JAMBOREE BUSINESS CASE

LINEAR REGRESSION

From company's perspective:

● Jamboree is a renowned educational institution that has successfully assisted numerous students in gaining admission to top colleges abroad. With their proven problem-solving methods, they have helped students achieve exceptional scores on exams like GMAT, GRE, and SAT with minimal effort.

● To further support students, Jamboree has recently introduced a new feature on their website. This feature enables students to assess their probability of admission to Ivy League colleges, considering the unique perspective of Indian applicants.

● By conducting a thorough analysis, we can assist Jamboree in understanding the crucial factors impacting graduate admissions and their interrelationships. Additionally, we can provide predictive insights to determine an individual's admission chances based on various variables.

Dataset link : https://drive.google.com/file/d/1UCnSk_NN02jlzj0bbSZ_j-gdGUDDJxy4/view

Column Dictionary

● Serial No.: This column represents the unique row identifier for each applicant in the dataset.

● GRE Scores: This column contains the GRE (Graduate Record Examination) scores of the applicants, which are measured on a scale of 0 to 340.

● TOEFL Scores: This column includes the TOEFL (Test of English as a Foreign Language) scores of the applicants, which are measured on a scale of 0 to 120.

● University Rating: This column indicates the rating or reputation of the university that the applicants are associated with.

○ The rating is based on a scale of 0 to 5, with 5 representing the highest rating.

● SOP: This column represents the strength of the applicant's statement of purpose, rated on a scale of 0 to 5, with 5 indicating a strong and compelling SOP.

● LOR: This column represents the strength of the applicant's letter of recommendation,

rated on a scale of 0 to 5, with 5 indicating a strong and compelling LOR.

● CGPA: This column contains the undergraduate Grade Point Average (GPA) of the

applicants, which is measured on a scale of 0 to 10.

● Research: This column indicates whether the applicant has research experience (1) or

not (0).

● Chance of Admit: This column represents the estimated probability or chance of

admission for each applicant, ranging from 0 to 1.

These columns provide relevant information about the applicants' academic qualifications, test

scores, university ratings, and other factors that may influence their chances of admission.

Expectations

primary objective is to analyse the given dataset and derive valuable insights from it. Additionally, utilize the dataset

to construct a predictive model capable of estimating an applicant's likelihood of admission

based on the available features.

Context

Jamboree has helped thousands of students like you make it to top colleges abroad. Be it GMAT, GRE or SAT, their unique problem-solving methods ensure maximum scores with minimum effort.
They recently launched a feature where students/learners can come to their website and check their probability of getting into the IVY league college. This feature estimates the chances of graduate admission from an Indian perspective.

How can you help here?

Your analysis will help Jamboree in understanding what factors are important in graduate admissions and how these factors are interrelated among themselves. It will also help predict one's chances of admission given the rest of the variables.

Concept Used:

- Exploratory Data Analysis
- Linear Regression

HIGH LEVEL CODE

1. Importing Libraries:

Use tools like pandas for data handling, numpy for numerical operations, seaborn and matplotlib for data visualization, sklearn for building model.

2. Loading the Data:
    a. Read a dataset named “jam.txt” into a DataFrame to analyze it.
3. Initial Data Check:
    a. Display the first few rows to get an overview of the data.
    b. Check for missing (null) values in the data.

```
df = pd.read_csv("jam.txt")
df.head()
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65

```
df.shape
```

```
(500, 9)
```

```
#  to check the missing values
df.isna().sum()
```

```
Serial No.           0
GRE Score            0
TOEFL Score          0
University Rating    0
SOP                  0
LOR                  0
CGPA                 0
Research             0
Chance of Admit      0
dtype: int64
```

Clearly there are no missing values in data frame

```
df.describe().T
```

	count	mean	std	min	25%	50%	75%	max
Serial No.	500.0	250.50000	144.481833	1.00	125.7500	250.50	375.25	500.00
GRE Score	500.0	316.47200	11.295148	290.00	308.0000	317.00	325.00	340.00
TOEFL Score	500.0	107.19200	6.081868	92.00	103.0000	107.00	112.00	120.00
University Rating	500.0	3.11400	1.143512	1.00	2.0000	3.00	4.00	5.00
SOP	500.0	3.37400	0.991004	1.00	2.5000	3.50	4.00	5.00
LOR	500.0	3.48400	0.925450	1.00	3.0000	3.50	4.00	5.00
CGPA	500.0	8.57644	0.604813	6.80	8.1275	8.56	9.04	9.92
Research	500.0	0.56000	0.496884	0.00	0.0000	1.00	1.00	1.00
Chance of Admit	500.0	0.72174	0.141140	0.34	0.6300	0.72	0.82	0.97

Observation from the above code:

- **While Observing the mean and 50% percentile of data there is no significant difference observed**
- **We can conclude there are no outliers in the dataset.**

NON GRAPHICAL ANALYSIS

