EE69210: Machine Learning for Signal Processing Laboratory
Department of Electrical Engineering, Indian Institute of Technology, Kharagpur

Decision Trees and Random Forests

Anirvan Krishna | Roll no. 21EE38002

Keywords: Decision Trees, Ensemble Models, Random Forests, Bagging

---

Grading Rubric

	Tick the best applicable per row			**Points**
	Below Expectations	Lacking in Some	Meets all Expectations	
Completeness of the report				
Organization of the report (5 pts)				
Quality of figures (5 pts)				
Implement a decision tree using the dataset and compute the accu- racy (30 pts)				
Create bags using the bootstrap al- gorithm (30 pts)				
Implement a random forest using 50 decision trees and compute the accu- racy (40 pts)				
			TOTAL (100 pts)	

1. Decision Trees

A **decision tree** is a supervised learning algorithm used for classification and regression tasks. It recursively partitions the dataset based on feature values to create a tree-like structure that makes predictions by following a sequence of decision rules.

Splitting Criterion (Information Gain): At each node, the dataset is split to maximize the homogeneity of the resulting subsets. The **Information Gain (IG)** is used to determine the optimal split by measuring the reduction in entropy before and after the split.

Entropy: Entropy quantifies the uncertainty in a dataset and is given by:

$$H(S) = -\sum_{i=1}^{C} p_i \log_2 p_i \tag{1}$$

where S represents the dataset at the current node, C is the number of classes, and p_i is the proportion of samples belonging to class i.

Information Gain: Information Gain measures the reduction in entropy after splitting the dataset based on a feature. It is defined as:

$$IG = H(S) - \sum_{j} w_j H(S_j) \tag{2}$$

where $H(S)$ is the entropy before splitting, $H(S_j)$ is the entropy of child node j, w_j is the proportion of samples in child node j. The split that results in the highest *Information Gain* is chosen at each step to construct the decision tree. The recursive algorithm to build a decision tree is as follows:

Algorithm 1 Build Decision Tree

[**Require:**]Feature matrix X, target vector Y, min split size min_samples_split, min information gain min_info_gain
[**Ensure:**]Root node of the decision tree
Create node $N(X,Y)$
Compute node entropy $H(Y)$
Compute class probabilities
if N is pure or $|Y| <$ min_samples_split **then**
    Mark N as a leaf **return** N
end if
left_data, right_data, feature, threshold, info_gain $\leftarrow$ FIND BEST SPLIT(X,Y)
if info_gain $<$ min_info_gain **then**
    Mark N as a leaf **return** N
end if
Set N's split feature, threshold, and information gain
Recursively build left and right subtrees:
N.left $\leftarrow$ BUILD DECISION TREE(left_data)
N.right $\leftarrow$ BUILD DECISION TREE(right_data) **return** N

Inference in a decision tree: The predicted label $\hat{y}_i$ for a given data sample $\mathbf{x}_i$.

$$\hat{y}_i = \arg\max_{\omega \in \Omega} \{p(\omega|\mathbf{x}_i)\} \tag{3}$$

where Ω is the set of all class labels. Recursively, it is implemented as follows:

Algorithm 2 Predict Using Decision Tree

1: [**Require:**]Trained tree root N, input sample X
2: [**Ensure:**]Predicted class label
3: **if** N is a leaf **then return** Majority class in N
4: **end if**
5: **if** $X_{\text{split\_feature}} <$ split_threshold **then return** PREDICT(N.left, X)
6: **elsereturn** PREDICT(N.right, X)
7: **end if**

2. Bagging and Random Forests

Random Forests is an ensemble learning method that combines multiple decision trees to improve prediction accuracy and reduce overfitting. Instead of relying on a single decision tree, it builds a collection (or "forest") of trees and aggregates their outputs, making it more robust and generalizable.

Bootstrap Sampling (Bagging): Given a dataset D with N samples, multiple subsets D_b are created by randomly sampling with replacement from D. Each decision tree in the forest is trained on a different subset D_b.

Random Feature Selection: When splitting a node, instead of considering all features, a random subset of m features (where $m < M$, and M is the total number of features) is chosen. This introduces further decorrelation between the trees.

Tree Construction using Information Gain: Each decision tree in the forest follows the standard splitting criterion based on Information Gain, calculated as:

$$IG(S,A) = H(S) - \sum_{v \in \text{values}(A)} \frac{|S_v|}{|S|} H(S_v)$$

where:

- $H(S)$ is the entropy of the dataset S,
- S_v is the subset of S after splitting on feature A,
- $H(S_v)$ is the entropy of subset S_v.

Voting (Classification) or Averaging (Regression)

- **For classification**, the final output is determined by majority voting among the trees:

$$\hat{y} = \arg\max_c \sum_{t=1}^{T} \mathbb{I}(h_t(x) = c)$$

  where $h_t(x)$ is the prediction of the t-th tree, and $\mathbb{I}(\cdot)$ is the indicator function.

- **For regression**, the output is the average of all tree predictions:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} h_t(x)$$

The recursive algorithms for building a random forest and inference of labels corresponding to given samples are discussed below.

Algorithm 3 Train a Random Forest

[**Require:**]Feature matrix X, target vector Y, number of trees T, minimum samples to split min_samples_split, minimum information gain min_info_gain
[**Ensure:**]Trained random forest with T decision trees
Initialize an empty forest: Forest $\leftarrow$ []
for $i = 1$ to T **do**
    Randomly sample feature indices S of size $\frac{|X|}{3}$ with replacement
    Train decision tree $T_i \leftarrow$ BUILD DECISION TREE$(X_S, Y_S, \text{min_samples_split}, \text{min_info_gain})$
    Add T_i to the forest: Forest $\leftarrow$ Forest $\cup \{T_i\}$
end for
return Forest

Algorithm 4 Predict Using a Random Forest

[**Require:**]Trained forest Forest, input sample x
[**Ensure:**]Predicted class label
Initialize an empty list of predictions: $P \leftarrow [\,]$
for each tree T_i in Forest **do**
    Predict class label $y_i \leftarrow$ PREDICT(T_i, x)
    Append y_i to P
end for
Compute the most frequent class in P:

$$y^* \leftarrow \arg\max \text{bincount}(P)$$

return y^*

3. Sample datasets for training and inference

3.1 XOR Dataset

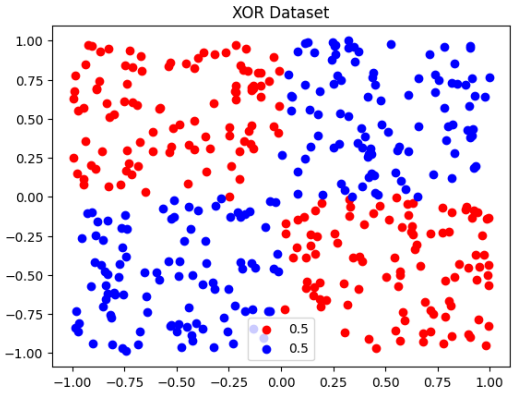


Fig. 1. XOR Dataset

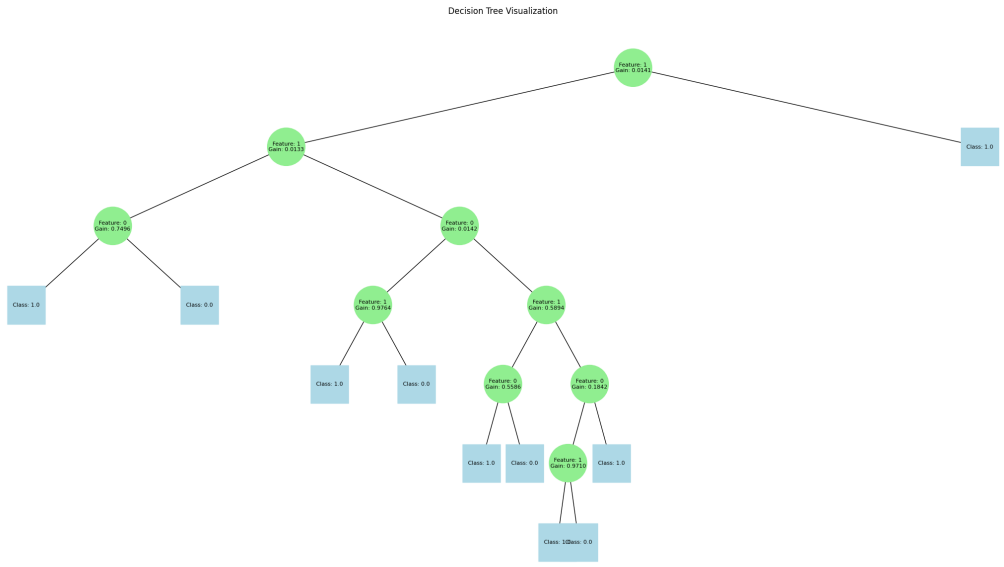


Fig. 2. Visualization of binary decision tree created on the XOR dataset

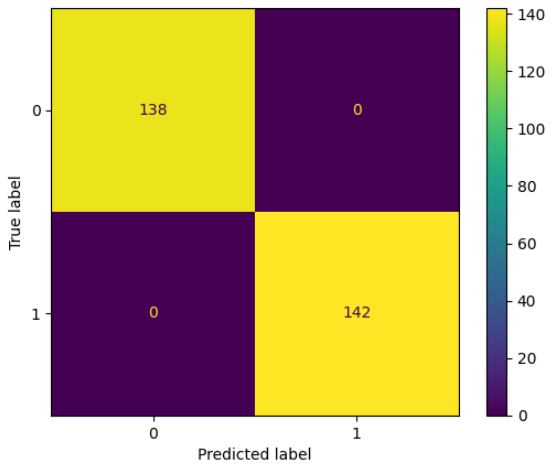


Fig. 3. Train data

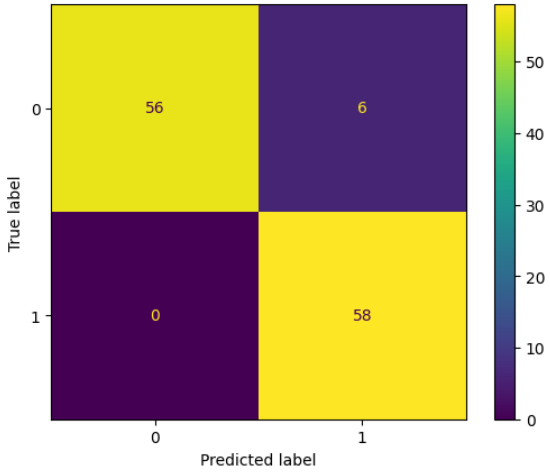
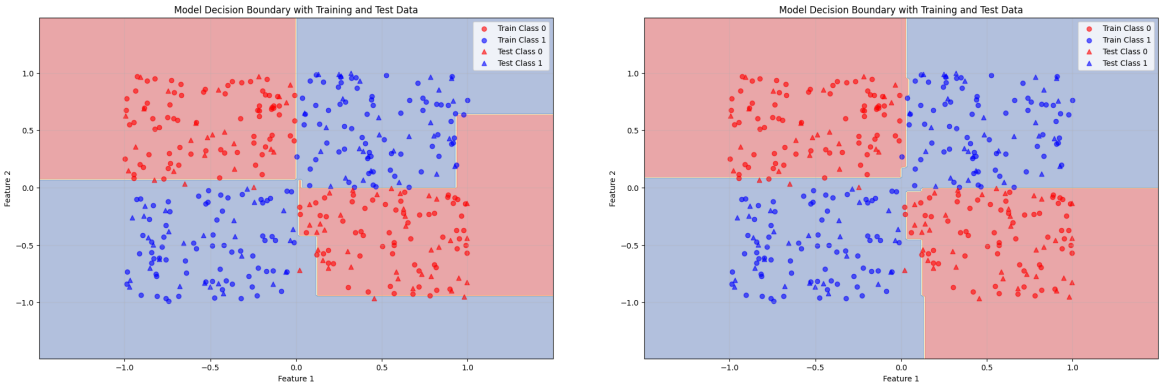


Fig. 4. Test data

Fig. 5. Confusion matrix for train and test data for decision tree trained on XOR Dataset

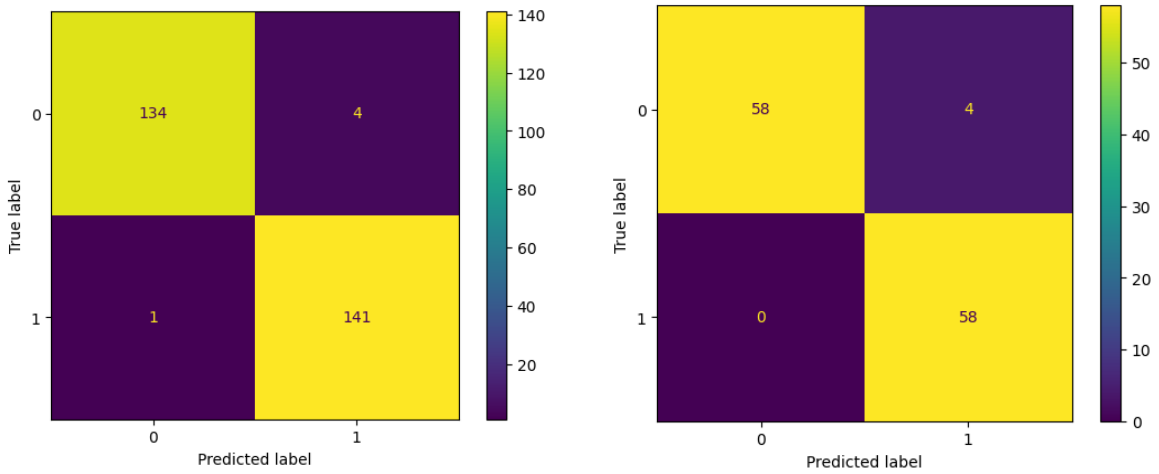
	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	1.00	1.00	1.00	0.90	0.95
1	1.00	1.00	1.00	0.91	1.00	0.95
Accuracy	1.00 (280 samples)			0.95 (120 samples)		
Macro Avg	1.00	1.00	1.00	0.95	0.95	0.95
Weighted Avg	1.00	1.00	1.00	0.95	0.95	0.95

Table 1. Classification report for decision tree trained on train and test sets from XOR dataset



Decision Tree Random Forest

Fig. 6. Decision boundaries for a decision tree and random forest with 10 trees trained on the XOR dataset



Train data Test data

Fig. 7. Confusion matrix for train and test data for a random forest with 10 decision trees trained on XOR Dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	0.99	0.97	0.98	1.00	0.94	0.97
1	0.97	0.99	0.98	0.94	1.00	0.97
Accuracy	0.98 (280 samples)			0.97 (120 samples)		
Macro Avg	0.98	0.98	0.98	0.97	0.97	0.97
Weighted Avg	0.98	0.98	0.98	0.97	0.97	0.97

