Lavina Golani

Batch A

16

**Experiment:5**

Aim:To implement logistic regression.

Code:

from sklearn.linear\_model import LogisticRegression

import pandas as pd

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

from sklearn.metrics import classification\_report,confusion\_matrix

file = '/home/lavina/Desktop/pima-indians-diabetes-database/diabetes.csv'

# load data set into frame

dataFrame = pd.read\_csv(file, names=['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'])

dataFrame = dataFrame.dropna()

print(dataFrame.describe())

print(dataFrame.columns)

# logistic regression

# seperating the data into training and testing

boundary =int (dataFrame.shape[0] \*0.6)

features = ['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age']

X = dataFrame.loc[:, ['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age']]

y = dataFrame.loc[:, ['Outcome']]

trainingData, testingData, trainingTarget, testingTarget = train\_test\_split(X, y)

print(testingData.shape)

logisticRegression = LogisticRegression()

#training the model

logisticRegression.fit(trainingData, trainingTarget)

#testing model

predict = logisticRegression.predict(testingData)

# Use score method to get accuracy of model

score = logisticRegression.score(testingData,testingTarget)

print(score)

#analysing output

print(confusion\_matrix(testingTarget, predict))

print(classification\_report(testingTarget, predict))

Output:

