**U19EC046 | ML | LAB 7**

**AIM**

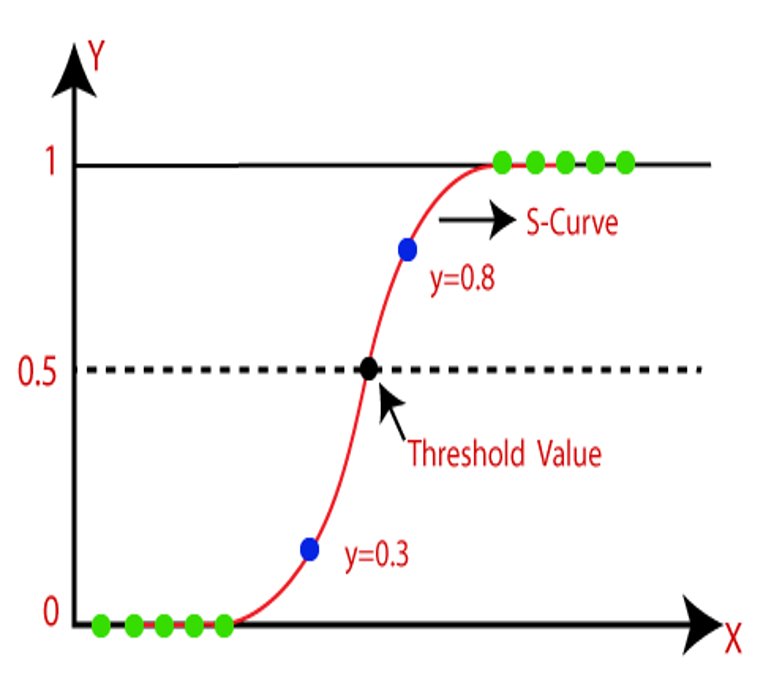
Write a program to implement the Logistic Regression for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets. ​

**THEORY:**

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.​

Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, **it gives the probabilistic values which lie between 0 and 1**.​

Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas **Logistic regression is used for solving the classification problems**.



The sigmoid function is a mathematical function used to map the predicted values to probabilities.​ It maps any real value into another value within a range of 0 and 1.​

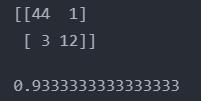
The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form. The S-form curve is called the Sigmoid function or the logistic function.​ In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

**CODE**

|  |
| --- |
| **import numpy as np**  **import matplotlib.pyplot as plt**  **import pandas as pd**  **dataset = pd.read\_csv('Social\_Network\_Ads.csv')**  **X = dataset.iloc[:, :-1].values**  **y = dataset.iloc[:, -1].values**  **from sklearn.model\_selection import train\_test\_split**  **X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.15, random\_state = 0)**  **from sklearn.preprocessing import StandardScaler**  **sc = StandardScaler()**  **X\_train = sc.fit\_transform(X\_train)**  **X\_test = sc.transform(X\_test)**  **from sklearn.linear\_model import LogisticRegression**  **classifier = LogisticRegression(random\_state = 100)**  **classifier.fit(X\_train, y\_train)**  **from sklearn.metrics import confusion\_matrix, accuracy\_score**  **cm = confusion\_matrix(y\_test, y\_pred)**  **print(cm)**  **accuracy\_score(y\_test, y\_pred)** |

**OUTPUT**

1.



**CONCLUSION**

In This Practical, we have studied and implemented Logistic regression using scikit learn library in python.