```
# Non Graphical analysis
df["University Rating"].value_counts(normalize = True)
```

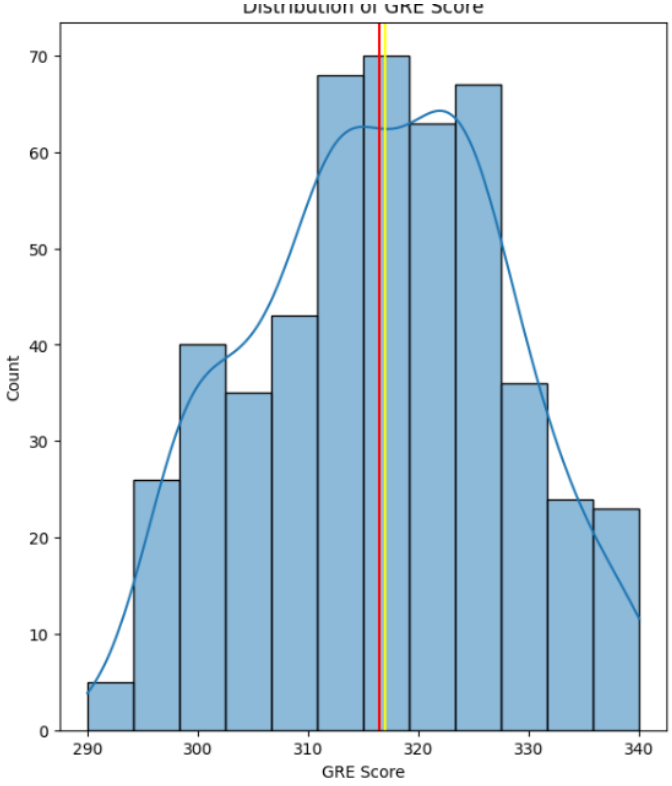
```
3    0.324
2    0.252
4    0.210
5    0.146
1    0.068
Name: University Rating, dtype: float64
```

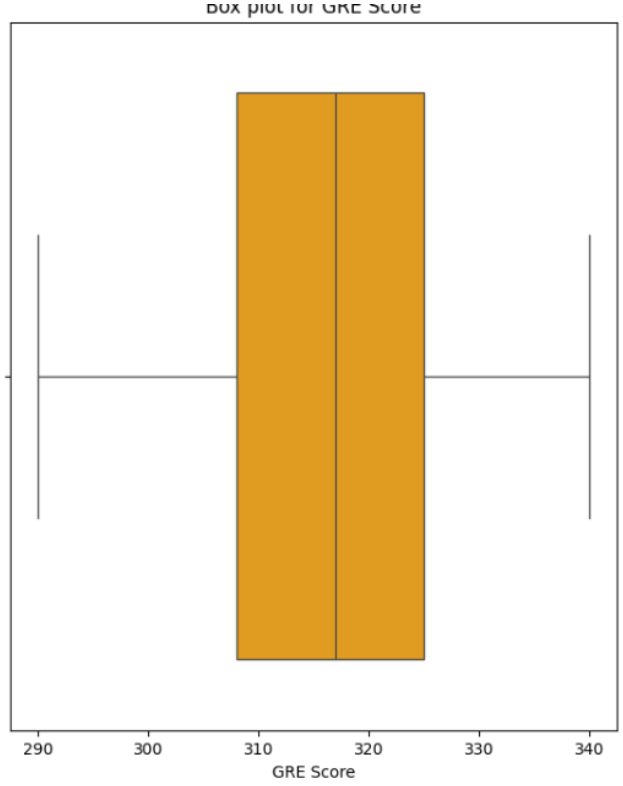
Observation :

Most of the universities average rated.

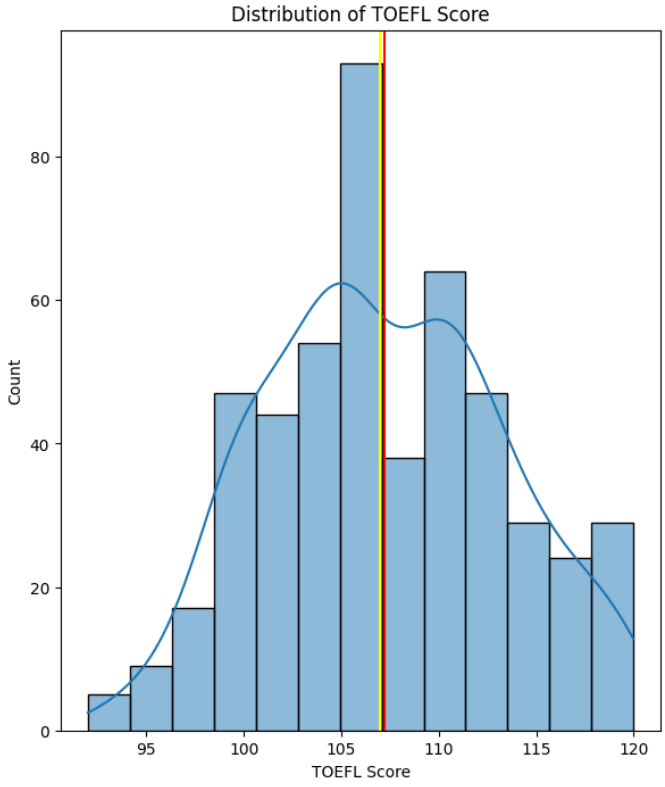
Also stats shows there are almost equal distribution among students who did research.

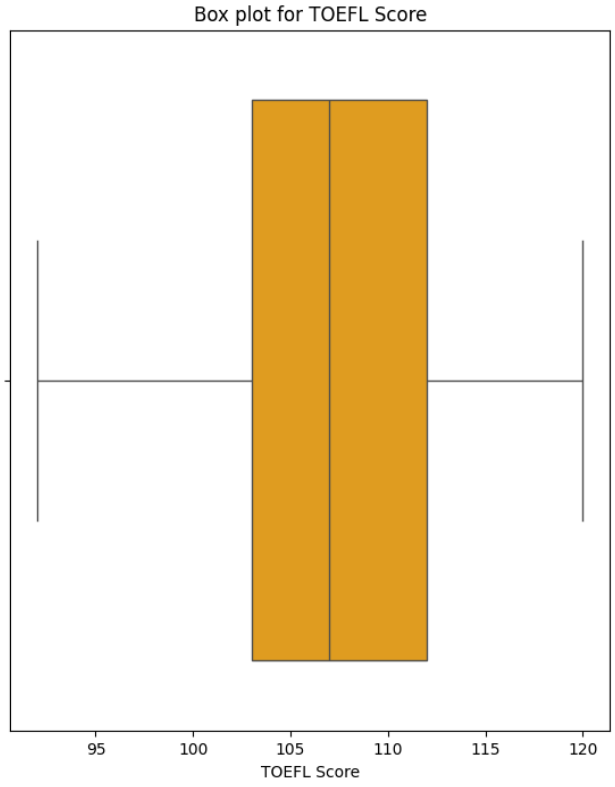
GRAPHICAL ANALYSIS: UNIVARIATE ANALYSIS



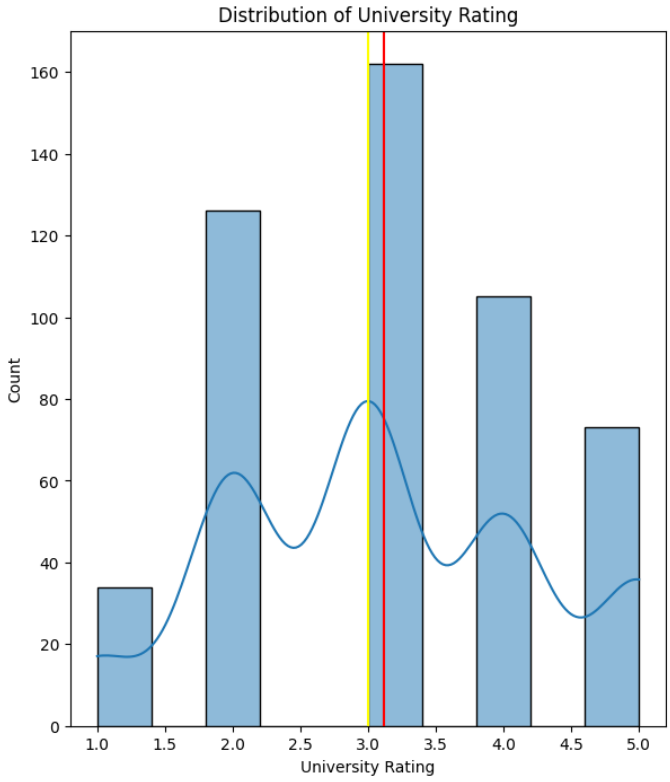


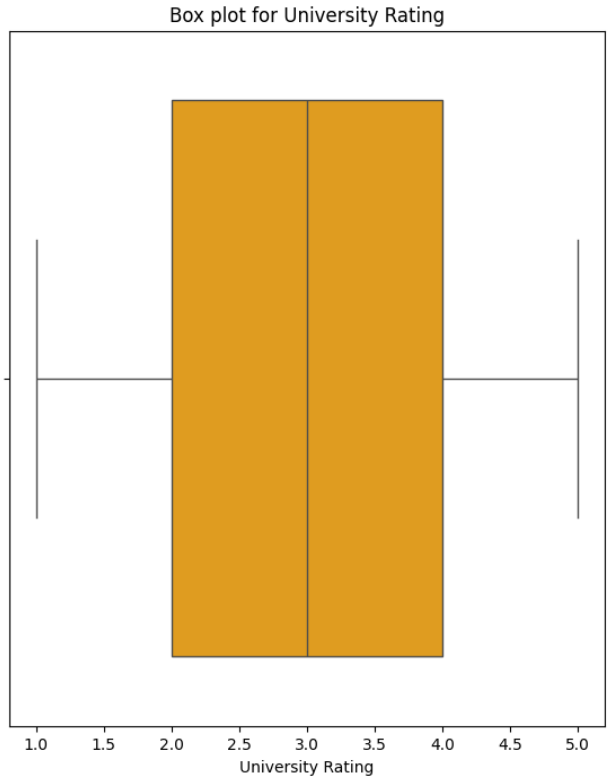
- **GRE Score Analysis**
  - Distribution of GRE resembles like Gaussian
  - Mean of GRE Score is approx 315
  - There is no outliers detected as mean and median overlaps



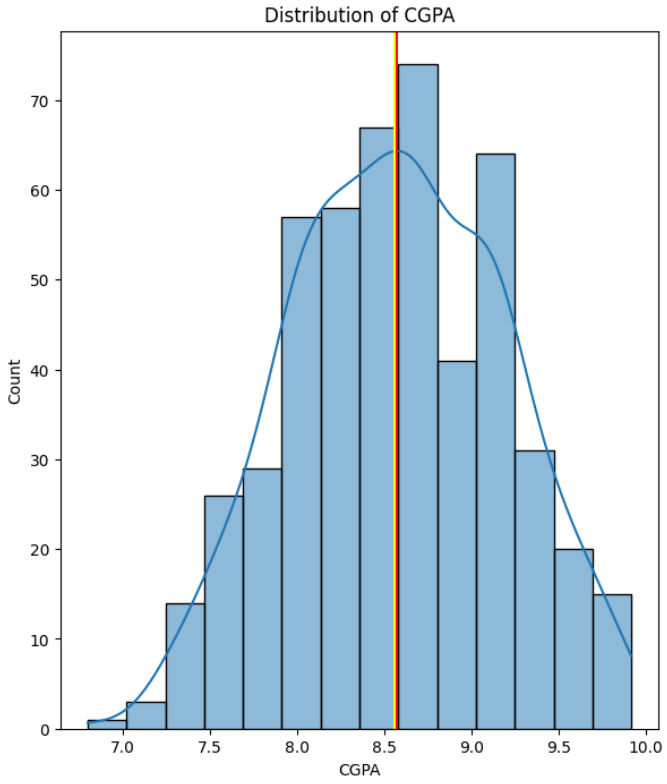


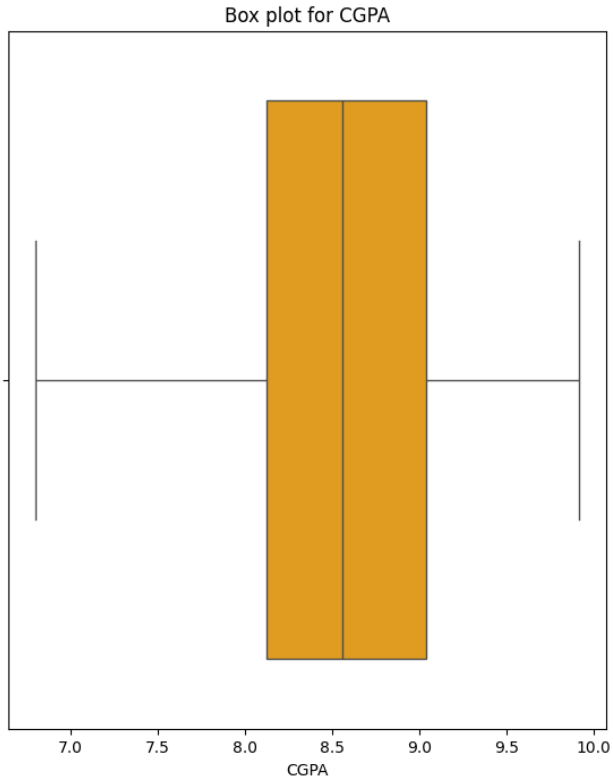
- **TOEFL Score Analysis**
