	0.035	-102.586	-3.836
color_G	-69.2001	24.353	-2.842	0.004	-116.933	-21.467
color_H	-643.4629	26.265	-24.499	0	-694.943	-591.983
color_I	-1029.116	29.293	-35.132	0	-1086.532	-971.701
color_J	-1594.7591	35.961	-44.347	0	-1665.245	-1524.274

Omnibus:	6421.403	**Durbin-Watson:**	2.005
Prob(Omnibus):	0	**Jarque-Bera (JB):**	43302.801
Skew:	1.008	**Prob(JB):**	0
Kurtosis:	8.963	**Cond. No.**	1.06E+04

- Basis above model we can that adjusted r square is 89.99%. Which seems to be good but dropping compared to above model 2. Also Checking the p values al the p values seems to significant variables. Except variable color_F. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to higher side which in turn shows that there is multi-collinearity.
- Let's check the VIF values.

	variables	VIF
0	carat	123.445387
1	table	717.346755
2	depth	995.217448
3	x	10258.65012
4	y	9220.487106
5	z	2970.85611
6	color_E	2.467627
7	color_F	2.414572
8	color_G	2.70249
9	color_H	2.26844
10	color_I	1.893794
11	color_J	1.49442

- Still we see that there are variables with VIF value above 5. Let's keep on dropping the variable above VIF 5.

Model 4 (Model with Dropping high infinity VIF values (i.e. clarity, cut & Depth).

- In fourth model will run the model using all the variables (i.e. price~carat+table+x+y+z+color_E+color_F+color_G+color_H+color_I+color_J).

OLS Regression Results

Dep. Variable:	price	R-squared:	0.899			
Model:	OLS	Adj. R-squared:	0.899			
Method:	Least Squares	F-statistic:	2.13E+04			
Date:	Mon, 19 Apr 2021	Prob (F-statistic):	0			
Time:	00:08:13	Log-Likelihood:	-2.21E+05			
No. Observations:	26228	AIC:	4.42E+05			
Df Residuals:	26216	BIC:	4.42E+05			
Df Model:	11					
Covariance Type:	nonrobust					
	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	6317.9942	233.308	27.08	0	5860.698	6775.29
carat	9783.7786	84.134	116.288	0	9618.871	9948.686
table	-76.2062	3.364	-22.652	0	-82.8	-69.612
x	-2108.3495	116.495	-18.098	0	-2336.686	-1880.013
y	2401.6581	111.899	21.463	0	2182.33	2620.986
z	-2069.6543	80.511	-25.706	0	-2227.46	-1911.848
color_E	-90.563	24.968	-3.627	0	-139.502	-41.624
color_F	-52.8264	25.194	-2.097	0.036	-102.209	-3.444
color_G	-68.4754	24.355	-2.811	0.005	-116.214	-20.737
color_H	-643.4703	26.269	-24.495	0	-694.959	-591.982
color_I	-1030.6601	29.293	-35.184	0	-1088.076	-973.244
color_J	-1595.8035	35.965	-44.371	0	-1666.297	-1525.31
Omnibus:	6424.709	Durbin-Watson:	2.004			
Prob(Omnibus):	0	Jarque-Bera (JB):	43695.197			
Skew:	1.006	Prob(JB):	0			
Kurtosis:	8.995	Cond. No.	2.07E+03			

- Basis above model we can that adjusted r square is 89.99%. Which seems to be good but dropping compared to above model 3. Also Checking the p values al the p values seems to significant variables. Except variable color_F. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to higher side which in turn shows that there is multi-collinearity.
- Let's check the VIF values.

	variables	VIF
0	carat	86.28756
1	table	266.136814
2	x	9980.010927
3	y	9194.625567

4	z	1571.724914
5	color_E	2.46172
6	color_F	2.407718
7	color_G	2.687687
8	color_H	2.254623
9	color_I	1.87998
10	color_J	1.489809

- Still we see that there are variables with VIF value above 5. Let's keep on dropping the variable above VIF 5.

Model 5 (Model with Dropping high infinity VIF values (i.e. clarity, cut, Depth & table)).

- In Fifth model will run the model using the variables (i.e.

  price~carat+x+y+z+color_E+color_F+color_G+color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.897
Method:	Least Squares	**F-statistic:**	2.29E+04
Date:	Mon, 19 Apr 2021	**Prob (F-statistic):**	0
Time:	00:08:14	**Log-Likelihood:**	-2.21E+05
No. Observations:	26228	**AIC:**	4.42E+05
Df Residuals:	26217	**BIC:**	4.42E+05
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	1987.0167	135.004	14.718	0	1722.401	2251.632
carat	9716.7723	84.899	114.451	0	9550.365	9883.179
x	-2557.8977	115.908	-22.068	0	-2785.083	-2330.712
y	2546.989	112.801	22.58	0	2325.893	2768.085
z	-1575.7656	78.256	-20.136	0	-1729.151	-1422.38
color_E	-100.7475	25.207	-3.997	0	-150.155	-51.34
color_F	-50.6219	25.439	-1.99	0.047	-100.484	-0.76
color_G	-56.7776	24.587	-2.309	0.021	-104.969	-8.586
color_H	-635.9057	26.522	-23.976	0	-687.891	-583.921
color_I	-1023.4693	29.576	-34.604	0	-1081.44	-965.498
color_J	-1599.9764	36.314	-44.059	0	-1671.154	-1528.798

Omnibus:	6469.011	**Durbin-Watson:**	2
Prob(Omnibus):	0	**Jarque-Bera (JB):**	42862.732
Skew:	1.022	**Prob(JB):**	0
Kurtosis:	8.92	**Cond. No.**	217

- Basis above model we can that adjusted r square is 89.7%. Which seems to be dropping down compared to above models. Also Checking the p values al the p values seems to significant variables. Except variable color_F & color_G. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to be decreasing compared to above models.

- Let's check the VIF values.

	variables	VIF
0	carat	11.324381
1	x	9570.4914
2	y	9193.516103
3	z	1564.980819
4	color_E	2.433057
5	color_F	2.394265
6	color_G	2.671584
7	color_H	2.23459
8	color_I	1.860607
9	color_J	1.478309

- Still we see that there are variables with VIF value above 5. Let's keep on dropping the variable above VIF 5.

Model 6 Model with Dropping high infinity VIF values (i.e. clarity, cut, Depth, table & z Variable).

- In Sixth model will run the model using the variables (i.e. price~carat+x+y+color_E+color_F+color_G+color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.896			
Model:	OLS	**Adj. R-squared:**	0.896			
Method:	Least Squares	**F-statistic:**	2.50E+04			
Date:	Mon, 19 Apr 2021	**Prob (F-statistic):**	0			
Time:	00:08:16	**Log-Likelihood:**	-2.21E+05			
No. Observations:	26228	**AIC:**	4.43E+05			
Df Residuals:	26218	**BIC:**	4.43E+05			
Df Model:	9					
Covariance Type:	nonrobust					

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	1242.7709	130.844	9.498	0	986.31	1499.232
carat	9245.5702	82.238	112.425	0	9084.38	9406.76
x	-3005.7346	114.628	-26.222	0	-3230.412	-2781.057
y	2217.3416	112.464	19.716	0	1996.906	2437.777
color_E	-96.68	25.4	-3.806	0	-146.465	-46.895
color_F	-49.9043	25.634	-1.947	0.052	-100.149	0.341
color_G	-61.0586	24.775	-2.465	0.014	-109.618	-12.499
color_H	-641.9958	26.724	-24.023	0	-694.377	-589.615
color_I	-1025.6703	29.803	-34.415	0	-1084.086	-967.254
color_J	-1606.3712	36.592	-43.9	0	-1678.093	-1534.649

Omnibus:	6155.626	**Durbin-Watson:**	2.002
Prob(Omnibus):	0	**Jarque-Bera (JB):**	40429.222
Skew:	0.968	**Prob(JB):**	0
Kurtosis:	8.766	**Cond. No.**	196

- Basis above model we can that adjusted r square is 89.6%. Which seems to be dropping down compared above models. Also Checking the p values al the p values seems to significant variables. Except variable color_F & color_G. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to be decreasing compared to above models.
- Let's check the VIF values.

	variables	VIF
0	carat	11.296452
1	x	9026.82853
2	y	8939.023996
3	color_E	2.431451
4	color_F	2.392727
5	color_G	2.667254
6	color_H	2.229572
7	color_I	1.857403
8	color_J	1.476089

- Still we see that there are variables with VIF value above 5. Let's keep on dropping the variable above VIF 5.

Model 7 Model with Dropping high infinity VIF values (i.e. clarity, cut, Depth, table, z & x Variable Variable).

- In seventh model will run the model using the variables (i.e. price~carat+y+color_E+color_F+color_G+color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	R-squared:	0.893
Model:	OLS	Adj. R-squared:	0.893
Method:	Least Squares	F-statistic:	2.73E+04
Date:	Mon, 19 Apr 2021	Prob (F-statistic):	0
Time:	00:08:17	Log-Likelihood:	-2.22E+05
No. Observations:	26228	AIC:	4.43E+05
Df Residuals:	26219	BIC:	4.44E+05
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	581.4979	130.061	4.471	0	326.572	836.424
carat	8744.9107	81.031	107.921	0	8586.086	8903.736
y	-603.0052	33.282	-18.118	0	-668.239	-537.771
color_E	-94.0882	25.73	-3.657	0	-144.521	-43.656
color_F	-49.9081	25.968	-1.922	0.055	-100.807	0.99
color_G	-62.5366	25.097	-2.492	0.013	-111.728	-13.345
color_H	-649.9867	27.07	-24.011	0	-703.045	-596.928
color_I	-1009.2519	30.184	-33.436	0	-1068.415	-950.089
color_J	-1584.7831	37.059	-42.764	0	-1657.42	-1512.146

Omnibus:	5749.28	Durbin-Watson:	1.998
Prob(Omnibus):	0	Jarque-Bera (JB):	37618.064
Skew:	0.895	Prob(JB):	0
Kurtosis:	8.587	Cond. No.	133

- Basis above model we can that adjusted r square is 89.3%. Which seems to be dropping down compared above models. Also Checking the p values all the p values seems to significant variables. Except variable color_F & color_G. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to be decreasing compared to above models.
- Let's check the VIF values.

	variables	VIF
0	carat	11.001789
1	y	22.930637
2	color_E	2.430476
3	color_F	2.391847
4	color_G	2.665555
5	color_H	2.226383
6	color_I	1.857345
7	color_J	1.476087

- Still we see that there are variables with VIF value above 5. Let's keep on dropping the variable above VIF 5.

Model 8 - Model with Dropping high infinity VIF values (i.e. clarity, cut, Depth, table, z & y Variable Variable).

- In eighth model will run the model using the variables (i.e. price~carat+color_E+color_F+color_G+color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	R-squared:	0.892
Model:	OLS	Adj. R-squared:	0.892
Method:	Least Squares	F-statistic:	3.08E+04
Date:	Mon, 19 Apr 2021	Prob (F-statistic):	0
Time:	00:08:18	Log-Likelihood:	-2.22E+05
No. Observations:	26228	AIC:	4.44E+05
Df Residuals:	26220	BIC:	4.44E+05
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1739.6345	22.58	-77.042	0	-1783.893	-1695.376
carat	7305.419	16.028	455.78	0	7274.003	7336.836
color_E	-91.6368	25.89	-3.539	0	-142.382	-40.891
color_F	-56.6515	26.127	-2.168	0.03	-107.861	-5.442
color_G	-60.9168	25.253	-2.412	0.016	-110.414	-11.42
color_H	-631.9695	27.22	-23.217	0	-685.322	-578.617
color_I	-980.9674	30.331	-32.342	0	-1040.419	-921.516
color_J	-1555.7648	37.254	-41.761	0	-1628.785	-1482.744

Omnibus:	5175.628	Durbin-Watson:	1.998
Prob(Omnibus):	0	Jarque-Bera (JB):	27366.15
Skew:	0.854	Prob(JB):	0
Kurtosis:	7.704	Cond. No.	11

- Basis above model we can that adjusted r square is 89.2%. Which seems to be coming down from above models. Also Checking the p values al the p values seems to significant variables. Except variable color_G. Let's check p values significance once we drop all the variable with high VIF Values. The condition number seems to be decreasing compared to above models.
- Let's check the VIF values.

	variables	VIF
0	carat	3.405663
1	color_E	1.318681
2	color_F	1.377663
3	color_G	1.50346
4	color_H	1.499119
5	color_I	1.429428
6	color_J	1.277313

- Now we see that all the variables are within VIF value of 5. Let's stop dropping.