Table 2. Classification report for train and test sets of XOR dataset for random forest

3.2 Concentric circles dataset

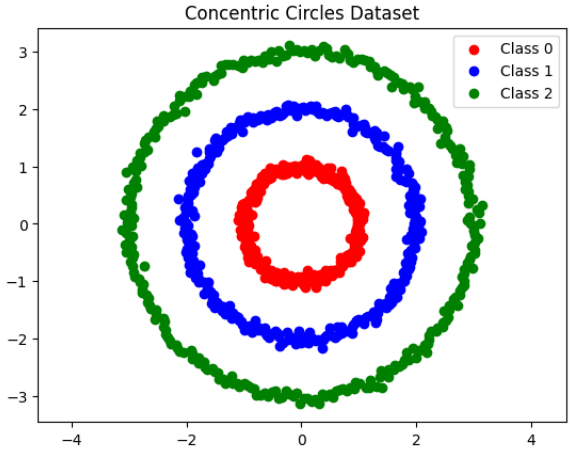


Fig. 8. Concentric circles dataset

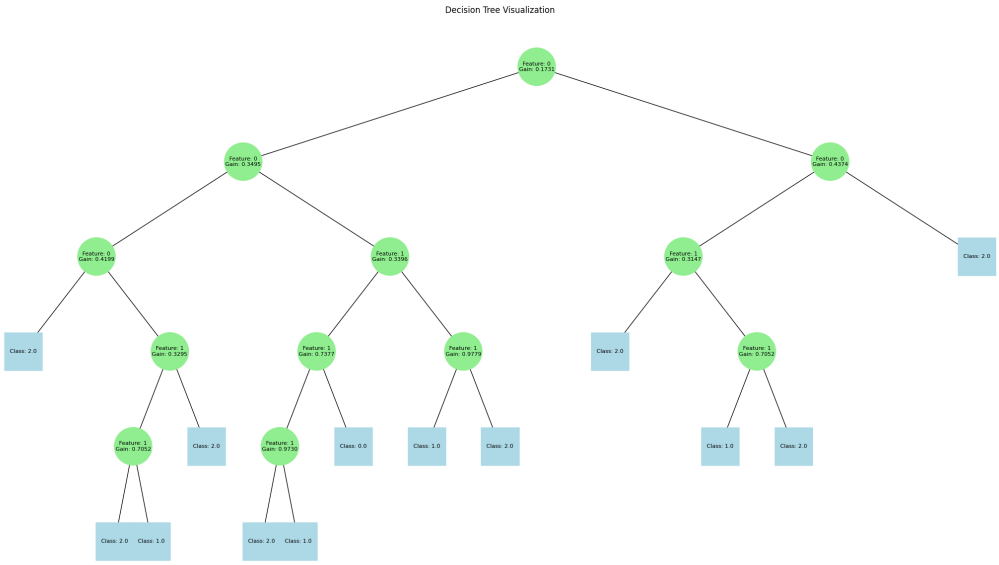
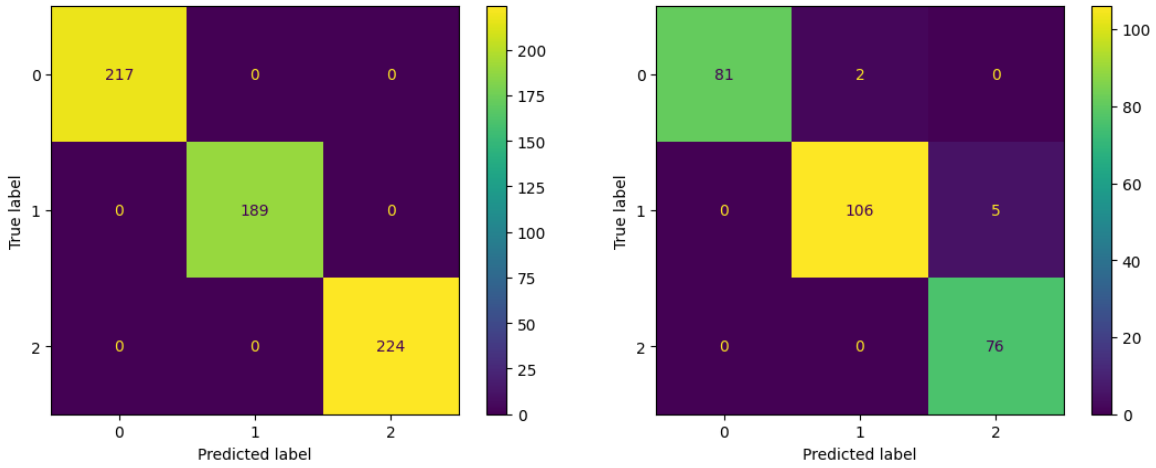


Fig. 9. Binary decision tree trained on the concentric circles dataset

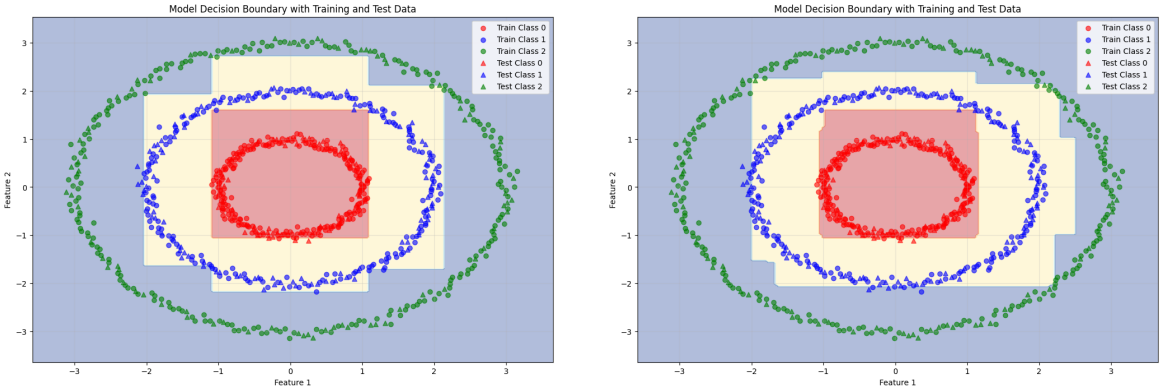
	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	1.00	1.00	1.00	0.98	0.99
1	1.00	1.00	1.00	0.98	0.95	0.97
2	1.00	1.00	1.00	0.94	1.00	0.97
Accuracy	1.00 (630 samples)			0.97 (270 samples)		
Macro Avg	1.00	1.00	1.00	0.97	0.98	0.97
Weighted Avg	1.00	1.00	1.00	0.98	0.97	0.97

Table 3. Classification report for train and test sets from concentric circles dataset with decision tree trained on it



Train data Test data

Fig. 10. Confusion matrix for train and test data for a decision tree trained on concentric circles dataset

