Iris Flower Classification

Problem Statement: Classify iris flowers into three species (Setosa, Versicolor, Virginica) based on their petal and sepal dimensions using the Iris dataset.

Student Details:

- **Name:** Badal Singh
- **Roll No.:** 202401100400071
- **Course:** Introduction of AI

Introduction

The **Iris Flower Classification** is a fundamental machine-learning problem used to classify flowers into three different species based on their morphological features. This dataset contains **sepal length, sepal width, petal length, and petal width** as input features, and the species as the output label.

The Iris dataset is widely used in machine learning as a beginner-friendly dataset because of its well-balanced nature and clear class separability. The goal is to train a classifier that can predict the species of a flower based on its given attributes.

We will implement a simple **k-Nearest Neighbors (k-NN) classifier** using **NumPy and Pandas** and visualize the dataset with **Matplotlib and Seaborn.**

Dataset Information:

- **Features:** Sepal length, Sepal width, Petal length, Petal width
- **Target:** Species (Setosa, Versicolor, Virginica)
- **Format:** CSV (Comma-Separated Values)
- **Size:** 150 samples, 50 samples per species

(An image of the iris flower species can be added for better visualization.)

Methodology

The following steps were followed to classify the flowers:

1. **Data Collection:** The Iris dataset was loaded from the UCI Machine Learning Repository.
2. **Data Preprocessing:** The categorical labels (species) were converted into numerical format.
3. **Data Splitting:** The dataset was split into **80% training and 20% testing.**
4. **Algorithm Implementation:** A **k-NN classifier (k=3)** was implemented to classify new flower samples.
5. **Model Evaluation:** The accuracy of the model was calculated on the test dataset.
6. **Data Visualization:** Various graphs were plotted to explore the dataset distribution and relationships between features.

Code

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

# Load the dataset
df = pd.read_csv("https://archive.ics.uci.edu/ml/machine-learning-
databases/iris/iris.data",
                 names=["sepal_length", "sepal_width", "petal_length",
"petal_width", "species"])

# Encode species as numbers
df["species"] = df["species"].astype("category").cat.codes

# Convert to NumPy arrays
X, y = df.iloc[:, :-1].values, df["species"].values

# Train-test split
split = int(0.8 * len(X))
X_train, X_test = X[:split], X[split:]
y_train, y_test = y[:split], y[split:]

# k-NN function
def knn_predict(X_train, y_train, x_test, k=3):
    distances = np.linalg.norm(X_train - x_test, axis=1)
    nearest_labels = y_train[np.argsort(distances)[:k]]
    return np.bincount(nearest_labels).argmax()

# Prediction
y_pred = np.array([knn_predict(X_train, y_train, x) for x in X_test])

# Accuracy
accuracy = np.mean(y_pred == y_test)
print(f"Model Accuracy: {accuracy:.2f}")

# Data Visualization
plt.figure(figsize=(6, 4))
sns.countplot(x=df["species"], palette="coolwarm")
plt.xlabel("Species")
plt.ylabel("Count")
plt.title("Count of Each Iris Species")
plt.xticks(ticks=[0, 1, 2], labels=["Setosa", "Versicolor",
"Virginica"])
plt.show()

plt.figure(figsize=(6, 4))
plt.hist(df["petal_length"], bins=10, color="purple",
edgecolor="black", alpha=0.7)
plt.xlabel("Petal Length")
plt.ylabel("Frequency")
plt.title("Distribution of Petal Length")
plt.grid(True)
plt.show()

plt.figure(figsize=(6, 4))
sns.scatterplot(x=df["sepal_length"], y=df["sepal_width"],
hue=df["species"], palette="viridis")
plt.xlabel("Sepal Length")
plt.ylabel("Sepal Width")
plt.title("Sepal Length vs. Sepal Width")
plt.show()