  - Distribution of TOEFL somewhat resembles like Gaussian
  - Mean of TOEFL Score is approx 108
  - There is no outliers detected as mean and median overlaps



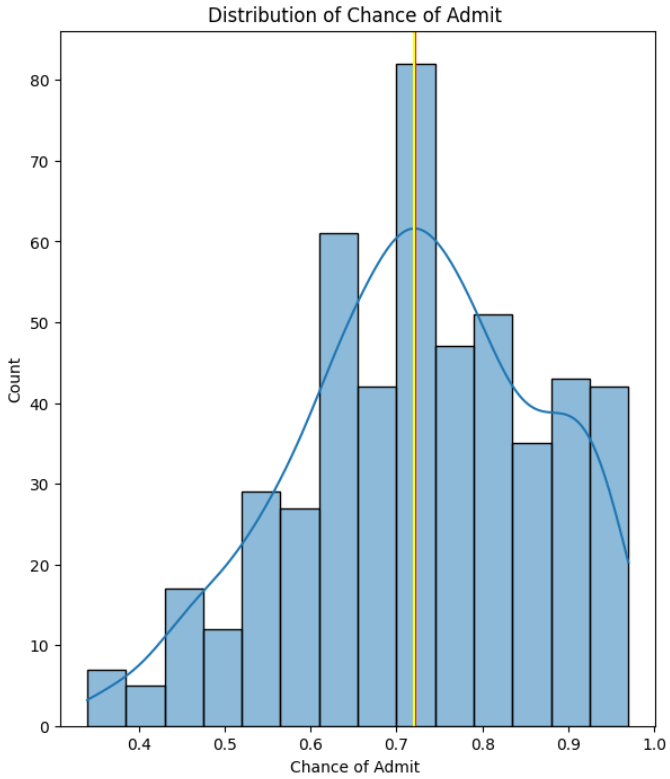


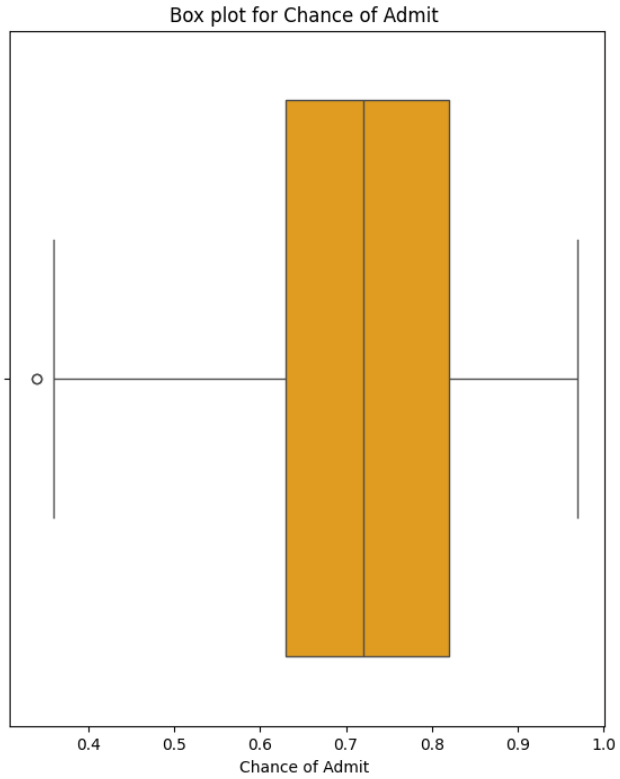
- **LOR Analysis**
  - Most of the students gets 3.5 out 5





- **CGPA Analysis**
  - Distribution of CGPA resembles like Gaussian
  - Mean of CGPA Score is approx. 8.5
  - There is no outliers detected as mean and median overlaps



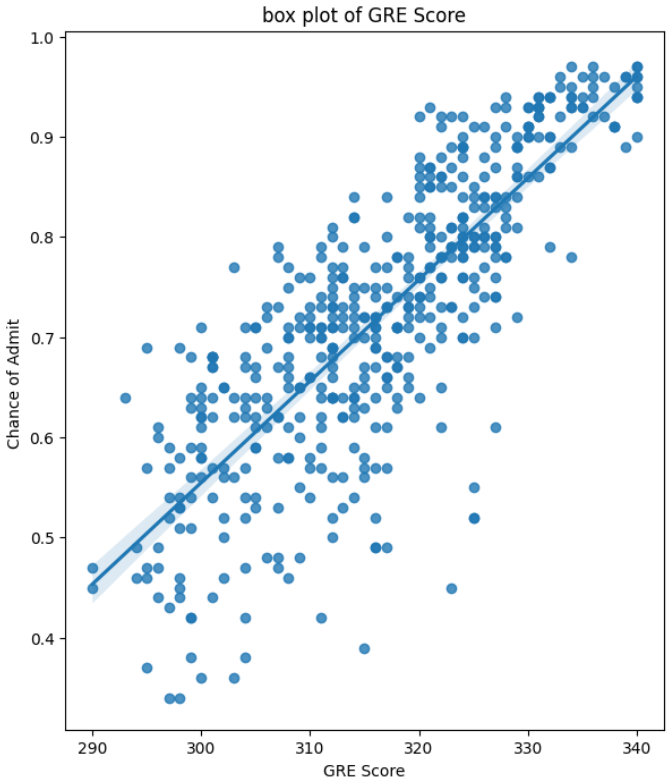


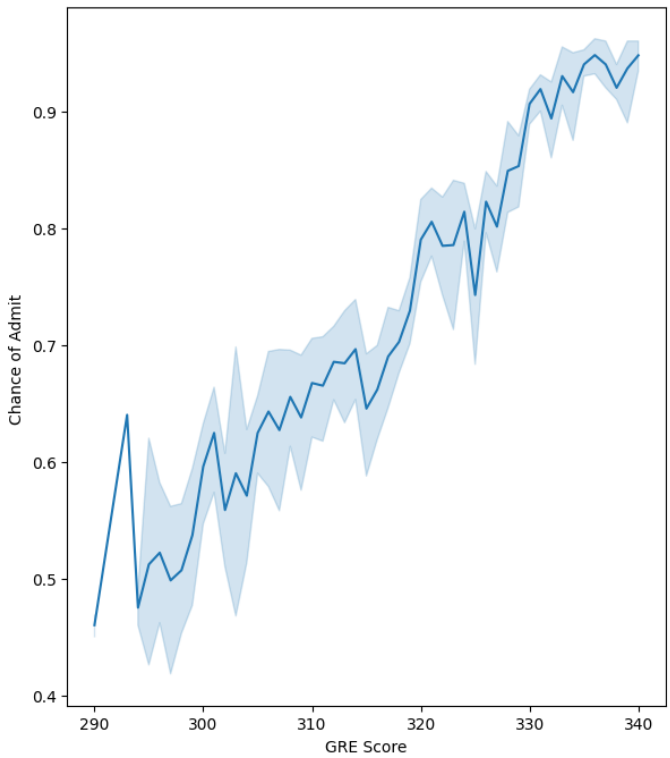
- **Chance of Admit**
  - Mean of chance of admission is 0.72
  - There is no outliers detected as mean and median overlaps

OUTLIER DETECTION

- **From the above observation, There is no outliers detected in the dataset**

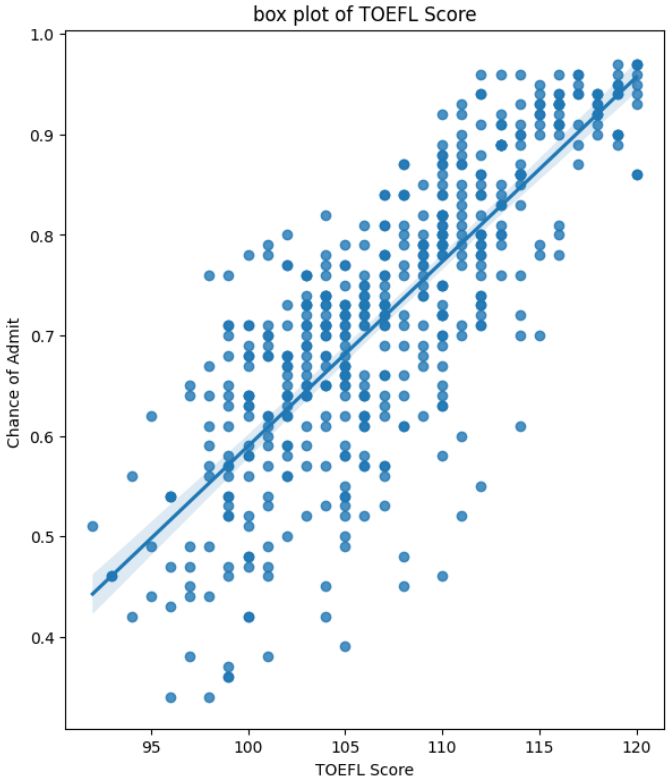
GRAPHICAL ANALYSIS: BIVARIATE ANALYSIS

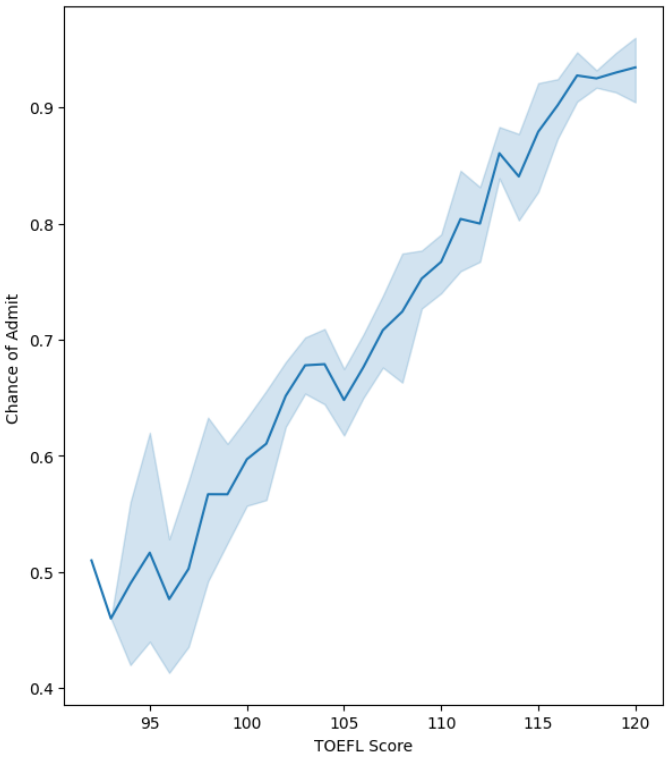




