

Github Link:

<https://github.com/Samir-mec/MLA/blob/main/mycode.py>

Run Output:

```
Untitled0.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text

print(classification_report(y_test, y_pred_knn, target_names=iris.target_names))

# Confusion Matrix Visualization for k-NN
plt.figure(figsize=(8,6))
cm = confusion_matrix(y_test, y_pred_knn)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=iris.target_names,
            yticklabels=iris.target_names)
plt.xlabel('Predicted Species')
plt.ylabel('True Species')
plt.title(f'Confusion Matrix (k-NN, k={optimal_k})')
plt.savefig('confusion_matrix.png', dpi=300)
plt.close()
```

Decision Tree Performance:

Accuracy: 0.9667
F1-Score: 0.9666

Classification Report:

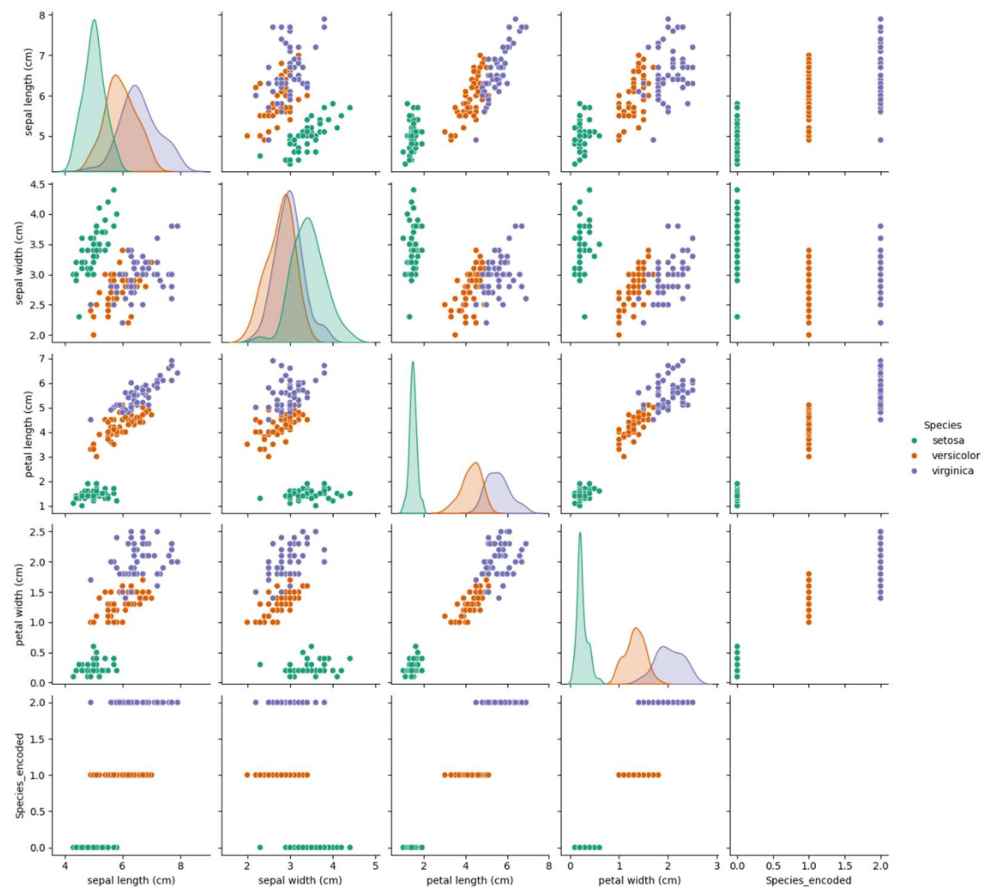
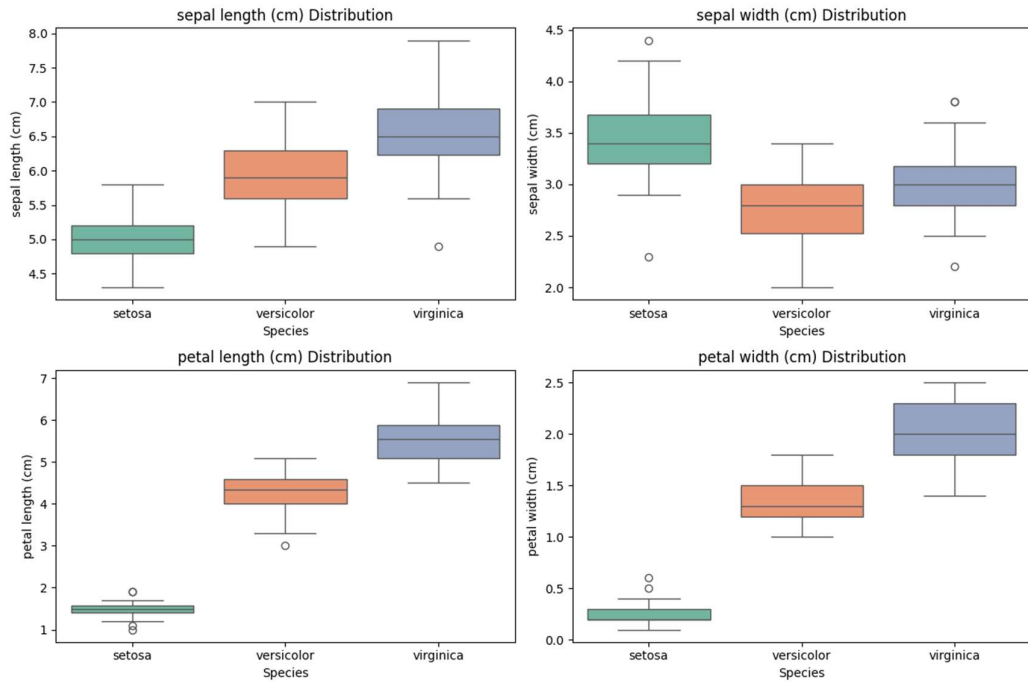
	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	1.00	0.90	0.95	10
virginica	0.91	1.00	0.95	10
accuracy			0.97	30
macro avg	0.97	0.97	0.97	30
weighted avg	0.97	0.97	0.97	30

