# Import necessary libraries

Import numpy as np

Import pandas as pd

Import matplotlib.pyplot as plt

From sklearn import datasets

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

From sklearn.preprocessing import StandardScaler

From sklearn.linear\_model import LogisticRegression

From sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

# Load the Iris dataset

Iris = datasets.load\_iris()

X = iris.data

Y = iris.target

# Split the dataset into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Standardize the features

Scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# Initialize and train the model

Model = LogisticRegression(max\_iter=200)

Model.fit(X\_train, y\_train)

# Make predictions

Y\_pred = model.predict(X\_test)

# Evaluate the model

Accuracy = accuracy\_score(y\_test, y\_pred)

Print(f”Accuracy: {accuracy:.2f}”)

Print(“Classification Report:”)

Print(classification\_report(y\_test, y\_pred))

Print(“Confusion Matrix:”)

Print(confusion\_matrix(y\_test, y\_pred))

# Optional: Visualize some data points

Plt.scatter(X\_test[:, 0], X\_test[:, 1], c=y\_pred, cmap=’viridis’, edgecolor=’k’, s=20)

Plt.xlabel(‘Feature 1’)

Plt.ylabel(‘Feature 2’)

Plt.title(‘Test Data Points with Predictions’)

Plt.show()