Team ID: PNT2022TMID11450

MODEL BUILDING

Model Evaluation

- There are many numbers of model evaluation techniques for the classification type of machine learning models.
- The following are widely used
  - Accuracy_score
  - Confusion matrix
  - Roc- Auc Curve

Accuracy

- One of the widely used metrics that computes the performance of classification models is accuracy. The percentage of labels that our model successfully predicted is represented by accuracy. For instance, if our model accurately classified 80 of 100 labels, its accuracy would be 0.80.

Accuracy_score:

- The accuracy_score () method of sklearn.metrics, accept the true labels of the sample and the labels predicted by the model as its parameters and computes the accuracy score as a float value, which can likewise be used to obtain the accuracy score in Python. There are several helpful functions to compute typical evaluation metrics in the sklearn.metrics class.

Recall_score:

- Recall_score computes a precision-recall curve from the ground truth label and a score given by the classifier by varying a decision threshold.

Roc_auc_score:

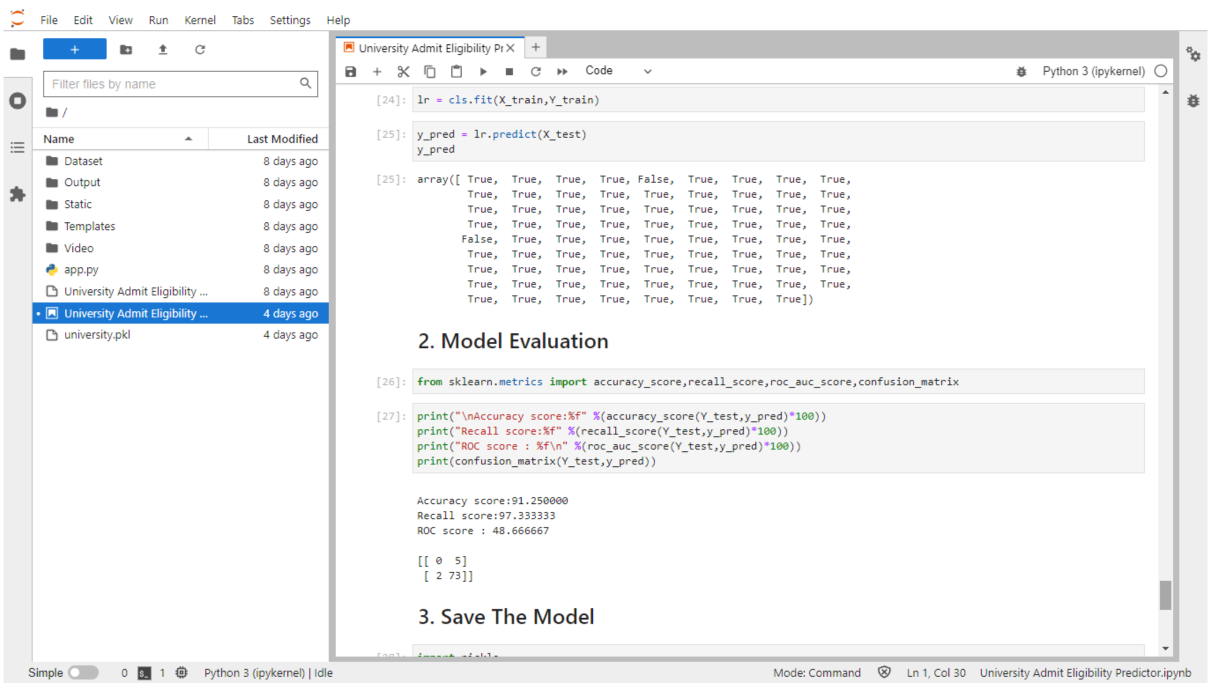
- Roc_auc_score compute Area Under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores.

Confusion_matrix:

- Confusion_matrix compute confusion matrix to evaluate the accuracy of a classification.

<u>Step-1:</u>

- After training the data, we need to check accuracy rate by sklearn.metrics library.
- Import **sklearn.metrics** and import **accuracy_score, roc_auc_score** and **confusion_matrix.**
- Then display the accuracy_score, roc_auc_score and confusion_matrix values as output from Trained models.



```
[24]: lr = cls.fit(X_train,Y_train)

[25]: y_pred = lr.predict(X_test)
      y_pred

[25]: array([ True,  True,  True,  True, False,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True,  True,
             False,  True,  True,  True,  True,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True,  True,
              True,  True,  True,  True,  True,  True,  True,  True])
```

2. Model Evaluation

```
[26]: from sklearn.metrics import accuracy_score,recall_score,roc_auc_score,confusion_matrix

[27]: print("\nAccuracy score:%f" %(accuracy_score(Y_test,y_pred)*100))
      print("Recall score:%f" %(recall_score(Y_test,y_pred)*100))
      print("ROC score : %f\n" %(roc_auc_score(Y_test,y_pred)*100))
      print(confusion_matrix(Y_test,y_pred))


      Accuracy score:91.250000
      Recall score:97.333333
      ROC score : 48.666667

      [[ 0  5]
       [ 2 73]]
```

3. Save The Model