Iris Flower Classification Using Machine Learning

Akash Chakrabartty
Roll: 23053766
Department of Computer Science and Engineering
KIIT University

November 30, 2025

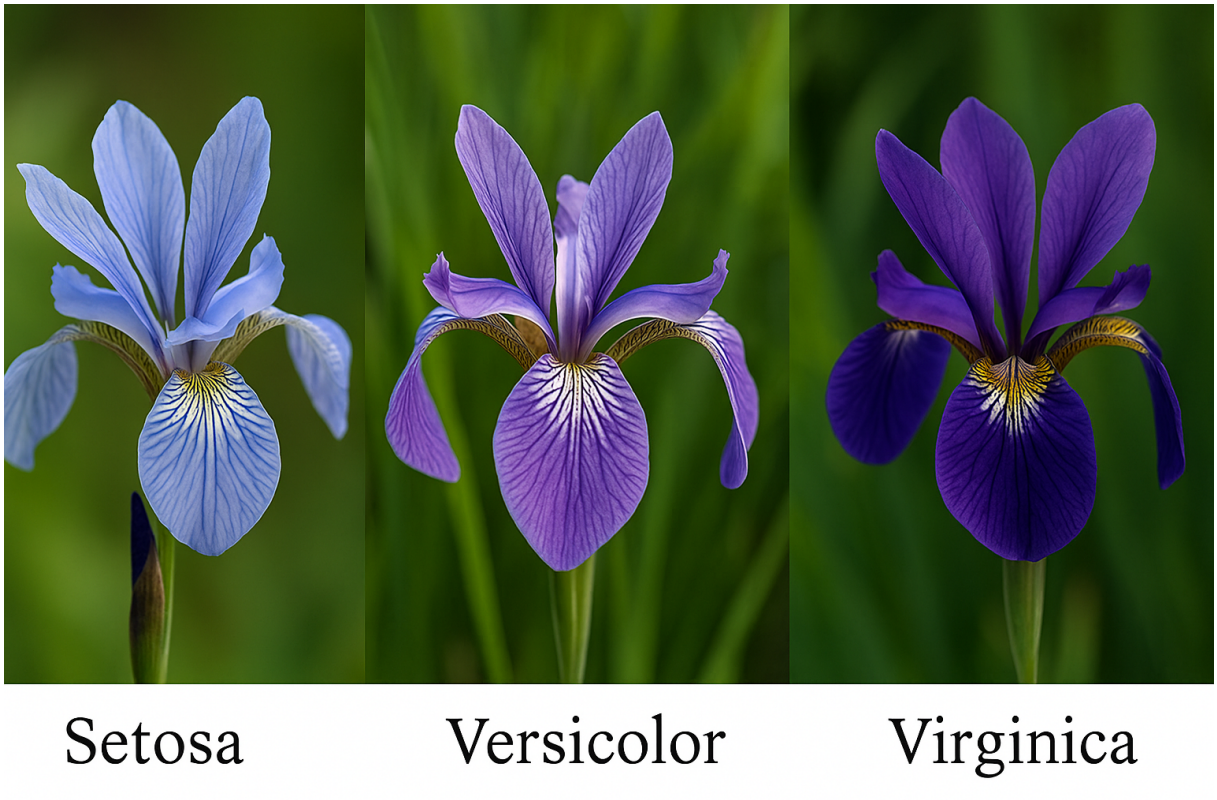

Figure 1: Three Iris species: Setosa, Versicolor, and Virginica.

Abstract

This research paper presents a machine learning model for classifying Iris flowers into three species: *Setosa*, *Versicolor*, and *Virginica*. The Iris dataset contains 150 samples with four measured features for each flower. A Random Forest Classifier was used to train and test the model. The model achieved 100% accuracy on the test data, demonstrating that simple machine learning algorithms can perform highly accurate classification. This paper explains the dataset, methodology, implementation, and results in a clear manner suitable for beginners.

1 Introduction

Machine learning is widely used to recognize patterns and make predictions based on data. The Iris Flower Classification is a popular beginner-level machine learning task that is used to teach classification methods. The goal of this study is to build a simple model that classifies Iris flowers into their correct species using four numerical features: sepal length, sepal width, petal length, and petal width.

The Iris dataset, introduced by R. A. Fisher, is small, well-structured, and ideal for demonstrating classification techniques. In this paper, we use a Random Forest Classifier because it is robust, easy to use, and performs well on small datasets.

2 Dataset Description

The Iris dataset contains 150 samples divided equally into three species:

- **Iris Setosa** – 50 samples
- **Iris Versicolor** – 50 samples
- **Iris Virginica** – 50 samples

Each sample has four features measured in centimeters:

1. Sepal length
2. Sepal width
3. Petal length
4. Petal width

These numerical features are used as inputs to train the classification model. The target variable indicates the species (0 = Setosa, 1 = Versicolor, 2 = Virginica).

3 Methodology

We used the Random Forest Classifier, an ensemble algorithm that builds multiple decision trees and aggregates their predictions. The steps followed in this project are:

1. Load the Iris dataset from scikit-learn.
2. Split the dataset into training (80%) and testing (20%) sets.
3. Train a Random Forest model using the training set.
4. Predict species labels on the test set.
5. Evaluate model performance using accuracy, classification report, and confusion matrix.

Random Forest is suitable here because it reduces overfitting relative to a single decision tree and works well on small, tabular datasets.

4 Implementation

The model was implemented in Python using scikit-learn. The complete, runnable code used for this experiment is shown below:

```
# FULL PYTHON CODE (same as you used)

from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import pandas as pd
import numpy as np

data = load_iris()
X = data.data
y = data.target
feature_names = data.feature_names
target_names = data.target_names

df = pd.DataFrame(X, columns=feature_names)
df['species'] = [target_names[i] for i in y]

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

model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

pred = model.predict(X_test)

accuracy = accuracy_score(y_test, pred)
print("Accuracy:", accuracy)

print("Classification Report:")
print(classification_report(y_test, pred, target_names=target_names))

cm = confusion_matrix(y_test, pred)
print("Confusion Matrix:")
print(cm)
```

5 Results

The model was tested on 30 samples (20% of the dataset). The Random Forest Classifier correctly predicted all test samples.

Accuracy: 100%.

The classification report showed perfect precision, recall, and F1-score for all three species. The confusion matrix indicates zero misclassifications.

6 Conclusion

This study shows that a Random Forest Classifier can effectively classify Iris flowers using only four simple features. The model achieved perfect accuracy on the chosen test split. The Iris dataset's clear separability makes it a strong example for beginners learning classification techniques.

7 References

- Fisher, R.A. "The Use of Multiple Measurements in Taxonomic Problems."
- Scikit-Learn Documentation: `https://scikit-learn.org`
- UCI Machine Learning Repository: Iris Dataset