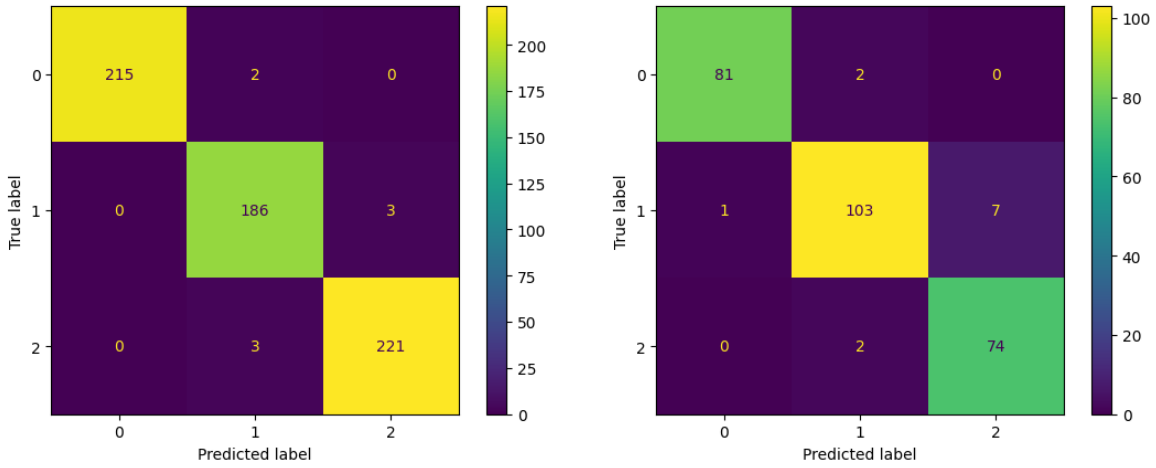


Decision Tree Random Forest

Fig. 11. Decision boundaries for a decision tree and random forest with 10 trees trained on the concentric circles dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	0.99	1.00	0.99	0.98	0.98
1	0.97	0.98	0.98	0.96	0.93	0.94
2	0.99	0.99	0.99	0.91	0.97	0.94
Accuracy	0.99 (630 samples)			0.96 (270 samples)		
Macro Avg	0.99	0.99	0.99	0.95	0.96	0.96
Weighted Avg	0.99	0.99	0.99	0.96	0.96	0.96

Table 4. Classification report for train and test sets for random forest with 10 trees trained on concentric circles dataset



Train data Test data

Fig. 12. Confusion matrix for train and test data for random forest with 10 trees trained on concentric circles dataset

3.3 Spiral dataset

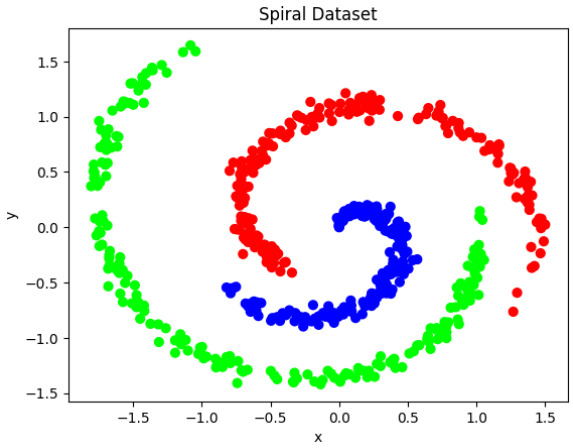


Fig. 13. Spiral dataset

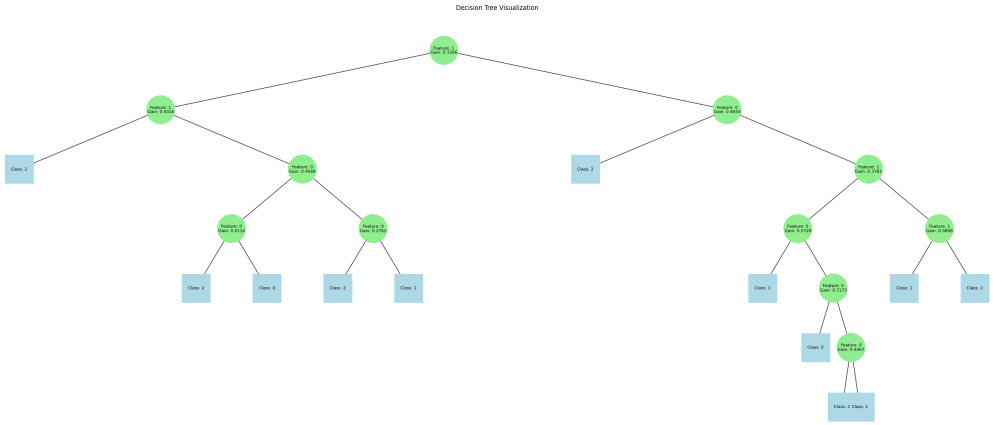
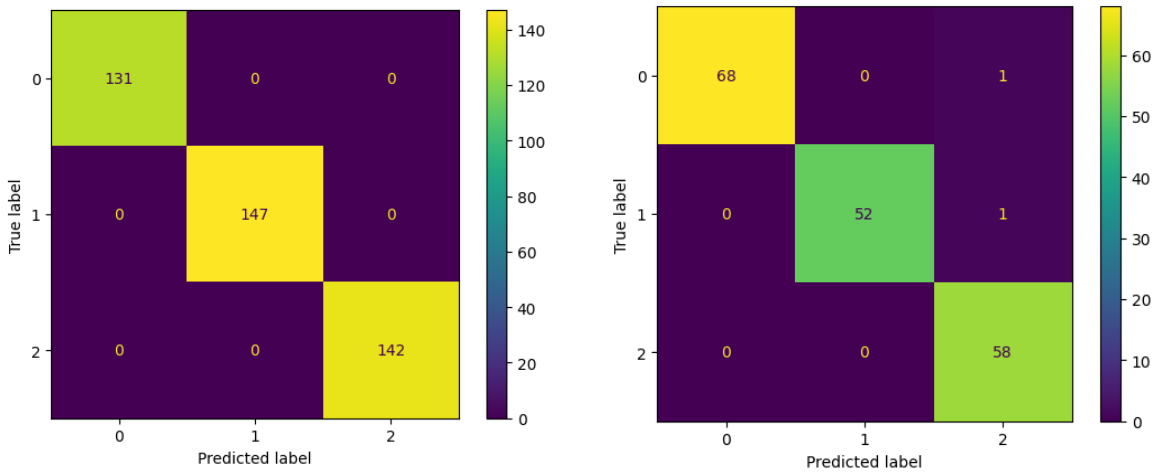


Fig. 14. Binary decision tree trained on spiral dataset

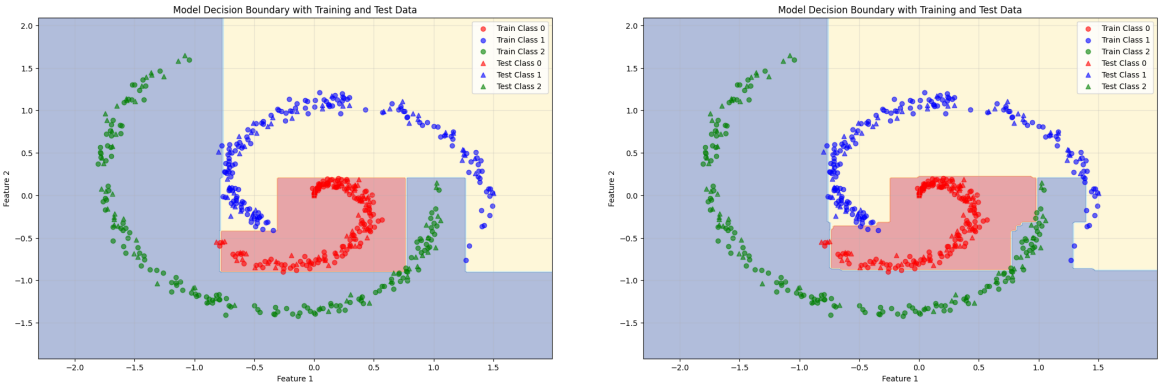


Train data Test data

Fig. 15. Confusion matrix for train and test data for decision tree trained on spiral dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	1.00	1.00	1.00	0.99	0.99
1	1.00	1.00	1.00	1.00	0.98	0.99
2	1.00	1.00	1.00	0.97	1.00	0.98
Accuracy	1.00 (420 samples)			0.99 (180 samples)		
Macro Avg	1.00	1.00	1.00	0.99	0.99	0.99
Weighted Avg	1.00	1.00	1.00	0.99	0.99	0.99

