UNIVERSITY OF SCIENCE, VNU-HCMC

FACULTY INFORMATION TECHNOLOGY

CSC14003 - Introduction to Artificial Intelligence

Lab 2

Decision Tree

Instructors: Nguyễn Trần Duy Minh, Nguyễn Ngọc Thảo,
Nguyễn Thanh Tình, Nguyễn Hải Đăng

Class: 22CLC06

Student: 22127154 – Nguyễn Gia Huy

Ho Chi Minh City – 2024

Table of contents

1. Introduction

1.1 Description of the Dataset

The UCI Breast Cancer Wisconsin (Diagnostic) dataset is utilized in this project to classify tumors as either malignant (M) or benign (B). The dataset comprises **569 samples** and **30 numerical features** derived from imaging data. Each sample is labeled as either malignant or benign, making it a binary classification problem.

1.2 Objective

The primary objective of this assignment is to build a Decision Tree classifier using the scikit-learn library to predict whether a tumor is malignant or benign based on the given features. The model's performance will be evaluated across different training and testing splits, and the effect of varying the tree depth on accuracy will also be explored.

2. Data Preparation

2.1 Dataset Preparation

The dataset was first fetched using the ucimlrepo library. The features and labels were extracted, and the dataset was then divided into multiple training and testing sets with the following proportions:

- **40/60**: 40% training, 60% testing
- **60/40**: 60% training, 40% testing
- **80/20**: 80% training, 20% testing
- **90/10**: 90% training, 10% testing

Each split was performed in a stratified manner to ensure that both training and testing sets maintained the original distribution of classes.

2.2 Visualization of Class Distribution

To ensure that the data splits were prepared correctly, the class distributions were visualized for the original dataset as well as each of the training and testing sets. This step verifies that the stratification was successful and that each split adequately represents the overall dataset.

