

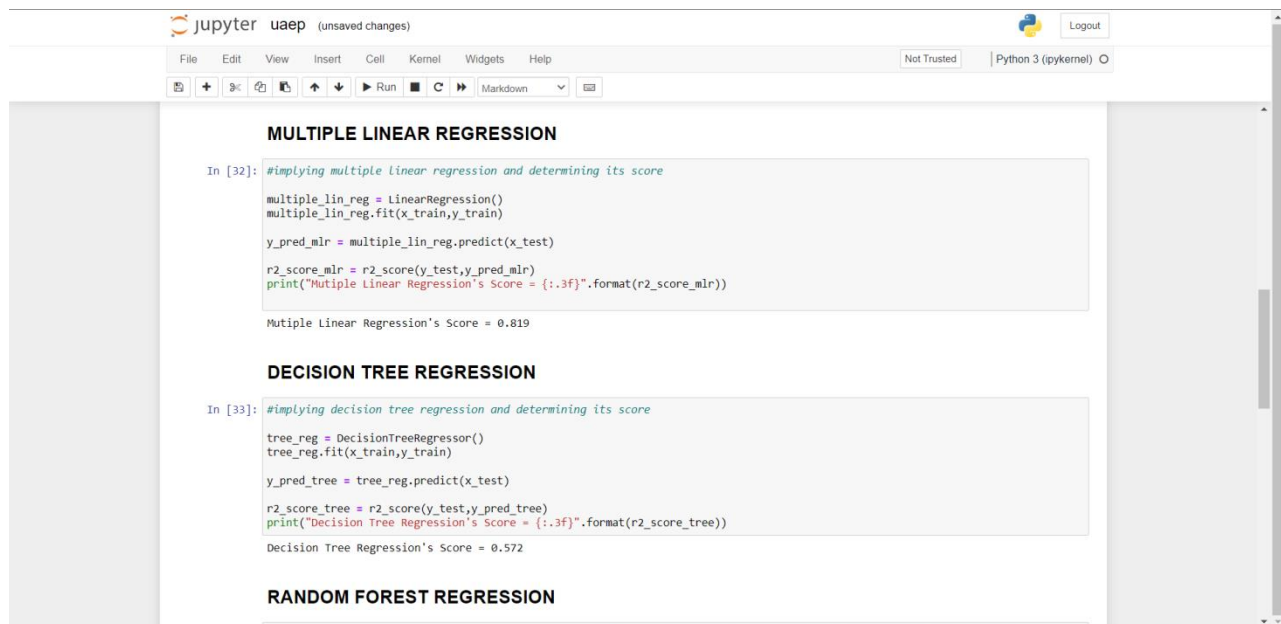
Project Development Phase Model Performance Test

Date	17 November 2022
Team ID	PNT2022TMID49628
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-, MSE-, RMSE-, R2 score- Classification Model: Confusion Matrix - Accuracy Score- & Classification Report-	See Below
2	Tune the Model	Hyper parameter Tuning-Validation Method-	See Below



The screenshot displays a Jupyter Notebook interface with the following content:

- Multiple Linear Regression**
In [32]: `#implying multiple linear regression and determining its score`
`multiple_lin_reg = LinearRegression()`
`multiple_lin_reg.fit(x_train,y_train)`
`y_pred_mlr = multiple_lin_reg.predict(x_test)`
`r2_score_mlr = r2_score(y_test,y_pred_mlr)`
`print("Multiple Linear Regression's Score = {:.3f}".format(r2_score_mlr))`
Output: `Multiple Linear Regression's Score = 0.819`
- Decision Tree Regression**
In [33]: `#implying decision tree regression and determining its score`
`tree_reg = DecisionTreeRegressor()`
`tree_reg.fit(x_train,y_train)`
`y_pred_tree = tree_reg.predict(x_test)`
`r2_score_tree = r2_score(y_test,y_pred_tree)`
`print("Decision Tree Regression's Score = {:.3f}".format(r2_score_tree))`
Output: `Decision Tree Regression's Score = 0.572`
- Random Forest Regression**

RANDOM FOREST REGRESSION

```
In [34]: #implying random forest regression and determining its score

ran_for_reg = RandomForestRegressor(n_estimators=100,random_state=42)
ran_for_reg.fit(x_train,y_train)

y_pred_rfr = ran_for_reg.predict(x_test)

r2_score_rfr = r2_score(y_test,y_pred_rfr)
print("Random Forest Regression's Score = {:.3f}".format(r2_score_rfr))

Random Forest Regression's Score = 0.787
```

CONCLUSION

R² score is an indicator of accuracy of Regression Models, and the accuracy is measured as close to 1 of this value. Therefore, as seen, Multiple Linear Regression Model is better than Decision Tree Regression and Random Forest Regression on this dataset when comparing their R² scores.

Persisting the Multiple Linear Regression Model in a Pickle File

```
In [35]: import pickle
pickle.dump(multiple_lin_reg, open("Multiple_Linear_Regression.pkl", "wb"))
```

PERFORMANCE TESTING

```
In [37]: #Finding Model Performance Metrics of the finalised model used
```

PERFORMANCE TESTING

```
In [37]: #Finding Model Performance Metrics of the finalised model used
from sklearn import metrics
x = df[["GRE Score","TOEFL Score","University Rating","SOP","LOR ","CGPA", "Research"]]
y = df["Chance of Admit "].values.reshape(-1,1)

x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.2,random_state=42)

multiple_lin_reg = LinearRegression()
multiple_lin_reg.fit(x_train,y_train)

y_pred = multiple_lin_reg.predict(x_test)
print("Multiple Linear Regression")
print("Mean Absolute Error      :", metrics.mean_absolute_error(y_test,y_pred))

print("Mean Squared Error       :", metrics.mean_squared_error(y_test,y_pred))

print("Root Mean Squared Error :", metrics.mean_squared_error(y_test,y_pred,squared=False))

print("R2 Score                  :", metrics.r2_score(y_test,y_pred))

Multiple Linear Regression
Mean Absolute Error      : 0.04272265427705367
Mean Squared Error       : 0.0037046553987884114
Root Mean Squared Error  : 0.06086580041578312
R2 Score                  : 0.8188432567829628
```