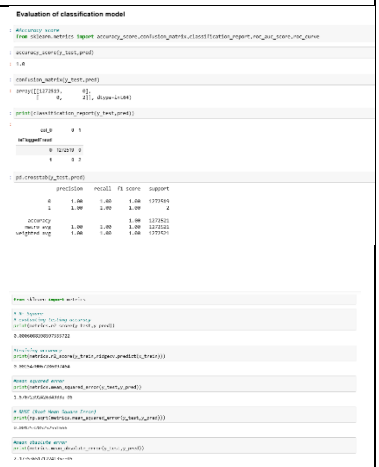
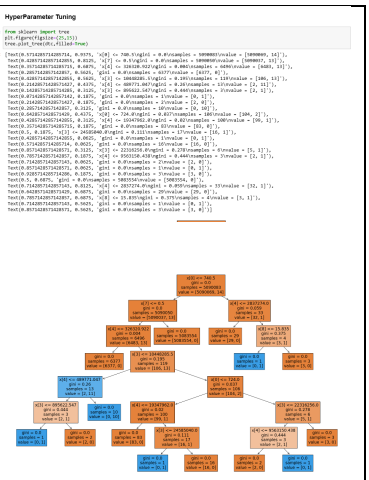
Project Development Phase
Model Performance Test

Date	17 November 2023
Team Leader	M. Akash
Project Name	Project – Online Payments Fraud Detection Using ML

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 – 2.1735365171224135e-05 MSE – 1.5707239242634933e-06 RMSE – 0.001253285252551666 R2 score –0.0006088398597555722 **Classification Model:** Confusion Matrix – Accuray Score- 1.0 Classification Report –	

2.	Tune the Model	Hyperparameter Tuning - Validation Method -	

Regression Model:

```
from sklearn import metrics
```

```
# R- Square
# evaluating testing accuracy
print(metrics.r2_score(y_test,y_pred))
```

```
0.0006088398597555722
```

```
#training accuracy
print(metrics.r2_score(y_train,ridgecv.predict(x_train)))
```

```
0.0025470067286097464
```

```
#mean squared error
print(metrics.mean_squared_error(y_test,y_pred))
```

```
1.5707239242634933e-06
```

```
# RMSE (Root Mean Square Error)
print(np.sqrt(metrics.mean_squared_error(y_test,y_pred)))
```

```
0.001253285252551666
```

```
#mean absolute error
print(metrics.mean_absolute_error(y_test,y_pred))
```

```
2.1735365171224135e-05
```

Classification Model:

Evaluation of classification model

```
#Accuracy score
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,roc_auc_score,roc_curve
```

```
accuracy_score(y_test,pred)
```

```
1.0
```

```
confusion_matrix(y_test,pred)
```

```
array([[1272519,       0],
       [      0,       2]], dtype=int64)
```

```
print(classification_report(y_test,pred))
```

```
pd.crosstab(y_test,pred)
```

col_0	0	1
isFlaggedFraud		
0	1272519	0
1	0	2

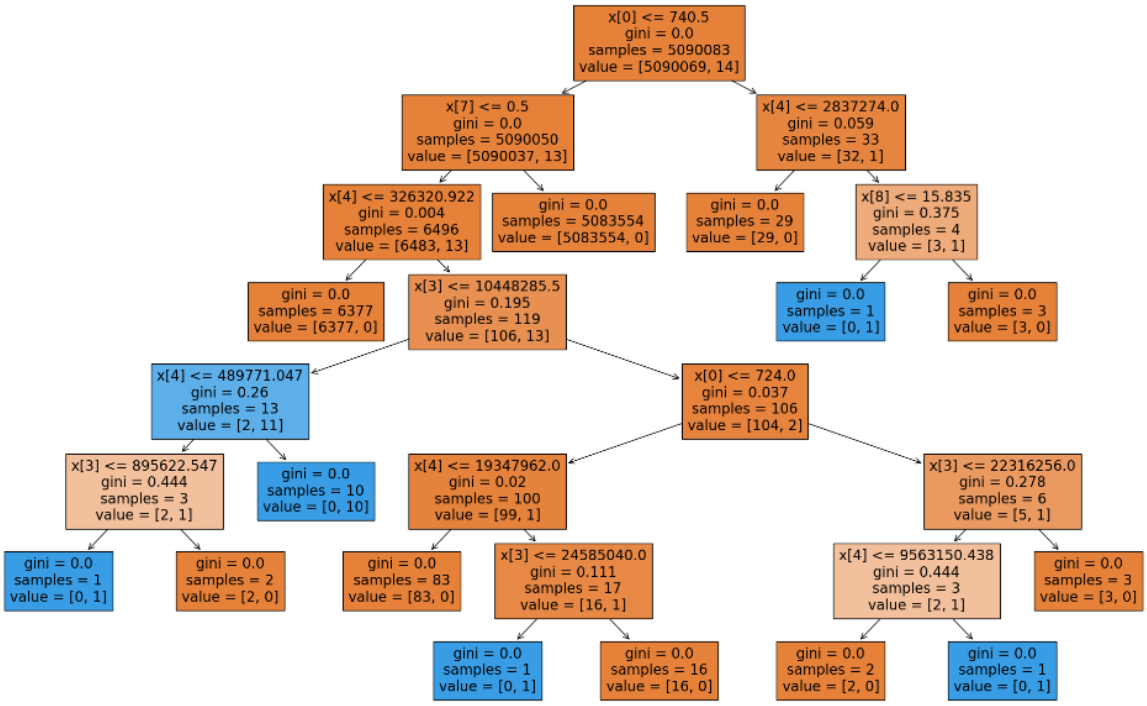
```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00   1272519
           1       1.00      1.00      1.00         2

    accuracy                           1.00   1272521
   macro avg       1.00      1.00      1.00   1272521
weighted avg       1.00      1.00      1.00   1272521
```