Original dataset's class distribution

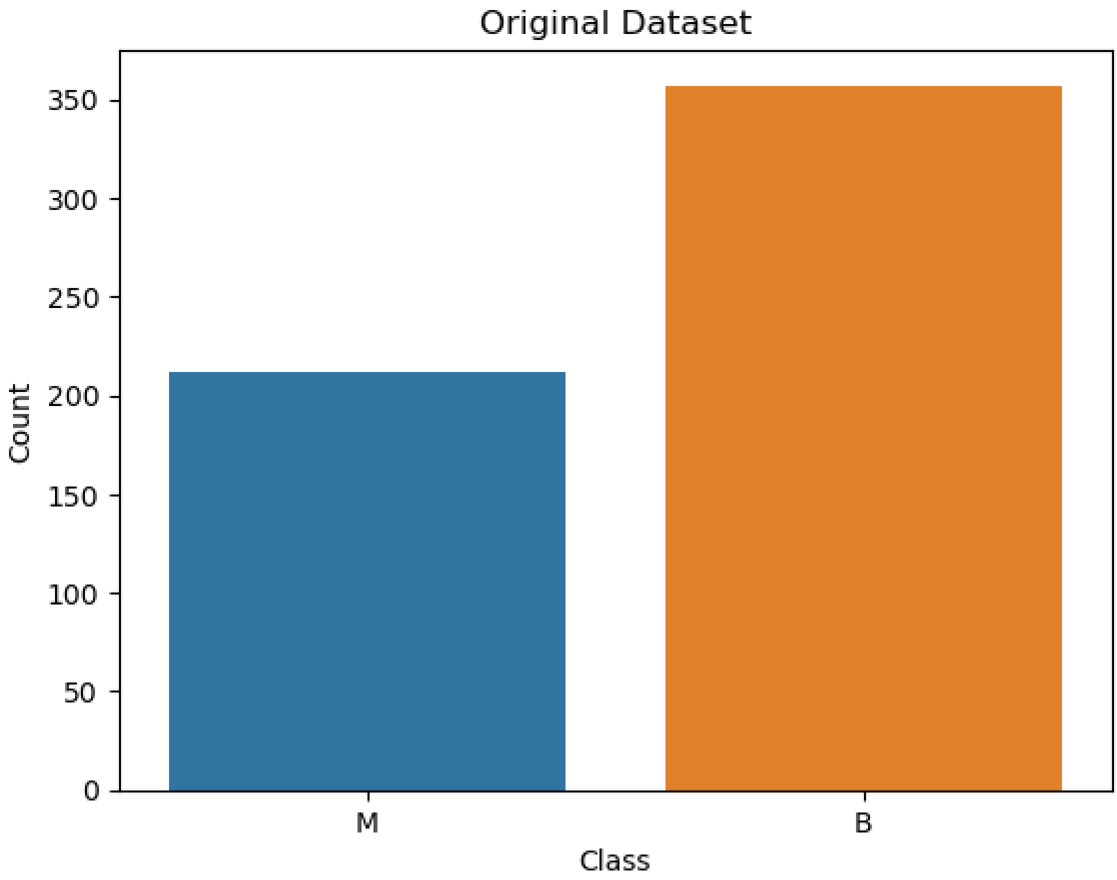


Figure 2.1: Original dataset's class distribution

Class distributions for each train/test split

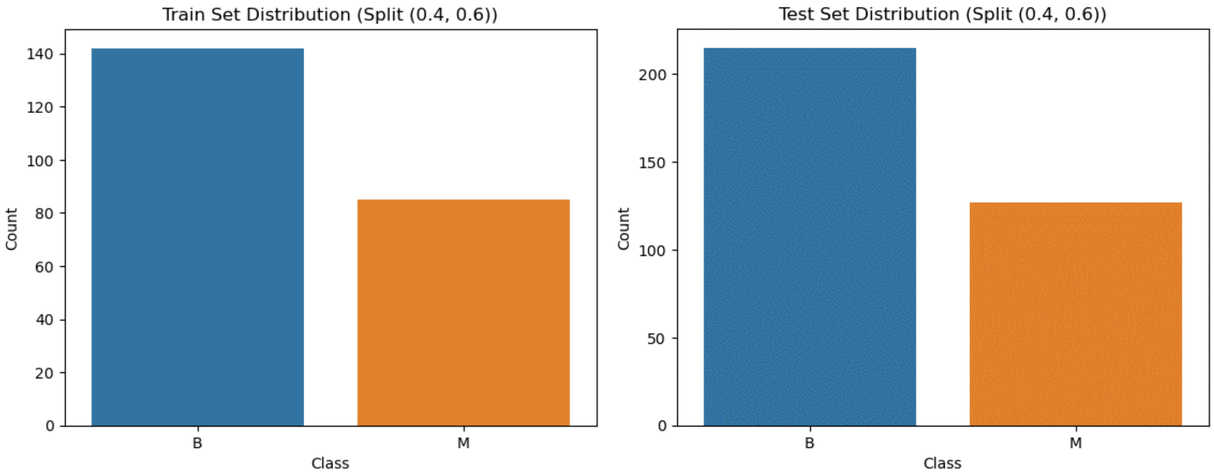


Figure 2.2: Train / Test Set Distribution for Split 40/60

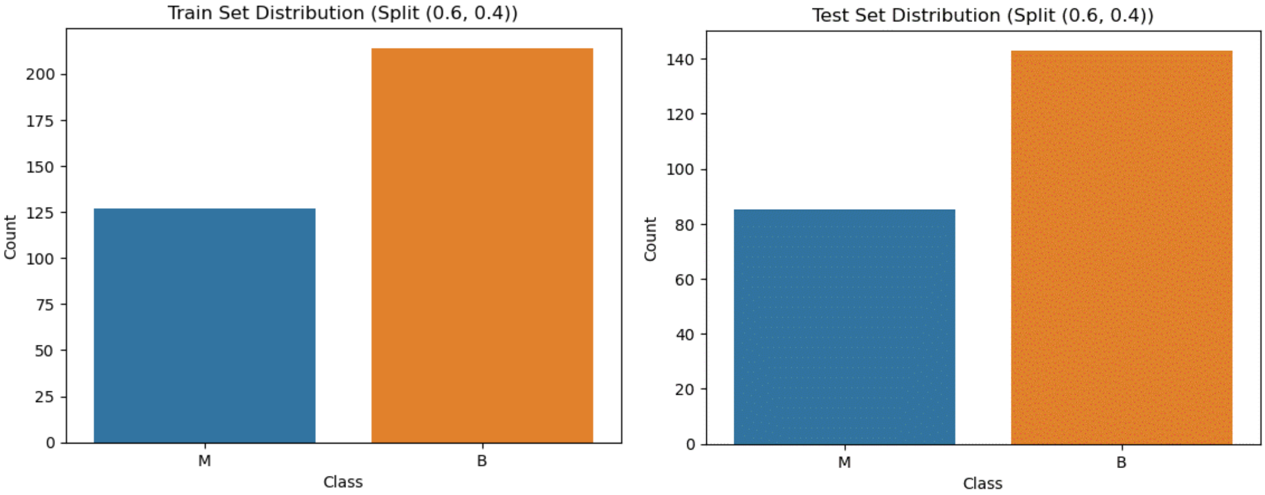


Figure 2.3: Train / Test Set Distribution for Split 60/40

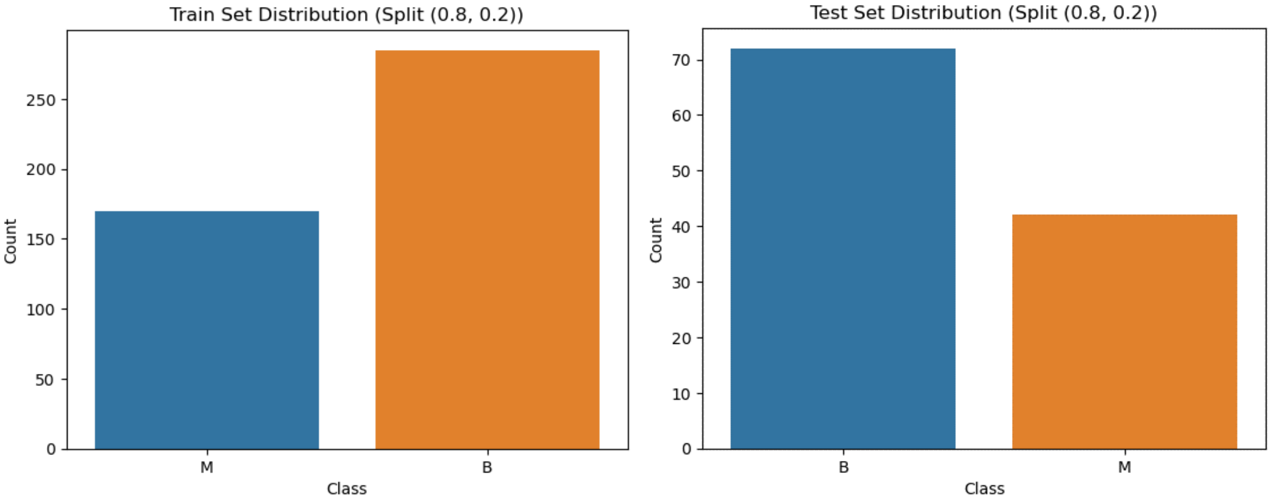


Figure 2.4: Train / Test Set Distribution for Split 80/20

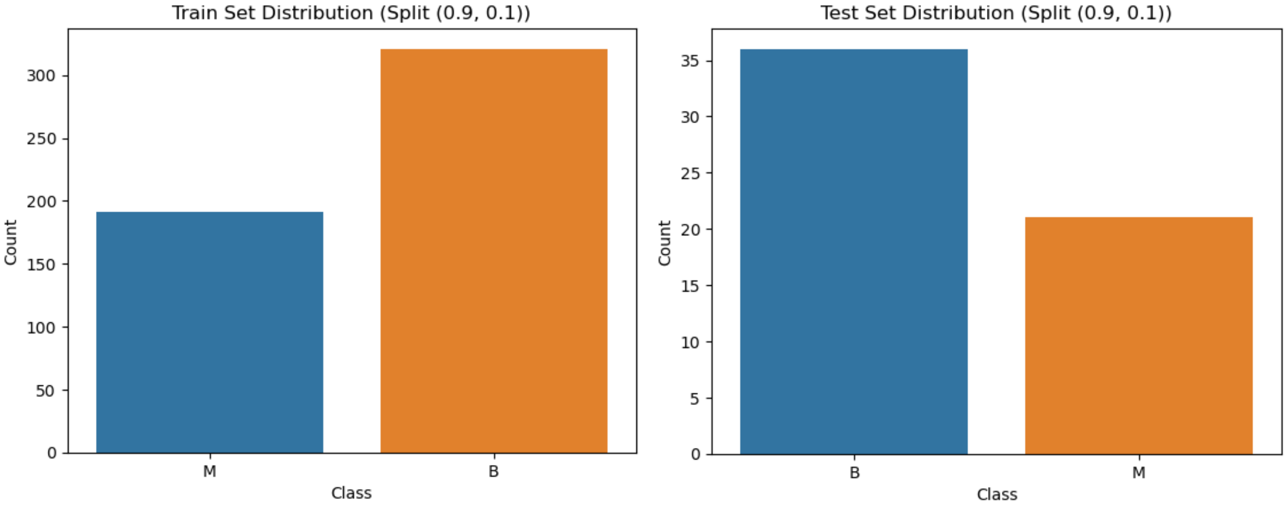


Figure 2.5: Train / Test Set Distribution for Split 90/10

3. Building the Decision Tree Classifiers

3.1 Model Construction

For each of the train/test splits, a Decision Tree classifier was built using the DecisionTreeClassifier from the scikit-learn library. The models were trained on the respective training sets, and the resulting trees were visualized using the graphviz library.

3.2 Visualization of Decision Trees

The following visualizations represent the structure of the decision trees built for each of the train/test splits. These trees show the decision paths taken by the classifier and provide insight into the model's logic at different stages of the decision process.