Table 5. Classification report for decision tree trained on spiral dataset

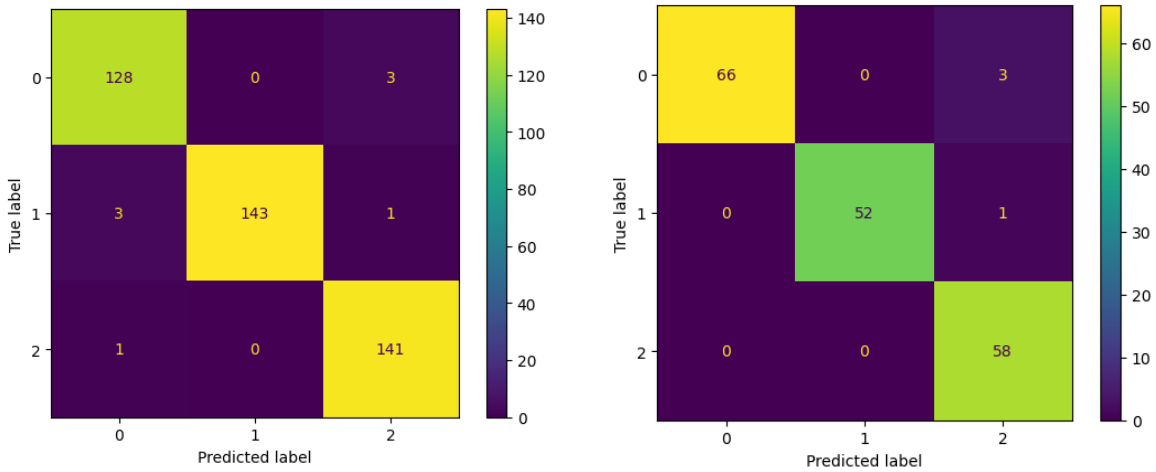


Decision Tree Random Forest

Fig. 16. Decision boundaries for a decision tree and random forest with 10 trees trained on the spiral dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	0.97	0.98	0.97	1.00	0.96	0.98
1	1.00	0.97	0.99	1.00	0.98	0.99
2	0.97	0.99	0.98	0.94	1.00	0.97
Accuracy	0.98 (420 samples)			0.98 (180 samples)		
Macro Avg	0.98	0.98	0.98	0.98	0.98	0.98
Weighted Avg	0.98	0.98	0.98	0.98	0.98	0.98

Table 6. Classification report for train and test sets for random forest with 10 trees trained on spiral dataset



Train data

Test data

Fig. 17. Confusion matrix for train and test data for random forest trained on spiral dataset

3.4 Balanced spiral dataset

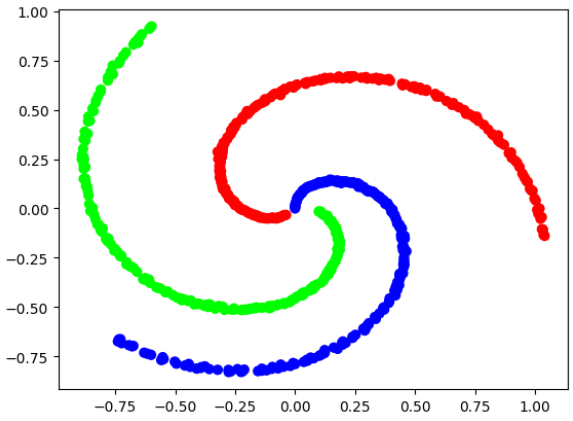


Fig. 18. Balanced spiral dataset

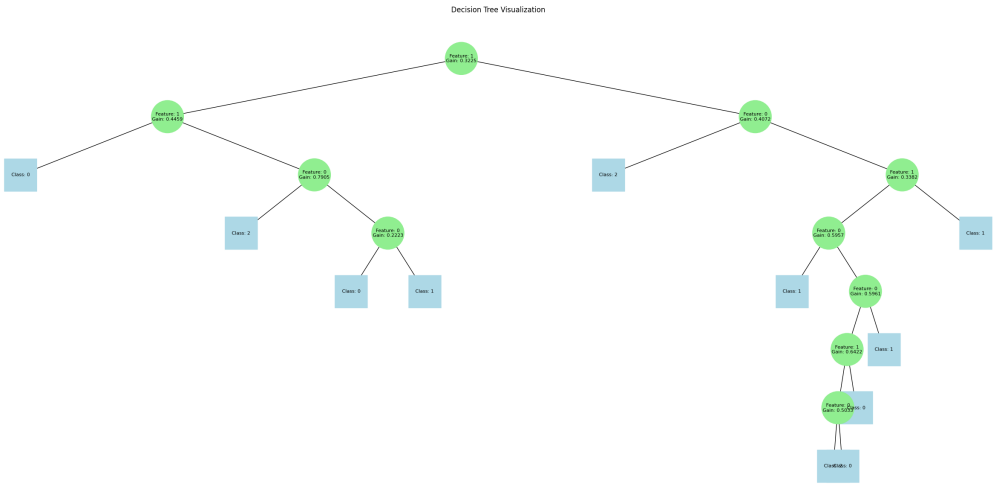
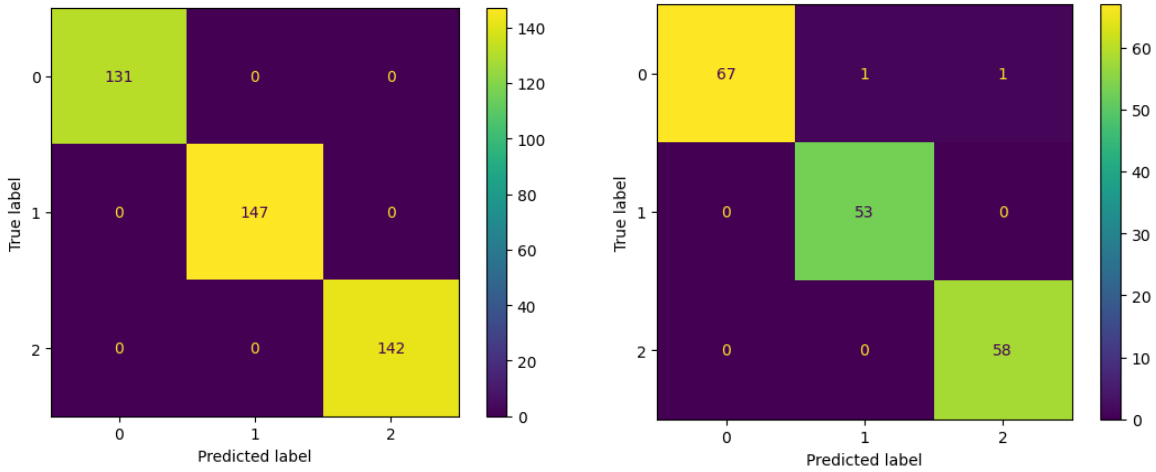


Fig. 19. Binary decision tree trained on balanced spiral dataset



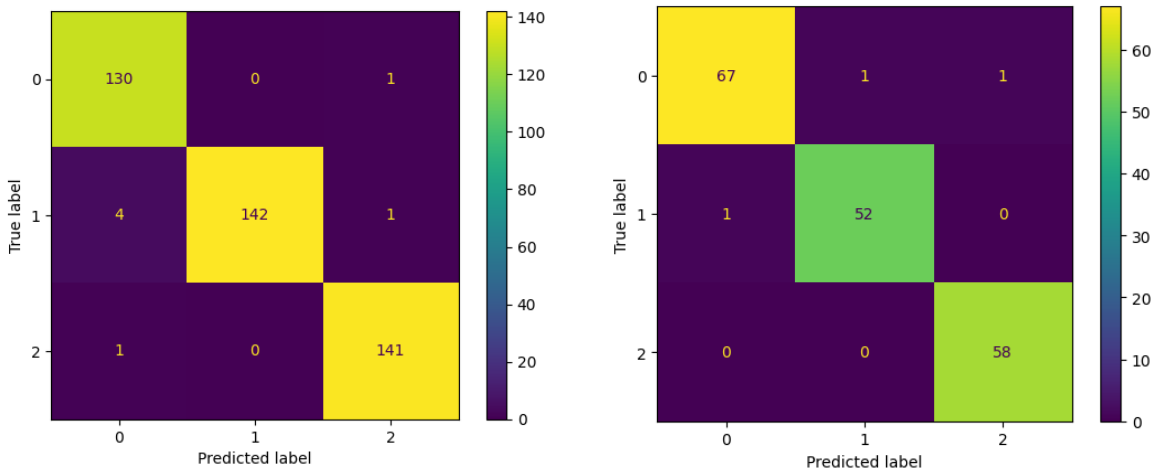
Train data

Test data

Fig. 20. Confusion matrix for train and test data for decision tree trained on spiral dataset

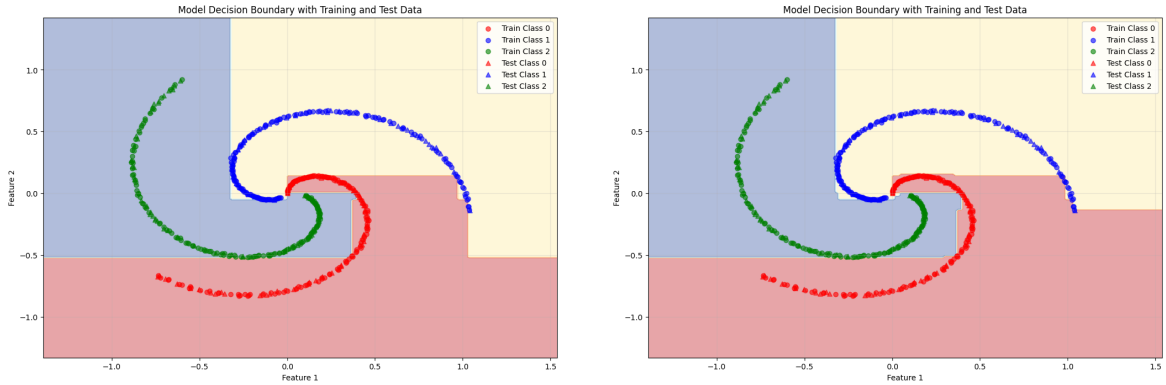
	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	1.00	1.00	1.00	0.97	0.99
1	1.00	1.00	1.00	0.98	1.00	0.99
2	1.00	1.00	1.00	0.98	1.00	0.99
Accuracy	1.00 (420 samples)			0.99 (180 samples)		
Macro Avg	1.00	1.00	1.00	0.99	0.99	0.99
Weighted Avg	1.00	1.00	1.00	0.99	0.99	0.99

Table 7. Classification Report for decision tree on balanced spiral dataset



Train data Test data

Fig. 21. Confusion matrix for train and test data for random forest trained on spiral dataset



Decision Tree Random Forest

Fig. 22. Decision boundary for decision tree and random forest trained on spiral dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	0.96	0.99	0.98	0.99	0.97	0.98
1	1.00	0.97	0.98	0.98	0.98	0.98
2	0.99	0.99	0.99	0.98	1.00	0.99
Accuracy	0.98 (420 samples)			0.98 (180 samples)		
Macro Avg	0.98	0.98	0.98	0.98	0.98	0.98
Weighted Avg	0.98	0.98	0.98	0.98	0.98	0.98

Table 8. Classification report for random forest on balanced spiral dataset

3.5 5-class spiral

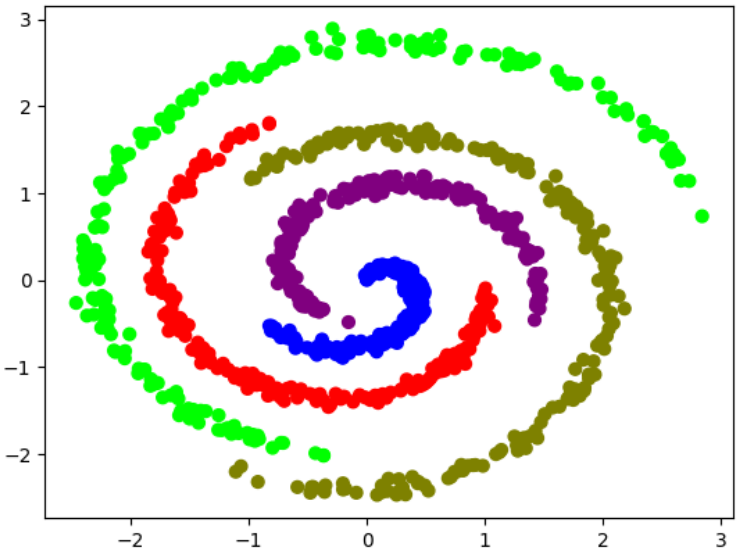


Fig. 23. Five-class unbalanced spiral dataset

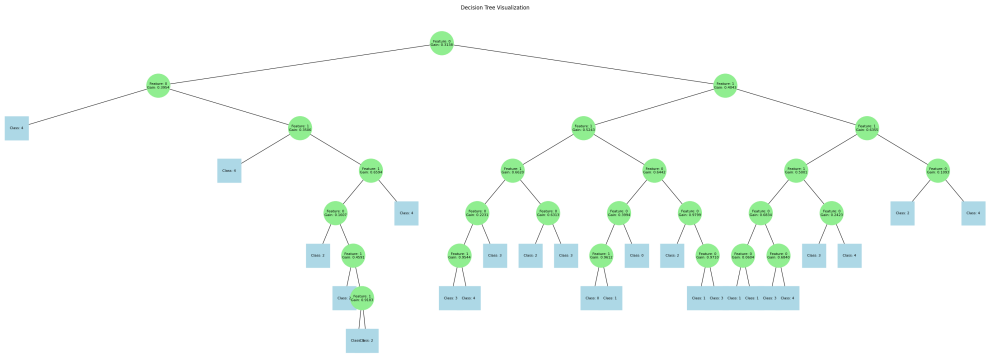
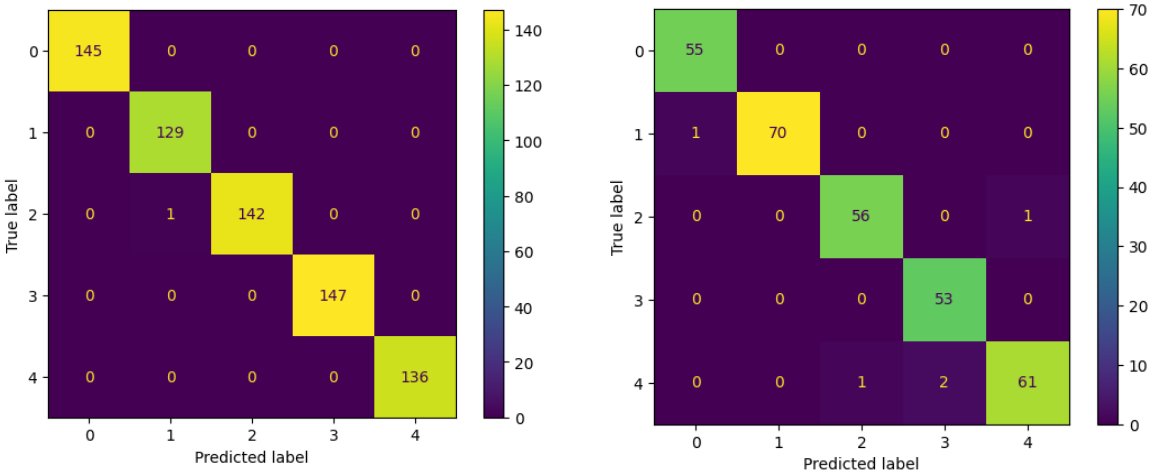


