

PRACTICAL NO.10

To perform and Data analysis with Confusion matrix

```
In [1]: import pandas as pd
import numpy as np
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, classification_report, ConfusionMatrixDisplay
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [2]: iris = load_iris()
X = iris.data
y = iris.target
```

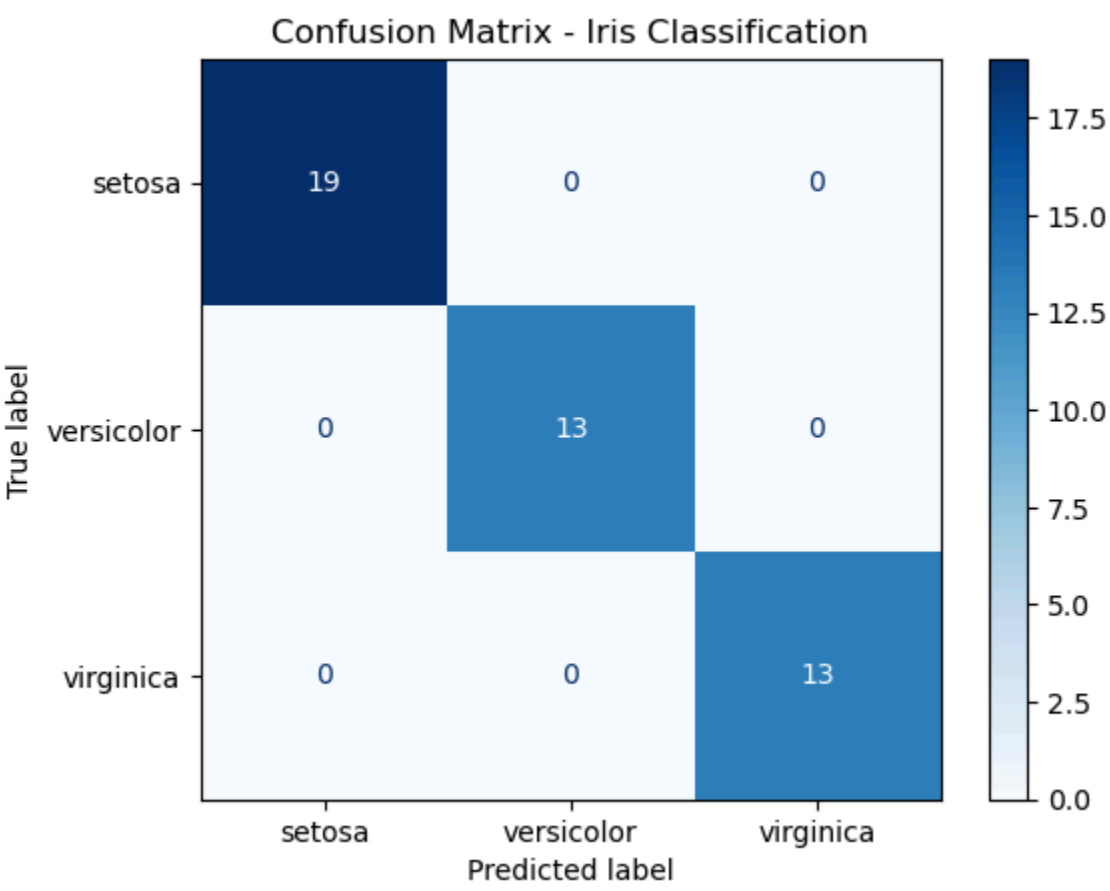
```
In [3]: X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=42)
```

```
In [4]: model = LogisticRegression(max_iter=200)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

```
In [5]: cm = confusion_matrix(y_test, y_pred)
print("Confusion Matrix:")
print(cm)
```

Confusion Matrix:
[[19 0 0]
 [0 13 0]
 [0 0 13]]

```
In [6]: disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=iris.target_names)
disp.plot(cmap=plt.cm.Blues)
plt.title("Confusion Matrix - Iris Classification")
plt.show()
```



```
In [7]: print("Classification Report:")
print(classification_report(y_test, y_pred, target_names=iris.target_names))
```

Classification Report:				
	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	19
versicolor	1.00	1.00	1.00	13
virginica	1.00	1.00	1.00	13
accuracy			1.00	45
macro avg	1.00	1.00	1.00	45
weighted avg	1.00	1.00	1.00	45