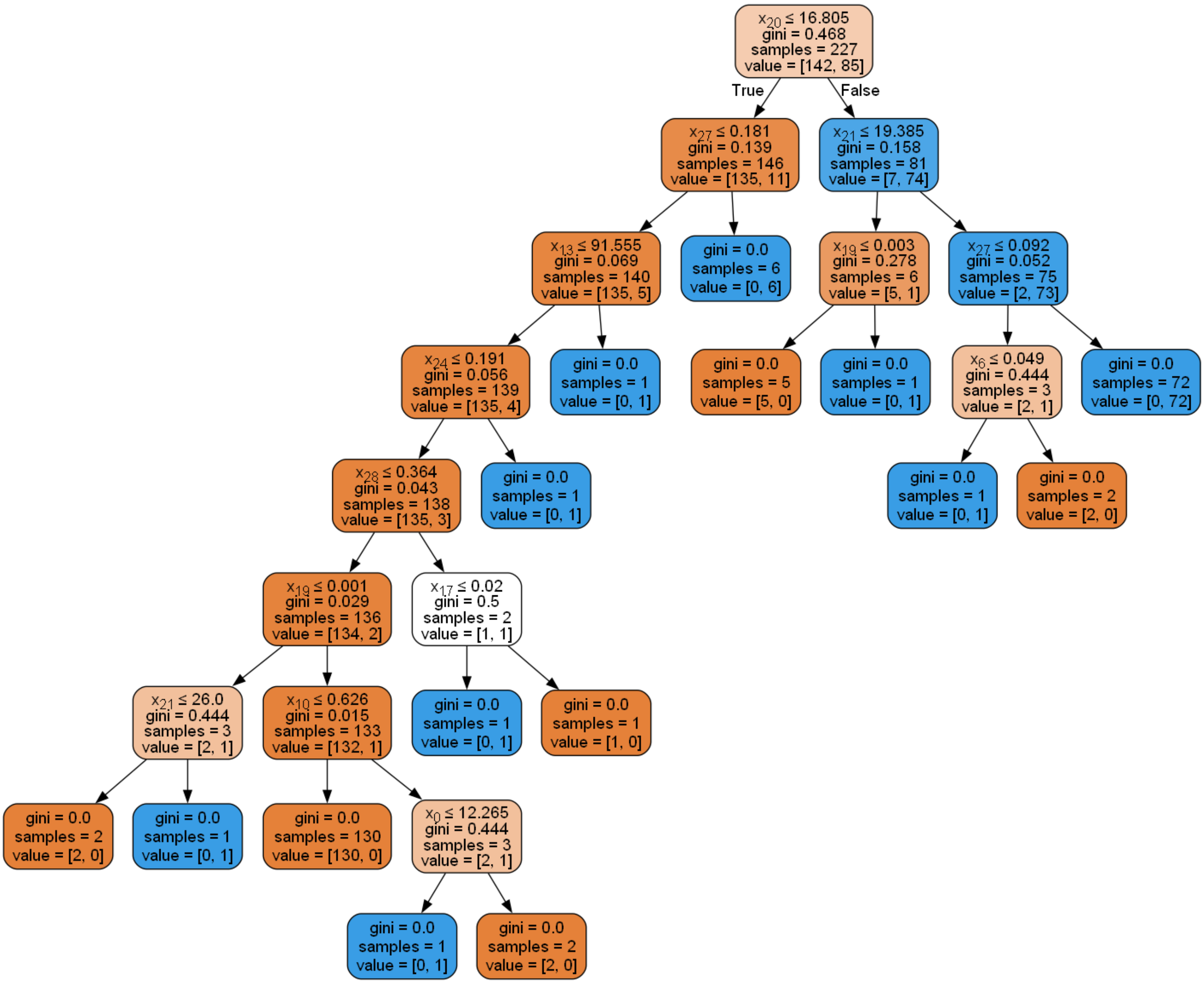

$x_{20} \leq 16.805$ gini = 0.468 samples = 227 value = [142, 85]
True
False
$x_{27} \leq 0.181$ gini = 0.139 samples = 146 value = [135, 11]
$x_{21} \leq 19.385$ gini = 0.158 samples = 81 value = [7, 74]
$x_{13} \leq 91.555$ gini = 0.069 samples = 140 value = [135, 5]
gini = 0.0 samples = 6 value = [0, 6]
$x_{19} \leq 0.003$ gini = 0.278 samples = 6 value = [5, 1]
$x_{27} \leq 0.092$ gini = 0.052 samples = 75 value = [2, 73]
$x_{24} \leq 0.191$ gini = 0.056 samples = 139 value = [135, 4]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 1 value = [0, 1]
$x_{6} \leq 0.049$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 72 value = [0, 72]
$x_{28} \leq 0.364$ gini = 0.043 samples = 138 value = [135, 3]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 2 value = [2, 0]
$x_{19} \leq 0.001$ gini = 0.029 samples = 136 value = [134, 2]
$x_{17} \leq 0.02$ gini = 0.5 samples = 2 value = [1, 1]
$x_{21} \leq 26.0$ gini = 0.444 samples = 3 value = [2, 1]
$x_{10} \leq 0.626$ gini = 0.015 samples = 133 value = [132, 1]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 1 value = [1, 0]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 130 value = [130, 0]
$x_{0} \leq 12.265$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 2 value = [2, 0]

Figure 3.1: Decision tree for train/test split 40/60

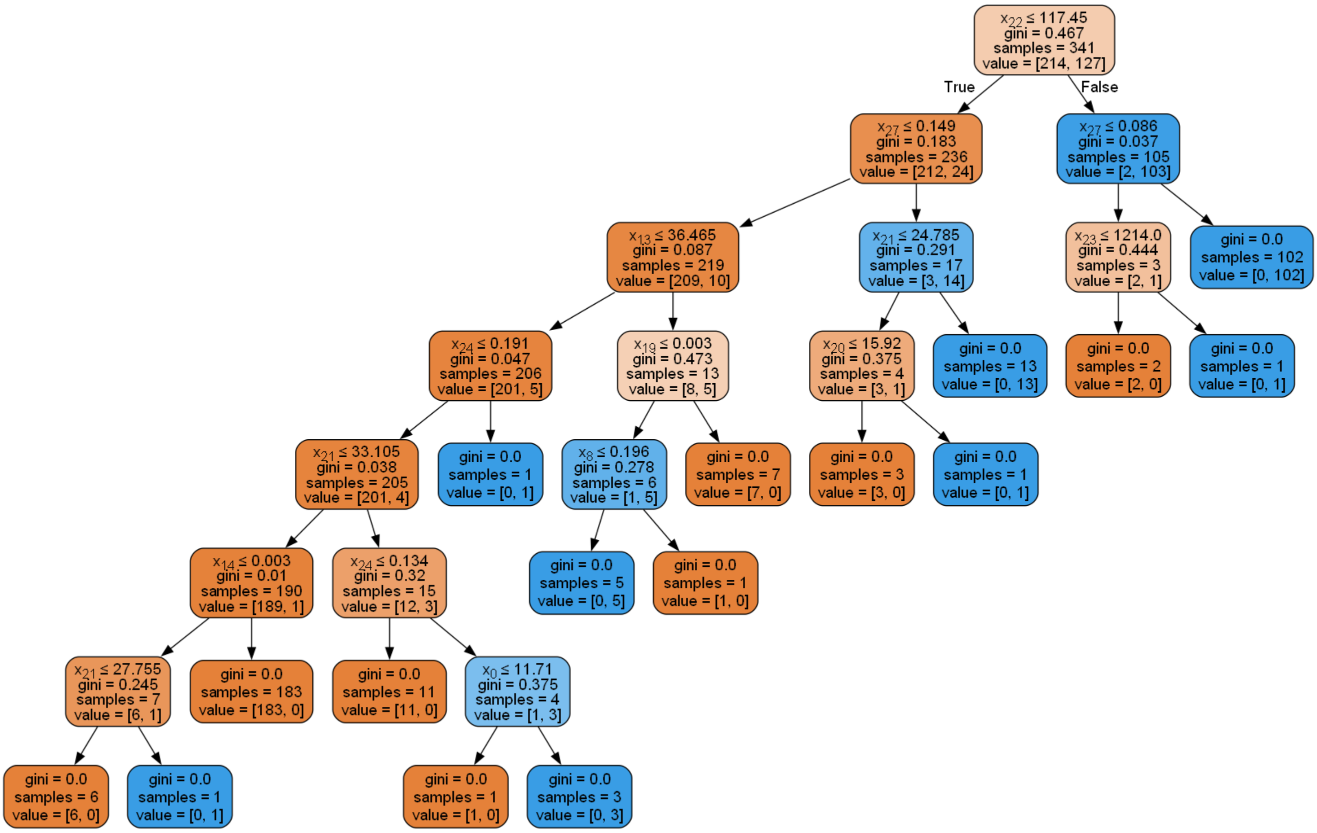

$x_{22} \le 117.45$ gini = 0.467 samples = 341 value = [214, 127]
True
False
$x_{27} \le 0.149$ gini = 0.183 samples = 236 value = [212, 24]
$x_{27} \le 0.086$ gini = 0.037 samples = 105 value = [2, 103]
$x_{13} \le 36.465$ gini = 0.087 samples = 219 value = [209, 10]
$x_{21} \le 24.785$ gini = 0.291 samples = 17 value = [3, 14]
$x_{23} \le 1214.0$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 102 value = [0, 102]
$x_{24} \le 0.191$ gini = 0.047 samples = 206 value = [201, 5]
$x_{19} \le 0.003$ gini = 0.473 samples = 13 value = [8, 5]
$x_{20} \le 15.92$ gini = 0.375 samples = 4 value = [3, 1]
gini = 0.0 samples = 13 value = [0, 13]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 1 value = [0, 1]
$x_{21} \le 33.105$ gini = 0.038 samples = 205 value = [201, 4]
gini = 0.0 samples = 1 value = [0, 1]
$x_{8} \le 0.196$ gini = 0.278 samples = 6 value = [1, 5]
gini = 0.0 samples = 7 value = [7, 0]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 1 value = [0, 1]
$x_{14} \le 0.003$ gini = 0.01 samples = 190 value = [189, 1]
$x_{24} \le 0.134$ gini = 0.32 samples = 15 value = [12, 3]
gini = 0.0 samples = 5 value = [0, 5]
gini = 0.0 samples = 1 value = [1, 0]
$x_{21} \le 27.755$ gini = 0.245 samples = 7 value = [6, 1]
gini = 0.0 samples = 183 value = [183, 0]
gini = 0.0 samples = 11 value = [11, 0]
$x_{0} \le 11.71$ gini = 0.375 samples = 4 value = [1, 3]
gini = 0.0 samples = 6 value = [6, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 1 value = [1, 0]
gini = 0.0 samples = 3 value = [0, 3]

