NAME : NAGAVENI L G

SEC : 5F

SRN :PES2UG21CS315

LAB2 -ANN

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score,
precision_score,recall_score,f1_score,confusion_matrix




def split_and_standardize(X,y):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=0)
    Standardize = StandardScaler()
    X_train = Standardize.fit_transform(X_train)
    X_test = Standardize.transform(X_test)
    return (X_train, X_test, y_train, y_test)







def create_model(X_train,y_train):
    model1 = MLPClassifier(hidden_layer_sizes=(44, 44, 44), activation='relu',
max_iter=45, random_state=1)
    model2 = MLPClassifier(hidden_layer_sizes=(33, 33, 33),
activation='logistic',  max_iter=100,random_state=1)
    model1.fit(X_train, y_train)
    model2.fit(X_train, y_train)
    return (model1, model2)







def predict_and_evaluate(model,X_test,y_test):
    y_predicted = model.predict(X_test)
```

```
    accuracy = accuracy_score(y_test, y_predicted)
    precision = precision_score(y_test, y_predicted, average='weighted',
zero_division=1)
    recall = recall_score(y_test, y_predicted, average='weighted',
zero_division=1)
    f_score = f1_score(y_test, y_predicted, average='weighted',
zero_division=1)
    confusion = confusion_matrix(y_test, y_predicted)
    return (accuracy, precision, recall, f_score, confusion)
```

OUTPUT :

```
PS C:\Users\Praka\OneDrive\Documents\5thSem\MI\ANN (Student)> python Test.py --ID EC_F_PES2UG21CS315_Lab2
Test Case 1 for the function split_and_standardize PASSED
Test Case 2 for the function create_model PASSED
Test Case 3 for the function predict_and_evaluate PASSED
Test Case 4 for the function predict_and_evaluate PASSED
PS C:\Users\Praka\OneDrive\Documents\5thSem\MI\ANN (Student)>
```