- Name: Sarvesh Surve
- Roll No. 60

Logistic Regression

```
import numpy as np
from sklearn.datasets import load_diabetes
from sklearn.model_selection import train_test_split


class LogisticRegression:
    def __init__(self):
        self.params = None

    def fit(self, X, y):
        ones_column = np.ones((X.shape[0], 1))
        X_bias = np.hstack((ones_column, X))
        self.params = np.linalg.inv(X_bias.T @ X_bias) @ X_bias.T @ y
        return self.params

    def predict(self, X):
        ones_column = np.ones((X.shape[0], 1))
        X_test = np.hstack((ones_column, X))
        z = X_test @ self.params
        sigmoid = 1 / (1 + np.exp(-z))
        y_hat = (sigmoid >= 0.5).astype(int)
        return sigmoid, y_hat


if __name__ == "__main__":
    dataset = load_diabetes()
    X = dataset.data
    y = (dataset.target > np.median(dataset.target)).astype(int)  # Convert to binary classification

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=42)

    model = LogisticRegression()
    parameters = model.fit(X_train, y_train)

    sig, y_pred = model.predict(X_test)
    print(f"The predicted outcome is {y_pred} and calculated sigmoid value is {sig}")

    print(f"First value of y_test: {y_test[0]} and first value of y_pred: {y_pred[0]}")
    print(f"The sigmoid probability for the tested value: {sig[0]}")
```

```
The predicted outcome is [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1] and calculated sigmoid value is [0.62208497 0.66118916 0.59345505 0.77307867 0.58371533 0.55245194
 0.72282697 0.65699936 0.55189212 0.5609232  0.5608303  0.66664598
 0.50859887 0.68860004 0.55627652 0.58653616 0.68855865 0.72127476
 0.67547208 0.69032301 0.70009756 0.55075378 0.51330212 0.66596491
 0.6312394  0.62995203 0.65981963 0.66076034 0.49300269 0.56859072
 0.64468836 0.55113876 0.5934533  0.66348858 0.65077909 0.66599142
 0.59018332 0.58407032 0.61613618 0.50250994 0.52173465 0.5521964
 0.64521382 0.619222   0.66009316]
First value of y_test: 1 and first value of y_pred: 1
The sigmoid probability for the tested value: 0.6220849698125
```