Experiment 3

Student Name: Rahul Saxena

UID: 24MCI10204

Branch: MCA (AI-ML)

Section/Group: MAM - 3(B)

Semester: II

Subject Name: Machine Learning Lab [24CAP-672]

Aim: Write a program to implement Logistic Regression using Python in Jupyter Notebook.

Steps:

- **1. Data Preparation**
  - Create the dataset with input features (X) and target labels (y).
  - Split the dataset into training and testing sets using train_test_split() from sklearn.model_selection.
- **2. Train the Logistic Regression Model**
  - Import LogisticRegression from sklearn.linear_model.
  - Initialize the Logistic Regression model.
  - Train the model using the fit() method on the training data.
- **3. Predict and Evaluate**
  - Predict the labels for the test dataset using the predict() method.
  - Evaluate the model performance using:
    - **Accuracy Score** (accuracy_score)
    - **Confusion Matrix** (confusion_matrix)

Source Code:

```
 import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

X = np.array([
 [2.5, 2.4],
 [1.0, 1.2],
 [3.5, 3.2],
 [5.0, 3.0],
 [4.5, 5.0],
 [6.0, 4.5],
 [7.8, 5.0],
 [4.8, 4.5]
])

y = np.array([0, 0, 0, 1, 1, 1, 1, 1])
```

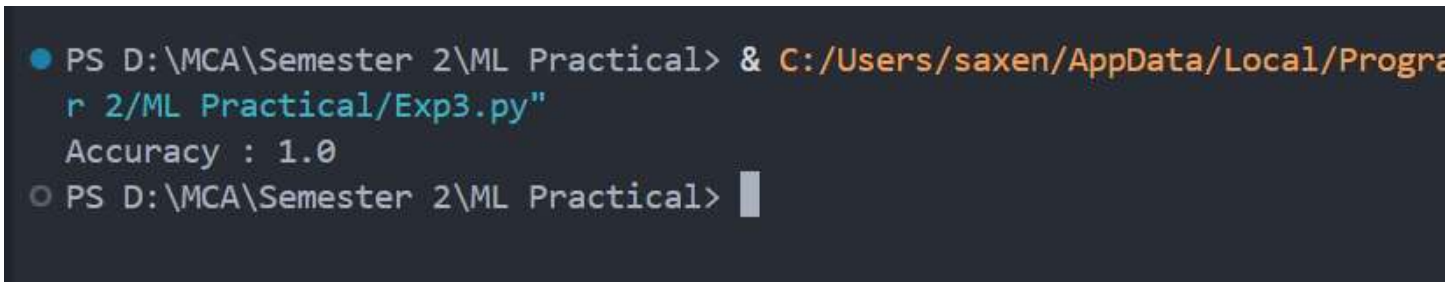
```
X_train, X_test, y_train, y_test = train_test_split(X, y)

model = LogisticRegression()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
print("Accuracy :", accuracy)
```

Output:



Learning Outcome:

1. **Understand Logistic Regression:**
   - Learn how to apply Logistic Regression for binary classification problems.
   - Understand the basic workflow of training and testing a machine learning model.
2. **Python Libraries:**
   - Practice using Python libraries such as NumPy, scikit-learn, and their various functionalities.
3. **Model Evaluation:**
   - Learn to interpret accuracy scores and confusion matrices to assess model performance.