Logistic Regression

Aim :

To implement the Logistic regression algorith using Python to classify given data into binary or multi class categories and evaluate the model's performance.

Algorithm :

1. Start

2. Import the required libraries (numpy, pandas, sklea

3. Load the dataset (Social_Network_Ads.csv) file.

4. Select the input features (Age, Estimated Salary) and the output label (Purchased)

5. Split the data into training and testing sets using train_test_split().

6. Create a Logistic Regression() model.

7. Train the model using fit() on training data.

8. Predict the output for test data using predict()

9. Evaluate the model using accuracy score and classification report

10. Stop.

Program:

```python
import numpy as nps
import pandas as pd
from sklearn.model-selection import train-test-split
from sklearn.linear_model import Logistic Regression
from sklearn.metrics import classification-report
df = pd.read_csv('Social_Network_Ads.csv')
print (df.head())
print ("\n Dataset Info:\n")
print (df.info())
features ("\n Features:\n", features[:5])
features = df.iloc[:, [2,3]].values
label = df.iloc[:, 4].values
print ("\n Features:\n", features[:5])
print ("\n Labels:\n", label[:10])
for i in range (1,40):
    x_train, x_test, y_train, y_test = train-test-split
    (features, label, test_size=0.2, random_state=i)
    model = LogisticRegression()
    model.fit(x_train, y_train)
    train_score = model.score(x_train, y_train)
    test_score = model.score(x_test, y_test)
    if test_score > train_score:
        print ("Test {} Train {} Random State {}".format
        (test_score, train_score, i))
```

```
x_train, x_test, y_train, y_test = train_test_split (featur
label, test_size = 0.2, random_state = 42)
final Model = Logistic Regression ()
final Model. fit (x_train, y_train)
print ("\n Training Accuracy: ", final Model. score(x_t
    y_train))
print ("Testing Accuracy: ", final Model. score(x_test, y_te
print ("\n Classification Report: \n")
print (classification_report (label, final Model. predict
    (features)))
```

Output:

Training Accuracy : 0.824375

Testing Accuracy : 0.9125

Classification Report:

	precision	recall	f1-score	support
0	0.85	0.93	0.89	257
1	0.54	0.71	0.77	143
accuracy			0.55	400
macro avg	0.85	0.82	0.83	400
weighted avg	0.85	0.55	0.85	400

Report:

Result:

    The Logistic Regression classifier model successfully implemented on the given dataset and predicted the purchase behaviour of users with an overall accuracy of 85% with training accuracy of 83.4% and testing accuracy of ~91.2%.