Let's Check features based on high P Value

For the $t\text{-}statistic$ for every co-efficient of the Linear Regression the null and alternate Hypothesis is as follows:

$H0$: The variable is significant.

$H1$: The variable is not significant.

Lower the p-value for the t-statistic more significant are the variables.

Model 9 - Model with Dropping insignificant P Values (i.e. clarity, cut, Depth, table, z, x, y & color_F Variable).

- In Ninth model will run the model using the variables (i.e. price~carat+color_E+color_G+color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	R-squared:	0.892
Model:	OLS	Adj. R-squared:	0.892
Method:	Least Squares	F-statistic:	3.59E+04
Date:	Tue, 20 Apr 2021	Prob (F-statistic):	0
Time:	17:29:49	Log-Likelihood:	-2.22E+05
No. Observations:	26228	AIC:	4.44E+05
Df Residuals:	26221	BIC:	4.44E+05
Df Model:	6		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1771.7235	17.056	-103.878	0	-1805.154	-1738.293
carat	7303.904	16.014	456.087	0	7272.515	7335.293
color_E	-58.5533	20.918	-2.799	0.005	-99.553	-17.553
color_G	-27.6629	20.064	-1.379	0.168	-66.99	11.664
color_H	-598.5149	22.427	-26.688	0	-642.472	-554.557
color_I	-947.3319	26.067	-36.343	0	-998.424	-896.24
color_J	-1521.9582	33.837	-44.979	0	-1588.281	-1455.635

Omnibus:	5172.729	Durbin-Watson:	1.998
Prob(Omnibus):	0	Jarque-Bera (JB):	27456.912
Skew:	0.852	Prob(JB):	0
Kurtosis:	7.714	Cond. No.	7.47

- Basis above model we can that adjusted r square is 89.2%. Which seems to be dropping down compared above models.
- Let's check the VIF values for the all the variables except clarity variable.

variables	VIF	
0	carat	2.472058
1	color_E	1.23132
2	color_G	1.365445
3	color_H	1.362294
4	color_I	1.311707
5	color_J	1.201292

Model 10- Model with Dropping insignificant P Values (i.e clarity, cut, Depth, table, z, x, y, color_F & color_G Variable).

- In Tenth model will run the model using the variables (i.e. price~carat+color_E+color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.892
Model:	OLS	**Adj. R-squared:**	0.892
Method:	Least Squares	**F-statistic:**	4.31E+04
Date:	Tue, 20 Apr 2021	**Prob (F-statistic):**	0
Time:	17:29:51	**Log-Likelihood:**	-2.22E+05
No. Observations:	26228	**AIC:**	4.44E+05
Df Residuals:	26222	**BIC:**	4.44E+05
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1782.2346	15.257	-116.812	0	-1812.14	-1752.329
carat	7302.681	15.99	456.704	0	7271.34	7334.022
color_E	-47.2394	19.241	-2.455	0.014	-84.953	-9.526
color_H	-586.9013	20.785	-28.237	0	-627.641	-546.161
color_I	-935.5723	24.632	-37.982	0	-983.852	-887.292
color_J	-1510.0606	32.719	-46.153	0	-1574.191	-1445.93

Omnibus:	5187.59	**Durbin-Watson:**	1.998
Prob(Omnibus):	0	**Jarque-Bera (JB):**	27641.47
Skew:	0.854	**Prob(JB):**	0
Kurtosis:	7.731	**Cond. No.**	6.5

- Basis above model we can that adjusted r square is 89.2%. Which seems to be coming down from above models in decimals.
- Let's check the VIF values.

	variables	VIF
0	carat	1.810442
1	color_E	1.16941
2	color_H	1.26533
3	color_I	1.228283
4	color_J	1.147419

Model 11- Model with Dropping insignificant P Values (i.e clarity, cut, Depth, table, z, x, y, color_F & color_G Variable & color_E)).

- In Eleventh model will run the model using the variables (i.e. price~carat+ color_H+color_I+color_J)

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.892			
Model:	OLS	**Adj. R-squared:**	0.892			
Method:	Least Squares	**F-statistic:**	5.39E+04			
Date:	Thu, 22 Apr 2021	**Prob (F-statistic):**	0			
Time:	00:09:46	**Log-Likelihood:**	-2.22E+05			
No. Observations:	26228	**AIC:**	4.44E+05			
Df Residuals:	26223	**BIC:**	4.44E+05			
Df Model:	4					
Covariance Type:	nonrobust					
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	-1796.3443	14.135	-127.086	0	-1824.049	-1768.639
carat	7305.0189	15.963	457.618	0	7273.73	7336.308
color_H	-574.8991	20.204	-28.455	0	-614.5	-535.298
color_I	-923.8492	24.167	-38.228	0	-971.218	-876.48
color_J	-1498.6014	32.387	-46.271	0	-1562.082	-1435.12
Omnibus:	5183.635	**Durbin-Watson:**	1.999			
Prob(Omnibus):	0	**Jarque-Bera (JB):**	27534.079			
Skew:	0.854	**Prob(JB):**	0			
Kurtosis:	7.72	**Cond. No.**	6.27			

- Basis above model we can that adjusted r square is 89.2%. Which seems to be coming down from above models in decimals.
- Let's check the VIF values.

	variables	VIF
0	carat	1.548167
1	color_H	1.226893
2	color_I	1.195212
3	color_J	1.126063

Model Evaluation.

model_name	model_perf	Adjusted R square
0	All Variables	0.93322
1	Dropping clarity	0.90105
2	Dropping clarity & cut	0.899229
3	Dropping clarity, cut & depth	0.899196
4	Dropping clarity, cut, Depth & table	0.897227
5	Dropping clarity, cut, Depth, table & z	0.895642
6	Dropping clarity, cut, Depth, table, z & x	0.892909
7	Dropping clarity, cut, Depth, table, z, x & y	0.891572
8	Dropping clarity, cut, Depth, table, z, x, y & color_F	0.891557
9	Dropping clarity, cut, Depth, table, z, x, y, color_F & color_G	0.891553
10	Dropping clarity, cut, Depth, table, z, x, y, color_F & color_G Variable & color_E	0.891533

Inference – Basis above iterations we can see that model No 10 seems to be giving decent results compared to model 1 (which is including all the variables). Also Model no 11 is the model free from multi-collinearity & following all the assumptions check.

Basis Above descriptive linear regression we can say that below variables are the important variables in predicting the prices of Cubic zirconia.

- carat
- color_H
- color_I
- color_J

- We will use Model 1 and Model 11 to predict and check the model evaluation.
- We Have chosen Model 11 because, it has a high Adjusted R Square, with least number of features. Also, for comparison we have taken model no 1. Which includes all the variables.

Model 1 Predictions & Model 11 Predictions

Figure 20

Distplot of Residuals for Model 1 & Model 11

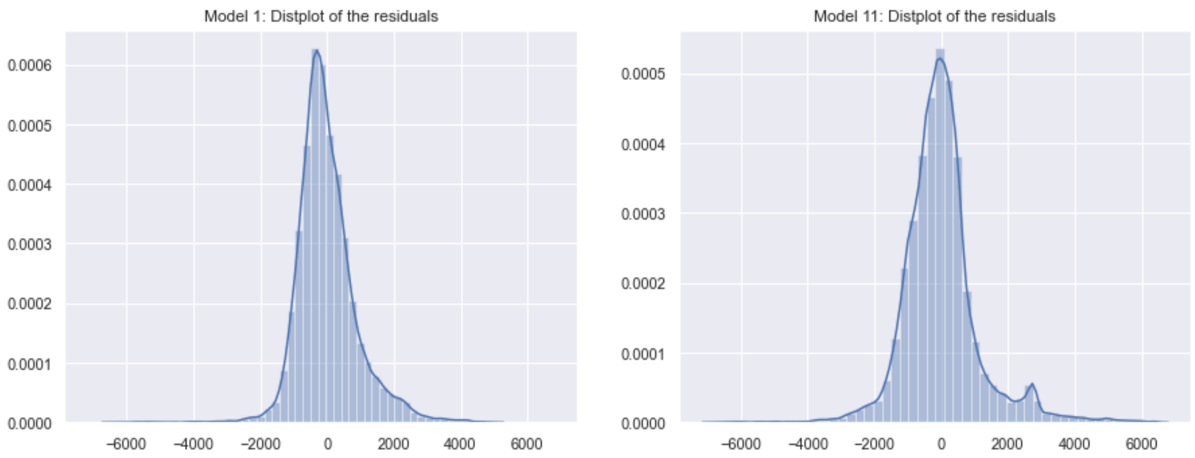


Figure 21

Boxplot of residuals for Model 1 & Model 11.

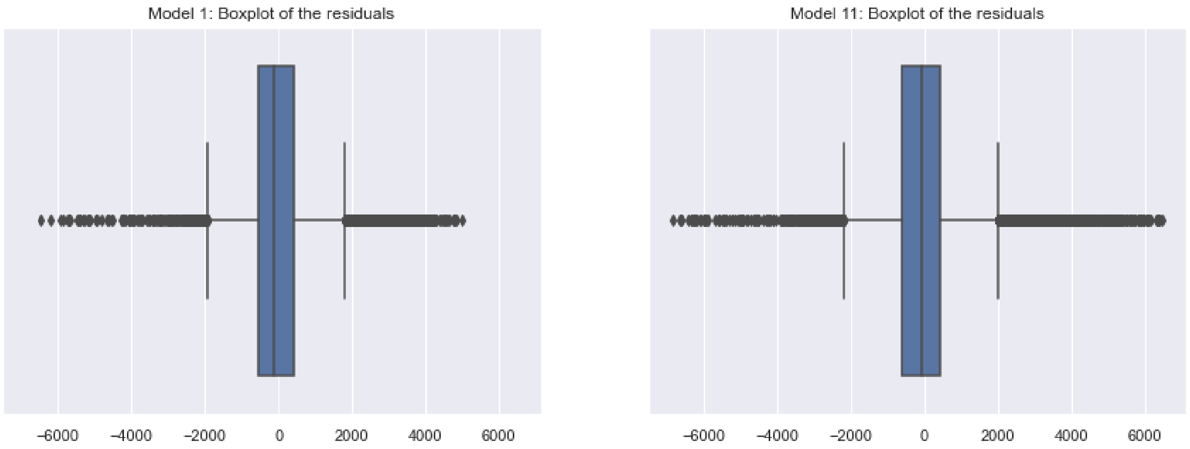


Figure 22

RMSE Scores for Model 1 & 11

	RMSE Score
Model 1	895.0999417
Model 11	1140.985589

Predictive Linear Regression

- Predictive Linear Regression – Main Objective of Predictive Linear Regression is predictive values for Features / Variables.
- In predictive Linear Regression we no need to look after assumptions. Whereas for predictive type assumptions are not important only metric is important.
- For predictive type we divide the dataset into train and test. Also, We Use sklearn & statmodel for as python library for predictive Linear Regression.

Predictive Approach using the Models 1, Model 10 & Model 11.

- from sklearn.linear_model import LinearRegression
- Splitting the data into the dependent and independent variables.
- from sklearn.model_selection import train_test_split
- Splitting the data into train (70%) and test (30%).
- Using only Model 1 variables to build the model on the training data and predict on the training as well as test data. Results for the model are mentioned below.
- Using only Model 10 variables to build the model on the training data and predict on the training as well as test data. Results for the model are mentioned below.
- Using only Model 11 variables to build the model on the training data and predict on the training as well as test data. Results for the model are mentioned below.

Model 1, Model 10 & Model 11 Train Prediction Scattered plot.