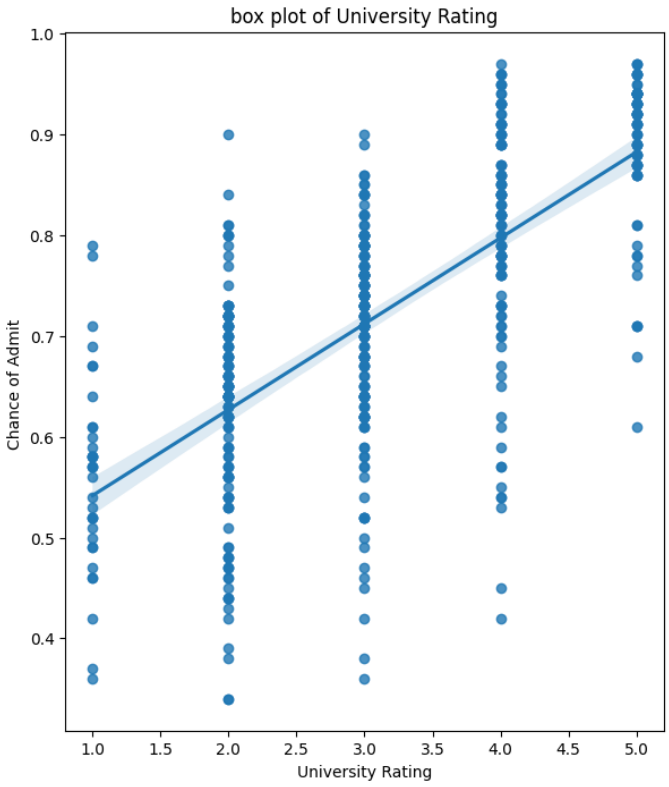
- **GRE vs Chance of Admit Analysis**

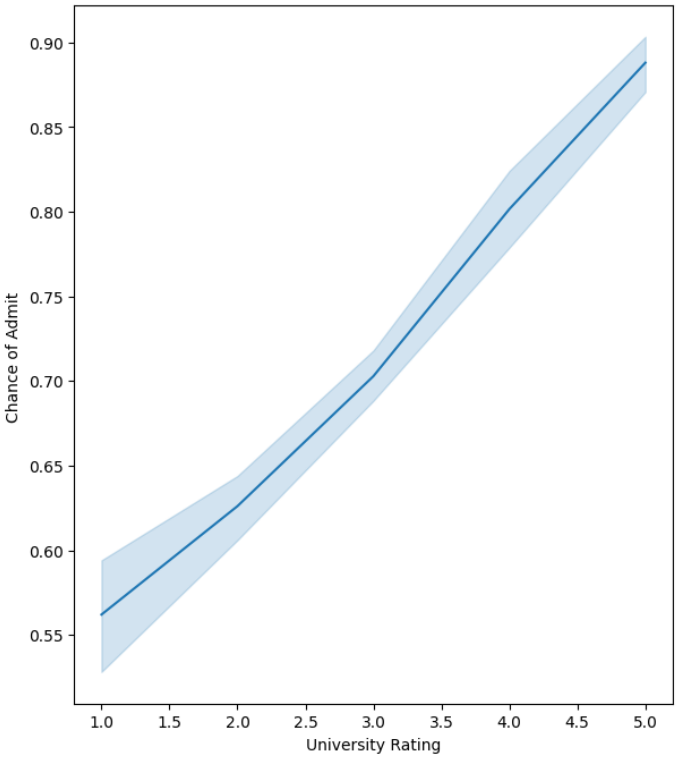
- There is linear relationship between GRE and Chance of Admission
- Higher the GRE -> Higher the chance of admission

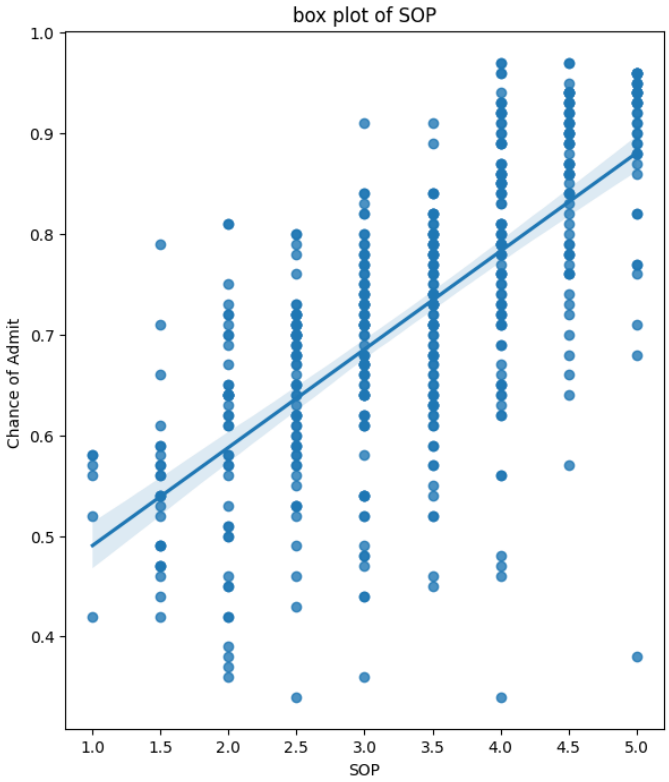


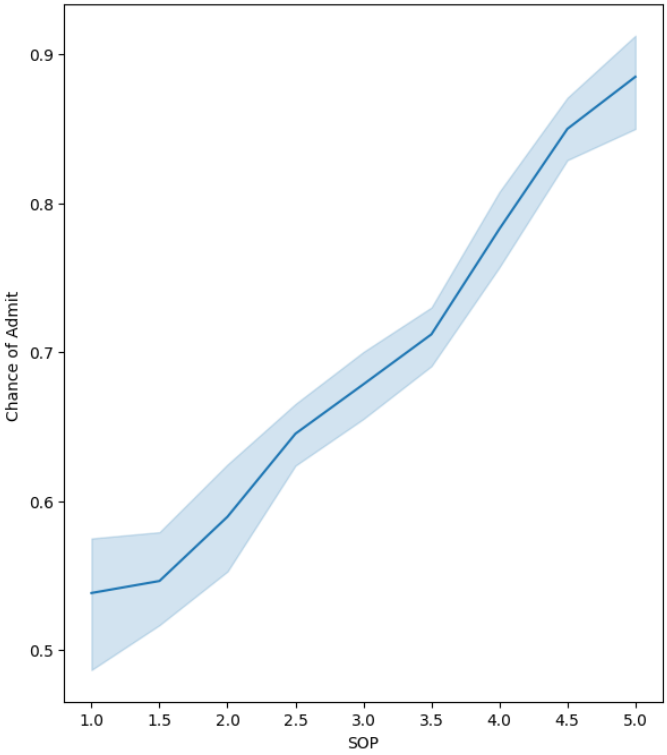


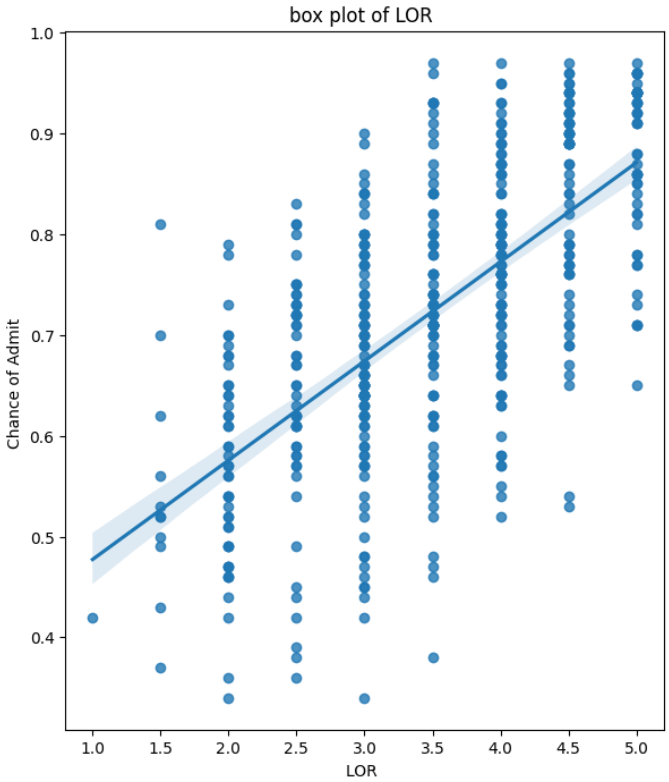
- **TOEFL vs Chance of Admit Analysis**
  - There is linear relationship between TOEFL and Chance of Admission
  - Higher the TOEFL -> Higher the chance of admission

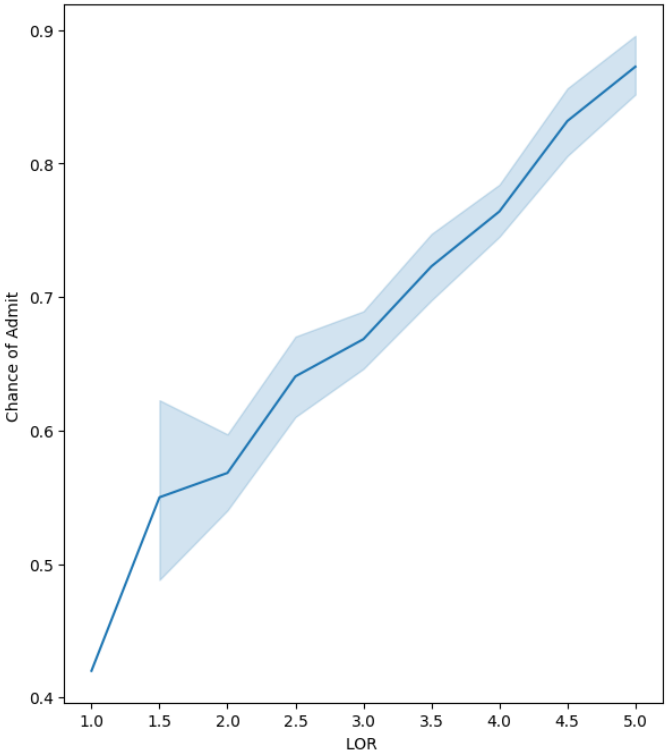




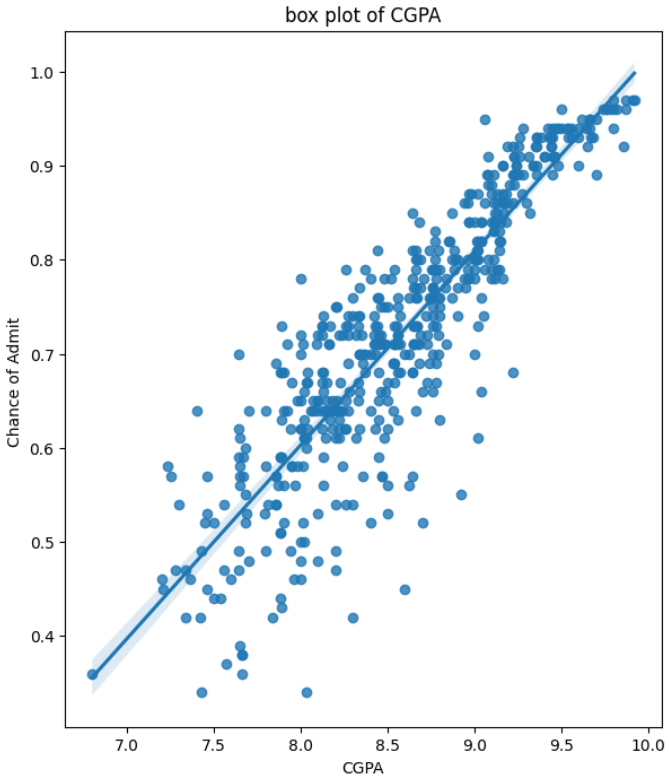


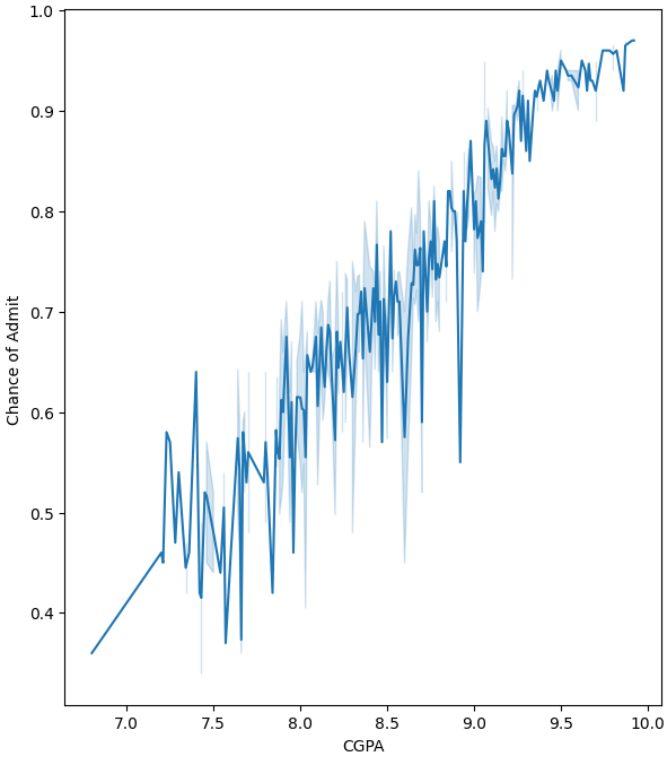




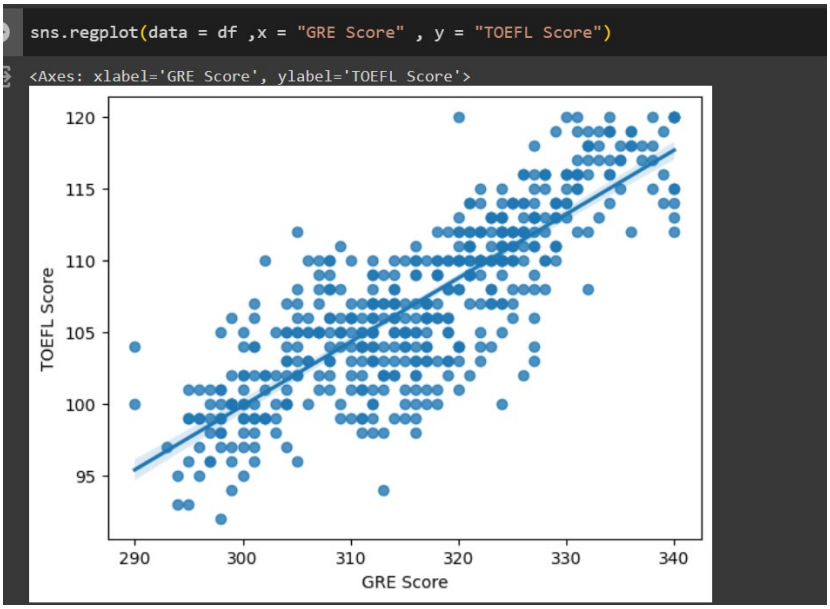


- **LOR / SOP / University Rating vs Chance of Admit Analysis**
  - There is no significant linear relationship between **LOR / SOP / University Rating** and Chance of Admission

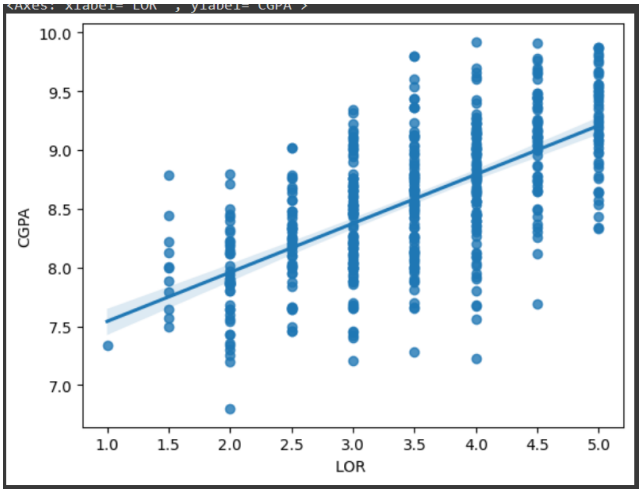