Figure 3.2: Decision tree for train/test split 60/40

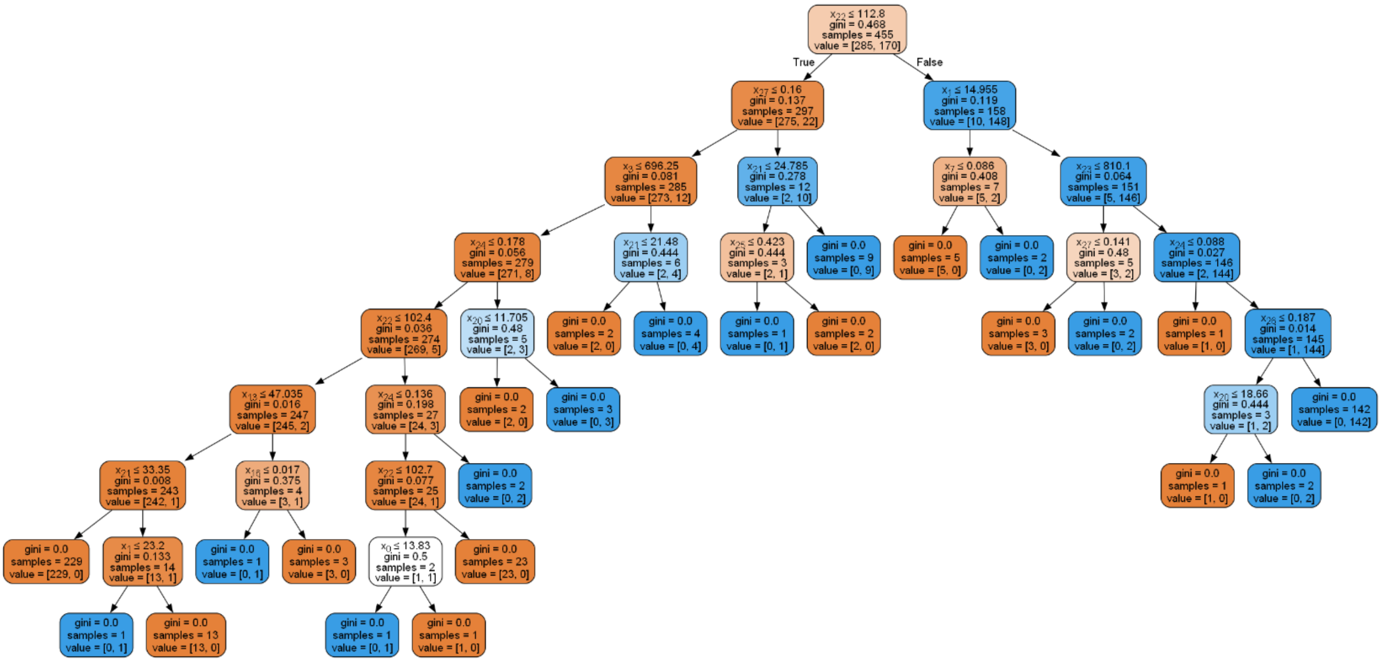

$x_{22} \le 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \le 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_{1} \le 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_{3} \le 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \le 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_{7} \le 0.086$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \le 810.1$ gini = 0.064 samples = 151 value = [5, 146]
$x_{24} \le 0.178$ gini = 0.056 samples = 279 value = [271, 8]
$x_{21} \le 21.48$ gini = 0.444 samples = 6 value = [2, 4]
$x_{25} \le 0.423$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 2 value = [0, 2]
$x_{27} \le 0.141$ gini = 0.48 samples = 5 value = [3, 2]
$x_{24} \le 0.088$ gini = 0.027 samples = 146 value = [2, 144]
$x_{22} \le 102.4$ gini = 0.036 samples = 274 value = [269, 5]
$x_{20} \le 11.705$ gini = 0.48 samples = 5 value = [2, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 4 value = [0, 4]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
$x_{26} \le 0.187$ gini = 0.014 samples = 145 value = [1, 144]
$x_{13} \le 47.035$ gini = 0.016 samples = 247 value = [245, 2]
$x_{24} \le 0.136$ gini = 0.198 samples = 27 value = [24, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [0, 3]
$x_{20} \le 18.66$ gini = 0.444 samples = 3 value = [1, 2]
gini = 0.0 samples = 142 value = [0, 142]
$x_{21} \le 33.35$ gini = 0.008 samples = 243 value = [242, 1]
$x_{16} \le 0.017$ gini = 0.375 samples = 4 value = [3, 1]
$x_{22} \le 102.7$ gini = 0.077 samples = 25 value = [24, 1]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 229 value = [229, 0]
$x_{1} \le 23.2$ gini = 0.133 samples = 14 value = [13, 1]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 3 value = [3, 0]
$x_{0} \le 13.83$ gini = 0.5 samples = 2 value = [1, 1]
gini = 0.0 samples = 23 value = [23, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 13 value = [13, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 1 value = [1, 0]