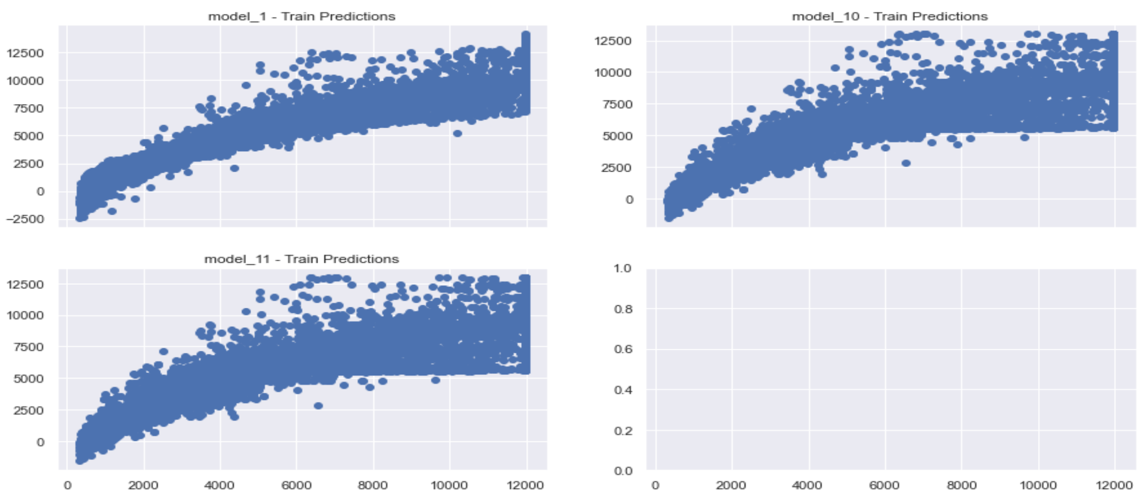


Figure 23

Model 1, Model 10 & Model 11 Test Prediction Scattered plot.

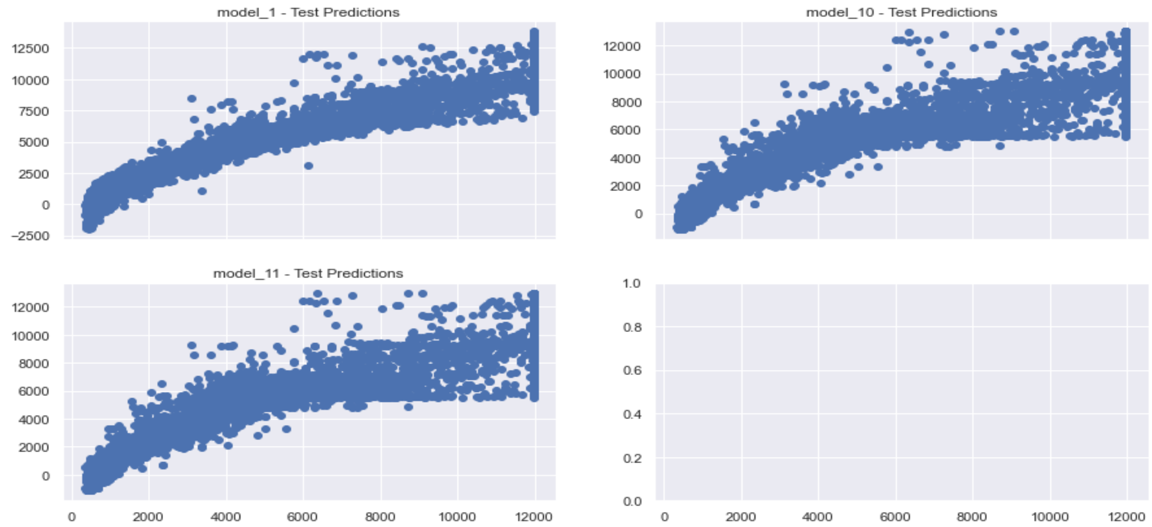


Figure 24

Model Results for Train & Test Dataset.

A good fitting model is one where the difference between the actual and observed values or predicted values for the selected model is small and unbiased for train, validation and test data sets.

There are three error metrics that are commonly used for evaluating and reporting the performance of a regression model:

- **Mean Squared Error (MSE).**
- **Root Mean Squared Error (RMSE).**
- **Mean Absolute Error (MAE)**

Here in the problem we are using RMSE Score as Metric -

- The most commonly used metric for regression tasks is **RMSE (root-mean-square error)**. This is defined as the square root of the average squared distance between the actual score and the predicted score:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- The RMSE is the square root of the variance of the residuals. It indicates the absolute fit of the model to the data–how close the observed data points are to the model's predicted values. Whereas R-squared is a relative measure of fit, RMSE is an absolute measure of fit. As the square root of a variance, RMSE can be interpreted as the standard deviation of the unexplained variance and has the useful property of being in the same units as the response variable. Lower values of RMSE indicate better fit. RMSE is a good measure of how accurately the model predicts the response, and it is the most important criterion for fit if the main purpose of the model is prediction.

	RMSE of training data	RMSE of test data
All Variables	892.696002	901.24668
Droppping clarity, Depth, table y, z, x, cut, color_F & color_G Variable	1138.301485	1146.979655
Dropping clarity, Depth, table y, z, x, cut, color_F, color_G & color_E Variable	1138.367736	1147.223326

Inference – Basis Model 1 we can see that it has a lowest RMSE Score for Train & test data when compared to the other two models (i.e. Model 10 & Model 11). Lower values of RMSE for Train & test data indicate better fit. RMSE is a good measure of how accurately the model predicts the response, and it is the most important criterion for fit if the main purpose of the model is prediction. The model 1 can be used when the prediction is important irrespective of checking which variables are important.

Whereas Model No 10 & Model 11 are the models which are free from multi-collinearity. Also, in case if client wants to understand which variables in dataset play an important role in changing the prices of cubic zirconia then we can go ahead with model 10 & Model No 11.

Problem 2 – Logistic Regression & Linear Discriminant Analysis (LDA)

Problem Statement -

You are hired by a sports analysis agency to understand the selection process of high school football players into college with a full or partial scholarship. You are provided details of 6215 high school graduates who have been inducted into 4-year degree colleges with either full or partial scholarships.

Objective

Objective is to help the agency in predicting whether a high school graduate will win a full scholarship on the basis of the information given in the data set. Also, find out the important factors which are instrumental in winning a full scholarship in colleges.

Football Scholarship Data

The data dictionary is given below.

1. Scholarship - Won a college scholarship: Full / Partial
2. Academic Score - High school academic performance of a candidate
3. Score on Plays Made - A composite score based on the achievements on the field
4. Missed Play Score - A composite score based on the failures on the field
5. Injury Propensity - This has 3 ordinal levels: High, Moderate, Normal and Low. It has been calculated based on what proportion of time a candidate had an injury problem
6. School Type - 3 types of schools based on their location
7. School Score - A composite score based on the overall achievement of the candidates' school, based on the school's academic, sports and community service performance
8. Overall Score - A composite score based on a candidate's family financial state, school performance, psychosocial attitude etc.
9. Region - Region of the country where the school is located.

Performing exploratory data analysis on the dataset. Showcasing some charts & graphs.

1. Loading the data set- We will be loading the "Football+Scholarship.csv" file using pandas library in python. For this, we will be using read_csv file.

2. The head function will tell us the top head records in the data set. By default, python shows you only the top 5 record.

	Academic_Score	Score_on_Plays_Made	Missed_Play_Score	Injury_Propensity	School_Type	School_Score	Overall_Score	Region	Scholarship
0	7	0.27	0.36	High	D	0.45	8.8	Eastern	Partial
1	6.3	0.3	0.34	Low	C	0.49	9.5	Eastern	Partial
2	8.1	0.28	0.4	Moderate	C	0.44	10.1	Eastern	Partial
3	7.2	0.23	0.32	Moderate	C	0.4	9.9	Eastern	Partial
4	7.2	0.23	0.32	Moderate	C	0.4	9.9	Eastern	Partial

3. The tail function will tell us the last entries records in the data set. By default, python shows you only the last 5 records. Let's check tail for the totals/subtotals if any. The Football+Scholarship data dataset doesn't contain any total/subtotals.

	Academic_Score	Score_on_Plays_Made	Missed_Play_Score	Injury_Propensity	School_Type	School_Score	Overall_Score	Region	Scholarship
6210	6.8	0.62	0.08	Low	C	0.82	9.5	Eastern	Full
6211	6.2	0.6	0.08	Low	C	0.58	10.5	Western	Full
6212	5.9	0.55	0.1	Low	C	0.76	11.2	Eastern	Full
6213	6.3	0.51	0.13	Low	C	0.75	11	Eastern	Full
6214	5.9	0.645	0.12	Low	C	0.71	10.2	Western	Full

4. The shape attribute tells us a number of observations and variables we have in the data set. It is used to check the dimension of data. The Football+Scholarship data set has 6215 observations and 9 variables in the data set.