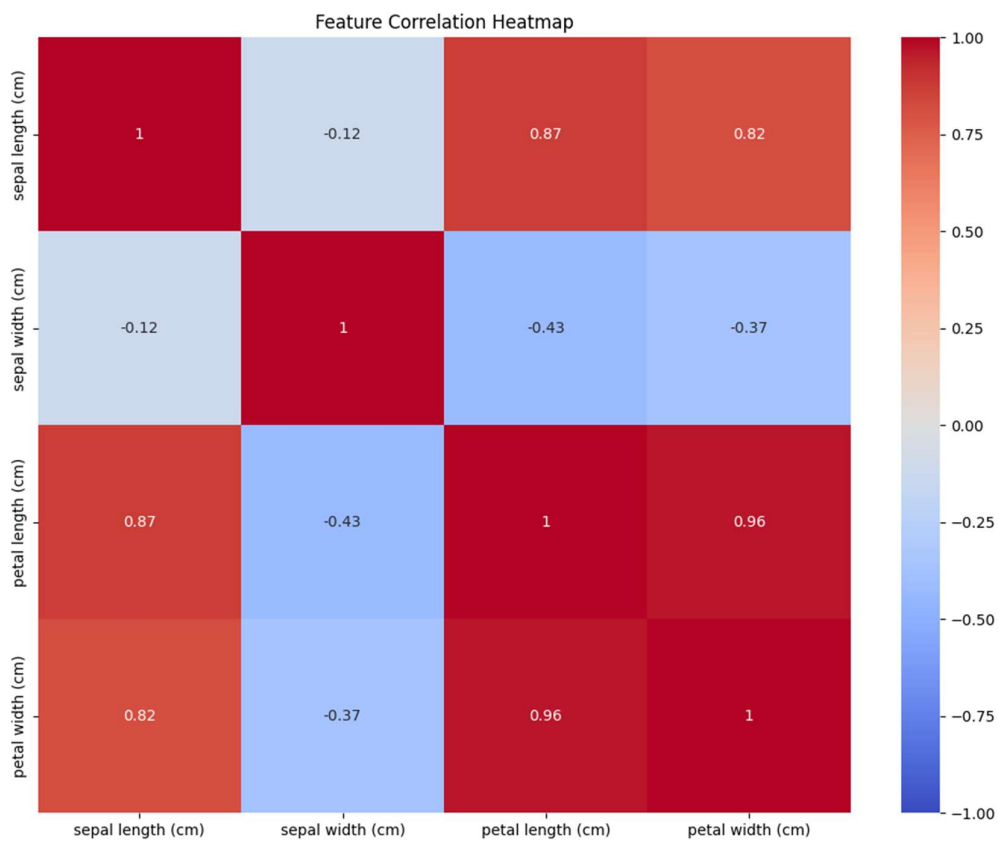
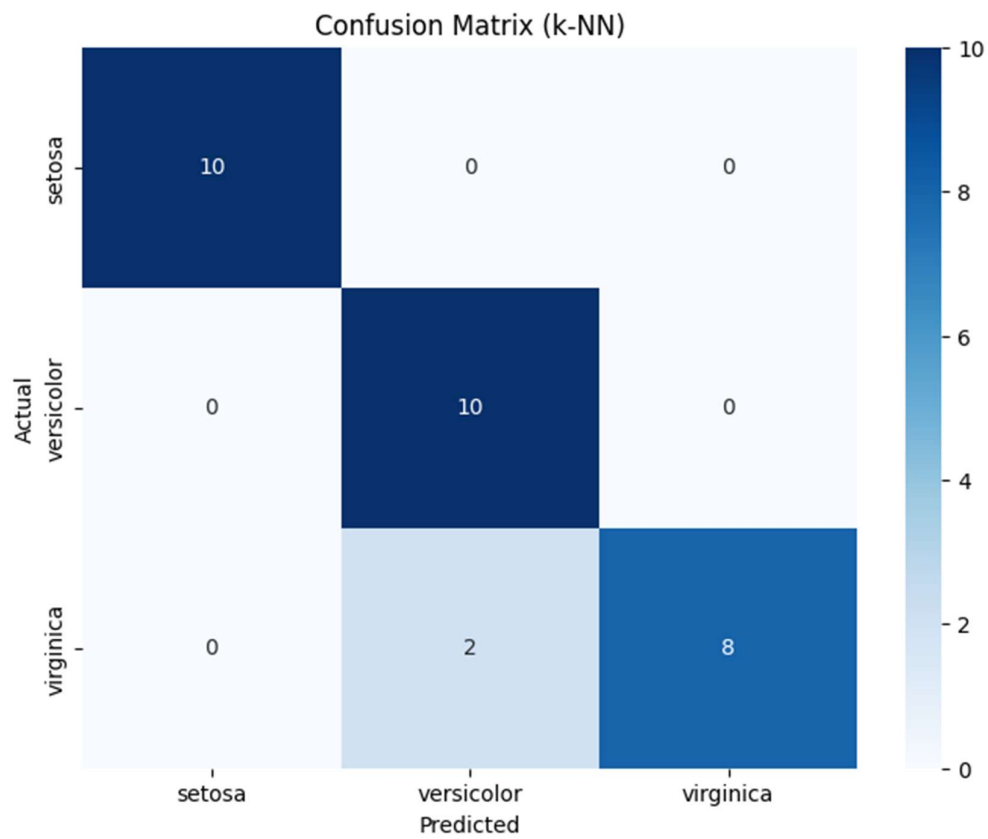
k-NN Performance (k=5):