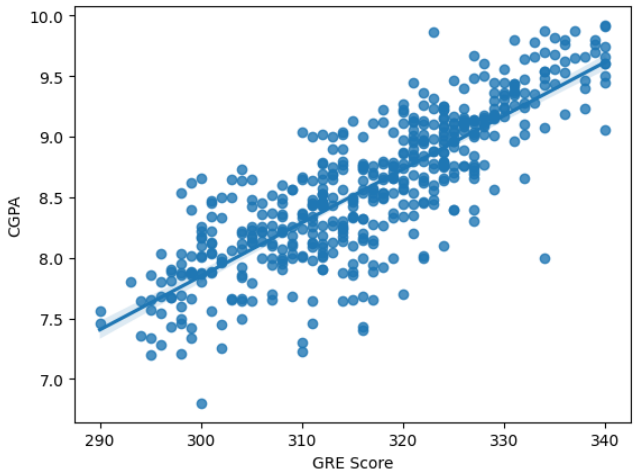
- **CGPA vs Chance of Admit Analysis**
  - There is linear relationship between CGPA and Chance of Admission
  - Higher the CGPA -> Higher the chance of admission



- People with higher GRE Scores also have higher TOEFL Scores which is justified because both TOEFL and GRE have a verbal section which although not similar are relatable
- Although there are exceptions, people with higher CGPA usually have higher GRE scores maybe because they are smart or hard working

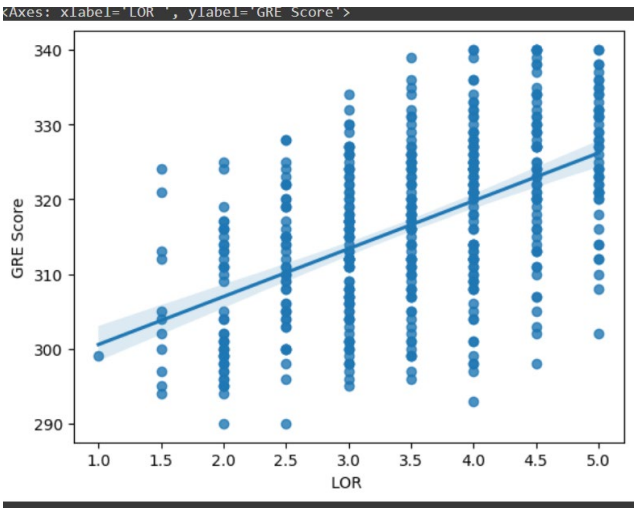


There is no relationship between LOR and CGPA as letter of recommendation does not depend academic excellence of student.

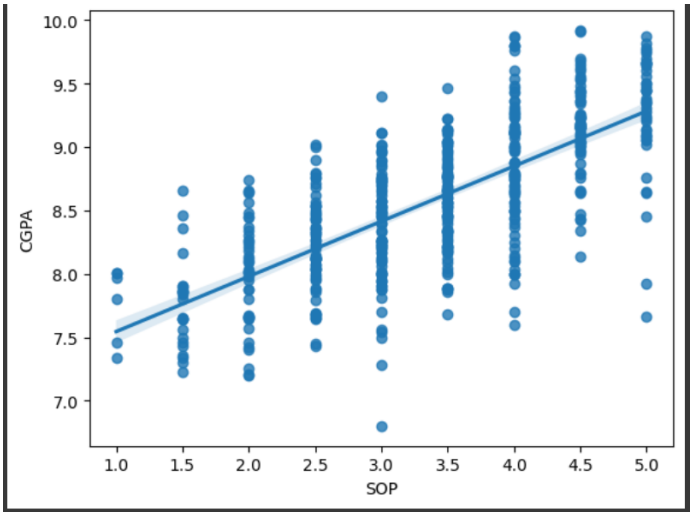


Clearly there is a positive relationship between GRE Score and CGPA as person who is academically strong will have high CGPA and GRE Score

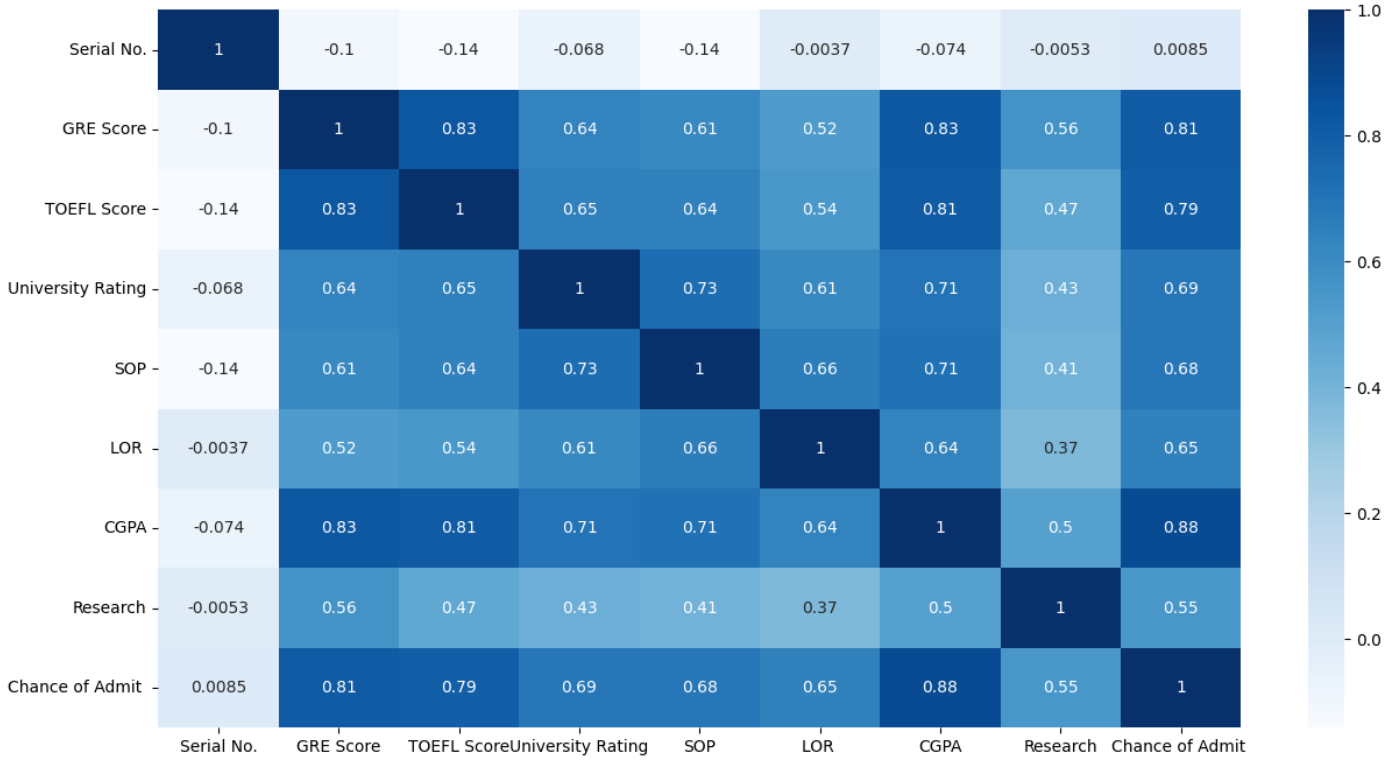
Similarly person with high CGPA will have good TOEFL Score.



There is no relationship between GRE Score and LOR as people with different kinds of LORs have all kinds of GRE scores.



- CGPA and SOP are not that related because Statement of Purpose is related to academic performance, but since people with good CGPA tend to be more hard working so they have good things to say in their SOP which might explain the slight move towards higher CGPA as along with good SOPs
- Similarly, GRE Score and SOP is only slightly related
- Applicants with different kinds of SOP have different kinds of TOEFL Score. So the quality of SOP is not always related to the applicants English skills.



- **High Correlation**
  1. GRE Score vs TOEFL Score
  2. CGPA vs TOEFL Score
  3. CGPA vs GRE Score
  4. Chance of Admit vs CGPA
  5. GRE Score vs Chance of Admit

DATA PREPROCESSING

Duplicate value check and missing values check

