Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester: (Spring, Year: 2026), B.Sc. in CSE (Day)

LAB REPORT NO : 01

Course Title : Machine Learning Lab
Course Code : CSE 404 Section : 231_D2

Student Details

Name	ID
Promod Chandra Das	231002005

Lab Date : 15-02-26
Submission Date : 22-02-26
Course Teacher's Name : Ms. Mayeesha Farjana

[For Teachers use only: Don't Write Anything inside this box]

Lab Report Status
Marks: ………………………………… Signature:.....................
Comments:... Date:..............................

1. TITLE :

Logistic Regression Classification on Iris Dataset (Setosa vs Non-Setosa).

2. TASK / PROBLEM :

I completed a binary classification task using the Iris dataset, focusing on distinguishing Iris-setosa from the other flower species. To accomplish this, I developed a Logistic Regression model based on two selected features: Sepal Length and Petal Width. After training the model, I evaluated its performance on test data to measure how effectively it could identify whether a flower belongs to the Setosa class or not.

3. OBJECTIVES :

- To explore the application of Logistic Regression in solving binary classification problems.
- To build and train a Logistic Regression model using specific input features.
- To assess the model's performance through metrics such as accuracy, confusion matrix, and classification report.
- To plot and analyze the decision boundary created by the classification model.

4. PROCEDURE :

The Iris dataset was loaded using the **scikit-learn dataset** library. The target variable was converted into binary form:

- **Setosa = 1**
- **Non-setosa = 0**

Only two features were selected:

- **Sepal Length (cm)**
- **Petal Width (cm)**

After feature selection, the dataset was split into:

- **80% training data**
- **20% testing data**

The Logistic Regression model was trained using the training data, and its performance was evaluated using:

- **Accuracy Score**
- **Confusion Matrix**

- **Classification Report**

Finally, the decision boundary was plotted to clearly visualize how the model separates Setosa from Non-setosa.

5. IMPLEMENTATION & OUTPUT :

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
from sklearn.linear_model import LogisticRegression
```

```
data = load_iris()
X = data.data
y = data.target

df = pd.DataFrame(X, columns=data.feature_names)
df['target'] = y
df.head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

Figure 5.1 : First five rows of the original Iris dataset showing all four features

```
X = data.data[:, [0, 3]]
y = (data.target == 0).astype(int)

df = pd.DataFrame(X, columns=["sepal length (cm)", "petal width (cm)"])
df['target'] = y
df.head()
```

	sepal length (cm)	petal width (cm)	target
0	5.1	0.2	1
1	4.9	0.2	1
2	4.7	0.2	1
3	4.6	0.2	1
4	5.0	0.2	1

Figure 5.2 : Iris dataset after feature selection and binary target conversion.

```
print('shape of feature : ', X.shape)
print('shape of target : ', y.shape)

print('\nClass Labels : ')
print(["Non-setosa", "Setosa"])
```

```
shape of feature :  (150, 2)
shape of target :  (150,)

Class Labels :
['Non-setosa', 'Setosa']
```

Figure 5.3 : Dataset shape after feature selection

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

```
scaler = StandardScaler()

X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

```
model = LogisticRegression(max_iter=5000)
model.fit(X_train_scaled, y_train)

y_pred = model.predict(X_test_scaled)
```

```
accuracy = accuracy_score(y_test, y_pred)
print('accuracy : ', accuracy)
```

```
accuracy :  1.0
```

Figure 5.4 : Accuracy result of the Logistic Regression model.

```
print('first 10 prediction : ', y_pred[0:10])
print('first 10 actual labels : ', y_test[0:10])
```

```
first 10 prediction :  [0 1 0 0 0 1 0 0 0 0]
first 10 actual labels :  [0 1 0 0 0 1 0 0 0 0]
```

Figure 5.5: First 10 predicted values compared with actual labels.

```
cm = confusion_matrix(y_test, y_pred)
print("confusion matrix : \n", cm)
```

```
confusion matrix :
 [[20  0]
 [ 0 10]]
```

Figure 5.6: Confusion matrix values showing model prediction results.

```
print("classification_report : \n")
print(classification_report(y_test, y_pred))
```

```
classification_report :

              precision    recall  f1-score   support

           0       1.00      1.00      1.00        20
           1       1.00      1.00      1.00        10

    accuracy                           1.00        30
   macro avg       1.00      1.00      1.00        30
weighted avg       1.00      1.00      1.00        30
```

Figure 5.7 : Classification report showing precision, recall and F1-score.

