EXPERIMENT 9

Logical Regression

Aim:

To understand the importance of logical regression.

Algorithm:

1. Import the required libraries — numpy, pandas, train\_test\_split, LogisticRegression, and classification\_report.

2. Load the dataset using pd.read\_csv() and display the first few records.

3. Select features (Age, EstimatedSalary) and labels (Purchased) from the dataset.

4. Split the data into training and testing sets using train\_test\_split().

5. Create a Logistic Regression model and train it on the training data using fit().

6. Evaluate model performance using training and testing accuracy scores.

7. Predict the output labels for the entire dataset using predict().

8. Generate and display the classification report to analyze precision, recall, and F1-score.

Program:

import numpy as np

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import classification\_report

df = pd.read\_csv("C:\Social\_Network\_Ads.csv")

print(df.head())

features = df.iloc[:, [2, 3]].values

label = df.iloc[:, 4].values

x\_train, x\_test, y\_train, y\_test = train\_test\_split(features, label, test\_size=0.2, random\_state=42)

model = LogisticRegression()

model.fit(x\_train, y\_train)

train\_score = model.score(x\_train, y\_train)

test\_score = model.score(x\_test, y\_test)

print(f"Training Accuracy: {train\_score:.4f}")

print(f"Testing Accuracy: {test\_score:.4f}")

y\_pred = model.predict(features)

print("\nClassification Report:")

print(classification\_report(label, y\_pred))

Output:

A screenshot of a computer screen

AI-generated content may be incorrect.

Result:

Hence a python program for logical regression is written and executed successfully.