5. info() is used to check the Information about the data and the datatypes of each respective attribute.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6215 entries, 0 to 6214
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Academic_Score       6215 non-null   float64
 1   Score_on_Plays_Made  6215 non-null   float64
 2   Missed_Play_Score    6215 non-null   float64
 3   Injury_Propensity    6215 non-null   object
 4   School_Type          6215 non-null   object
 5   School_Score         6215 non-null   float64
 6   Overall_Score        6215 non-null   float64
 7   Region               6215 non-null   object
 8   Scholarship          6215 non-null   object
dtypes: float64(5), object(4)
```

Figure 25

- Looking at the data in the head function and in info, we come to know that the variables comprise of float and object data types. sklearn in Python does not take the input of object data types while building Logistic Regression models & LDA Models. So, we need to convert these variables into some numerical form.

- Basis Figure 1 we can see that there are three object type variables (Injury_Propensity, School_Type, Region & Scholarship) which has the object data types which we need to convert into numerical form. Since the variable Scholarship & Injury_Propensity are in ordinal range, we are replacing the categorical variables with the numbers. Further we shall perform one hot encoding for School_Type & Region variable.
- Further basis above figure 1 we can see that there are 0 null values in the dataset.
- Also, it was identified that there were 947 duplicated rows in the dataset. Will go ahead and drop the duplicated values from the dataset. Post Dropping the duplicated values we are left out with 5268 Entries. We have lost approx. 15.24% of data.

6. The described method will help to see how data has been spread for numerical values. We can clearly see the minimum value, mean values, different percentile values, and maximum values for the Income data set.

	count	mean	std	min	25%	50%	75%	max
Academic_Score	5268	7.134045	1.075858	4.45	6.4	6.9	7.7	9.65
Score_on_Plays_Made	5268	0.331494	0.143066	0.08	0.23	0.29	0.4	0.655
Missed_Play_Score	5268	0.317783	0.135136	0.025	0.25	0.31	0.4	0.625
Injury_Propensity	5268	1.066439	1.132292	0	0	1	2	3
School_Score	5268	0.526795	0.129673	0.22	0.43	0.51	0.6	0.855
Overall_Score	5268	10.502685	1.168602	8	9.5	10.3	11.3	14
Scholarship	5268	0.366553	0.481909	0	0	0	1	1

Figure 26

7). Check Proportion of observations in each of the target classes (Scholarship Variable).

	Numbers	Percentage
Partial Scholarship (0)	3337	63%
Full Scholarship (1)	1931	37%

- Since 63% & 37% is balanced data. Hence, We Are not changing the threshold value / Probability value. We are going ahead considering it as 0.5 itself.

Exploratory Data Analysis

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations. We performed uni-variate and bi-variate analysis to get a better overview and to find outliers in our dataset. Outliers can occur due to some kind of errors while collecting the data and need to be removed so that it doesn't affect the performance of our model.

Uni-Variate Analysis.

Target / Predictor Variable Analysis - price.

Analyzing Feature: Scholarship

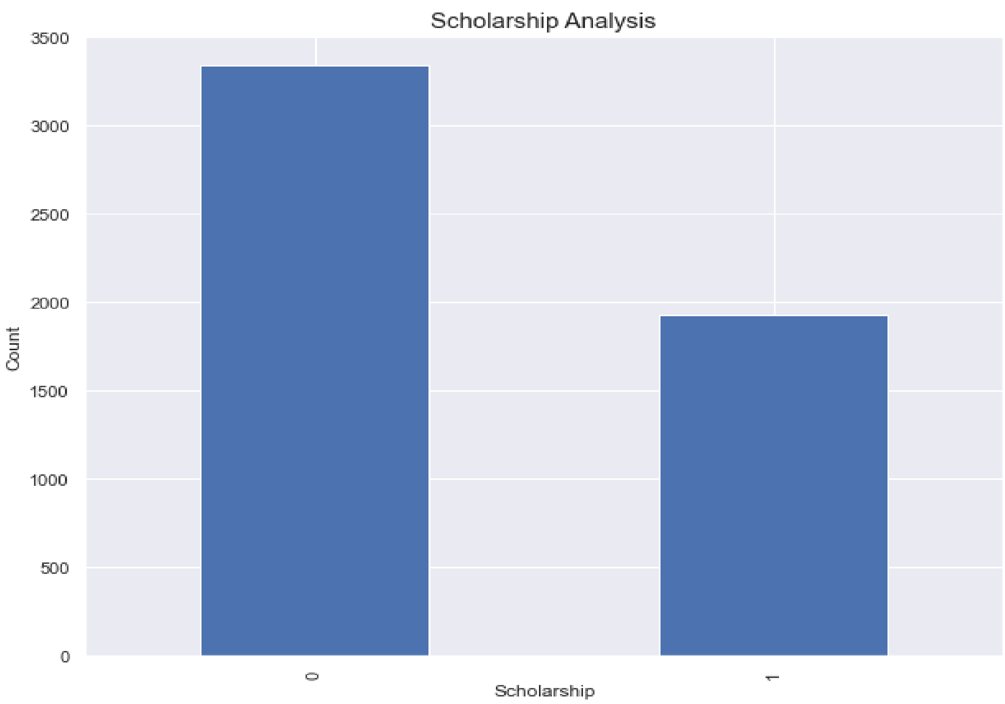


Figure 27

- Basis above figure we can see that 0 (Partial Scholarship) are high in number compared to 1 (Full Scholarship)

Response / Dependent Variable Analysis

Analyzing Feature: Academic_Score

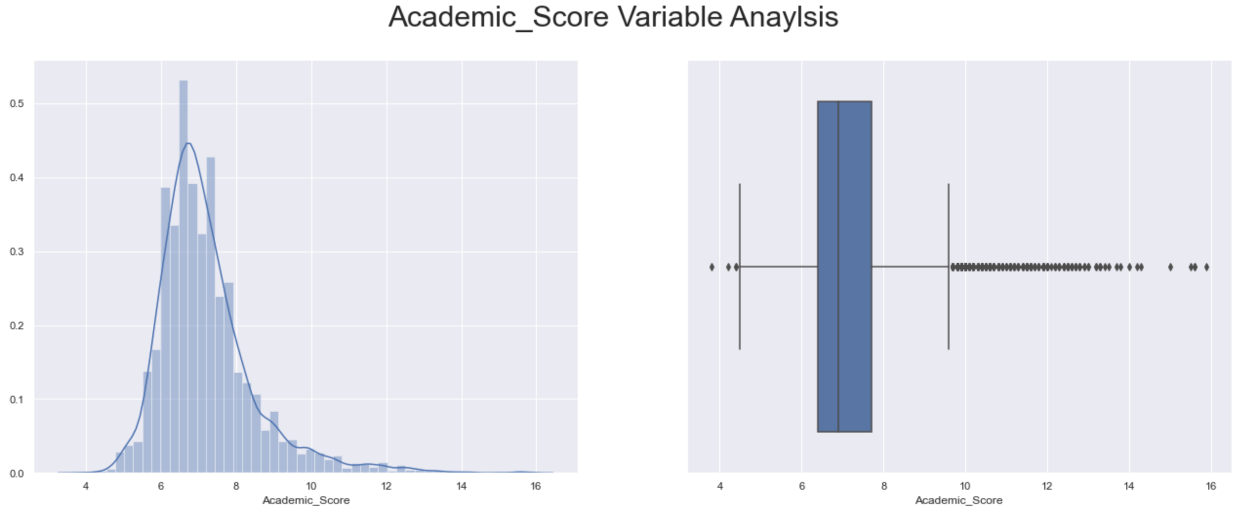


Figure 28

- The Academic_Score Variable distribution seems to be Right skewed.
- Basis skewness value we can see that distribution is highly skewed.
- Basis box plot we can see that there are outliers in the variable. Let's treat the outliers in the further process.

Analyzing Feature: Score_on_Plays_Made

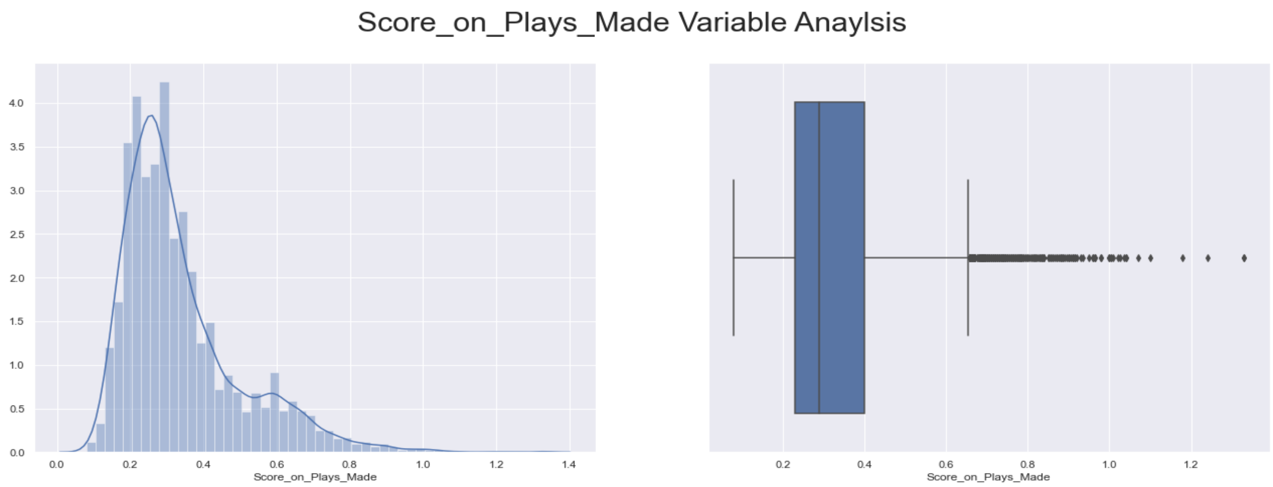


Figure 29

- The Score_on_Plays_Made Variable distribution seems to be Right skewed.
- Basis skewness value we can see that distribution is highly skewed

- Basis box plot we can see that there are outliers in the varaible. Lets treat the outliers in the further process.

Analyzing Feature: Missed_Play_Score

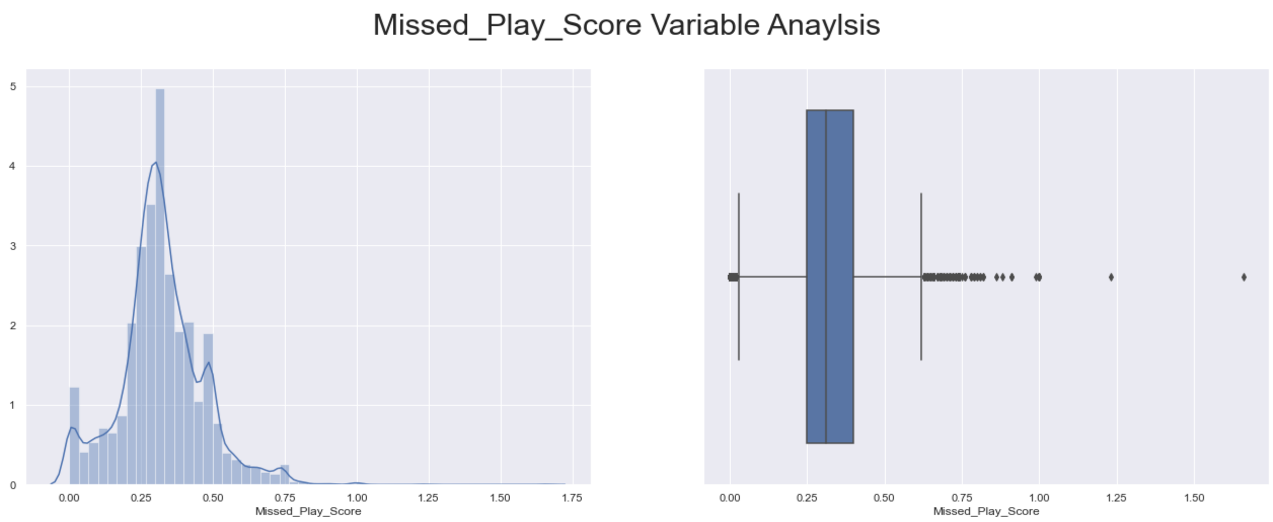


Figure 30

- The Missed_Play_Score Variable distribution seems to be Normally skewed.
- Basis skewness value we can see that distribution seems to be approximately symmetric
- Basis box plot we can see that there are outliers in the varaible. Lets treat the outliers in the further process.

Analyzing Feature: School_Score

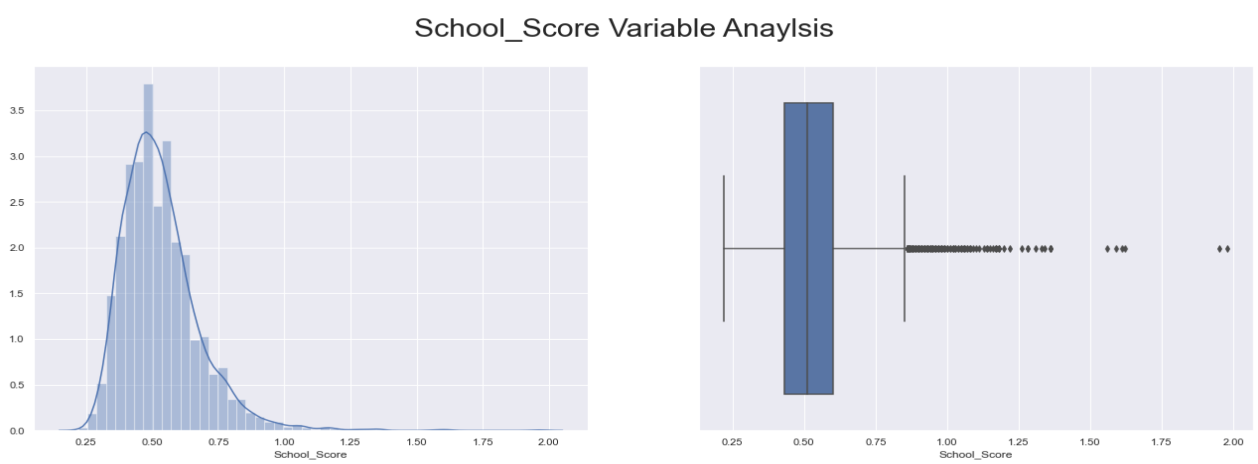


Figure 31

- The School_Score Variable distribution seems to be slightly right skewed.

- Basis skewness value we can see that distribution seems to be approximately symmetric
- Basis box plot we can see that there are outliers in the variable. Let's treat the outliers in the further process.

Analyzing Feature: Overall_Score

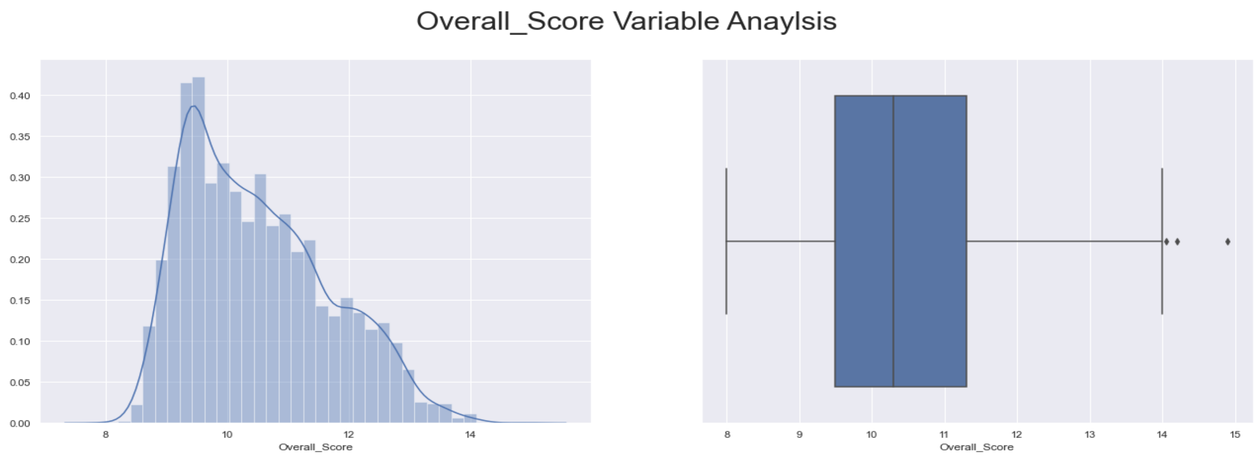


Figure 32

- The Overall_Score Variable distribution seems to be right skewed.
- Basis skewness value we can see that distribution seems to be approximately symmetric
- Basis box plot we can see that there are couple of outliers in the varaible. Lets treat the outliers in the further process.