Hyperparameter Tuning:

HyperParameter Tuning

```
from sklearn import tree
plt.figure(figsize=(25,15))
tree.plot_tree(dtc,filled=True)
```

```
[Text(0.5714285714285714, 0.9375, 'x[0] <= 740.5\ngini = 0.0\nsamples = 5090083\nvalue = [5090069, 14]'),
 Text(0.42857142857142855, 0.8125, 'x[7] <= 0.5\ngini = 0.0\nsamples = 5090050\nvalue = [5090037, 13]'),
 Text(0.35714285714285715, 0.6875, 'x[4] <= 326320.922\ngini = 0.004\nsamples = 6496\nvalue = [6483, 13]'),
 Text(0.2857142857142857, 0.5625, 'gini = 0.0\nsamples = 6377\nvalue = [6377, 0]'),
 Text(0.42857142857142855, 0.5625, 'x[3] <= 10448285.5\ngini = 0.195\nsamples = 119\nvalue = [106, 13]'),
 Text(0.21428571428571427, 0.4375, 'x[4] <= 489771.047\ngini = 0.26\nsamples = 13\nvalue = [2, 11]'),
 Text(0.14285714285714285, 0.3125, 'x[3] <= 895622.547\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.07142857142857142, 0.1875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.21428571428571427, 0.1875, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.2857142857142857, 0.3125, 'gini = 0.0\nsamples = 10\nvalue = [0, 10]'),
 Text(0.6428571428571429, 0.4375, 'x[0] <= 724.0\ngini = 0.037\nsamples = 106\nvalue = [104, 2]'),
 Text(0.42857142857142855, 0.3125, 'x[4] <= 19347962.0\ngini = 0.02\nsamples = 100\nvalue = [99, 1]'),
 Text(0.35714285714285715, 0.1875, 'gini = 0.0\nsamples = 83\nvalue = [83, 0]'),
 Text(0.5, 0.1875, 'x[3] <= 24585040.0\ngini = 0.111\nsamples = 17\nvalue = [16, 1]'),
 Text(0.42857142857142855, 0.0625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.5714285714285714, 0.0625, 'gini = 0.0\nsamples = 16\nvalue = [16, 0]'),
 Text(0.8571428571428571, 0.3125, 'x[3] <= 22316256.0\ngini = 0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.7857142857142857, 0.1875, 'x[4] <= 9563150.438\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.7142857142857143, 0.0625, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.8571428571428571, 0.0625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.9285714285714286, 0.1875, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.5, 0.6875, 'gini = 0.0\nsamples = 5083554\nvalue = [5083554, 0]'),
 Text(0.7142857142857143, 0.8125, 'x[4] <= 2837274.0\ngini = 0.059\nsamples = 33\nvalue = [32, 1]'),
 Text(0.6428571428571429, 0.6875, 'gini = 0.0\nsamples = 29\nvalue = [29, 0]'),
 Text(0.7857142857142857, 0.6875, 'x[8] <= 15.835\ngini = 0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.7142857142857143, 0.5625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.8571428571428571, 0.5625, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]')]
```




```
from sklearn.model_selection import GridSearchCV
parameter={
 'criterion':['gini','entropy'],
  'splitter':['best','random'],
  'max_depth':[1,2,3,4,5],
  'max_features':['auto', 'sqrt', 'log2']
}
```

```
grid_search=GridSearchCV(estimator=dtc,param_grid=parameter,cv=5,scoring="accuracy")
```

```
grid_search.fit(x_train,y_train)
```

```
C:\Users\HP\anaconda3\Lib\site-packages\sklearn\model_selection\_validation.py:425: FitFailedWarning:
100 fits failed out of a total of 300.
The score on these train-test partitions for these parameters will be set to nan.
If these failures are not expected, you can try to debug them by setting error_score='raise'.

Below are more details about the failures:
--------------------------------------------------------------------------------
100 fits failed with the following error:
Traceback (most recent call last):