Figure 3.3: Decision tree for train/test split 80/20

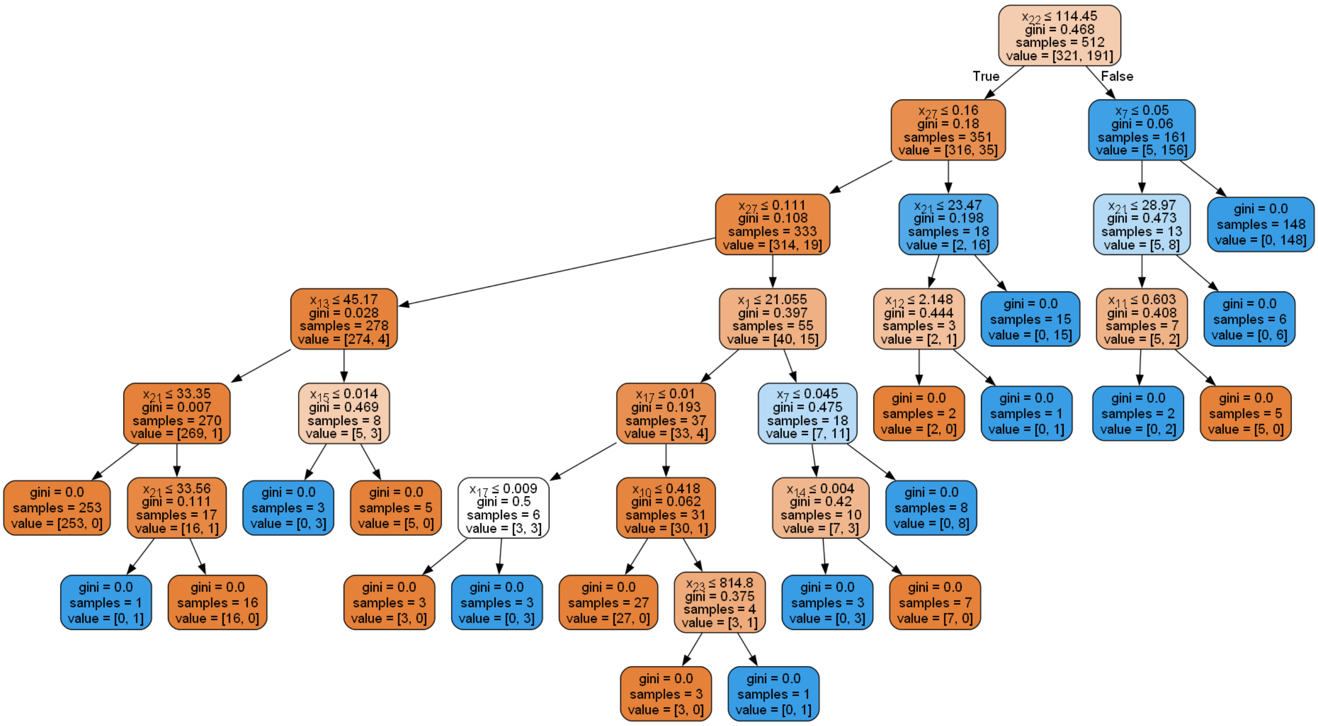

$x_{22} \le 114.45$ gini = 0.468 samples = 512 value = [321, 191]
True
False
$x_{27} \le 0.16$ gini = 0.18 samples = 351 value = [316, 35]
$x_{7} \le 0.05$ gini = 0.06 samples = 161 value = [5, 156]
$x_{27} \le 0.111$ gini = 0.108 samples = 333 value = [314, 19]
$x_{21} \le 23.47$ gini = 0.198 samples = 18 value = [2, 16]
$x_{21} \le 28.97$ gini = 0.473 samples = 13 value = [5, 8]
gini = 0.0 samples = 148 value = [0, 148]
$x_{13} \le 45.17$ gini = 0.028 samples = 278 value = [274, 4]
$x_{1} \le 21.055$ gini = 0.397 samples = 55 value = [40, 15]
$x_{12} \le 2.148$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 15 value = [0, 15]
$x_{11} \le 0.603$ gini = 0.408 samples = 7 value = [5, 2]
gini = 0.0 samples = 6 value = [0, 6]
$x_{21} \le 33.35$ gini = 0.007 samples = 270 value = [269, 1]
$x_{15} \le 0.014$ gini = 0.469 samples = 8 value = [5, 3]
$x_{17} \le 0.01$ gini = 0.193 samples = 37 value = [33, 4]
$x_{7} \le 0.045$ gini = 0.475 samples = 18 value = [7, 11]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 253 value = [253, 0]
$x_{21} \le 33.56$ gini = 0.111 samples = 17 value = [16, 1]
gini = 0.0 samples = 3 value = [0, 3]
gini = 0.0 samples = 5 value = [5, 0]
$x_{17} \le 0.009$ gini = 0.5 samples = 6 value = [3, 3]
$x_{10} \le 0.418$ gini = 0.062 samples = 31 value = [30, 1]
$x_{14} \le 0.004$ gini = 0.42 samples = 10 value = [7, 3]
gini = 0.0 samples = 8 value = [0, 8]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 16 value = [16, 0]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 3 value = [0, 3]
gini = 0.0 samples = 27 value = [27, 0]
$x_{23} \le 814.8$ gini = 0.375 samples = 4 value = [3, 1]
gini = 0.0 samples = 3 value = [0, 3]
gini = 0.0 samples = 7 value = [7, 0]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 1 value = [0, 1]

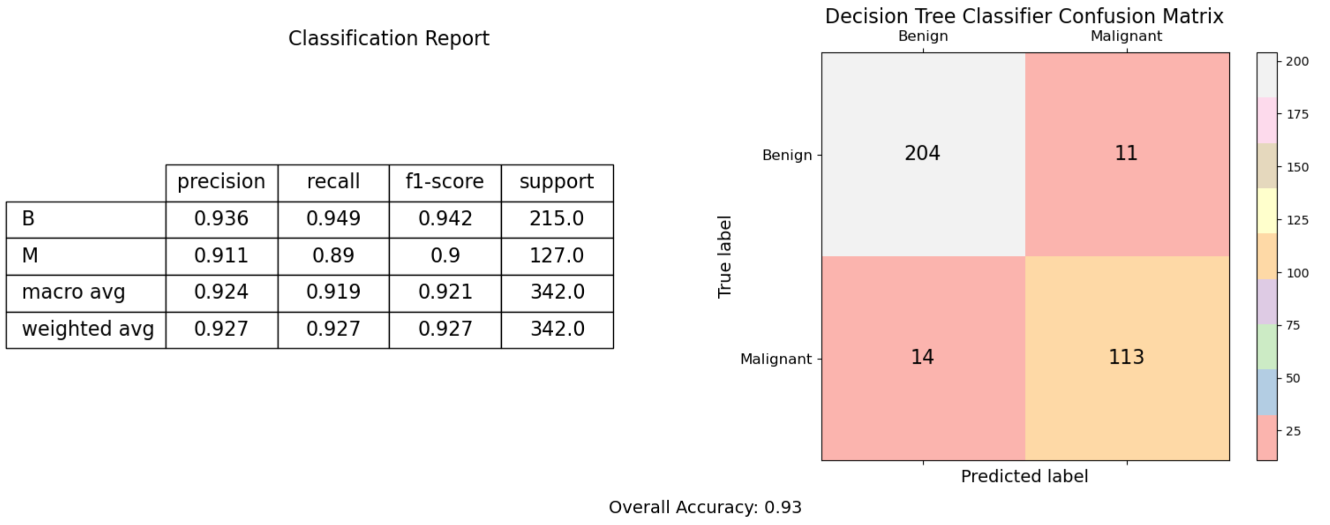
Figure 3.4: Decision tree for train/test split 90/10

4. Evaluation of the Decision Tree Classifiers

4.1 Performance Evaluation

For each decision tree classifier, the model's performance was evaluated on the corresponding test set. The evaluation was conducted using the classification_report and confusion_matrix functions from scikit-learn.

Classification report and confusion matrix



Classification Report

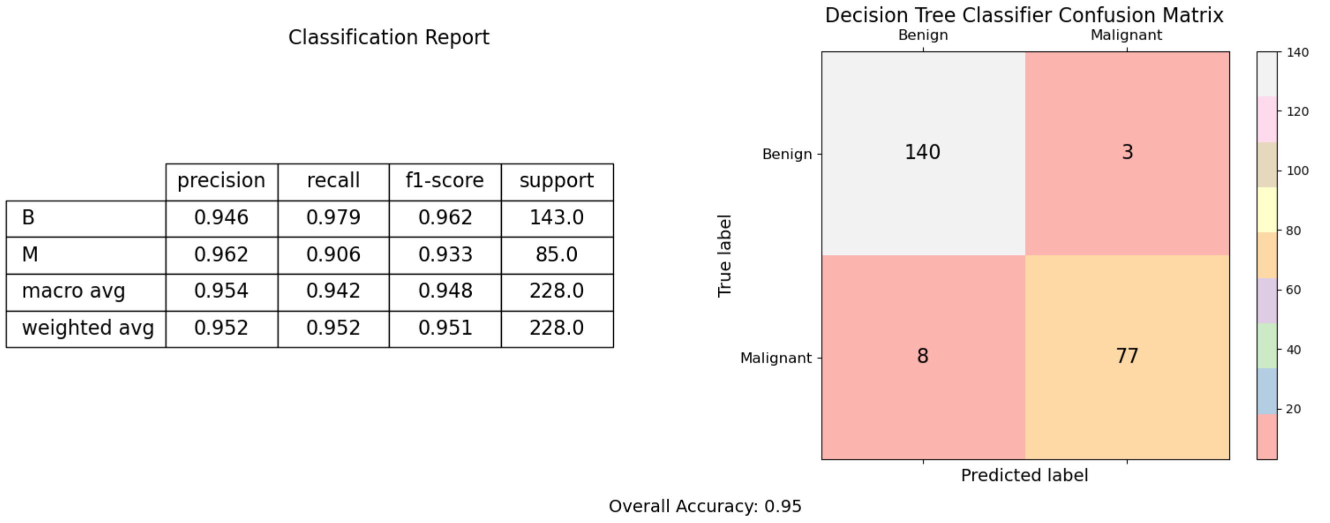
	precision	recall	f1-score	support
B	0.936	0.949	0.942	215.0
M	0.911	0.89	0.9	127.0
macro avg	0.924	0.919	0.921	342.0
weighted avg	0.927	0.927	0.927	342.0