```
plt.figure()
plt.imshow(cm, cmap='Blues')
plt.title("Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.colorbar()
plt.show()
```

...

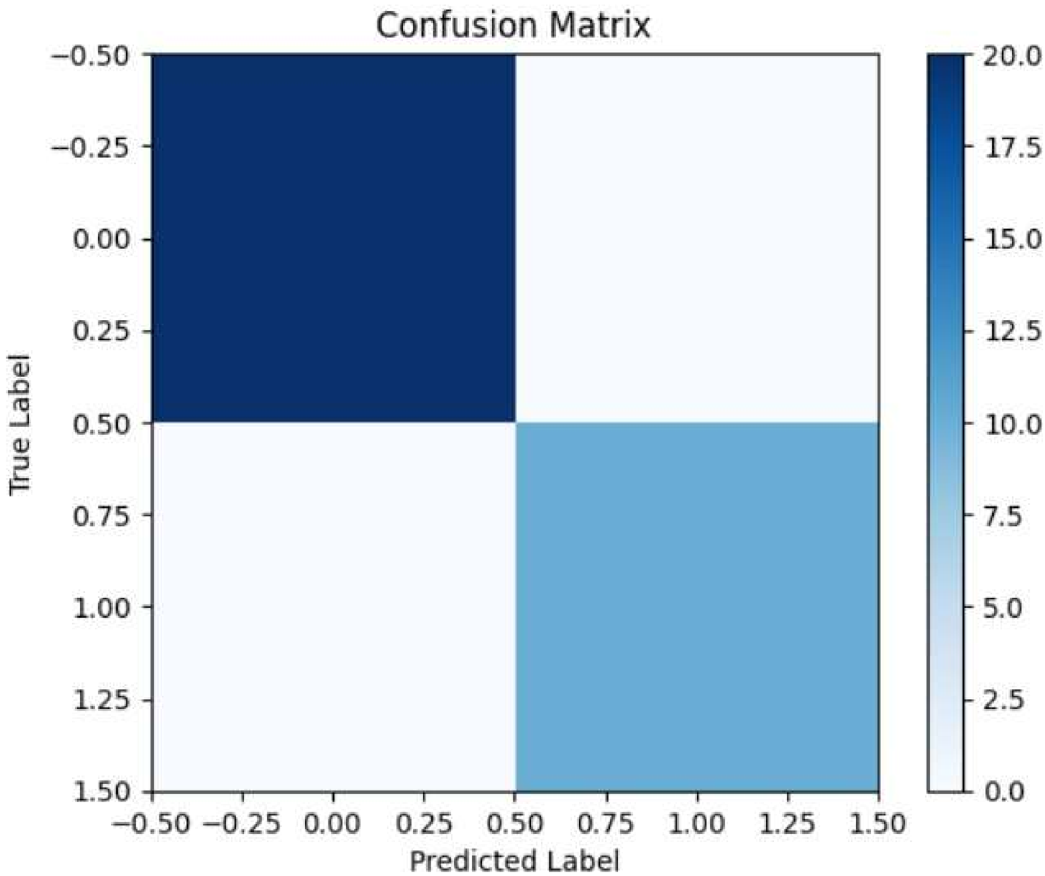


Figure 5.8 : Confusion matrix heatmap visualization of prediction performance.

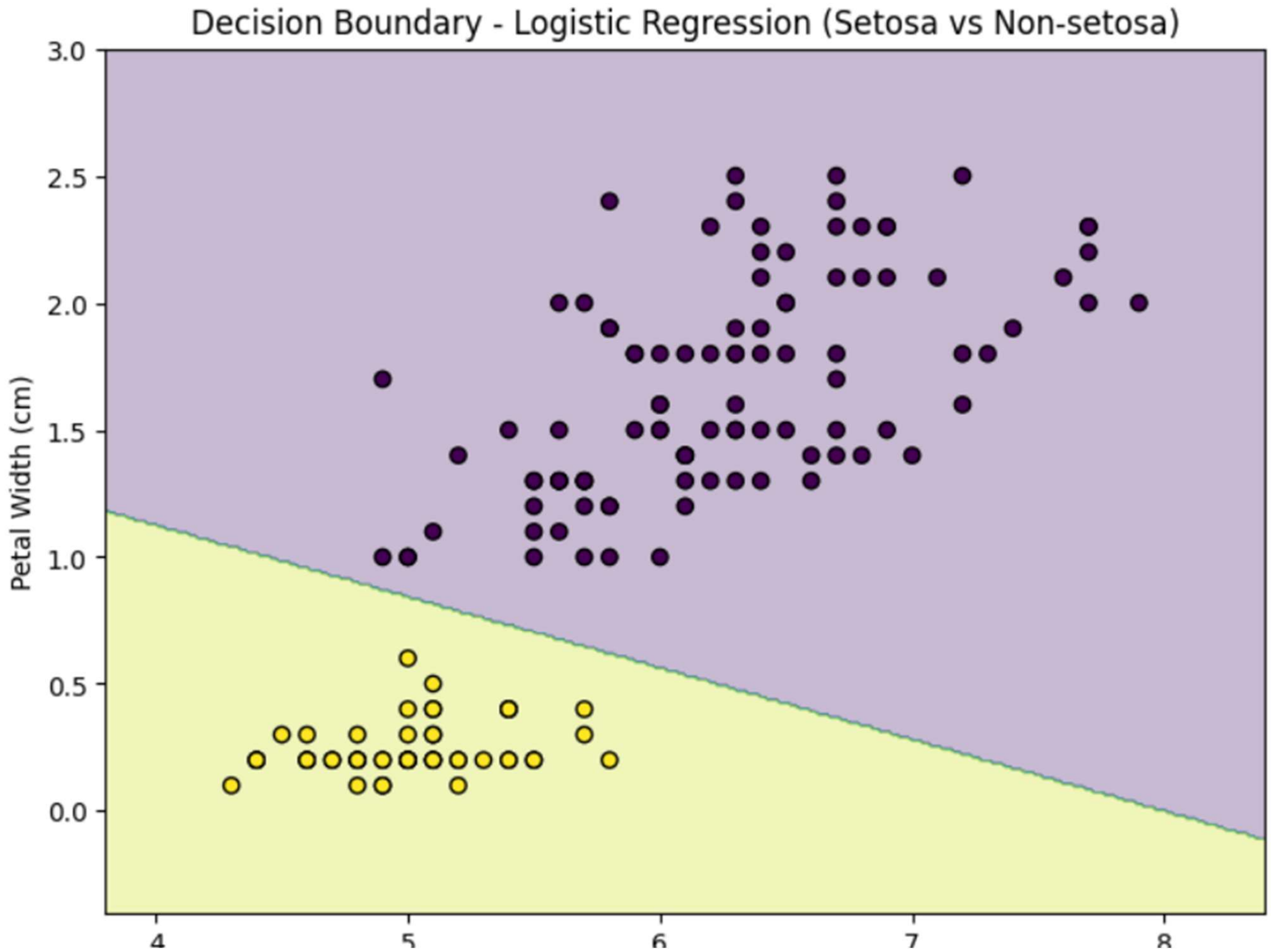
```
x_min, x_max = X[:, 0].min() - 0.5, X[:, 0].max() + 0.5
y_min, y_max = X[:, 1].min() - 0.5, X[:, 1].max() + 0.5

