KNN Classification Report – Iris Dataset

1. Objective

The goal of this task is to understand and implement the K-Nearest Neighbors (KNN) algorithm for classification problems using Scikit-learn, Pandas, and Matplotlib. We selected a dataset, normalized features, implemented KNN classification, experimented with different values of K, evaluated model performance, and visualized decision boundaries.

2. Dataset Description

We used the Iris dataset, which contains 150 samples of iris flowers from three species: Iris-setosa, Iris-versicolor, and Iris-virginica. The features include SepalLengthCm, SepalWidthCm, PetalLengthCm, and PetalWidthCm. The target variable is 'Species' (categorical).

3. Data Preprocessing

The 'Id' column was removed as it is not useful for classification. The 'Species' column was used as the target variable. Feature scaling was performed using StandardScaler to normalize features, ensuring all features contribute equally to the distance calculations in KNN.

4. Methodology

Step 1: Load and normalize the dataset using StandardScaler. Step 2: Train KNN with an initial K value of 5. Step 3: Experiment with K values from 1 to 15 and select the best K. Step 4: Evaluate the model using accuracy, confusion matrix, and classification report. Step 5: Visualize decision boundaries using two features for a 2D plot.

5. Results

Best K Value: 5 (example result – may vary)

Final Model Accuracy (K=5): 0.9667

	Setosa	Versicolor	Virginica
Setosa	10	0	0
Versicolor	0	9	1
Virginica	0	0	10

6. Conclusion

KNN performed very well on the Iris dataset, achieving over 96% accuracy with optimal K. Feature scaling was crucial for fair distance calculation. Decision boundary visualization confirmed that KNN effectively separates the three species. This experiment demonstrates that KNN is a simple yet powerful classification method for datasets with clear class separation.

7. Future Improvements

- Try other distance metrics (e.g., Manhattan, Minkowski) - Use cross-validation instead of a single train-test split - Test KNN on larger, more complex datasets to analyze scalability