
# IRIS FLOWER CLASSIFICATION REPORT

## 2025

**Project: Iris Flower Classification**

**Name: Rohan Sharma**

**Roll No.: 2024011004400157**

**Institution: KIET**

**Date: 11 MARCH 2025**

# TABLE OF CONTENTS

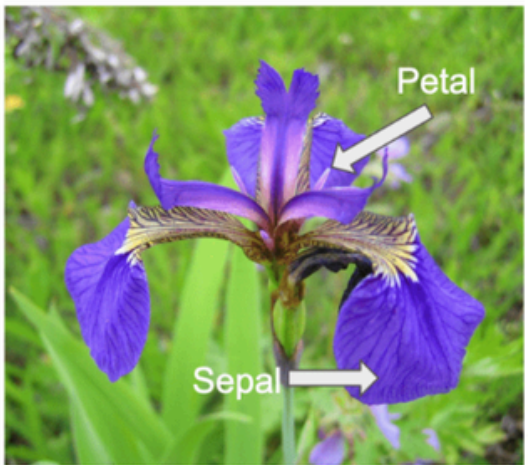

# INTRODUCTION

The objective of this project is to classify different species of the Iris flower based on petal and sepal dimensions. The Iris dataset, which contains information on three flower species (Setosa, Versicolor, Virginica), is used to train a machine learning model to make accurate predictions.

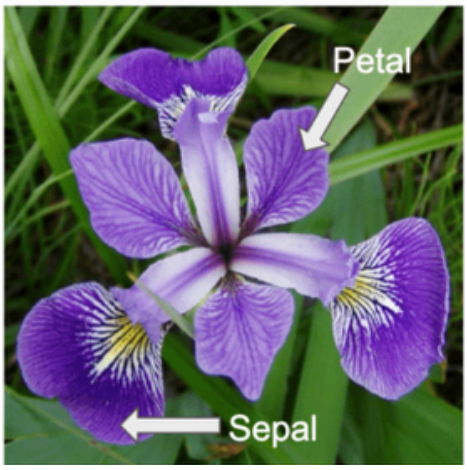
The classification is performed using the Random Forest Classifier, which is a powerful ensemble learning method that improves accuracy by combining multiple decision trees.

