```
Assignment no. 05
Aim [
1.Logistic Regression
2. Differentiate between Linear and Logistic Regression
3. Sigmoid Function
4. Types of LogisticRegression
5. Confusion Matrix Evaluation Metrics

import numpy as np
import pandas as pd

df = pd.read_csv("C:/Users/CNLAB04/Desktop/diabetes.csv")
df.head()
```

```
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin
BMI  \
0            6      148             72             35        0  33.6

1            1       85             66             29        0  26.6

2            8      183             64              0        0  23.3

3            1       89             66             23       94  28.1

4            0      137             40             35      168  43.1


   DiabetesPedigreeFunction  Age  Outcome
0                     0.627   50        1
1                     0.351   31        0
2                     0.672   32        1
3                     0.167   21        0
4                     2.288   33        1
```

```
df.isnull()
```

```
     Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin
BMI  \
0          False    False          False          False    False
False
1          False    False          False          False    False
False
2          False    False          False          False    False
False
3          False    False          False          False    False
False
4          False    False          False          False    False
False
..           ...      ...            ...            ...      ...    ..
.
763        False    False          False          False    False
```

```
False
764        False    False          False          False    False
False
765        False    False          False          False    False
False
766        False    False          False          False    False
False
767        False    False          False          False    False
False

     DiabetesPedigreeFunction    Age  Outcome
0                       False  False    False
1                       False  False    False
2                       False  False    False
3                       False  False    False
4                       False  False    False
..                        ...    ...      ...
763                     False  False    False
764                     False  False    False
765                     False  False    False
766                     False  False    False
767                     False  False    False

[768 rows x 9 columns]
```

```
from sklearn.model_selection import train_test_split

X = data1.drop('Outcome', axis=1)
Y = data1['Outcome']

X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
test_size=0.2, random_state=42)

print(f"Training data shape (X_train): {X_train.shape}")
print(f"Testing data shape (X_test): {X_test.shape}")
print(f"Training data shape (Y_train): {Y_train.shape}")
print(f"Testing data shape (Y_test): {Y_test.shape}")
```

```
Training data shape (X_train): (614, 8)
Testing data shape (X_test): (154, 8)
Training data shape (Y_train): (614,)
Testing data shape (Y_test): (154,)
```

```
from sklearn.linear_model import LogisticRegression

logreg = LogisticRegression(max_iter=800)

logreg.fit(X_train,Y_train)
```

```
LogisticRegression(max_iter=800)
```

```python
y_testpred=logreg.predict(X_test)
y_trainpred = logreg.predict(X_train)

from sklearn.metrics import precision_score, confusion_matrix,
accuracy_score, recall_score

train_accuracy = accuracy_score(Y_train, y_trainpred)
train_precision = precision_score(Y_train, y_trainpred)
train_recall = recall_score(Y_train, y_trainpred)
train_cm = confusion_matrix(Y_train, y_trainpred)
test_accuracy = accuracy_score(Y_test, y_testpred)
test_precision = precision_score(Y_test, y_testpred)
test_recall = recall_score(Y_test, y_testpred)
test_cm = confusion_matrix(Y_test, y_testpred)
print("Training Accuracy: ", train_accuracy)
print("Training Precision: ", train_precision)
print("Training Recall: ", train_recall)
print("Training Confusion Matrix:\n", train_cm)
print("\nTesting Accuracy: ", test_accuracy)
print("Testing Precision: ", test_precision)
print("Testing Recall: ", test_recall)
print("Testing Confusion Matrix:\n", test_cm)
```

```
Training Accuracy:  0.7703583061889251
Training Precision:  0.7142857142857143
Training Recall:  0.5633802816901409
Training Confusion Matrix:
 [[353  48]
 [ 93 120]]

Testing Accuracy:  0.7467532467532467
Testing Precision:  0.6379310344827587
Testing Recall:  0.6727272727272727
Testing Confusion Matrix:
 [[78 21]
 [18 37]]
```

```
Name: Karan More   Rollno: 13234
```