Fig. 24. Binary decision tree for 5-class spiral dataset



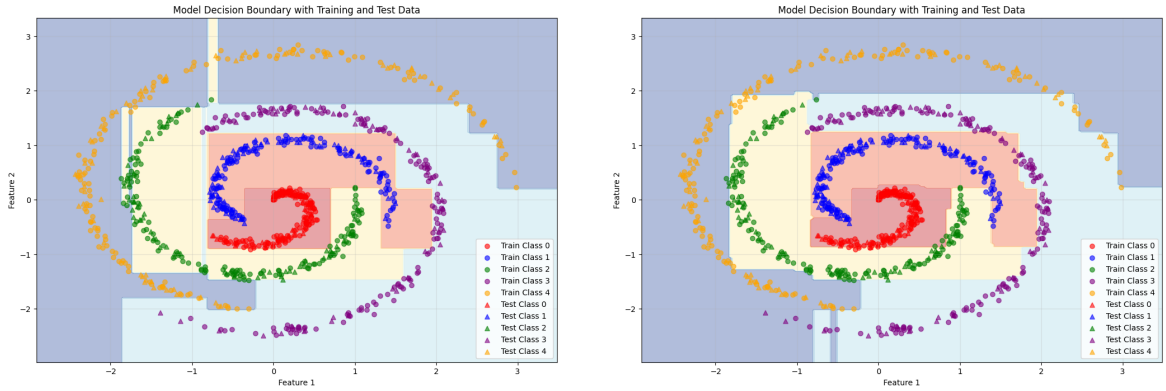
Train set

Test set

Fig. 25. Confusion matrix for decision tree trained on 5 class spiral dataset

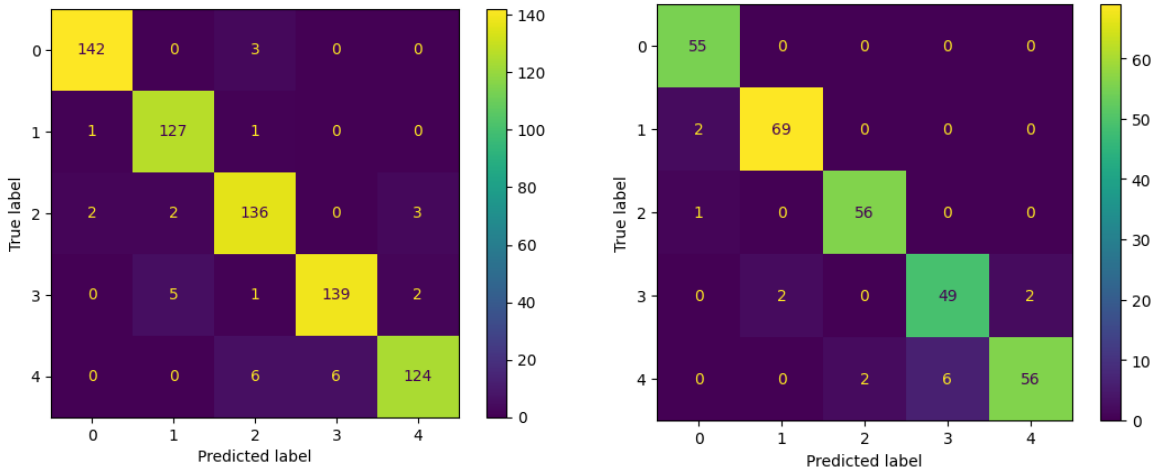
	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	1.00	1.00	0.98	1.00	0.99
1	0.99	1.00	1.00	1.00	0.99	0.99
2	1.00	0.99	1.00	0.98	0.98	0.98
3	1.00	1.00	1.00	0.96	1.00	0.98
4	1.00	1.00	1.00	0.98	0.95	0.97
Accuracy	1.00 (700 samples)			0.98 (300 samples)		
Macro Avg	1.00	1.00	1.00	0.98	0.98	0.98
Weighted Avg	1.00	1.00	1.00	0.98	0.98	0.98

Table 9. Classification report for train and test sets of 5-class spiral dataset for a decision tree



Decision Tree Random Forest

Fig. 26. Decision boundary for decision tree and random forest trained on 5 class spiral dataset



Train set Test set

Fig. 27. Confusion matrix for random forest trained on 5 class spiral dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	0.98	0.98	0.98	0.95	1.00	0.97
1	0.95	0.98	0.97	0.97	0.97	0.97
2	0.93	0.95	0.94	0.97	0.98	0.97
3	0.96	0.95	0.95	0.89	0.92	0.91
4	0.96	0.91	0.94	0.97	0.88	0.92
Accuracy	0.95 (700 samples)			0.95 (300 samples)		
Macro Avg	0.95	0.95	0.95	0.95	0.95	0.95
Weighted Avg	0.95	0.95	0.95	0.95	0.95	0.95

Table 10. Classification report for random forest trained on 5-class spiral dataset