# RESEARCH METHODOLOGY

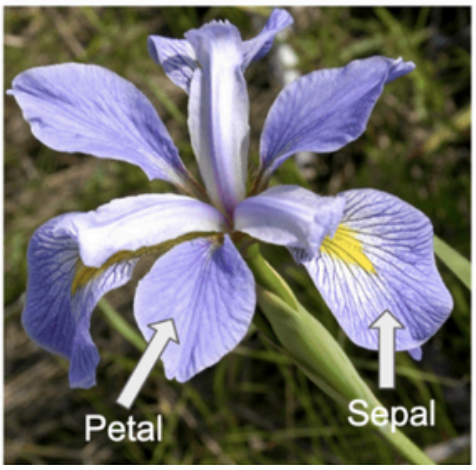
*Iris setosa*




*Iris versicolor*




*Iris virginica*

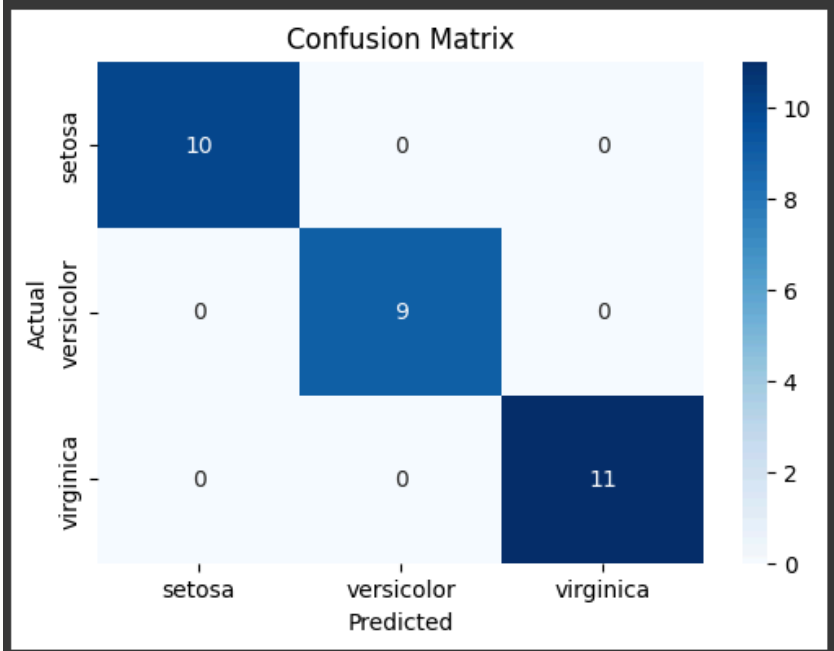



1. ***LOAD THE DATASET:*** THE IRIS DATASET IS LOADED USING SKLEARN.DATASETS.
2. ***PREPROCESS THE DATA:*** FEATURE SCALING IS APPLIED TO NORMALIZE THE INPUT VALUES.
3. ***SPLIT THE DATA:*** THE DATASET IS DIVIDED INTO TRAINING (80%) AND TESTING (20%) SETS.
4. ***TRAIN THE MODEL:*** A RANDOM FOREST CLASSIFIER IS TRAINED USING THE TRAINING DATA.
5. ***EVALUATE THE MODEL:*** PERFORMANCE IS MEASURED USING ACCURACY, CONFUSION MATRIX, AND CLASSIFICATION REPORT.
6. ***VISUALIZATION:*** A CONFUSION MATRIX HEATMAP IS PLOTTED TO VISUALIZE CLASSIFICATION PERFORMANCE.

# CODE

```
IMPORT NUMPY AS NP
IMPORT PANDAS AS PD
IMPORT MATPLOTLIB.PYPLOT AS PLT
IMPORT SEABORN AS SNS
FROM SKLEARN IMPORT DATASETS
FROM SKLEARN.MODEL_SELECTION IMPORT TRAIN_TEST_SPLIT
FROM SKLEARN.PREPROCESSING IMPORT STANDARDSCALER
FROM SKLEARN.ENSEMBLE IMPORT RANDOMFORESTCLASSIFIER
FROM SKLEARN.METRICS IMPORT ACCURACY_SCORE, CONFUSION_MATRIX, CLASSIFICATION_REPORT

# LOAD THE IRIS DATASET
IRIS = DATASETS.LOAD_IRIS()
DATA = PD.DATAFRAME(DATA=IRIS.DATA, COLUMNS=IRIS.FEATURE_NAMES)
DATA['SPECIES'] = IRIS.TARGET

# MAPPING TARGET NUMBERS TO SPECIES NAMES
DATA['SPECIES'] = DATA['SPECIES'].MAP({0: 'SETOSA', 1: 'VERSICOLOR', 2: 'VIRGINICA'})

# SPLITTING DATA INTO FEATURES AND TARGET
X = DATA.ILOC[:, :-1]
Y = DATA.ILOC[:, -1]

# SPLIT INTO TRAINING AND TESTING SETS
X_TRAIN, X_TEST, Y_TRAIN, Y_TEST = TRAIN_TEST_SPLIT(X, Y, TEST_SIZE=0.2, RANDOM_STATE=42)

# STANDARDIZING THE FEATURES
SCALER = STANDARDSCALER()
X_TRAIN = SCALER.FIT_TRANSFORM(X_TRAIN)
X_TEST = SCALER.TRANSFORM(X_TEST)

# TRAIN THE MODEL USING RANDOM FOREST
MODEL = RANDOMFORESTCLASSIFIER(N_ESTIMATORS=100, RANDOM_STATE=42)
MODEL.FIT(X_TRAIN, Y_TRAIN)

# MAKE PREDICTIONS
Y_PRED = MODEL.PREDICT(X_TEST)

# EVALUATE THE MODEL
ACCURACY = ACCURACY_SCORE(Y_TEST, Y_PRED)
CONF_MATRIX = CONFUSION_MATRIX(Y_TEST, Y_PRED)
CLASS_REPORT = CLASSIFICATION_REPORT(Y_TEST, Y_PRED)

# PRINT RESULTS
PRINT(F'ACCURACY: {ACCURACY:.2F}')
PRINT('CONFUSION MATRIX:\N', CONF_MATRIX)
PRINT('CLASSIFICATION REPORT:\N', CLASS_REPORT)

# VISUALIZING THE CONFUSION MATRIX
PLT.FIGURE(FIGSIZE=(6,4))
SNS.HEATMAP(CONF_MATRIX, ANNOT=TRUE, CMAP='BLUES', FMT='D', XTICKLABELS=IRIS.TARGET_NAMES,
YTICKLABELS=IRIS.TARGET_NAMES)
PLT.XLABEL('PREDICTED')
PLT.YLABEL('ACTUAL')
PLT.TITLE('CONFUSION MATRIX')
PLT.SHOW()
```


```
Accuracy: 1.00
Confusion Matrix:
 [[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
Classification Report:
              precision    recall  f1-score   support

      setosa       1.00      1.00      1.00        10
  versicolor       1.00      1.00      1.00         9
   virginica       1.00      1.00      1.00        11

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




- Model accuracy
- Confusion matrix
- Classification report
- Confusion matrix heatma


1. Dataset: The Iris dataset is sourced from the UCI Machine Learning Repository.
2. Libraries Used: numpy, pandas, matplotlib, seaborn, and scikit-learn.
3. Scikit-Learn
4. Documentation: https://scikit-learn.org/
5. Author: Rohan Sharma

# Thank You