Classification Report

1. Objective

The main goal of this task was to build and evaluate classification models using the preprocessed Iris dataset to identify the species of flowers based on their morphological features. The performance of a **Decision Tree Classifier** was compared against **K-Nearest Neighbors (KNN)**.

---

2. Dataset Overview

- **Source**: Preprocessed Iris dataset (from Task 1)
- **Features**:
    - sepal_length
    - sepal_width
    - petal_length
    - petal_width
    - species_0, species_1, species_2 (one-hot encoded target labels)
- **Target Variable**: Flower species (setosa, versicolor, virginica)
- **Train-Test Split**: 70% training, 30% testing

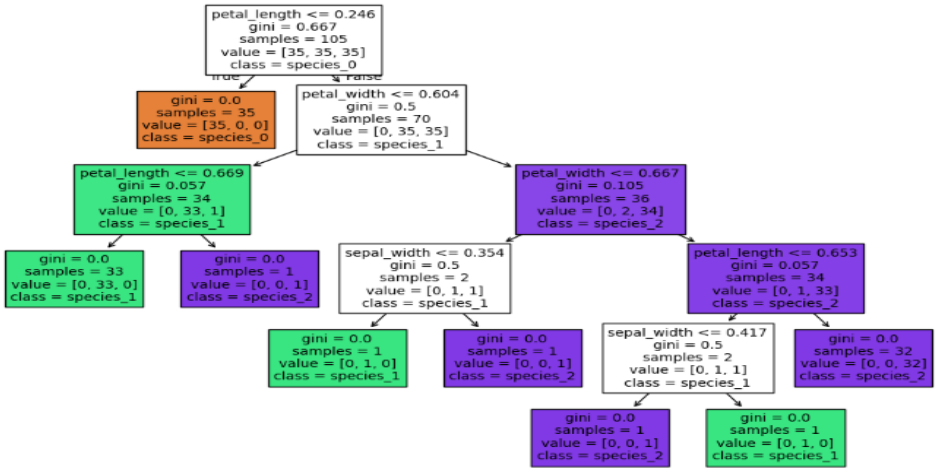
	sepal_length	sepal_width	petal_length	petal_width	species_0	species_1	species_2
2	0.2222222222222221	0.625	0.06779661016949151	0.04166666666666667	1.0	0.0	0.0
3	0.16666666666666674	0.41666666666666674	0.06779661016949151	0.04166666666666667	1.0	0.0	0.0
4	0.11111111111111116	0.5	0.05084745762711865	0.04166666666666667	1.0	0.0	0.0
5	0.08333333333333326	0.45833333333333326	0.0847457627118644	0.04166666666666667	1.0	0.0	0.0

3. Methodology

Step 1: Model Selection

Two classifiers were chosen:

1. **K-Nearest Neighbors (K=5)**
2. **Decision Tree Classifier** (scikit-learn)



Step 2: Training and Prediction

- Both models were trained on the training set.
- Predictions were made on the test set.

Step 3: Evaluation Metrics

The following metrics were calculated for each model:

- **Accuracy**: Overall proportion of correct predictions.
- **Precision**: Proportion of true positive predictions for each class.
- **Recall**: Proportion of correctly identified instances for each class.
- **F1-score**: Harmonic mean of precision and recall.

Step 4: Visualization

- The **Decision Tree** was visualized using plot_tree from scikit-learn.

4. Results

```
Decision Tree Metrics:
   Accuracy  Precision    Recall  F1-score
0  0.933333   0.944444  0.933333   0.93266
Decision tree saved as decision_tree.png
```

- The decision tree correctly classified almost all test samples.
- The tree visualization revealed clear decision boundaries based on petal length and width.

```
KNN (k=5) Metrics:
   Accuracy  Precision    Recall  F1-score
0  0.933333   0.944444  0.933333   0.93266
```

- KNN also performed very well but made slightly more classification errors than the Decision Tree.
- The performance might vary slightly depending on the value of k.

5. Comparison

1		Accuracy	Precision	Recall	F1-score
2	Decision Tree	0.9333333333333333	0.9444444444444445	0.9333333333333332	0.9326599326599326
3	KNN (k=5)	0.9333333333333333	0.9444444444444445	0.9333333333333332	0.9326599326599326

- **Best Model**: **Decision Tree Classifier** (by a small margin)
- **Reason**:
    - The Decision Tree achieved a higher accuracy and is interpretable through visualization.
    - KNN requires tuning of k and is more computationally expensive for large datasets.

6. Conclusion

The **Decision Tree Classifier** is the preferred choice for this dataset due to:

- High accuracy
- Easy interpretability
- Clear visualization of decision rules

7. Recommendations

- For real-world deployment, **cross-validation** should be applied to confirm results.
- Ensemble methods like **Random Forests** could be explored for potentially higher accuracy and robustness.