```
[77] np.any(df.duplicated())

     False
```

to check missing values

```
[78] df.isnull().sum()

     Serial No.           0
     GRE Score            0
     TOEFL Score          0
     University Rating    0
     SOP                  0
     LOR                  0
     CGPA                 0
     Research             0
     Chance of Admit      0
     dtype: int64
```

Also there are no outliers as from the distributions plotted in the prior figures. Mean and medians are almost overlapping.

FEATURE ENGINEERING, DATA MODELLING AND STANDARDIZATION

```
[82] X = df.drop(["Chance of Admit "], axis =1)
     y = df["Chance of Admit "]

     print("shape if X: ", X.shape)
     print("shape if y: ", y.shape)

     shape if X:  (500, 8)
     shape if y:  (500,)
```

TRAIN TEST SPLIT

```
X_train , X_test , y_train , y_test =train_test_split(X, y ,test_size = 0.2, shuffle = True)
print("X_train shape: {}".format(X_train.shape))
print("X_test shape: {}".format(X_test.shape))
print("y_train shape: {}".format(y_train.shape))
print("y_test shape: {}".format(y_test.shape))

X_train shape: (400, 8)
X_test shape: (100, 8)
y_train shape: (400,)
y_test shape: (100,)
```

```
X_train = pd.DataFrame(data = X_train_std, columns = X_train_columns)
X_train.head(5)
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research
0	0.499137	0.048755	-0.183643	-0.070888	0.648929	0.019999	-0.116474	0.904534
1	-0.599035	0.487991	0.460719	1.701315	1.156897	0.553296	0.662781	-1.105542
2	1.158040	0.048755	-0.989096	-0.070888	-0.367007	-1.579895	-1.044947	0.904534
3	-0.117256	-1.093258	-1.150187	-0.956990	-0.874975	0.019999	-0.829409	-1.105542
4	-0.230616	0.400144	0.299628	0.815213	0.648929	0.553296	0.928059	0.904534

MODEL BUILDING- LINEAR REGRESSION

```
from sklearn.linear_model import LinearRegression, Lasso, Ridge
from sklearn.metrics import mean_squared_error
from statsmodels.stats.outliers_influence import variance_inflation_factor
import statsmodels.api as sm
lm = LinearRegression()
lm.fit(X_train.values , y_train)
predictions = lm.predict(std.transform(X_test))
print("RMSE Linear Regression " , np.sqrt(mean_squared_error(y_test,predictions)))
list(zip(X_train.columns , lm.coef_))
```

```
RMSE Linear Regression  0.05108373668730406
[('Serial No.', 0.011767979966348301),
 ('GRE Score', 0.020994926059454903),
 ('TOEFL Score', 0.019892225051393236),
 ('University Rating', 0.005308731129562885),
 ('SOP', 0.0036165286256913676),
 ('LOR ', 0.014259848966300776),
 ('CGPA', 0.07223178164167804),
 ('Research', 0.011972433498967953)]
```

```
RMSE Lasso 0.13201267221369317
[('Serial No.', -0.0),
 ('GRE Score', 0.0),
 ('TOEFL Score', 0.0),
 ('University Rating', 0.0),
 ('SOP', 0.0),
 ('LOR ', 0.0),
 ('CGPA', 0.0),
 ('Research', 0.0)]
```

```
RMSE Ridge 0.051043401273448474
[('Serial No.', 0.011733133315892956),
 ('GRE Score', 0.02121524872543078),
 ('TOEFL Score', 0.020035900073730517),
 ('University Rating', 0.005413828363200945),
 ('SOP', 0.003751843185769286),
 ('LOR ', 0.014304247742259864),
 ('CGPA', 0.0715442821763735),
 ('Research', 0.011974923704461267)]
```

Lasso and Ridge regularisations are performed for a more accurate prediction. Lasso model uses shrinkage. Shrinkage is where data values are shrunk towards a central point as the mean.

MODEL SUMMARY USING STATS MODEL

```
==============================================================================
Dep. Variable:                      y   R-squared:                       0.825
Model:                            OLS   Adj. R-squared:                  0.821
Method:                 Least Squares   F-statistic:                     230.0
Date:                Tue, 06 Feb 2024   Prob (F-statistic):          9.64e-143
Time:                        13:39:48   Log-Likelihood:                 558.26
No. Observations:                 400   AIC:                            -1099.
Df Residuals:                     391   BIC:                            -1063.
Df Model:                           8
Covariance Type:            nonrobust
==============================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const              0.7205      0.003    237.739      0.000       0.715       0.726
Serial No.         0.0118      0.003      3.797      0.000       0.006       0.018
GRE Score          0.0210      0.007      3.178      0.002       0.008       0.034
TOEFL Score        0.0199      0.006      3.216      0.001       0.008       0.032
University Rating  0.0053      0.005      1.041      0.298      -0.005       0.015
SOP                0.0036      0.005      0.703      0.482      -0.006       0.014
LOR                0.0143      0.004      3.272      0.001       0.006       0.023
CGPA               0.0722      0.007     10.641      0.000       0.059       0.086
Research           0.0120      0.004      3.244      0.001       0.005       0.019
==============================================================================
Omnibus:                       68.942   Durbin-Watson:                   1.917
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              133.904
Skew:                          -0.946   Prob(JB):                     8.38e-30
Kurtosis:                       5.111   Cond. No.                         5.88
==============================================================================

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

✓ 0s completed at 19:11

CHECKING VIF

```
check_vif(X_train_new)
```

	Features	VIF
7	CGPA	5.02
2	GRE Score	4.75
3	TOEFL Score	4.16
5	SOP	2.88
4	University Rating	2.83
6	LOR	2.07
8	Research	1.48
1	Serial No.	1.05
0	const	1.00

INFERENCE: p_value of SOP is significantly higher than alpha(0.05), hence dropping it as it is insignificant in presence of other variables.

p_value of University Rating is significantly higher than alpha(0.05), hence dropping it as it is insignificant in presence of other variables.

AFTER DROPPING BOTH THE FEATURES CHECKING FOR MULTICOLLINEARITY USING VIF

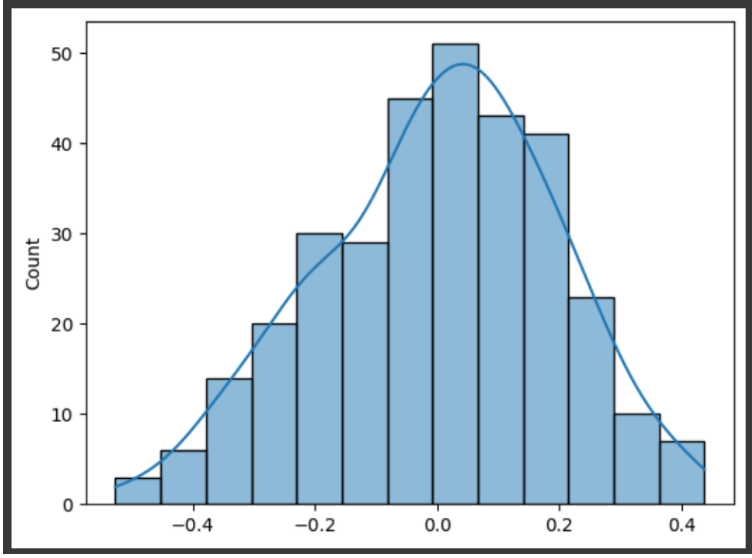
```
check_vif(X_train_new)
```

	Features	VIF
2	GRE Score	4.69
5	CGPA	4.66
3	TOEFL Score	4.09
4	LOR	1.70
6	Research	1.47
1	Serial No.	1.03
0	const	1.00

VIF is less than 5 hence there is no multicollinearity . Hence , we can go with the predictions.

RESIDUAL ANALYSIS OF MODEL