plt.figure(figsize=(6, 4))
sns.boxplot(x=df["species"], y=df["petal_length"], palette="pastel")
```

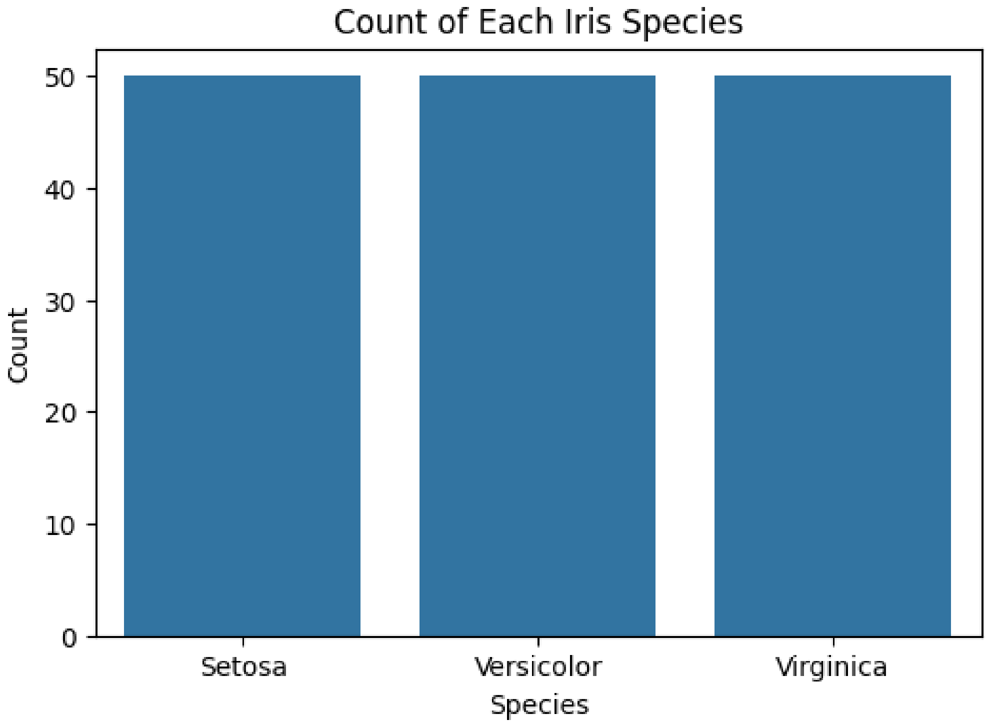
```
plt.xlabel("Species")
plt.ylabel("Petal Length")
plt.title("Petal Length Across Different Species")
plt.xticks(ticks=[0, 1, 2], labels=["Setosa", "Versicolor",
"Virginica"])
plt.show()
```

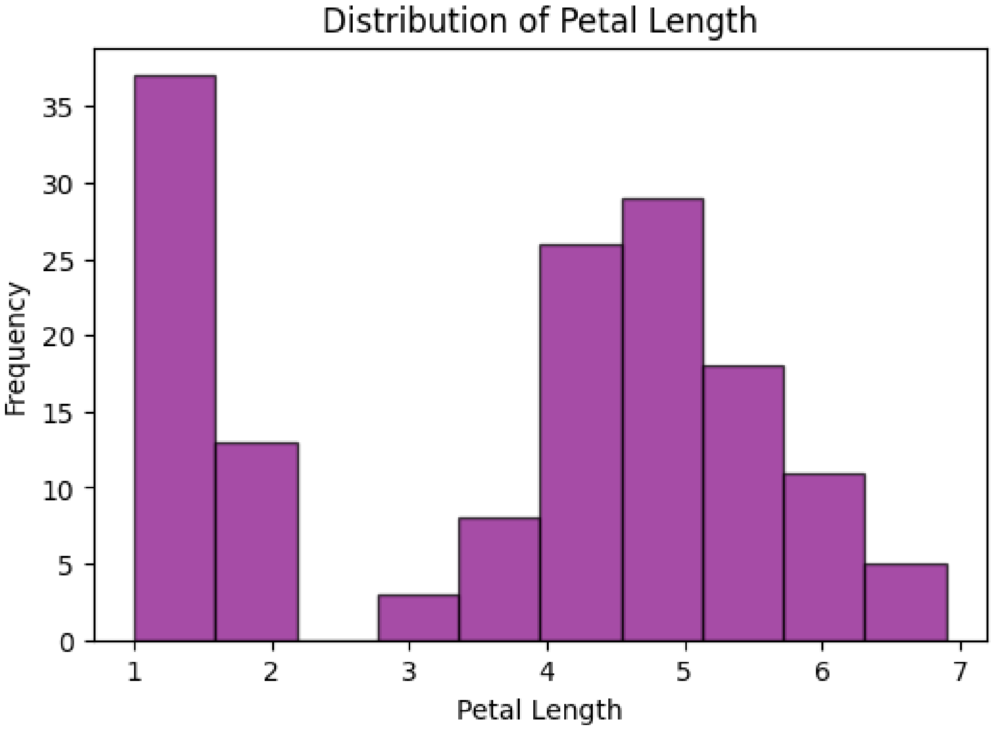
Output / Result

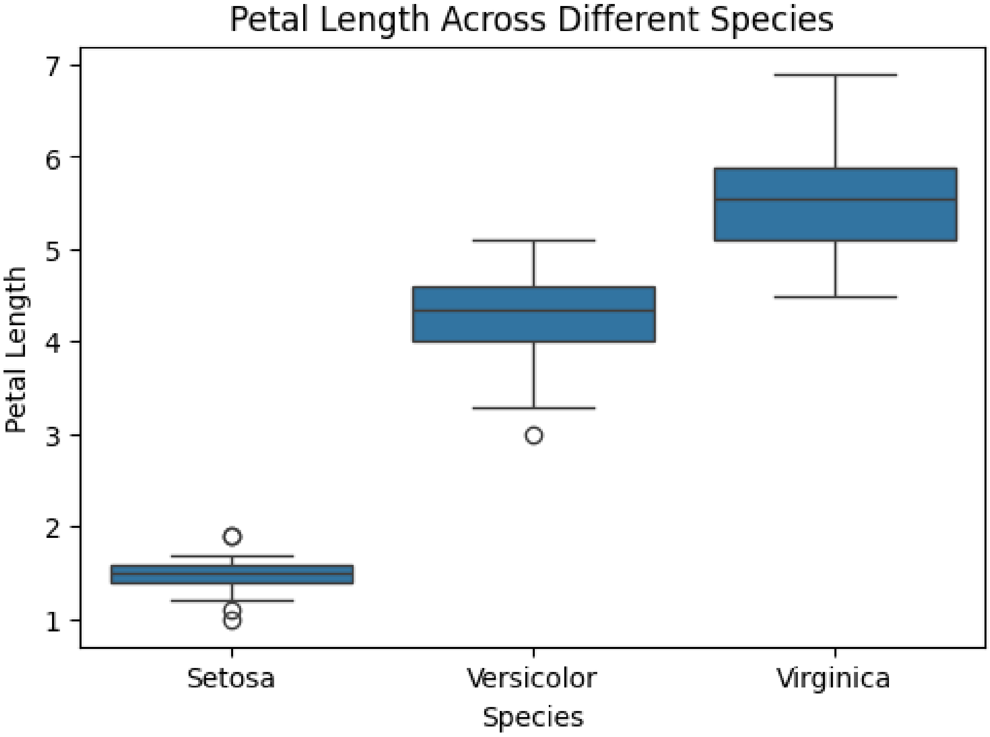
- **Model Accuracy:** [Paste your result here]
- **Visualizations:** *(Paste screenshots of your plotted graphs here)*

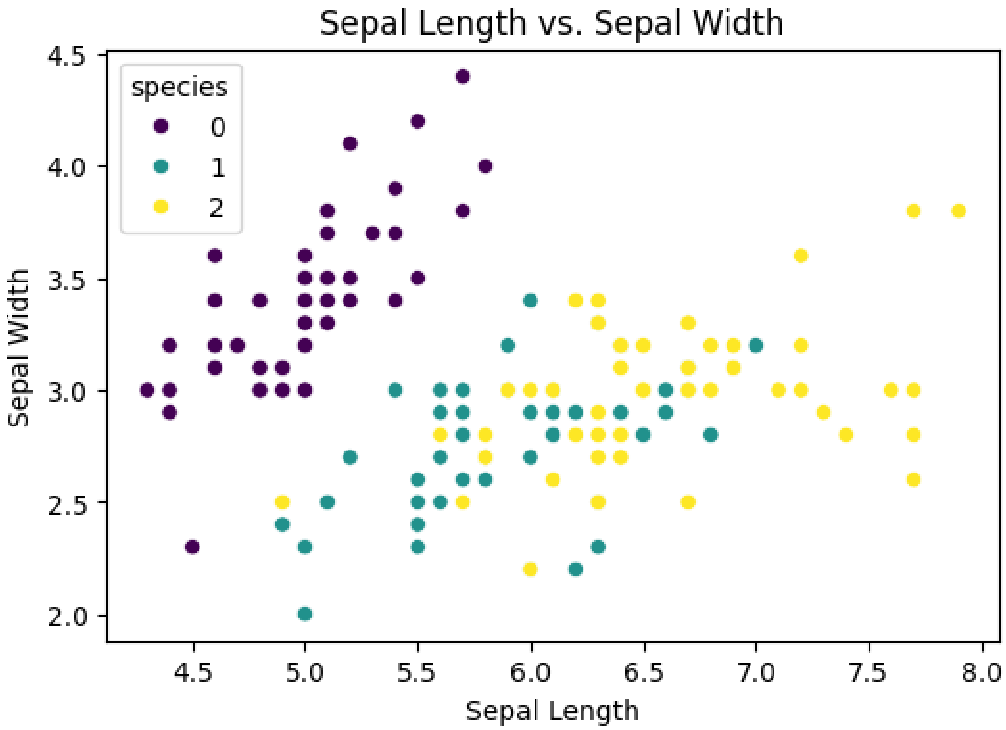
The results indicate that the k-NN classifier performs well on the Iris dataset due to its well-separated classes. The visualizations further help in understanding the data distribution, feature relationships, and class separability.

Output




Distribution of Petal Length


Petal Length Across Different Species



Conclusion

The **Iris Flower Classification** project successfully demonstrates the use of machine learning for simple classification tasks. The k-NN model achieved a good accuracy score by effectively utilizing four simple features of the iris flowers.

This project highlights the importance of **data preprocessing, visualization, and model evaluation** in machine learning. Future improvements could include trying different classification algorithms such as **decision trees, support vector machines (SVM), or neural networks** to compare performance.

References / Credits

- **Dataset:** UCI Machine Learning Repository - Iris Dataset (https://archive.ics.uci.edu/ml/datasets/Iris)
- **Images (if used):** [Provide source links if images were taken from external sources]
- **Libraries Used:** NumPy, Pandas, Matplotlib, Seaborn