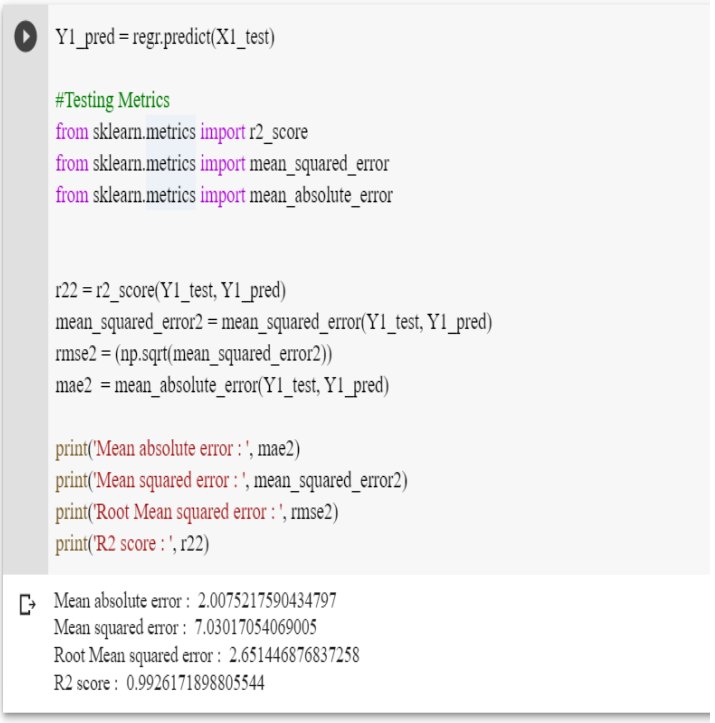
Project Development Phase
Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID35251
Project Name	Developing a Flight delay prediction model using Machine Learning
Maximum Marks	10 Marks

Model Performance Testing:

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Random Forest:** MAE - 2.00 MSE - 7.03 RMSE - 2.65 R2 score - 0.99	

		Logistic Regression MAE - 6.88 MSE - 6.77 RMSE - 8.23 R2 score - 1.00 It is clear from the results that the model is overfitting.	

2.	Tune the Model	Hyperparameter Tuning - The depth of the regressor is taken as the hyperparameter. We can see that the results improve by increasing its value.	``` from sklearn.ensemble import RandomForestRegressor from sklearn.datasets import make_regression regr = RandomForestRegressor(max_depth=2, random_state=0) regr.fit(X1_train, Y1_train) #r_sq3 = regr.score(X1_train, Y1_train) r_sq3 = regr.score(X1_test, Y1_test) print(f"Determination coefficient: {r_sq3}") ``` Determination coefficient: 0.8385558200636558 ``` [78] Y1_pred = regr.predict(X1_test) #Testing Metrics from sklearn.metrics import r2_score from sklearn.metrics import mean_squared_error from sklearn.metrics import mean_absolute_error r22 = r2_score(Y1_test, Y1_pred) mean_squared_error2 = mean_squared_error(Y1_test, Y1_pred) rmse2 = (np.sqrt(mean_squared_error2)) mae2 = mean_absolute_error(Y1_test, Y1_pred) print('Mean absolute error : ', mae2) print('Mean squared error : ', mean_squared_error2) print('Root Mean squared error : ', rmse2) print('R2 score : ', r22) ``` Mean absolute error : 9.44220570961539 Mean squared error : 153.73280626098187 Root Mean squared error : 12.398903429778855 R2 score : 0.8385558200636558 Model with depth 2

Model with depth 3

```
[79] from sklearn.ensemble import RandomForestRegressor
     from sklearn.datasets import make_regression
     regr = RandomForestRegressor(max_depth=3, random_state=0)
     regr.fit(X1_train, Y1_train)
     #r_sq3 = regr.score(X1_train, Y1_train)
     r_sq3 = regr.score(X1_test, Y1_test)
     print(f'Determination coefficient: {r_sq3}')
```

```
Determination coefficient: 0.961670182382008
```

```
[80] Y1_pred = regr.predict(X1_test)

     #Testing Metrics
     from sklearn.metrics import r2_score
     from sklearn.metrics import mean_squared_error
     from sklearn.metrics import mean_absolute_error


     r22 = r2_score(Y1_test, Y1_pred)
     mean_squared_error2 = mean_squared_error(Y1_test, Y1_pred)
     rmse2 = (np.sqrt(mean_squared_error2))
     mae2  = mean_absolute_error(Y1_test, Y1_pred)

     print('Mean absolute error : ', mae2)
     print('Mean squared error : ', mean_squared_error2)
     print('Root Mean squared error : ', rmse2)
     print('R2 score : ', r22)
```

```
Mean absolute error :  4.72516024026673
Mean squared error :  36.498995678933255
Root Mean squared error :  6.041439868022627
R2 score :  0.961670182382008
```

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import make_regressi
regr = RandomForestRegressor(max_depth=4, random_state=0)
regr.fit(X1_train, Y1_train)
#r_sq3 = regr.score(X1_train, Y1_train)
r_sq3 = regr.score(X1_test, Y1_test)
print(f"Determination coefficient: {r_sq3}")
```

Loading...

```
Determination coefficient: 0.9926171898805544
```

```
[ ]  Y1_pred = regr.predict(X1_test)

#Testing Metrics
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_error


r22 = r2_score(Y1_test, Y1_pred)
mean_squared_error2 = mean_squared_error(Y1_test, Y1_pred)
rmse2 = (np.sqrt(mean_squared_error2))
mae2  = mean_absolute_error(Y1_test, Y1_pred)

print('Mean absolute error : ', mae2)
print('Mean squared error : ', mean_squared_error2)
print('Root Mean squared error : ', rmse2)
print('R2 score : ', r22)
```

```
Mean absolute error :  2.0075217590434797
Mean squared error :  7.03017054069005
Root Mean squared error :  2.651446876837258
R2 score :  0.9926171898805544
```

Model with depth 4