```
fig = plt.figure() # plot histogram of error terms
sns.histplot((y_train - y_train_admit), kde = True)
plt.show()
```

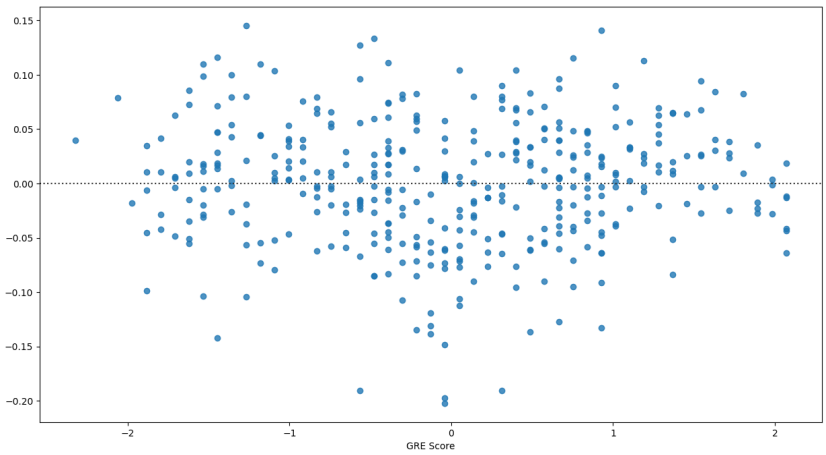


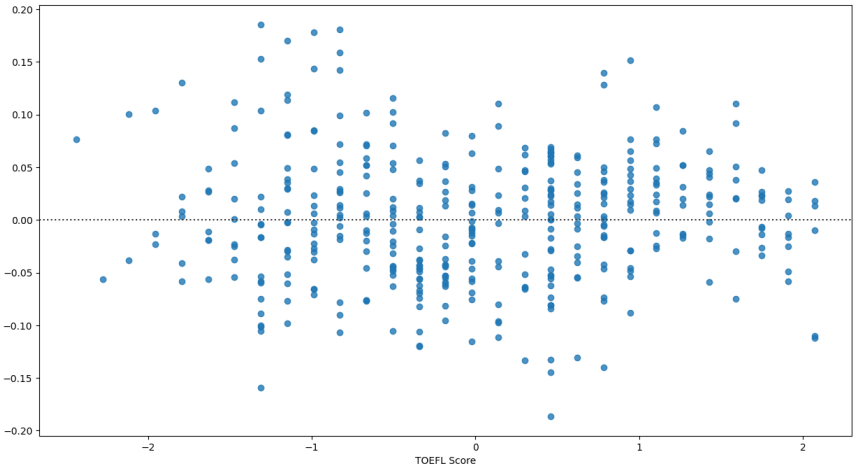
All the errors seems normally distributed, so assumptions of linear regression seems to be fulfilled.

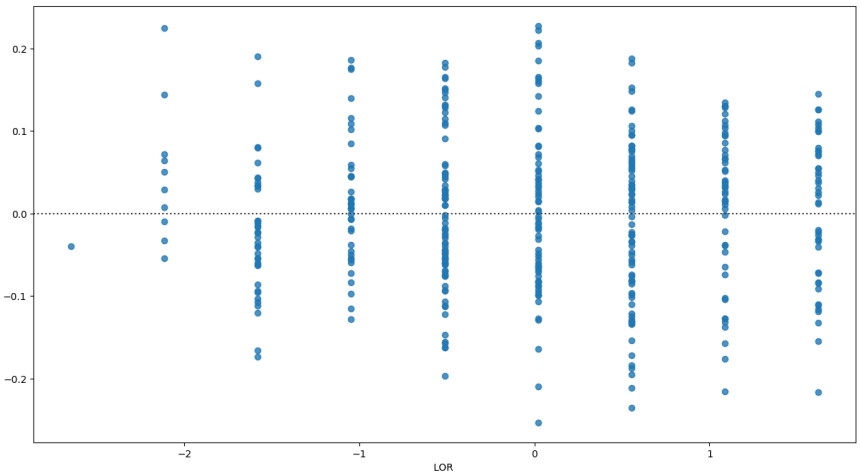
MEANS OF RESIDUALS

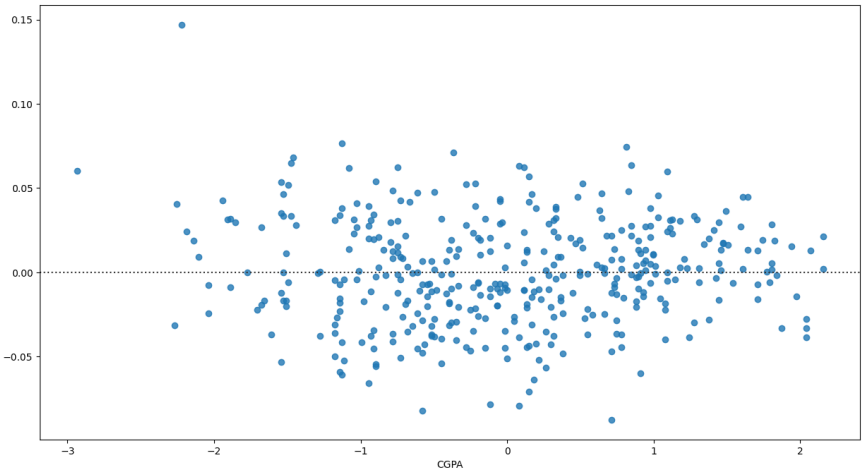
```
[113] residuals = y_train - y_train_admit
      np.mean(residuals)

      0.0004772549377836469
```

LINEARITY OF VARIABLES : RESIDUAL PLOT

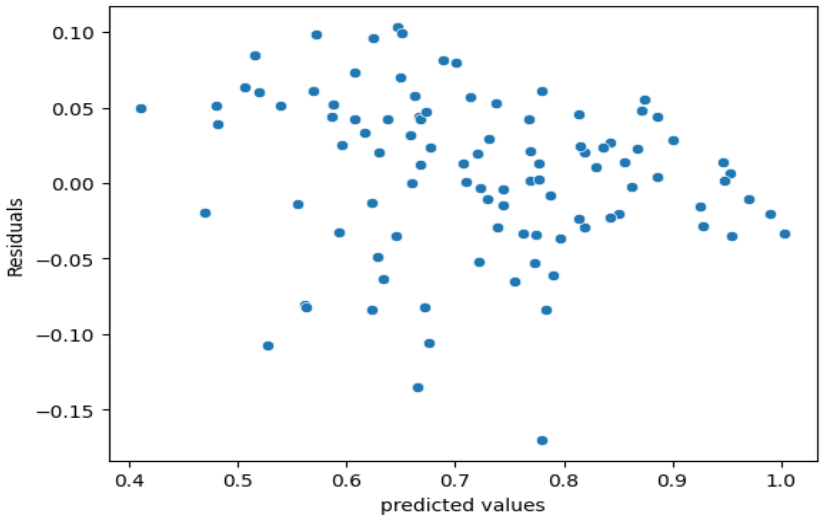








TEST FOR HOMOSCEDACITY




```
import statsmodels.stats.api as sas
from statsmodels.compat import lzip
name=['F statistics','p-value']
test=sas.het_goldfeldquandt(residuals,X_test)
lzip(name,test)
```

```
[('F statistics', 1.40945847797264), ('p-value', 0.1350494380216088)]
```

INFERENCE : Here null hypothesis is ;; errors terms are homoscedastic. Since, p_value >0.05 . hence , we fail to reject null hypothesis.

Hence error terms are homoscedastic.

MODEL PERFORMANCE EVALUATION

METRICS TO BE CHECKED: MAE , RMSE,R2, Adj. R2

```
Mean Absolute Error:  0.04211448792841083
============================================================
Root Mean Square Error:  0.05266759746087557
============================================================
R2 Score:  0.8404942127800776
============================================================
Adjusted. R2 Score:  0.8404942127800776
============================================================
```

PERFORMANCE TEST :TRAIN AND TEST DATASET

```
trainr2score  = r2_score(y_train, y_train_admit)
print("train r2 score :" ,trainr2score)
testr2score  = r2_score(y_test , y_pred)
print("test r2 score :", testr2score)
```

```
train r2 score : 0.8236769335970521
test r2 score : 0.8404942127800776
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.824
Model:                            OLS   Adj. R-squared:                  0.821
Method:                 Least Squares   F-statistic:                     306.0
Date:                Tue, 06 Feb 2024   Prob (F-statistic):          1.06e-144
Time:                        14:28:33   Log-Likelihood:                 557.02
No. Observations:                 400   AIC:                            -1100.
Df Residuals:                     393   BIC:                            -1072.
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.7205      0.003    237.605      0.000       0.715       0.726
Serial No.     0.0114      0.003      3.724      0.000       0.005       0.017
GRE Score      0.0217      0.007      3.299      0.001       0.009       0.035
TOEFL Score    0.0212      0.006      3.454      0.001       0.009       0.033
LOR            0.0170      0.004      4.303      0.000       0.009       0.025
CGPA           0.0750      0.007     11.452      0.000       0.062       0.088
Research       0.0125      0.004      3.403      0.001       0.005       0.020
==============================================================================
Omnibus:                       65.598   Durbin-Watson:                   1.918
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              125.015
Skew:                          -0.911   Prob(JB):                     7.13e-28
Kurtosis:                       5.044   Cond. No.                         4.97
==============================================================================
```

ACTIONABLE INSIGHTS AND RECOMMENDATIONS

1. R-sqaured and Adjusted R-squared (extent of fit) - 0.83 and 0.82 - 85% variance explained.
2. F-stats and Prob(F-stats) (overall model fit) - 387.9 and 1.03e-149(approx. 0.0) - Model fit is significant and explained 82% variance is just not by chance.
3. p-values - p-values for all the coefficients seem to be less than the significance level of 0.05. - meaning that all the predictors are statistically significant.
4. There is lot of chance for the model improvement by tunining the parameters.
5. Currently this models attains accuracy around 80%. This can be improved further by doing some feature engineering.
6. This model is not generalized, there is scope for the generalization of this model.
7. LogLikelihood is around 570 which indicates model is significantly fit.
8. Performance of training and test data is almost same indicates the model will work significantly on unseen data.
9. While observing the model and according to test assumptions - We can infer errors are homoscedasticity according to p-value
10. While observing the linearity of residual there is no significant pattern found which indicates the residual plots are not correlated
11. While observing the normality of residual - the distribution resembles like bell-shaped and the reg. line fits almost every point

COLAB LINK:

https://colab.research.google.com/drive/1CXCQkRhdOnbiQFtAOJ0qdKYEMyswh_xw?usp=sharing