xx, yy = np.meshgrid(
    np.linspace(x_min, x_max, 300),
    np.linspace(y_min, y_max, 300)
)

grid_points = np.c_[xx.ravel(), yy.ravel()]
grid_points_scaled = scaler.transform(grid_points)

Z = model.predict(grid_points_scaled)
Z = Z.reshape(xx.shape)
```

Figure 5.9 : Decision boundary showing separation between Setosa and Non-setosa.



```
plt.figure(figsize=(8,6))
plt.contourf(xx, yy, Z, alpha=0.3)

plt.scatter(X[:, 0], X[:, 1], c=y, edgecolor="k")

plt.xlabel("Sepal Length (cm)")
plt.ylabel("Petal Width (cm)")
plt.title("Decision Boundary - Logistic Regression (Setosa vs
Non-setosa)")
plt.show()
```

6. ANALYSIS AND DISCUSSION :

The Logistic Regression model achieved perfect classification accuracy (100%). This indicates that Sepal Length and Petal Width are highly effective features for distinguishing Iris-setosa from the other species.

The confusion matrix confirms that:

● **All Setosa samples were correctly classified.**
● **All Non-setosa samples were correctly classified**.

There were no incorrect predictions in the test dataset.

This result occurs because Iris-setosa forms a clearly distinct cluster compared to the other two species. In particular, Petal Width plays a significant role in making the separation straightforward. Since Logistic Regression is a linear classifier, it performs exceptionally well when the data is linearly separable.

The decision boundary plot clearly illustrates how the model draws a straight line to separate Setosa and Non-setosa classes. Instead of using an ROC curve, the decision boundary plot was chosen because it visually demonstrates how the two classes are separated based on the selected features.

Overall, this task helped me understand how to implement Logistic Regression for a binary classification problem and evaluate the model using accuracy, confusion matrix, and classification report. Initially, I was slightly confused about selecting only two features and converting the target variable into binary form. I also encountered minor issues, such as missing imports. However, by resolving each problem step by step, I gained a clear understanding of the entire process. This lab improved my practical knowledge of classification models and boosted my confidence in applying machine learning algorithms.

7. SUMMARY :

The Logistic Regression model for classifying Iris-setosa versus non-setosa was successfully developed using only two features: Sepal Length and Petal Width. The model was properly trained and tested, and its performance was evaluated using metrics such as accuracy score, confusion matrix, and classification report.

The decision boundary plot clearly illustrates how the model distinguishes Setosa from the other species. This lab enhanced my understanding of binary classification, the Logistic Regression algorithm, feature selection, and different model evaluation methods.