Decision Tree Classifier Confusion Matrix

	Benign	Malignant
Benign	204	11
Malignant	14	113

Overall Accuracy: 0.93

Figure 4.1: Classification Report and confusion matrix for train/test split 40/60



Classification Report

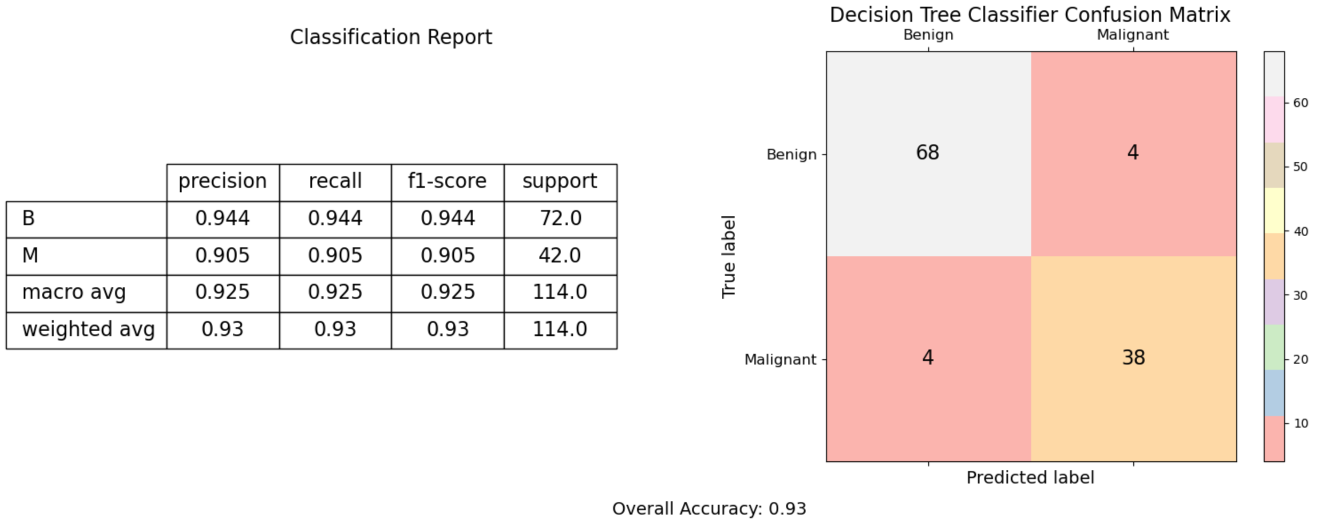
	precision	recall	f1-score	support
B	0.946	0.979	0.962	143.0
M	0.962	0.906	0.933	85.0
macro avg	0.954	0.942	0.948	228.0
weighted avg	0.952	0.952	0.951	228.0

Decision Tree Classifier Confusion Matrix

	Benign	Malignant
Benign	140	3
Malignant	8	77

Overall Accuracy: 0.95

Figure 4.2: Classification Report and confusion matrix for train/test split 60/40

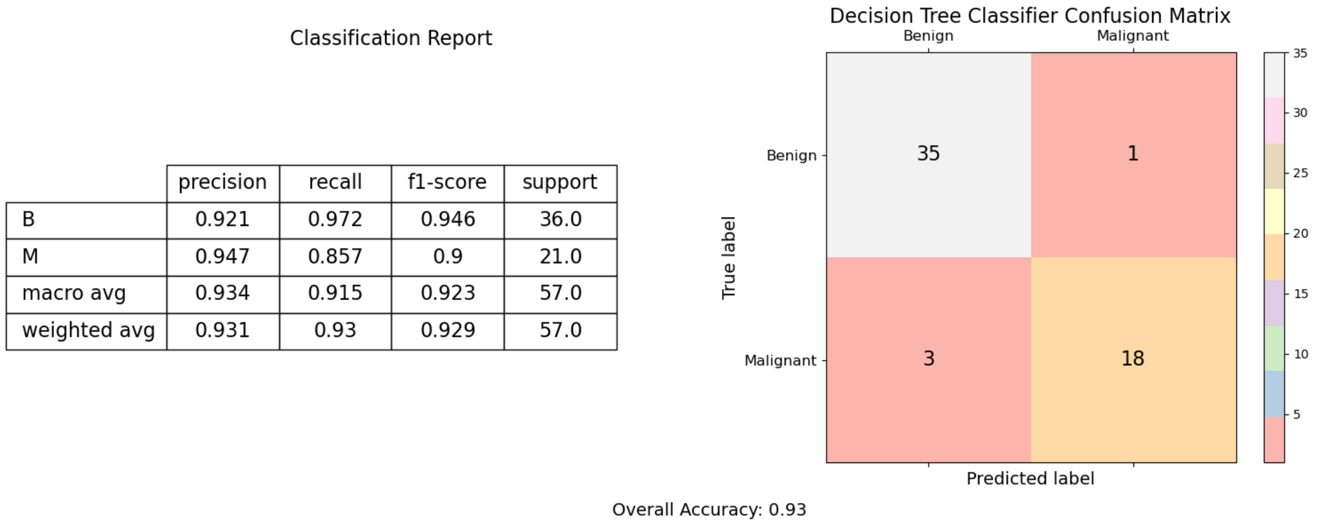


Classification Report

	precision	recall	f1-score	support
B	0.944	0.944	0.944	72.0
M	0.905	0.905	0.905	42.0
macro avg	0.925	0.925	0.925	114.0
weighted avg	0.93	0.93	0.93	114.0

Overall Accuracy: 0.93

Figure 4.3: Classification Report and confusion matrix for train/test split 80/20



Classification Report

	precision	recall	f1-score	support
B	0.921	0.972	0.946	36.0
M	0.947	0.857	0.9	21.0
macro avg	0.934	0.915	0.923	57.0
weighted avg	0.931	0.93	0.929	57.0

Overall Accuracy: 0.93

Figure 4.4: Classification Report and confusion matrix for train/test split 90/10

Comments on the Results

The decision tree classifiers were evaluated using four different train/test splits: 40/60, 60/40, 80/20, and 90/10. The performance of each model was assessed through classification reports and confusion matrices, as illustrated in Figures 4.1 to 4.4.

Train / Test Split	Accuracy	Precision and Recall	Confusion Matrix
40/60	The model achieved an overall accuracy of 0.93, which is a solid performance given that the model was trained on only 40% of the data.	The precision for the benign class (B) was 0.936, while for the malignant class (M), it was slightly lower at 0.911. This indicates that the model was slightly more likely to incorrectly classify a benign case as malignant. The recall for benign cases was high at 0.949, showing that most benign cases were correctly identified. However, the recall for malignant cases was lower at 0.89, indicating that some malignant cases were misclassified as benign.	The model correctly classified 204 benign and 113 malignant cases. However, it misclassified 11 benign cases as malignant and 14 malignant cases as benign. These misclassifications could potentially have significant consequences in a clinical setting.
60/40	The accuracy increased to 0.952 with a larger training set, demonstrating improved performance with more data.	Both precision and recall were higher for both classes compared to the 40/60 split. The model achieved a precision of 0.946 for benign cases and 0.962 for malignant cases. The recall was exceptionally high for benign cases at 0.979, and still strong for malignant cases at 0.906.	The confusion matrix shows that the model made fewer errors in this configuration. It correctly identified 140 benign and 77 malignant cases, while only misclassifying 3 benign cases and 8 malignant cases. This split provided the best balance between precision and recall across both classes.
80/20	The overall accuracy slightly decreased to 0.93, similar to the 40/60 split, suggesting that adding more training data does not always improve the model's accuracy.	The precision for benign cases was 0.944 and for malignant cases was 0.905. The recall values were identical at 0.944 for benign and 0.905 for malignant cases, indicating consistent performance in identifying both classes.	The model correctly classified 68 benign and 38 malignant cases, with 4 misclassifications in each class. Although the performance is still good, the smaller test set likely contributed to the slight decrease in accuracy and the balanced precision/recall.
90/10	The model maintained an accuracy of 0.93, consistent with other splits, but with much less test data.	The precision for benign cases decreased slightly to 0.921, while the precision for malignant cases increased to 0.947. The recall for benign cases was very high at 0.972, indicating that the model identified almost all benign cases correctly. However, the recall for malignant cases was lower at 0.857, showing that some malignant cases were missed.	With a smaller test set, the model correctly identified 35 benign and 18 malignant cases, with only 1 benign case misclassified as malignant and 3 malignant cases misclassified as benign. The high recall for benign cases suggests the model is more cautious, potentially leading to more false positives (benign classified as malignant).

