Iris Classification Project

Overview

This project builds a machine learning model to classify iris flowers into three species: **Setosa**, **Versicolour**, and **Virginica** using features like sepal length, sepal width, petal length, and petal width. The project uses a K-Nearest Neighbors (KNN) classifier trained on the Iris dataset.

Folder Structure

```
projects/
├── data/
│   └── Iris.csv            # Dataset file (originally in Excel but saved as
CSV)
├── iris_classification.ipynb  # Main Jupyter Notebook with all code
├── knn_model.pkl           # (Optional) Saved trained model
└── README.md               # Project documentation (this file)
```

Requirements

- Python 3.10+
- Libraries:
- pandas
- numpy
- matplotlib
- seaborn
- scikit-learn
- joblib (optional, for saving the model)

Install with:

```
pip install numpy pandas matplotlib seaborn scikit-learn joblib
```

Steps Performed

1. **Import Libraries**
2. **Load Dataset**: Used `pandas.read_csv()` to load Iris.csv
3. **Explore Data**:
4. Viewed info, statistics
5. Visualized pairplots and correlation heatmaps
6. **Preprocess Data**:
7. Dropped 'Id' column

8. Split into features (`X`) and target (`y`)
9. Scaled features with `StandardScaler`
10. **Train Model**:
11. Used `KNeighborsClassifier` with k=3
12. **Make Predictions**: On test set
13. **Evaluate Model**:
14. Printed confusion matrix and classification report
15. Plotted heatmap for confusion matrix
16. **Tune k (Optional)**: Plotted error rate vs different k values
17. **Save Model (Optional)**: Used `joblib.dump()` to save KNN model

Results

- Model achieved high accuracy (>95%) on test data
- Plots show strong feature separation and minimal misclassification

Future Improvements

- Try different classifiers: SVM, Decision Tree
- Build a web app using Streamlit or Flask
- Deploy as a microservice or web application

License

Created by Maddineni Harshitha