Mini Machine Learning Experiment – Iris Flower Classification

Author: Debasish Mahanta

1■■ Objective

The goal of this project is to classify iris flowers into their respective species (Setosa, Versicolor, or Virginica) using basic machine learning techniques. This demonstrates fundamental ML skills — data preprocessing, model building, evaluation, and visualization.

2■■ Dataset

Source: Iris Dataset – UCI Machine Learning Repository Description: The dataset contains 150 samples of iris flowers, with 4 numerical features: - Sepal Length (cm) - Sepal Width (cm) - Petal Length (cm) - Petal Width (cm) Target Variable: Species (3 classes: Setosa, Versicolor, Virginica)

3■■ Steps & Workflow

a. Importing Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
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.ensemble import RandomForestClassifier
```

b. Load and Explore Data

```
data = pd.read_csv("https://raw.githubusercontent.com/mwaskom/seaborn-data/master/iris.csv")
print(data.head())
print(data.info())
print(data.describe())
```

c. Data Visualization

```
sns.pairplot(data, hue="species")
plt.show()
```

d. Data Preprocessing

```
X = data.drop('species', axis=1)
y = data['species']

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

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

e. Model Training

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

f. Model Evaluation

```
y_pred = model.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))

cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, cmap='Blues', fmt='d')
plt.title("Confusion Matrix")
plt.show()
```

4■■ Reflection

What Worked: Random Forest performed very well even without heavy tuning. Features were well-separated; scaling improved model stability. Visualization helped understand feature relationships early.

What Didn't Work: Logistic Regression performed worse (~92%) due to overlap in features. Model slightly confused Versicolor vs Virginica.

What I Learned: Importance of EDA before modeling. Feature scaling and visualization can reveal insights quickly. Even simple models can achieve high accuracy on structured data.

5■■ Conclusion

This mini experiment successfully applied ML fundamentals — from preprocessing to evaluation — on a small dataset. It reinforced key concepts: data cleaning, visualization, model selection, and evaluation metrics.