  File "C:\Users\HP\anaconda3\Lib\site-packages\sklearn\model_selection\_validation.py", line 729, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "C:\Users\HP\anaconda3\Lib\site-packages\sklearn\base.py", line 1145, in wrapper
    estimator._validate_params()
  File "C:\Users\HP\anaconda3\Lib\site-packages\sklearn\base.py", line 638, in _validate_params
    validate_parameter_constraints(
  File "C:\Users\HP\anaconda3\Lib\site-packages\sklearn\utils\_param_validation.py", line 96, in validate_parameter_constraints
    raise InvalidParameterError(
sklearn.utils._param_validation.InvalidParameterError: The 'max_features' parameter of DecisionTreeClassifier must be an int in
the range [1, inf), a float in the range (0.0, 1.0], a str among {'sqrt', 'log2'} or None. Got 'auto' instead.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
C:\Users\HP\anaconda3\Lib\site-packages\sklearn\model_selection\_search.py:979: UserWarning: One or more of the test scores are
non-finite: [       nan        nan 0.99999725 0.99999725 0.99999725 0.99999725
        nan        nan 0.99999725 0.99999725 0.99999725 0.99999725
        nan        nan 0.99999725 0.99999725 0.99999725 0.99999725
        nan        nan 0.99999705 0.99999725 0.99999725 0.99999764
        nan        nan 0.99999823 0.99999745 0.99999705 0.99999725
        nan        nan 0.99999725 0.99999725 0.99999725 0.99999725
        nan        nan 0.99999725 0.99999725 0.99999725 0.99999725
        nan        nan 0.99999725 0.99999764 0.99999745 0.99999725
        nan        nan 0.99999764 0.99999725 0.99999764 0.99999725
        nan        nan 0.99999823 0.99999725 0.99999745 0.99999745]
  warnings.warn(
```

```
GridSearchCV(cv=5, estimator=DecisionTreeClassifier(),
             param_grid={'criterion': ['gini', 'entropy'],
                         'max_depth': [1, 2, 3, 4, 5],
                         'max_features': ['auto', 'sqrt', 'log2'],
                         'splitter': ['best', 'random']},
             scoring='accuracy')
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

```
pred=dtc_cv.predict(x_test)
```

```
print(classification_report(y_test,pred))
```

```
C:\Users\HP\anaconda3\Lib\site-packages\sklearn\metrics\_classification.py:1471: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavio
r.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\HP\anaconda3\Lib\site-packages\sklearn\metrics\_classification.py:1471: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavio
r.
  _warn_prf(average, modifier, msg_start, len(result))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1272519
1	0.00	0.00	0.00	2
accuracy			1.00	1272521
macro avg	0.50	0.50	0.50	1272521
weighted avg	1.00	1.00	1.00	1272521

Validation Method:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

▾ RandomForestClassifier

RandomForestClassifier()

```
y_test_predict2=rfc.predict(x_test)
test_accuracy=accuracy_score (y_test,y_test_predict2)
test_accuracy
```

1.0

```
y_train_predict2=rfc.predict(x_train)
train_accuracy=accuracy_score (y_train,y_train_predict2)
train_accuracy
```

1.0

]: `pd.crosstab(y_test,y_test_predict2)`

]:

col_0	0	1
isFlaggedFraud		
0	1272519	0
1	0	2

]: `print(classification_report(y_test,y_test_predict2))`

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00   1272519
           1       1.00      1.00      1.00         2

    accuracy                           1.00   1272521
   macro avg       1.00      1.00      1.00   1272521
weighted avg       1.00      1.00      1.00   1272521
```