Project Development Phase
Model Performance Test

Date	19 November 2022
Team ID	PNT2022TMID30140
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
1.	Metrics	**Classification Model:** Confusion Matrix – [115,15,15,92] Accuracy Score-87.34 Classification Report – 89.85
	Tune the Model	Hyperparameter Tuning – 88.22 Validation Method – RandomizedCV

Screenshots:

Confusion Matrix:

```
    #Creating the Confusion matrix
    from sklearn.metrics import confusion_matrix,accuracy_score
    cm= confusion_matrix(y_test, y_pred)
    cm
]
```

```
array([[115,  15],
       [ 15,  92]], dtype=int64)
```

Accuracy Score:

```
    accuracy_score(y_pred,y_test)  * 100
```

```
87.34177215189874
```

```
    classifier.score(x_train,y_train) *100
```

```
89.85507246376811
```

```
    classifier.score(x_test,y_test)*100
```

```
87.34177215189874
```

Hyper Parameter Tuning:

```
In [75]: from sklearn.model_selection import RandomizedSearchCV
         rs = RandomizedSearchCV(svm.SVC(gamma='auto'), {
                 'C': [1,10,20],
                 'kernel': ['rbf','linear']
             },
             cv=5,
             return_train_score=False,
             n_iter=2
         )
         rs.fit(x_train,y_train)
```

```
In [62]: (rs.best_score_ ) * 100
Out[62]: 88.22113022113022

In [63]: (rs.best_params_)
Out[63]: {'kernel': 'linear', 'C': 10}

In [67]: rs_tuned = SVC(C =10,kernel = 'linear')
         rs_tuned
Out[67]: SVC(C=10, kernel='linear')

In [68]: rs_tuned.fit(x_train,y_train)
         C:\Users\A.Afrinbanu\anaconda3\lib\site-packages\sklearn\utils\validation.py:72: DataConversionWarning: A column-vector y was p
         assed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
           return f(**kwargs)
Out[68]: SVC(C=10, kernel='linear')

In [71]: Pred_tuned=rs_tuned.predict(x_test)
         Pred_tuned
Out[71]: array([1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1,
                1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,
                1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0,
                1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0,
                0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0,
                0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0,
                1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0,
```

```
Pred_tuned
```

Out[71]:
```
array([1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1,
       1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,
       1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0,
       1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0,
       0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0,
       0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0,
       1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0,
       1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1,
       1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
       0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1,
       0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1])
```

In [73]:
```
accuracy_score(Pred_tuned,y_test)
```

Out[73]:
```
0.8354430379746836
```

In [74]:
```
confusion_matrix(Pred_tuned,y_test)
```

Out[74]:
```
array([[110,  19],
       [ 20,  88]], dtype=int64)
```