Experiment-3.2

Student Name: Ashish Kumar **UID: 23MAI10008**
Branch: ME CSE AIML **Section/Group: 23MAI-1**
Semester: 02 **Date of Performance: 03/04/2024**
Subject Name: Machine Learning Lab **Subject Code: 23CSH-651**

Aim of the Experiment :

Implementing Logistic Regression using Python.

Theory :

Logistic regression is a supervised machine learning algorithm used for classification tasks where the goal is to predict the probability that an instance belongs to a given class or not. Logistic regression is a statistical algorithm which analyze the relationship between two data factors.

The equation for Logistic Regression is:

$$\log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + \ldots + b_nx_n$$

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. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

Types of Logistic Regression:

1) Binomial: There can be only 2 possible types of dependent variables. Ex: 0 or 1, Pass or Fail, etc.

2) Multinomial: There can be 3 or more possible unordered types of the dependent variable.

Example: "cat", "dogs", or "sheep".

3) Ordinal: There can be 3 or more possible ordered types of dependent variables.

Example: "low", "Medium", or "High".

Logistic Function (Sigmoid Function):

1) The sigmoid function is a mathematical function used to map the predicted values to probabilities.

2) It maps any real value into another value within a range of 0 and 1.

3) 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.

4) 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 for Experiment :

```
# Import necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
diabetes = pd.read_csv('diabetes.csv')
diabetes

# Split Dataset into X and Y
X = diabetes.iloc[:, [0, 1, 2, 3, 4, 5, 6, 7]].values
y = diabetes.iloc[:, 8].values

# Split the X and Y Dataset into the Training set and Test set
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.25, random_state = 42)
```

```
# Perform Feature Scaling as all values are not in the same range
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

# Training the KNN model on the Training set
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
model.fit(X_train, y_train)

# Predict the Test Set Results
y_pred = model.predict(X_test)
print("Actual values:")
print(y_test)
print("Predicted values:")
print(y_pred)

# Make the Confusion Matrix
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report

cm = confusion_matrix(y_test, y_pred)
print("Confusion Matrix: ")
print(cm)

print("\nAccuracy Score: ",accuracy_score(y_test,y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
```

Result/Output :

jupyter ML_Experiment9_ASHISH_23MAI10008 Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help Trusted

Run Code

Out[1]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

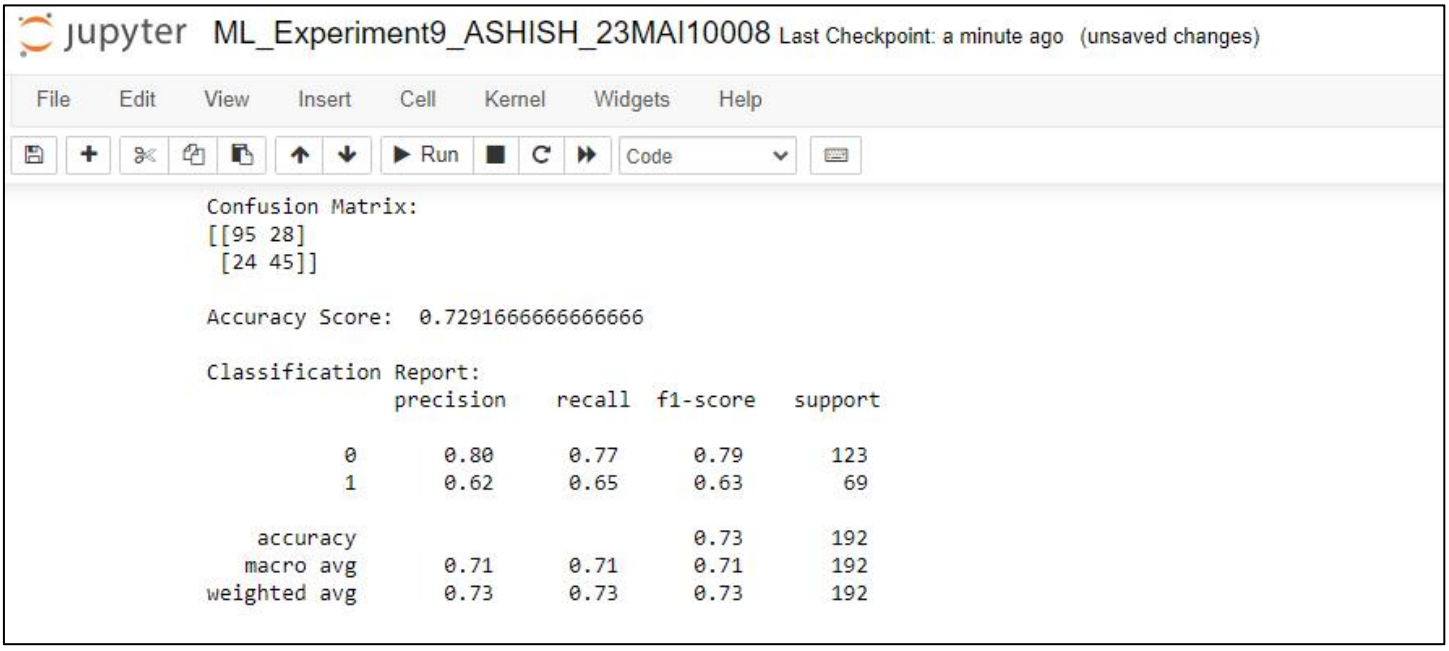
768 rows × 9 columns

jupyter ML_Experiment9_ASHISH_23MAI10008 Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Run Code

```
Actual values:
[0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 1 1 0 0 0 0 0 1 0 0 1 0 1 1 1 1 0 1 1
 1 0 1 0 0 0 1 0 1 1 0 0 0 0 1 1 1 0 0 0 0 0 1 1 0 0 1 0 0 0 1 0 1 0 0 0 1
 0 0 0 0 0 0 1 1 0 0 0 1 1 1 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 1 0 0 1 1 1 0
 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 1 1 1 0 1 1 0 1 1 0 1 1 1 0 0 0 0 0 0 0 1 0
 0 1 0 0 1 0 0 1 0 0 1 0 1 1 0 0 0 1 0 0 0 1 0 1 1 0 0 0 0 0 1 1 0 0 0 1 1
 0 0 0 1 0 1 0]
Predicted values:
[0 0 0 0 0 0 0 1 1 1 0 1 0 0 0 0 0 0 1 1 0 0 1 0 1 1 0 0 0 0 1 1 1 1 1 1 1
 0 1 1 0 1 1 0 0 1 1 0 0 1 0 1 1 0 0 0 1 0 0 1 1 0 0 0 0 1 0 1 0 1 1 0 0 0
 0 1 0 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 0 1 1 1 0 0 1 0 1 0 1 1 1 0 0 1 0 1 0
 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 1 1 1 1 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0
 0 1 0 0 0 0 0 0 0 1 1 0 1 1 0 0 0 1 0 0 1 1 1 0 0 1 1 0 0 0 0 0 1 1 0 1 1
 0 0 0 1 0 0 0]
```



Learning outcomes (What I have learnt):

1. I learnt about various python libraries like pandas, numpy and sklearn.
2. I learnt about the concept of Logistic Regression Algorithm.
3. I learnt about different types of Logistic Regression algorithm.
4. I learnt about the Sigmoid function used in Logistic Regression.
5. I learnt about Confusion Matrix and Accuracy Score metrics.