Analyzing Feature: Injury_Propensity

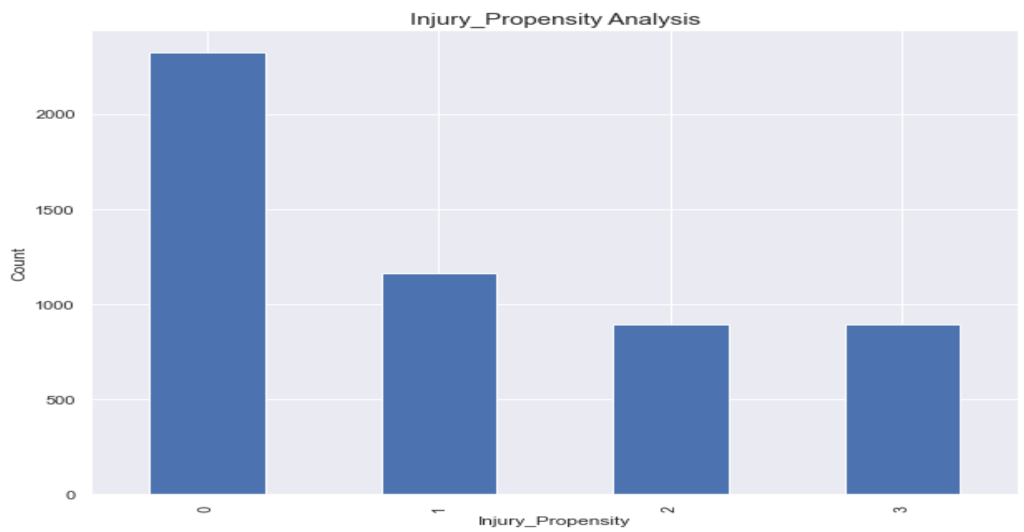


Figure 33

- Basis above figure we can see that there are many students with low (0) Injury_Propensity.
- Basis above figure we can see that there are low students with High (3) Injury_Propensity.

*5). **Analyzing Feature: School_Type***

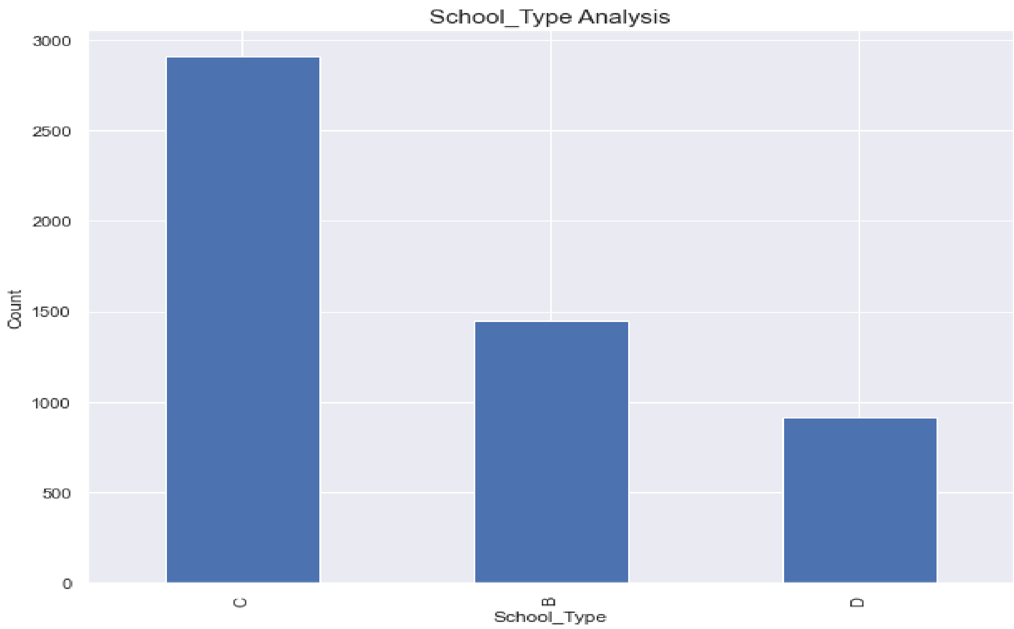


Figure 34

- Basis above figure we can see that 'C' Type school are more in number Followed by School Type 'B'.
- Basis above figure we can see that D' Type school are Low in number.

*6). **Analyzing Feature: Region***

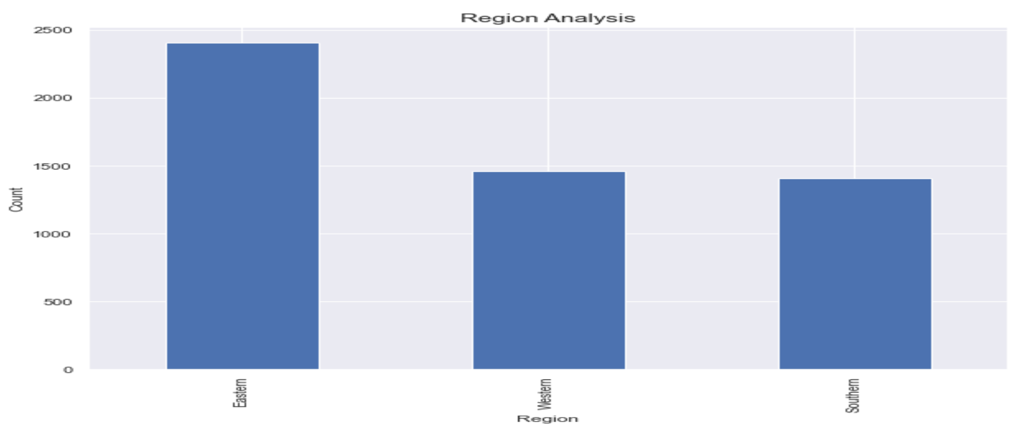


Figure 35

- Basis above figure we can see that there are many students from Easter Region.

Bi-Variate Analysis.

Analyzing Feature: scholarship Vs ACADEMIC_SCORE

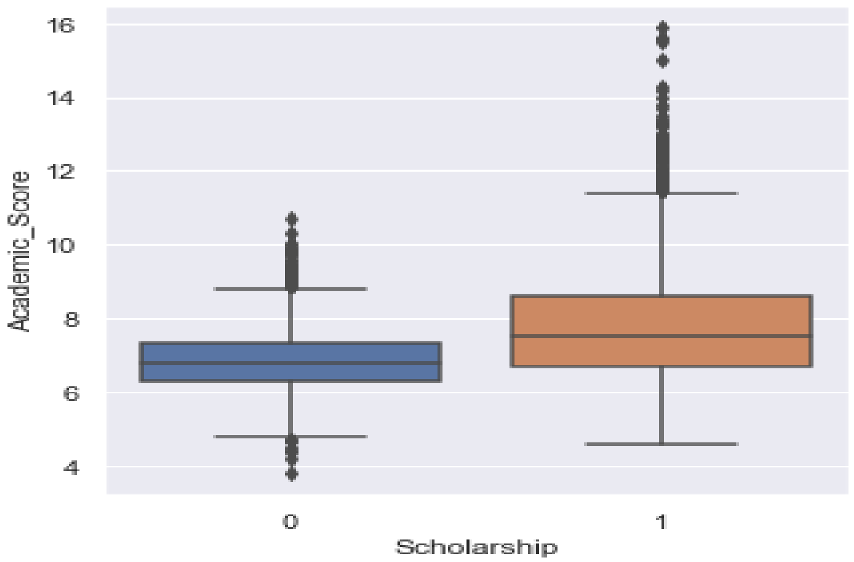


Figure 36

- Basis above figure we can see that students with high academic score have high chances to get full scholarship.
- Basis above figure we can also infer that as the academic score falls below 5 there are most of chances that he will not be eligible for full scholarship

Analyzing Feature: scholarship Vs INJURY_PROPENSITY

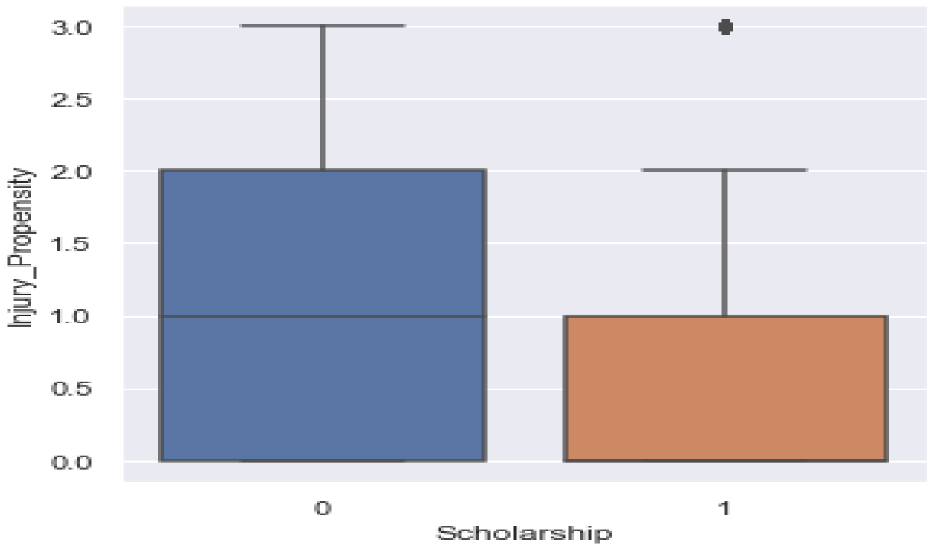


Figure 37

- Basis above figure we can see that students with INJURY_PROPENSITY have low chances to get full scholarship.

Analyzing Feature: scholarship Vs SCHOOL_SCORE

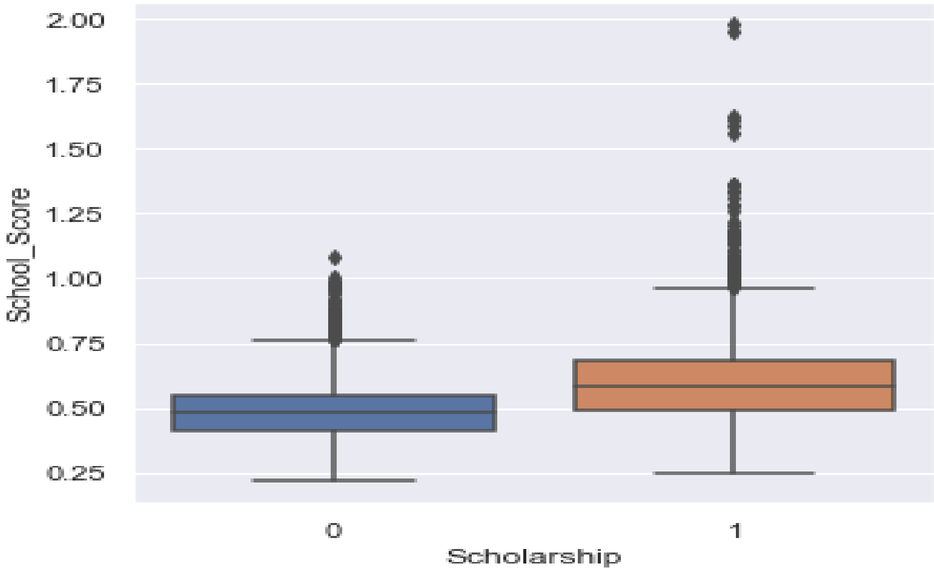


Figure 38

- Basis above figure we can see that students with school score above 1.5 have High chances to get full scholarship.

Data Pre-Processing

- Basis analysis of above we can see that there are outliers in the variables. Let's go ahead and treat the outliers.
- sklearn in Python does not take the input of object data types when building linear regression model. So, we need to convert these variables into some numerical form. We shall perform one hot encoding for them (i.e. colour variable). Post Data Pre-processing the head of data looks like

	Academic_Score	Score_on_Plays_Mad	Missed_Play_Score	Injury_Propensity	School_Score	Overall_Score	Scholarship	School_Type_C	School_Type_D	Region_Southern	Region_Western
0	7	0.27	0.36	3	0.45	8.8	0	0	1	0	0
1	6.3	0.3	0.34	0	0.49	9.5	0	1	0	0	0
2	8.1	0.28	0.4	2	0.44	10.1	0	1	0	0	0
3	7.2	0.23	0.32	2	0.4	9.9	0	1	0	0	0
6	6.2	0.32	0.16	2	0.47	9.6	1	1	0	0	0

- We are not scaling the dataset. And proceeded with dataset as it is.

Logistic Regression

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).

3. **Descriptive Logistic Regression**
4. **Predictive Logistic Regression**

Descriptive Logistic Regression

- Descriptive Logistic Regression – Main Objective of Descriptive Logistic Regression is to understand the relation between Features / Variables.
- In Descriptive Logistic Regression we need to look after assumptions i.e. VIF Values (for Multicollinearity). And we select variables basis significance of p value. Whereas for descriptive type assumptions and metric both stand important.
- For Descriptive type we don't divide the dataset into train and test. Also, We Use statmodel for as python library for Descriptive Logistic Regression.

Firstly, we will use descriptive logistic regression to understand which all variables are significant variables that impact the scholarship variable.

Descriptive logistic regression Models

Model 1 (Using all the variables)

- Firstly, we will import statsmodels.formula.api as SM model
- In first model will run the model using all the variables (i.e. 'Scholarship~Academic_Score+Score_on_Plays_Made+Missed_Play_Score+School_Score+Overall_Score+Injury_Propensity+School_Type_C+School_Type_D+Region_Southern+Region_Western').

Logit Regression Results

Dep. Variable:	Scholarship	No. Observations:	5268
Model:	Logit	Df Residuals:	5257
Method:	MLE	Df Model:	10
Date:	Mon, 19 Apr 2021	Pseudo R-squ.:	0.3240
Time:	12:28:53	Log-Likelihood:	-2340.1
converged:	True	LL-Null:	-3461.6
Covariance Type:	nonrobust	LLR p-value:	0.000

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-9.0594	0.551	-16.430	0.000	-10.140	-7.979
Academic_Score	0.4294	0.044	9.825	0.000	0.344	0.515
Score_on_Plays_Made	5.1342	0.322	15.931	0.000	4.503	5.766
Missed_Play_Score	-1.4835	0.345	-4.303	0.000	-2.159	-0.808
School_Score	2.8561	0.312	9.159	0.000	2.245	3.467
Overall_Score	0.2083	0.044	4.705	0.000	0.122	0.295
Injury_Propensity	-0.5544	0.043	-12.869	0.000	-0.639	-0.470
School_Type_C	1.3066	0.125	10.494	0.000	1.063	1.551
School_Type_D	2.3399	0.204	11.443	0.000	1.939	2.741
Region_Southern	-0.4837	0.091	-5.297	0.000	-0.663	-0.305
Region_Western	0.0227	0.092	0.248	0.804	-0.157	0.202

adj_pseudo_r2 = (model.llf-model.df_model)/model.llnull

- adj_pseudo_r2 = 0.32108239111187487
- Basis above model we can see that adjusted pseudo r square is 32.1%, this value cannot be read individually since it does not signify the model performance, but it can be used to compare different models.
- Let's check the VIF values for the all the variables to identify multi-collinearity among the independent variables.

Variables	VIF Value
Academic_Score VIF	= 1.68
Score_on_Plays_Made VIF	= 1.61
Missed_Play_Score VIF	= 1.52
Injury_Propensity VIF	= 1.73
School_Score VIF	= 1.29
