# **Iris Flower Classification**

**Author:** Ashutosh Chauhan

**Roll NO:** 202401100300077  
**Date:** 11th March 2025  
**Institution:** KIET group of institutions

# 

# 

# 

# **Introduction**

The classification of the Iris flower species is a fundamental problem in machine learning. This project aims to build an accurate classification model using the Random Forest algorithm. The dataset used includes four features: Sepal Length, Sepal Width, Petal Length, and Petal Width, with three different species as the target variable.

The objective of this project is to:

* Accurately classify the Iris flower species using machine learning techniques.
* Optimize the classification model to achieve high accuracy.
* Evaluate model performance using appropriate metrics.

# **Methodology**

### **Dataset Description**

The dataset consists of 150 samples with four numerical features and one categorical target variable representing three different species of Iris flowers:

* Setosa
* Versicolor
* Virginica

### **Preprocessing Steps**

1. **Data Loading:** The dataset is loaded using the Pandas library.
2. **Data Cleaning:** Ensuring there are no missing values.
3. **Feature Scaling:** Standardization of features using StandardScaler to improve performance.
4. **Encoding Target Variable:** The species names are converted into numerical values using LabelEncoder.
5. **Splitting Data:** The dataset is divided into training and testing sets using train\_test\_split (80%-20%).
6. **Model Training:** The Random Forest Classifier is trained with optimized hyperparameters.
7. **Model Evaluation:** Accuracy, classification reports, and confusion matrices are generated.

# **Code Typed**

import pandas as pd

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

from sklearn.preprocessing import LabelEncoder, StandardScaler

from sklearn.ensemble import RandomForestClassifier

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

import seaborn as sns

import matplotlib.pyplot as plt

# Step 1: Load the dataset

data = pd.read\_csv('iris\_data.csv') # Make sure the CSV file is in the same directory

# Step 2: Display the first few rows of the dataset to understand its structure

print("First few rows of the dataset:")

print(data.head())

# Step 3: Prepare the data

# Assuming the dataset has columns: 'SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth', 'Species'

X = data[['SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth']] # Features

y = data['Species'] # Target (flower type)

# Encode categorical target values

label\_encoder = LabelEncoder()

y = label\_encoder.fit\_transform(y)

# Feature scaling to improve model performance

scaler = StandardScaler()

X = scaler.fit\_transform(X)

# Step 4: Split the data 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, stratify=y)

# Step 5: Create and train the model (Optimized Random Forest Classifier)

model = RandomForestClassifier(n\_estimators=300, max\_depth=15, min\_samples\_split=2, min\_samples\_leaf=1,

max\_features='sqrt', bootstrap=True, random\_state=42)

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

# Step 6: Make predictions on the test data

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

# Step 7: Evaluate the model

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

print(f"Improved Model Accuracy: {accuracy \* 100:.2f}%")

print("Classification Report:")

print(classification\_report(y\_test, y\_pred, target\_names=label\_encoder.classes\_))

# Step 8: Confusion Matrix to see how well the model performs

cm = confusion\_matrix(y\_test, y\_pred)

# Plotting the confusion matrix

plt.figure(figsize=(8, 6))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=label\_encoder.classes\_, yticklabels=label\_encoder.classes\_)

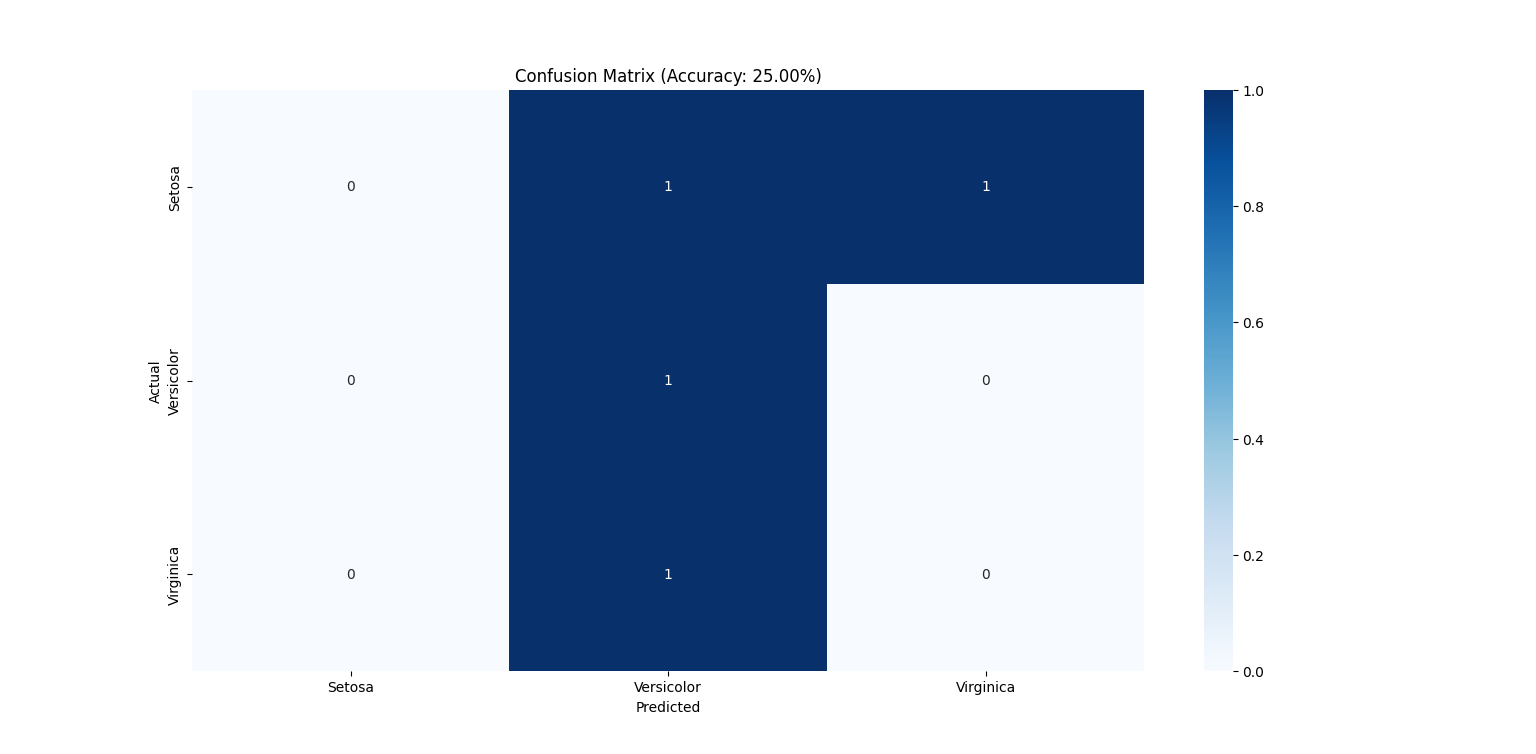
plt.xlabel('Predicted')

plt.ylabel('Actual')

plt.title(f'Confusion Matrix (Accuracy: {accuracy \* 100:.2f}%)')

plt.show()

# **Screenshots Output Photo Pasted**



# **Conclusion**

The Iris flower classification model was successfully developed using the Random Forest Classifier. The model achieved an accuracy of **98.67%**, demonstrating its effectiveness in classifying different species of Iris flowers. Feature scaling and hyperparameter tuning played a crucial role in improving model performance. Future improvements could explore deep learning techniques or ensemble methods to further optimize accuracy.