Summary

Overall, the decision tree models performed consistently across different train/test splits, with accuracies hovering around 0.93 to 0.95. The 60/40 split provided the best performance, with the highest overall accuracy and balanced precision and recall for both classes. The model tends to have a higher recall for benign cases, suggesting it is more effective at correctly identifying benign tumors but at the risk of missing some malignant cases. The performance stability across different data splits indicates that the decision tree is a robust model for this dataset, although the slightly lower performance on the 90/10 split suggests that very small test sets might introduce some instability in the results.

The confusion matrices across all splits highlight the importance of considering both false positives and false negatives, especially in a sensitive domain like cancer diagnosis, where misclassifications can have significant implications.

5. The Depth and Accuracy of a Decision Tree

5.1 Experiment with Tree Depth

The effect of varying the tree depth on the model's accuracy was explored by adjusting the max_depth parameter of the decision tree classifier. The depths tested were: None, 2, 3, 4, 5, 6, and 7. For each depth, the tree was trained using the 80/20 train/test split, and the accuracy was recorded.

5.2 Results and Visualization

The results of this experiment are summarized in the table below, which shows the accuracy for each tested depth. Additionally, visualizations of the decision trees at different depths are included to illustrate how the tree structure changes as the depth increases.

Table of accuracy vs. max depth

Max Depth	None	2	3	4	5	6	7
Accuracy	0.93	0.921	0.904	0.912	0.921	0.921	0.939

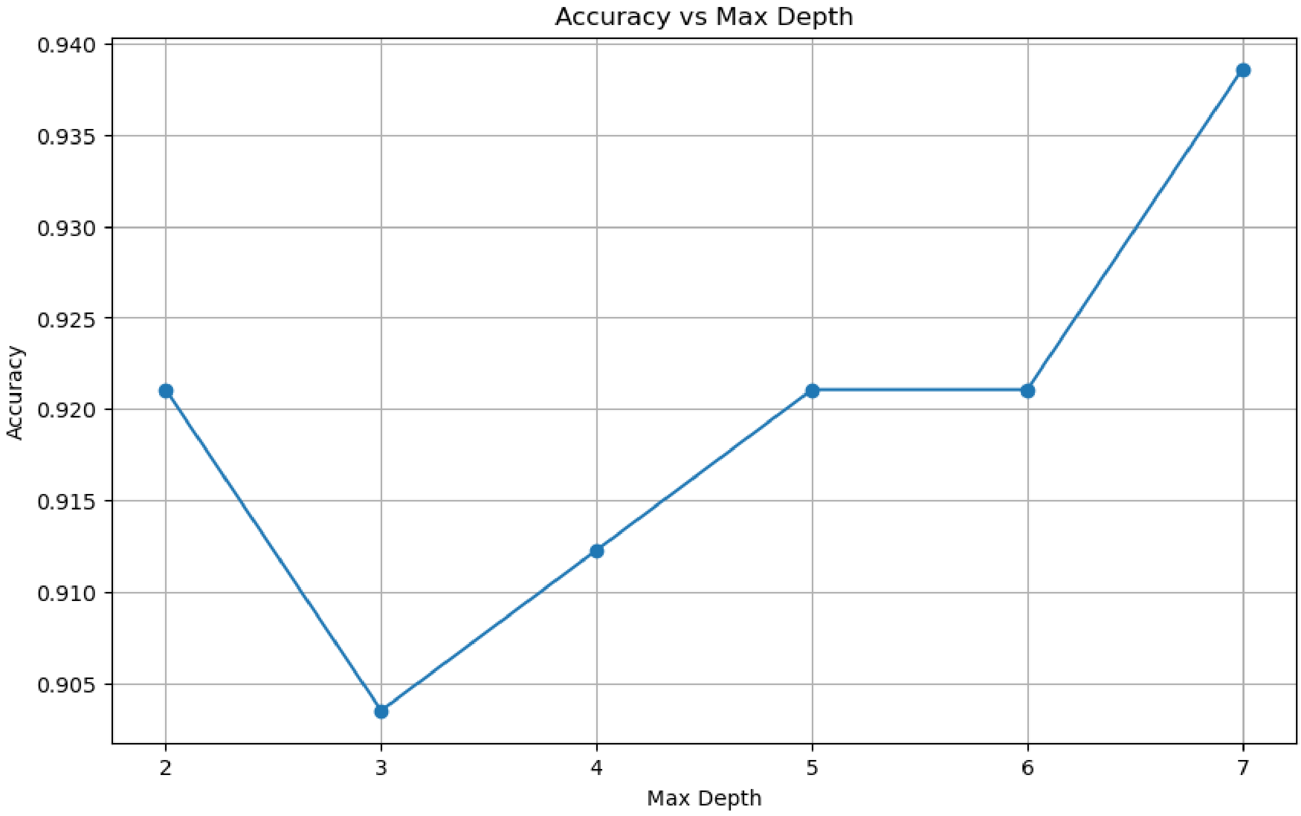


Figure 5.1: Max depth vs Accuracy graph

Visualizations of decision trees at different depths

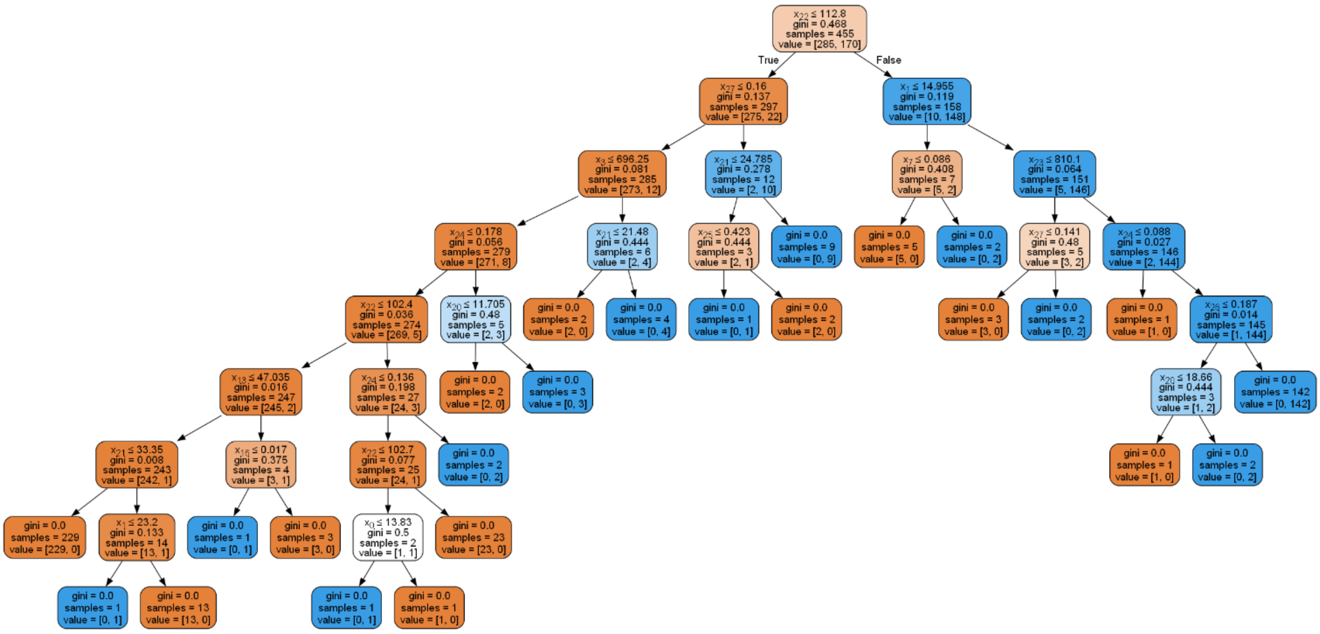

$x_{22} \le 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \le 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_{1} \le 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_{3} \le 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \le 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_{7} \le 0.086$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \le 810.1$ gini = 0.064 samples = 151 value = [5, 146]
$x_{24} \le 0.178$ gini = 0.056 samples = 279 value = [271, 8]
$x_{21} \le 21.48$ gini = 0.444 samples = 6 value = [2, 4]
$x_{25} \le 0.423$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 2 value = [0, 2]