Overall_Score VIF	= 1.97
School_Type_C VIF	= 2.92
School_Type_D VIF	= 4.12
Region_Southern VIF	= 1.23
Region_Western VIF	= 1.25

Considering the threshold value of 2 for VIF, we go ahead and drop the variable School_Type_D which has the maximum VIF value of 4.12

Model 2 (Dropping School_Type_D)

- In second model will run the model using the variables (i.e. 'Scholarship~Academic_Score+Score_on_Plays_Made+Missed_Play_Score+School_Score+Overall_Score+Injury_Propensity+School_Type_C+Region_Southern+Region_Western').

Logit Regression Results

Dep. Variable:	Scholarship	No. Observations:	5268
Model:	Logit	Df Residuals:	5258
Method:	MLE	Df Model:	9
Date:	Mon, 19 Apr 2021	Pseudo R-squ.:	0.3040
Time:	12:28:54	Log-Likelihood:	-2409.3
converged:	True	LL-Null:	-3461.6
Covariance Type:	nonrobust	LLR p-value:	0.000

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-7.3399	0.513	-14.316	0.000	-8.345	-6.335
Academic_Score	0.6178	0.040	15.510	0.000	0.540	0.696
Score_on_Plays_Made	5.9450	0.313	18.999	0.000	5.332	6.558
Missed_Play_Score	-1.2916	0.339	-3.813	0.000	-1.955	-0.628
School_Score	3.9312	0.294	13.357	0.000	3.354	4.508
Overall_Score	-0.0994	0.035	-2.865	0.004	-0.167	-0.031
Injury_Propensity	-0.2990	0.036	-8.418	0.000	-0.369	-0.229
School_Type_C	0.2332	0.078	2.974	0.003	0.080	0.387
Region_Southern	-0.4937	0.090	-5.494	0.000	-0.670	-0.318
Region_Western	0.0011	0.090	0.012	0.990	-0.175	0.177

- adj_pseudo_r2 = 0.3014006091437501
- Basis above model we can see that adjusted pseudo r square is 30.1%, the model performance has gone down compared to model1, but p Value looks insignificant for variable' Region_Western'.
- Let us now check multi-collinearity using VIF value.

  Variable VIF Value

  Academic_Score VIF = 1.39

  Score_on_Plays_Made VIF = 1.51

  Missed_Play_Score VIF = 1.52

  School_Score VIF = 1.19

Overall_Score VIF = 1.28

Injury_Propensity VIF = 1.24

School_Type_C VIF = 1.13

Region_Southern VIF = 1.23

Region_Western VIF = 1.25

- All the variables are below threshold of 2. Hence, we will go ahead and drop the insignificant variables by checking P value. (i.e,Region_Western)

Model 3 (Dropping School_Type_D & Region_Western)

- In third model will run the model using the variables (i.e. 'Scholarship~Academic_Score+Score_on_Plays_Made+Missed_Play_Score+School_Score+Overall_Score+Injury_Propensity+School_Type_C+ Region_Southern').

Logit Regression Results

Dep. Variable:	Scholarship	No. Observations:	5268
Model:	Logit	Df Residuals:	5259
Method:	MLE	Df Model:	8
Date:	Mon, 19 Apr 2021	Pseudo R-squ.:	0.3040
Time:	12:28:55	Log-Likelihood:	-2409.3
converged:	True	LL-Null:	-3461.6
Covariance Type:	nonrobust	LLR p-value:	0.000

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-7.3396	0.512	-14.328	0.000	-8.344	-6.336
Academic_Score	0.6178	0.040	15.516	0.000	0.540	0.696
Score_on_Plays_Made	5.9457	0.308	19.273	0.000	5.341	6.550
Missed_Play_Score	-1.2918	0.339	-3.815	0.000	-1.955	-0.628
School_Score	3.9312	0.294	13.358	0.000	3.354	4.508
Overall_Score	-0.0994	0.035	-2.865	0.004	-0.167	-0.031
Injury_Propensity	-0.2990	0.035	-8.458	0.000	-0.368	-0.230
School_Type_C	0.2331	0.078	2.990	0.003	0.080	0.386
Region_Southern	-0.4941	0.084	-5.863	0.000	-0.659	-0.329

- adj_pseudo_r2 - 0.30168947223257847
- Let us now check multi-collinearity using VIF value.

<u>Variable</u> <u>VIF Value</u>

Academic_Score VIF = 1.39

Score_on_Plays_Made VIF = 1.46

Missed_Play_Score VIF = 1.52

School_Score VIF = 1.18

Overall_Score VIF = 1.28

Injury_Propensity VIF = 1.23

School_Type_C VIF = 1.12

Region_Southern VIF = 1.09

All the variables VIF Value is below threshold of 2. And by looking at p Value, all the variables are significant.

Model Evaluation

	model_name	model_perf
1	Mod_1	0.321082
2	Mod_2	0.301401
3	Mod_3	0.301689

Inference - Based on Adjusted Pseudo R square we can see that Model 3 seems to be good model with important variables and free from multi collinearity.

A pseudo R-squared only has meaning when compared to another pseudo R-squared of the same type, on the same data, predicting the same outcome. In this situation, the higher pseudo R-squared indicates which model better predicts the outcome. Its not used as Evaluation Metric

Basis Above descriptive Logistic regression we can say that below variables are the important variables in predicting the partial or full scholarship of students.

- Academic_Score
- Score_on_Plays_Made
- Missed_Play_Score
- School_Score
- Overall_Score
- Injury_Propensity
- School_Type_C
- Region_Southern'

- We will use Model 1, Model 2 and Model 3 to predict and check the model evaluation.

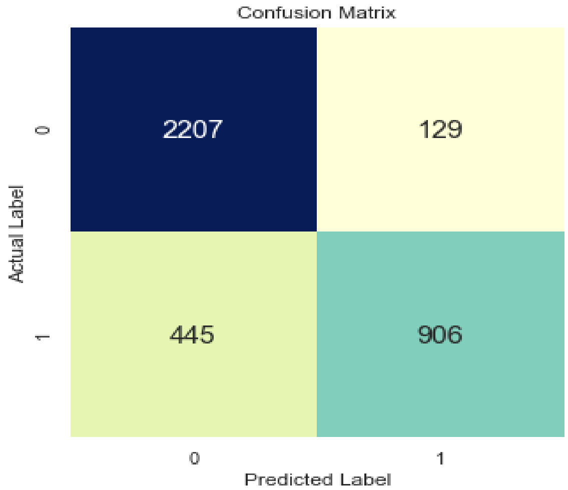
Predictive Logistic Regression

- Predictive Logistic Regression – Main Objective of Predictive Logistic Regression is predictive values for Features / Variables.
- In predictive Logistic Regression we no need to look after assumptions. Whereas for predictive type assumptions are not important only metric is important.

  For predictive type we divide the dataset into train and test. Also, We Use sklearn & statmodel for as python library for predictive Logistic Regression.

Logistic Regression - Model 1

- Let's import classification_report from sklearn.metrics.
