PROJECT DEVELOPMENT PHASE

MODEL PERFORMANCE TEST

Date	18 November 2022
Team ID	PNT2022TMID21492
Project Name	University Admit Eligibility Predictor
Maximum Marks	10 Marks

Model Performance Testing:

S.NO	PARAMETERS	VALUES	SCREENSHOTS
1	Metrics	**Regression Model:** MAE - , MSE - , RMSE - , R2 score - **Classification Model:** Confusion Matrix - , Accuray Score- & Classification Report	

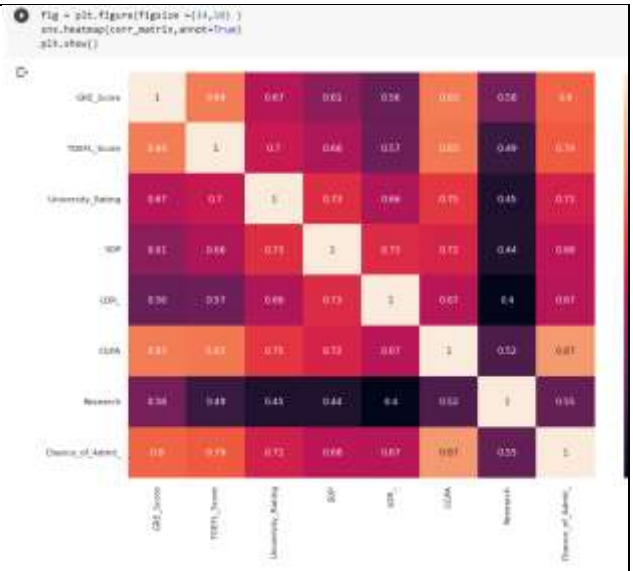
```
from sklearn.metrics import r2_score,mean_squared_error,mean_absolute_error
from math import sqrt
RMSE=float(format(np.sqrt(mean_squared_error(y_test,rf_y_pred))))
MAE=mean_absolute_error(y_test,rf_y_pred)
MSE=mean_squared_error(y_test,rf_y_pred)
R2=r2_score(y_test,rf_y_pred)

print('RMSE :', RMSE, '\nMSE :' ,MSE, '\nMAE :',MAE, '\nR2 score:',R2)
```

```
RMSE : 0.25
MSE : 0.0625
MAE : 0.0625
R2 score: 0.686520376175548 6
```

```
[53] corr_matrix = df.corr()
     corr_matrix
```

	GRE_Score	TOEFL_Score	University_Rating	SOP	LOR_	CGPA	Research	Chance_of_Admit_
GRE_Score	1.000000	0.835977	0.668976	0.612831	0.557555	0.833060	0.580391	0.802610
TOEFL_Score	0.835977	1.000000	0.695590	0.657981	0.567721	0.828417	0.489858	0.791594
University_Rating	0.668976	0.695590	1.000000	0.734523	0.660123	0.746479	0.447783	0.711250
SOP	0.612831	0.657981	0.734523	1.000000	0.729593	0.718144	0.444029	0.675732
LOR_	0.557555	0.567721	0.660123	0.729593	1.000000	0.670211	0.396858	0.669889
CGPA	0.833060	0.828417	0.746479	0.718144	0.670211	1.000000	0.521654	0.873289
Research	0.580391	0.489858	0.447783	0.444029	0.396858	0.521654	1.000000	0.553202
Chance_of_Admit_	0.802610	0.791594	0.711250	0.675732	0.669889	0.873289	0.553202	1.000000

			
			[56] `from sklearn.ensemble import RandomForestClassifier`
			`from sklearn.metrics import accuracy_score`
			`rf = RandomForestClassifier()`
			`trained_model = rf.fit(X_train,y_train)`
			[57] `import pickle`
			`pickle.dump(rf,open('university.pkl','wb'))`
			[58] `rf_y_pred = rf.predict(X_test)`
			`rf_accuracy = accuracy_score(rf_y_pred,y_test)`
			`print("Accuracy: ",rf_accuracy*100,"%")`
			Accuracy: 92.8125 %

2	Tune the Model	Hyperparameter Tuning - Validation Method	`from sklearn.linear_model import LogisticRegression`
			`from sklearn.model_selection import GridSearchCV`
			`c_space = np.logspace(-5, 8, 15)`
			`param_grid = {'C': c_space}`
			`logreg = LogisticRegression()`
			`logreg_cv = GridSearchCV(logreg, param_grid, cv = 5)`
			`logreg_cv.fit(X_train, y_train)`
			`print("Tuned Logistic Regression Parameters: {}".format(logreg`
			`print("Best score is {}".format(logreg_cv.best_score_))`
			Tuned Logistic Regression Parameters: {'C': 0.4393970560760
			Best score is 0.9