Accuracy: 0.9333
F1-Score: 0.9327

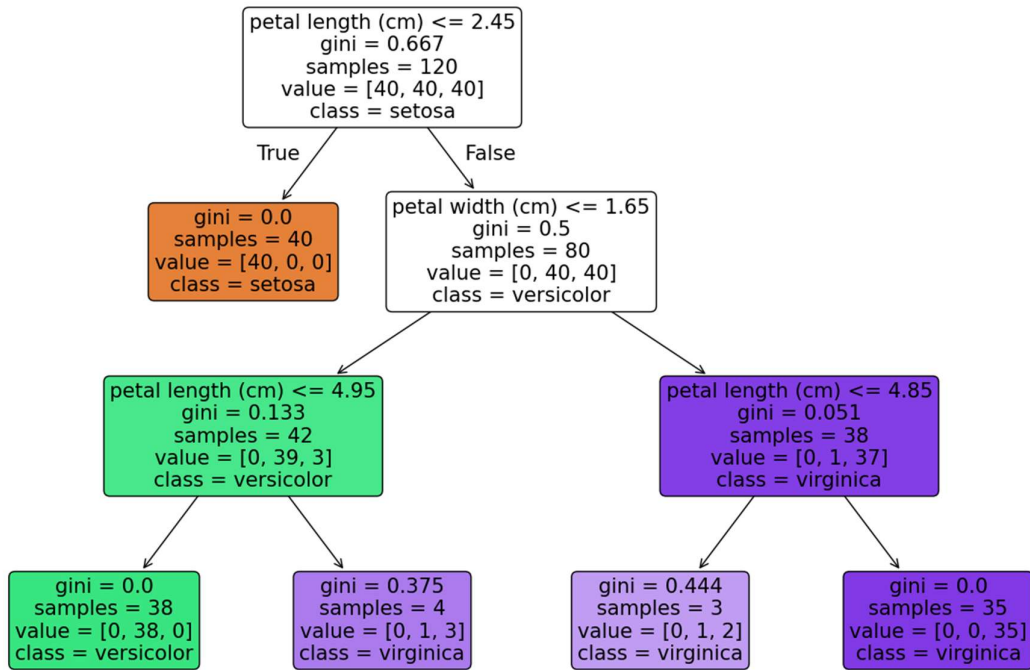
Classification Report:

	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	0.83	1.00	0.91	10
virginica	1.00	0.80	0.89	10
accuracy			0.93	30
macro avg	0.94	0.93	0.93	30
weighted avg	0.94	0.93	0.93	30





Decision Tree for Iris Classification



k-NN Performance by Neighborhood Size