- Let us first evaluate on the **training data**.
- We will start by checking the confusion matrix and then the classification report as well.
- Firstly, Train Accuracy for Model is 0.844317873609981.
- Confusion_matrix for train data.



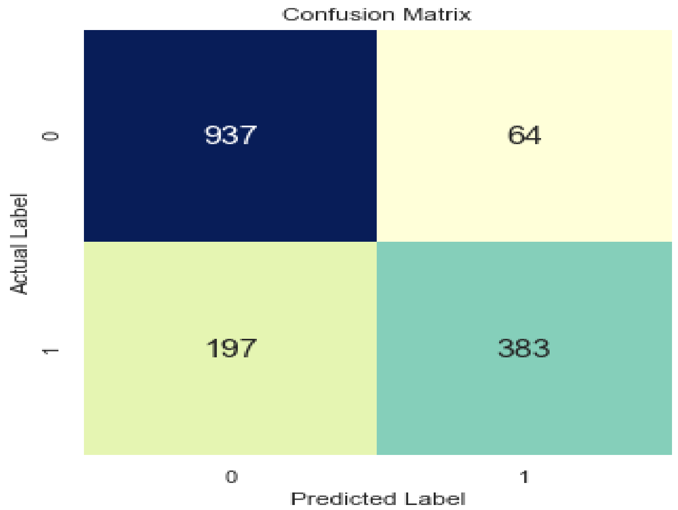
- Classification Report.

	precision	recall	f1-score	support
0	0.83	0.94	0.88	2336
1	0.88	0.67	0.76	1351
accuracy			0.84	3687
macro avg	0.85	0.81	0.84	3687
weighted avg	0.85	0.84	0.84	3687

- We can see 84% overall accuracy on the Training data.

Let us now evaluate on the testing data

- Test Accuracy for Model is 0.8349146110056926
- Confusion_matrix for train data.



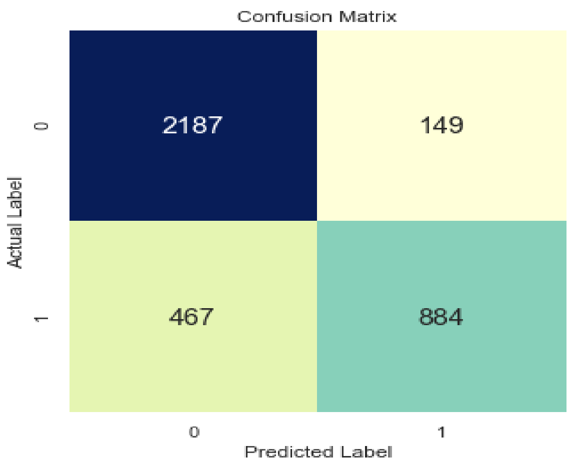
- Classification Report.

	precision	recall	f1-score	support
0	0.83	0.94	0.88	1001
1	0.86	0.66	0.75	580
accuracy			0.83	1581
macro avg	0.84	0.80	0.81	1581
weighted avg	0.84	0.83	0.83	1581

- We can see 83% overall accuracy on the Training data.

Logistic Regression - Model 2

- Let us first evaluate on the **training data**.
- Train Accuracy for Model is 0.8329264985082723.
- Confusion_matrix for train data



- Classification Report.

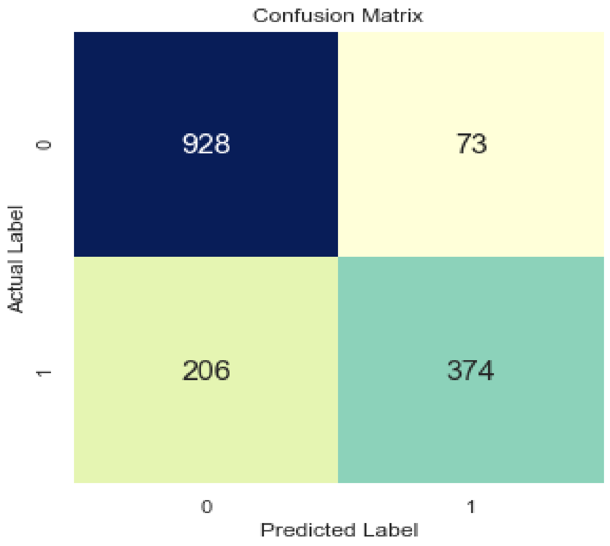
	precision	recall	f1-score	support
0	0.82	0.94	0.88	2336
1	0.86	0.65	0.74	1351
accuracy			0.83	3687
macro avg	0.84	0.80	0.81	3687
weighted avg	0.84	0.83	0.83	3687

- We can see 83% overall accuracy on the Training data.

Let us now evaluate on the testing data

- Test Accuracy for Model is 0.8235294117647058

- Confusion_matrix for test data.



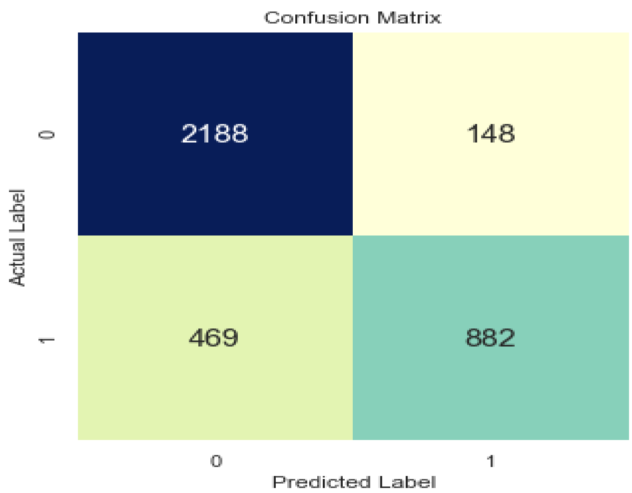
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.93	0.87	1001
1	0.84	0.64	0.73	580
accuracy			0.82	1581
macro avg	0.83	0.79	0.80	1581
weighted avg	0.83	0.82	0.82	1581

- We can see 82% overall accuracy on the Testing data.

Logistic Regression - Model 3

- Let us first evaluate on the **training data**.
- Train Accuracy for Model is 0.8326552752915649
- Confusion_matrix for train data



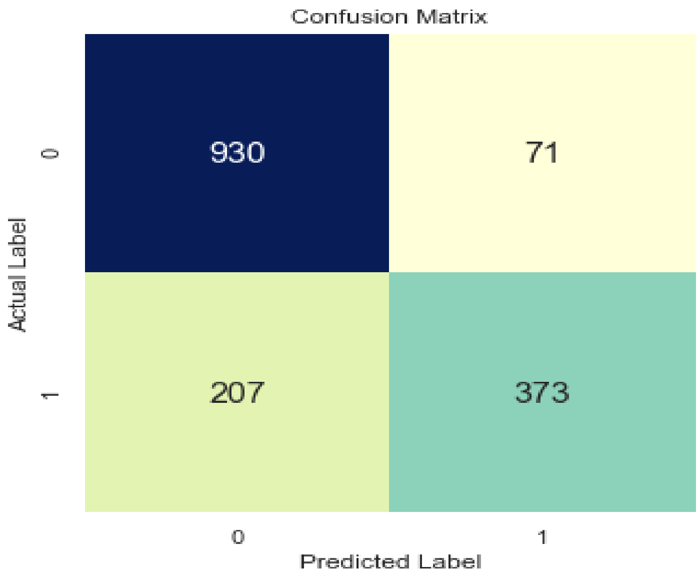
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.94	0.88	2336
1	0.86	0.65	0.74	1351
accuracy			0.83	3687
macro avg	0.84	0.79	0.81	3687
weighted avg	0.84	0.83	0.83	3687

- We can see a 83% overall accuracy on the Training data.

Let us now evaluate on the testing data

- Test Accuracy for Model is 0.8241619228336496
- Confusion_matrix for test data.

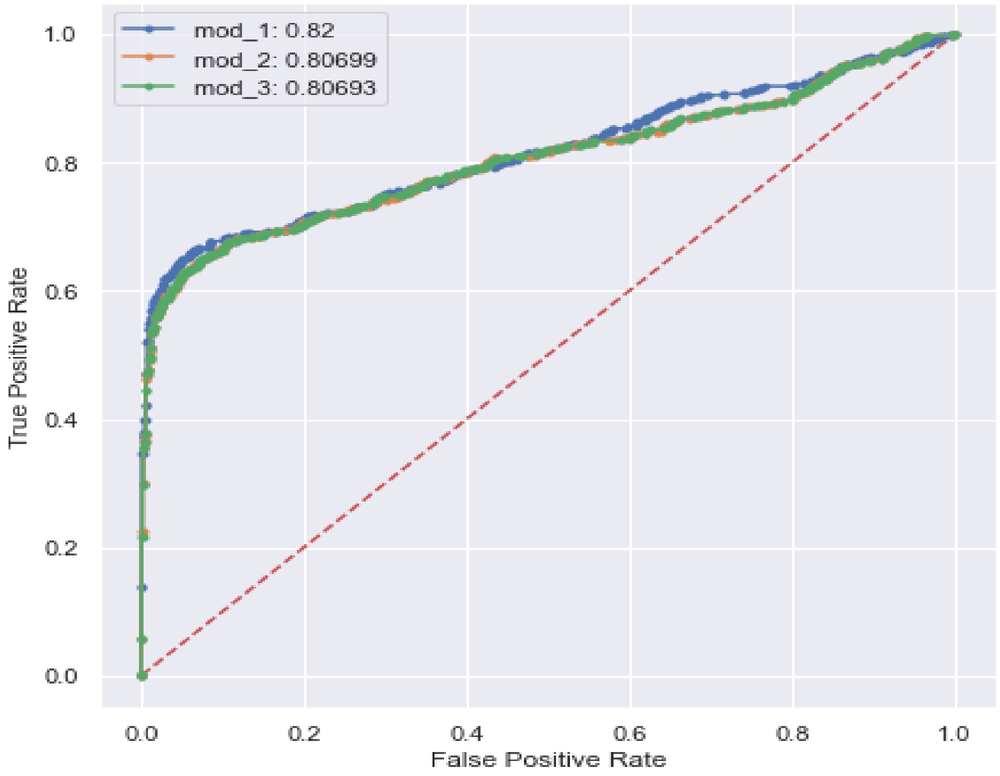


- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.93	0.87	1001
1	0.84	0.64	0.73	580
accuracy			0.82	1581
macro avg	0.83	0.79	0.80	1581
weighted avg	0.83	0.82	0.82	1581

- We can see a 82% overall accuracy on the Testing data.

Check the summary statistics of the AUC-ROC curve for all the three Logistic Regression Models built. This is for the test data



AUC stands for "Area under the ROC Curve." That is, AUC measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1).

AUC provides an aggregate measure of performance across all possible classification thresholds. One way of interpreting AUC is as the probability that the model ranks a random positive example more highly than a random negative example.

AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0. Random forest has the highest AUC for test of all the models.

Logistic Model 1 has the Highest AUC amongst the all the Logistic Models.

Linear Discriminant Analysis (LDA)

LDA uses linear combinations of independent variables to predict the class in the response variable of a given observation. LDA assumes that the independent variables(p) are normally distributed and there is equal variance / covariance for the classes. LDA is popular because it can be used for both classification and dimensionality reduction.

When these assumptions are satisfied, LDA creates a linear decision boundary. Note that based on many research studies, it is observed that LDA performs well when these assumptions are violated.

LDA is based upon the concept of searching for a linear combination of predictor variables that best separates the classes of the target variable.

Key Assumptions for LDA are

- Independent variables should be normally distributed.
- Each Independent variable must have the same variance across classes.
- LDA Does well even if these assumptions are flouted

Linear Discriminant Analysis Model using the same models 1, Model 2 & Model 3.

- Import LinearDiscriminantAnalysis from sklearn.discriminant_analysis. Import confusion_matrix from sklearn.metrics & import scale from sklearn.preprocessing
- Before building the model, we should split the data into Train and Test. We will thus build a model on the training data and use this model to predict on the test data.
- We will be doing a 70:30 split. 70% of the whole data will be used to train the data and then 30% of the data will be used for testing the model thus built.
- Importing train_test_split from sklearn.model_selection to splitting data into training and test set for independent attributes.
- We Use Stratify in train_test_split. **stratification** means that the **train_test_split** method returns training and test subsets that have the same proportions of class labels as the input dataset.

- In LDA we use Bayes Theorem for calculating the probabilities. Using threshold, on probabilities calculated by Baysian Rules