3.6 5-class balanced spiral

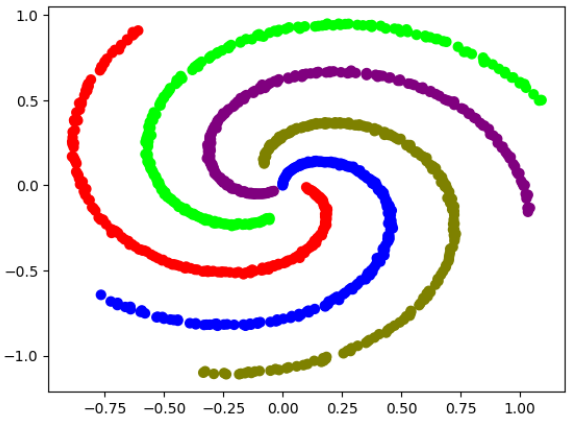


Fig. 28. Five class balanced spiral dataset

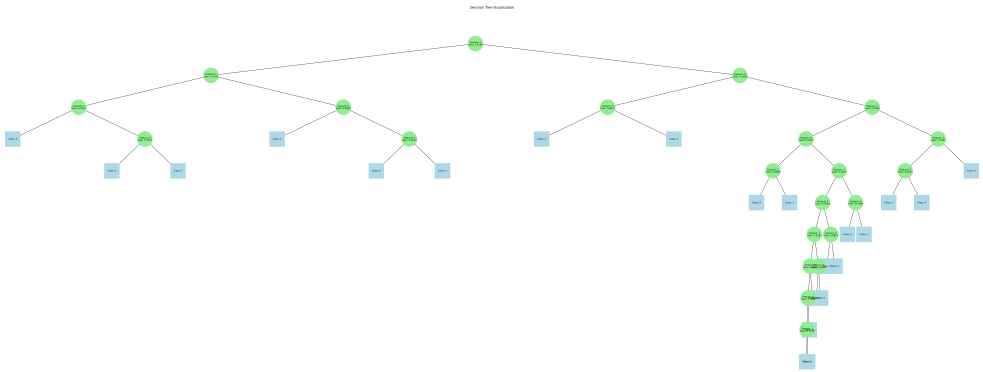
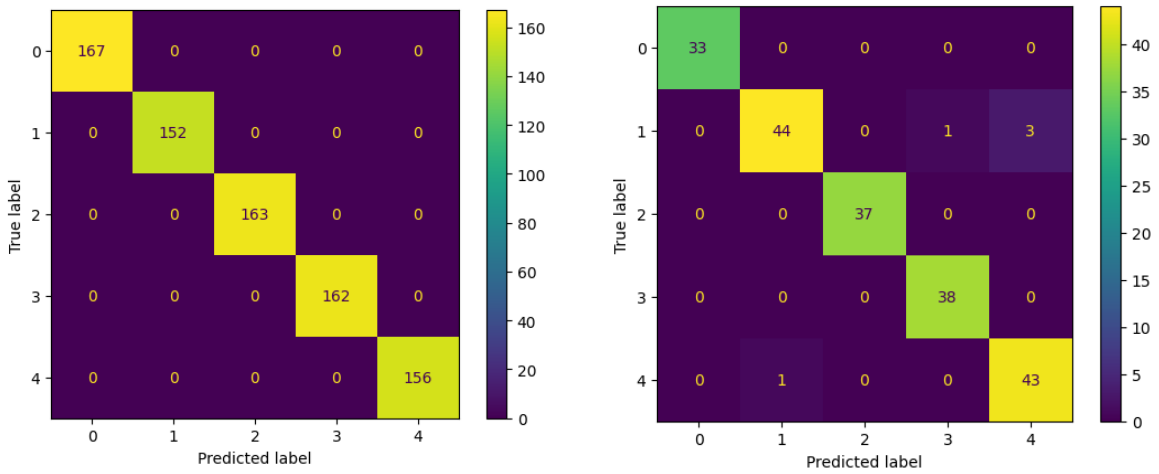


Fig. 29. Binary decision tree for 5-class balanced spiral dataset

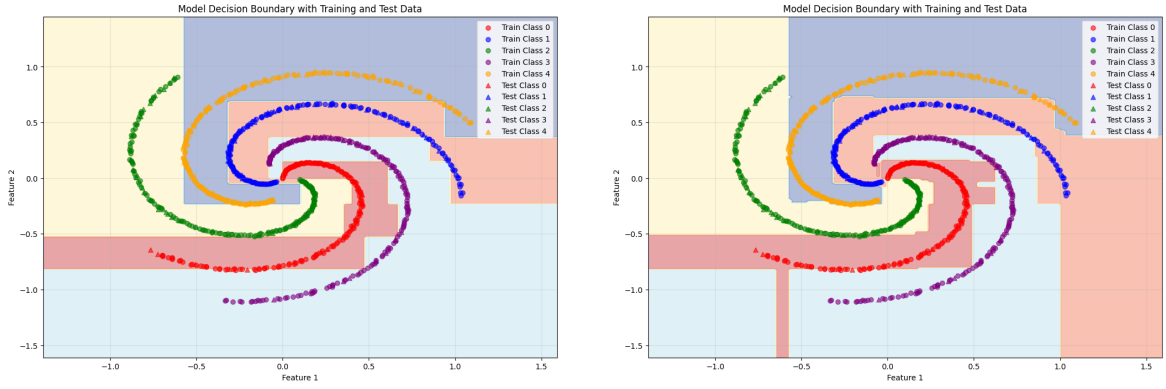


Train set Test set

Fig. 30. Confusion matrix for decision tree trained on 5 class balanced spiral dataset

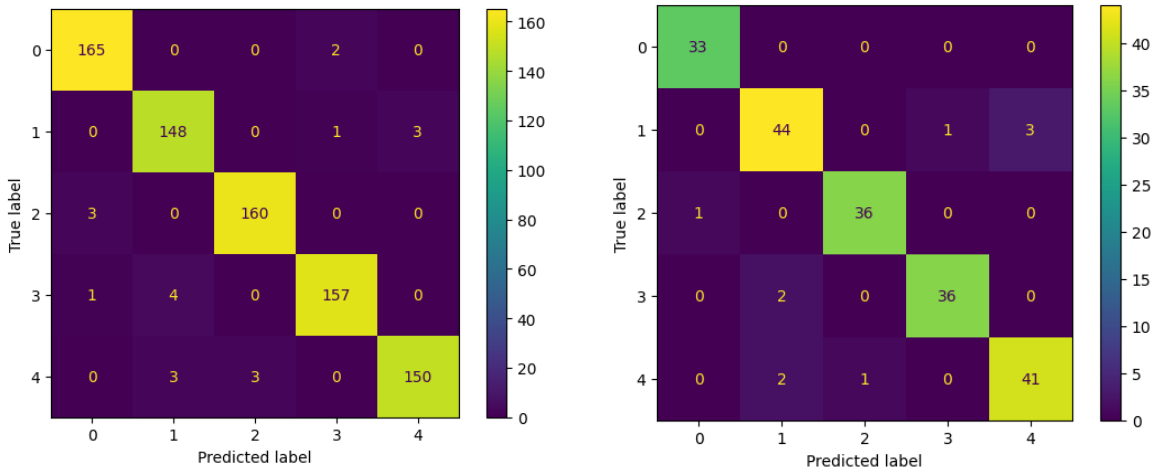
	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	1.00	1.00	1.00	1.00	1.00	1.00
1	1.00	1.00	1.00	0.98	0.92	0.95
2	1.00	1.00	1.00	1.00	1.00	1.00
3	1.00	1.00	1.00	0.97	1.00	0.99
4	1.00	1.00	1.00	0.93	0.98	0.96
Accuracy	1.00 (800 samples)			0.97 (200 samples)		
Macro Avg	1.00	1.00	1.00	0.98	0.98	0.98
Weighted Avg	1.00	1.00	1.00	0.98	0.97	0.97

Table 11. Classification report for decision tree trained on balanced 5-class spiral dataset



Train set Test set

Fig. 31. Decision boundary for decision tree and random forest trained on 5 class balanced spiral dataset



Train set Test set

Fig. 32. Confusion matrix for random forest trained on 5 class balanced spiral dataset

	Train Set			Test Set		
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
0	0.98	0.99	0.98	0.97	1.00	0.99
1	0.95	0.97	0.96	0.92	0.92	0.92
2	0.98	0.98	0.98	0.97	0.97	0.97
3	0.98	0.97	0.98	0.97	0.95	0.96
4	0.98	0.96	0.97	0.93	0.93	0.93
Accuracy	0.97 (800 samples)			0.95 (200 samples)		
Macro Avg	0.97	0.97	0.97	0.95	0.95	0.95
Weighted Avg	0.98	0.97	0.98	0.95	0.95	0.95

Table 12. Classification report for random forest trained on 5-class balanced spiral dataset