Project Development Phase
Model Performance Test

Date	18 NOvember 2022
Team ID	PNT2022TMID32941
Project Name	Project – UNIVERSITY ADMIT ELIGIBILITY PREDICTOR
Maximum Marks	10 Marks

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE -0.043051, MSE - 0.003313, RMSE - 0.057560, R2 score - 0.807216 **Classification Model:** Confusion Matrix , Accuray Score- 0.872& Classification Report.	```In [47]: mae = metrics.mean_absolute_error(y_test, predlinear)
mse = metrics.mean_squared_error(y_test, predlinear)
rmse = np.sqrt(mse) # or mse**(0.5)
r2 = metrics.r2_score(y_test, predlinear)``` ```In [48]: chart = {
        'Metric':["MAE", "MSE", "RMSE", "R2-SCORE"],
        'LINEAR_REGRESSION':[mae,mse,rmse,r2],
        }
chart = pd.DataFrame(chart)``` ```In [49]: display(chart)```
 Metric LINEAR_REGRESSION
 0 MAE 0.043051
 1 MSE 0.003313
 2 RMSE 0.057560
 3 R2-SCORE 0.807216

 ```In [43]: model = LinearRegression(normalize=True)
model.fit(X_test, y_test)
# model.score(X_test, y_test)
predlinear = model.predict(X_test)
print ("Accuracy : ",model.score(X_test, y_test)*100)
methodDict = {}
methodDict['Linear Regression'] = model.score(X_test, y_test)*100

Accuracy : 80.7216438856893``` |
| 2. | Tune the Model | Hyperparameter Tuning. Validation Method. | ```In [64]: scores = cross_val_score(model, X_train, y_train, scoring='r2', cv=5)
scores``` ```Out[64]: array([0.81813967, 0.77169539, 0.83989563, 0.74719974, 0.78589678])``` ```In [65]: avg_score=scores.mean()``` ```In [67]: print ("Cross Validation Scores : ",scores)
print ("Average CV Score : ",avg_score)
print ("Number of CV Scores used in Average : ",len(scores))

Cross Validation Scores : [0.81813967 0.77169539 0.83989563 0.74719974 0.78589678]
Average CV Score : 0.7925654408790849
Number of CV Scores used in Average : 5``` |