Bayes Theorem: ¶

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

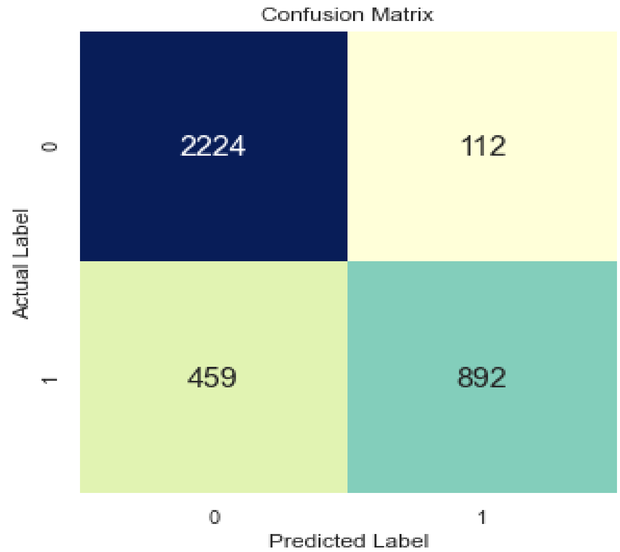
$$P(y = k|x) = \frac{P(x|y = k) \times P(y = k)}{P(x)}$$

LDA makes predictions by estimating the probability that a new set of inputs belongs to each class. The class that gets the highest probability is the output class and a prediction is made.

The model uses Bayes Theorem to estimate the probabilities. Briefly Bayes' Theorem can be used to estimate the probability of the output class (k) given the input (x) using the probability of each class and the probability of the data belonging to each class:

Linear Discriminant analysis - Model 1

- Let us first evaluate on the **training data**.
- Train Accuracy for Model is 0.8451315432601031.
- Confusion_matrix for train data.



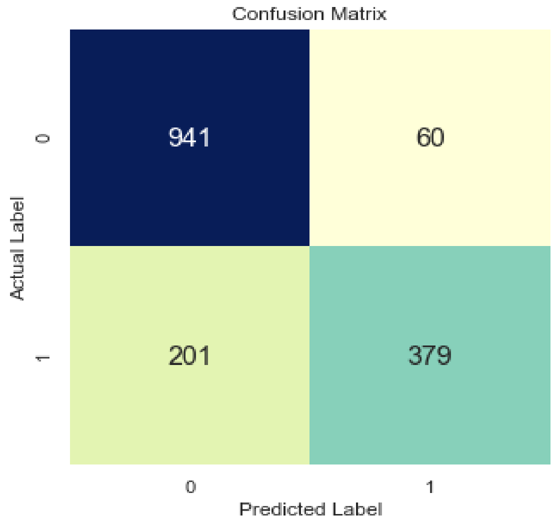
- Classification Report.

	precision	recall	f1-score	support
0	0.83	0.95	0.89	2336
1	0.89	0.66	0.76	1351
accuracy			0.85	3687
macro avg	0.86	0.81	0.82	3687
weighted avg	0.85	0.85	0.84	3687

- We can see 85% overall accuracy on the Training data.

Let us now evaluate on the testing data

- Test Accuracy for Model is 0.8349146110056926
- Confusion_matrix for test data.



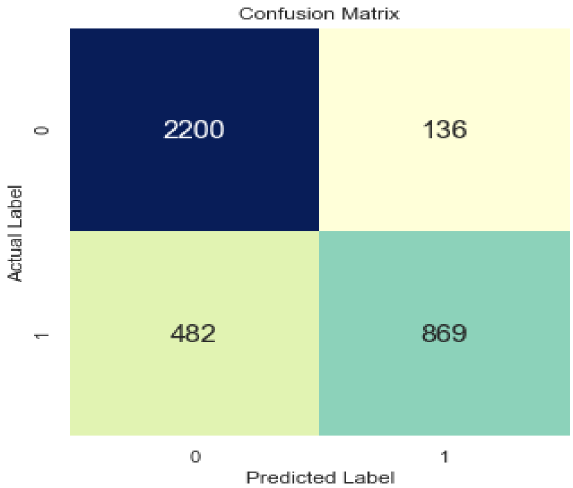
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.94	0.88	1001
1	0.86	0.65	0.74	580
accuracy			0.83	1581
macro avg	0.84	0.80	0.81	1581
weighted avg	0.84	0.83	0.83	1581

- We can see 83% overall accuracy on the Testing data.

Linear Discriminant analysis - Model 2

- Let us first evaluate on the **training data**.
- Train Accuracy for Model is 0.8323840520748577.
- Confusion_matrix for train data.



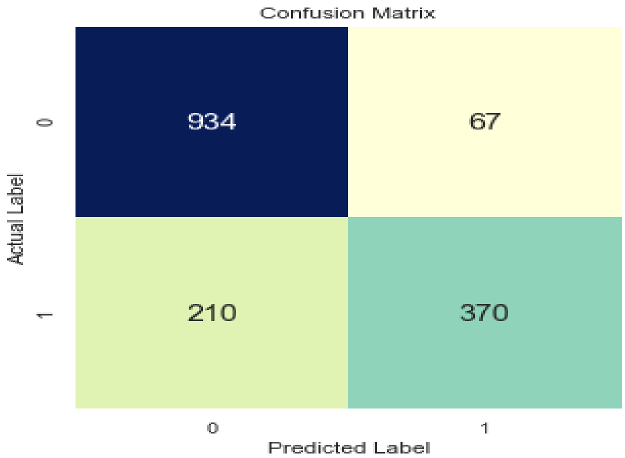
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.94	0.88	2336
1	0.86	0.64	0.74	1351
accuracy			0.83	3687
macro avg	0.84	0.79	0.81	3687
weighted avg	0.84	0.83	0.83	3687

- We can see a 83% overall accuracy on the Training data.

Let us now evaluate on the testing data

- Test Accuracy for Model is 0.8247944339025933
- Confusion_matrix for test data.



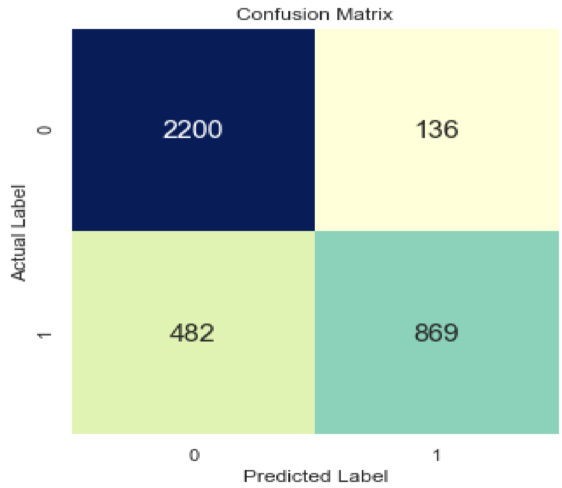
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.93	0.87	1001
1	0.85	0.64	0.73	580
accuracy			0.82	1581
macro avg	0.83	0.79	0.80	1581
weighted avg	0.83	0.82	0.82	1581

- We can see a 82% overall accuracy on the Testing data.

Linear Discriminant analysis - Model 3

- Let us first evaluate on the **training data**.
- Train Accuracy for Model is 0.8323840520748577.
- Confusion_matrix for train data.



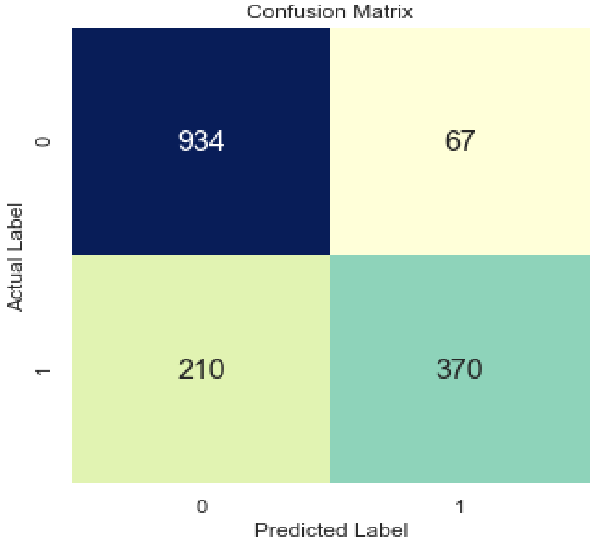
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.94	0.88	2336
1	0.86	0.64	0.74	1351
accuracy			0.83	3687
macro avg	0.84	0.79	0.81	3687
weighted avg	0.84	0.83	0.83	3687

- We can see a 83% overall accuracy on the Training data.

Let us now evaluate on the testing data

- Test Accuracy for Model is 0.8247944339025933
- Confusion_matrix for test data.



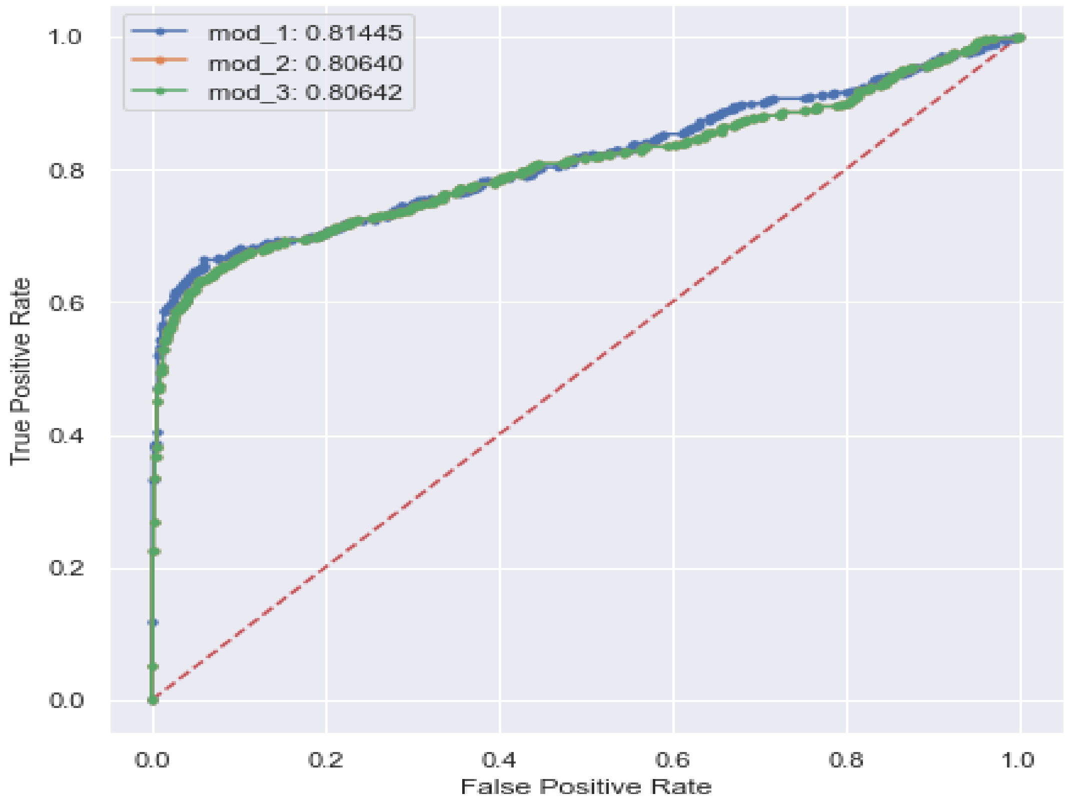
- Classification Report.

	precision	recall	f1-score	support
0	0.82	0.93	0.87	1001
1	0.85	0.64	0.73	580
accuracy			0.82	1581
macro avg	0.83	0.79	0.80	1581
weighted avg	0.83	0.82	0.82	1581

- We can see a 82% overall accuracy on the Testing data.

Check the summary statistics of the AUC-ROC curve for all the three LDA Models built. This is for the test data.

.



AUC stands for "Area under the ROC Curve." That is, AUC measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1).

AUC provides an aggregate measure of performance across all possible classification thresholds. One way of interpreting AUC is as the probability that the model ranks a random positive example more highly than a random negative example.

AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0. Random forest has the highest AUC for test of all the models.

LDA Model 1 has the Highest AUC amongst the all the three LDA Models

Model Evaluation

	LR_Model_1	LR_Model_2	LR_Model_3	LDA_Model_1	LDA_Model_2	LDA_Model_3
Train-Accuracy	0.844318	0.832927	0.832655	0.845132	0.832384	0.832384
Test-Accuracy	0.834915	0.823529	0.823529	0.834915	0.824794	0.824794

Inference –

- For this problem we are using accuracy as metric for evaluation of the models.
- In comparison between Logistic regression model's Logistic regression Model 1 has good Accuracy for both train and test compared to other models.
- In comparison between LDA Models basis above results we can see that LDA Model has good accuracy for both train and test models.
- When in comparison between Logistic model & Linear Discriminant Model We can see that LDA Model has a very good accuracy amongst the all the logistic & LDA Models. As the dataset is small ones. LDA Seems to out-preforming than logistic Regression.
- Logistic regression does not have as many assumptions and restrictions as discriminant analysis. But Logistic Regression lacks stability when the classes are well separated, Whereas the LDA Is good when the classes are well separated. this is when LDA comes into pictures However, when discriminant analysis' assumptions are met, it is more powerful than logistic regression. Unlike logistic regression, discriminant analysis can be used with small sample sizes.