$x_{27} \le 0.141$ gini = 0.48 samples = 5 value = [3, 2]
$x_{24} \le 0.088$ gini = 0.027 samples = 146 value = [2, 144]
$x_{22} \le 102.4$ gini = 0.036 samples = 274 value = [269, 5]
$x_{20} \le 11.705$ gini = 0.48 samples = 5 value = [2, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 4 value = [0, 4]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
$x_{28} \le 0.187$ gini = 0.014 samples = 145 value = [1, 144]
$x_{13} \le 47.035$ gini = 0.016 samples = 247 value = [245, 2]
$x_{24} \le 0.136$ gini = 0.198 samples = 27 value = [24, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [0, 3]
$x_{20} \le 18.66$ gini = 0.444 samples = 3 value = [1, 2]
gini = 0.0 samples = 142 value = [0, 142]
$x_{21} \le 33.35$ gini = 0.008 samples = 243 value = [242, 1]
$x_{16} \le 0.017$ gini = 0.375 samples = 4 value = [3, 1]
$x_{22} \le 102.7$ gini = 0.077 samples = 25 value = [24, 1]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 229 value = [229, 0]
$x_{1} \le 23.2$ gini = 0.133 samples = 14 value = [13, 1]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 3 value = [3, 0]
$x_{0} \le 13.83$ gini = 0.5 samples = 2 value = [1, 1]
gini = 0.0 samples = 23 value = [23, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 13 value = [13, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 1 value = [1, 0]

Figure 5.2: Full decision tree

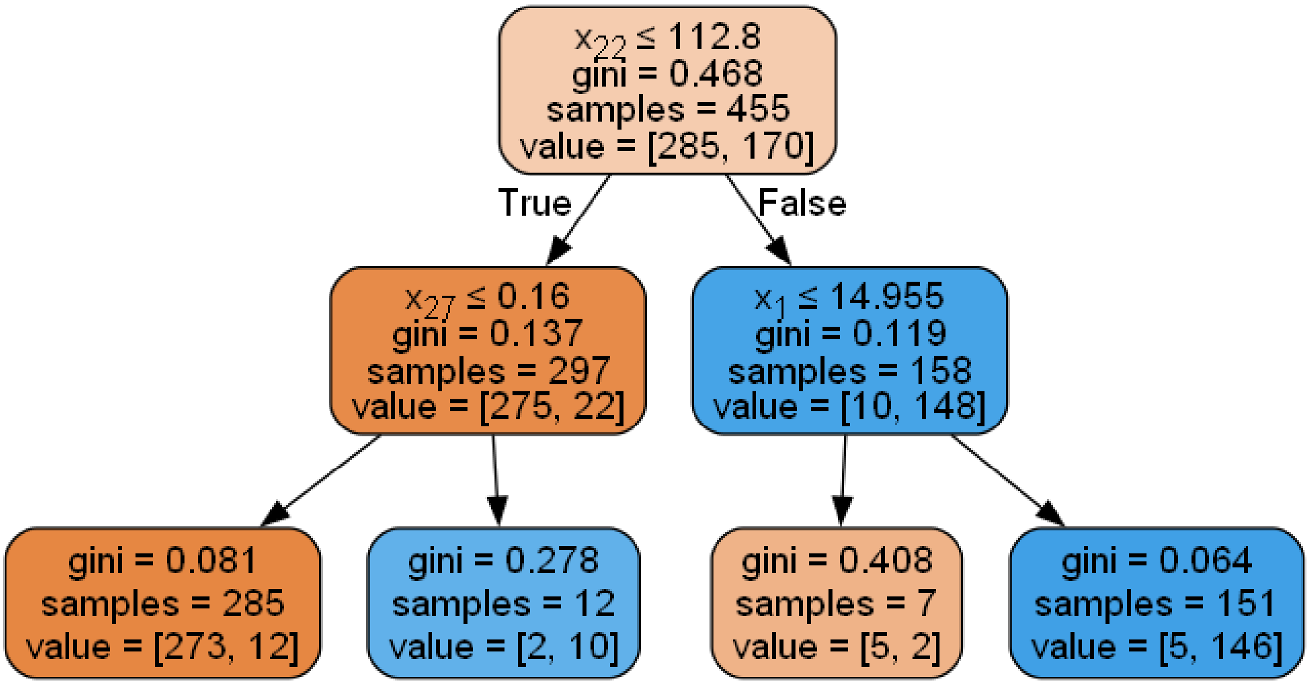

$x_{22} \le 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \le 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_{1} \le 14.955$ gini = 0.119 samples = 158 value = [10, 148]
gini = 0.081 samples = 285 value = [273, 12]
gini = 0.278 samples = 12 value = [2, 10]
gini = 0.408 samples = 7 value = [5, 2]
gini = 0.064 samples = 151 value = [5, 146]

Figure 5.3: Decision tree for depth 2

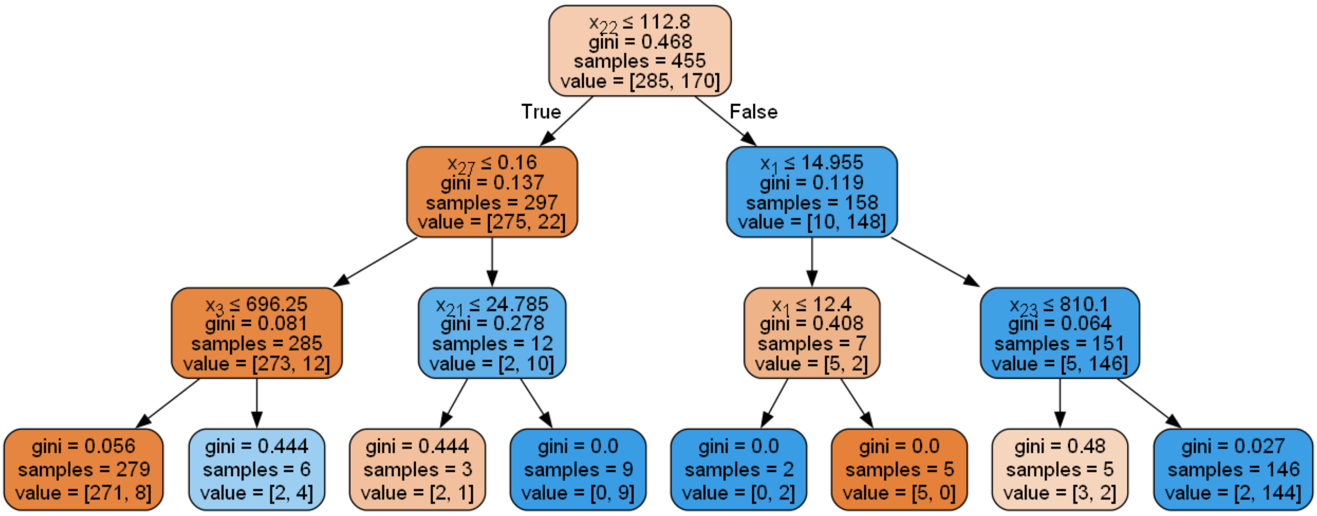

$x_{22} \leq 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \leq 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_{1} \leq 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_{3} \leq 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \leq 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_{1} \leq 12.4$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \leq 810.1$ gini = 0.064 samples = 151 value = [5, 146]
gini = 0.056 samples = 279 value = [271, 8]
gini = 0.444 samples = 6 value = [2, 4]
gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.48 samples = 5 value = [3, 2]
gini = 0.027 samples = 146 value = [2, 144]

Figure 5.4: Decision tree for depth 3

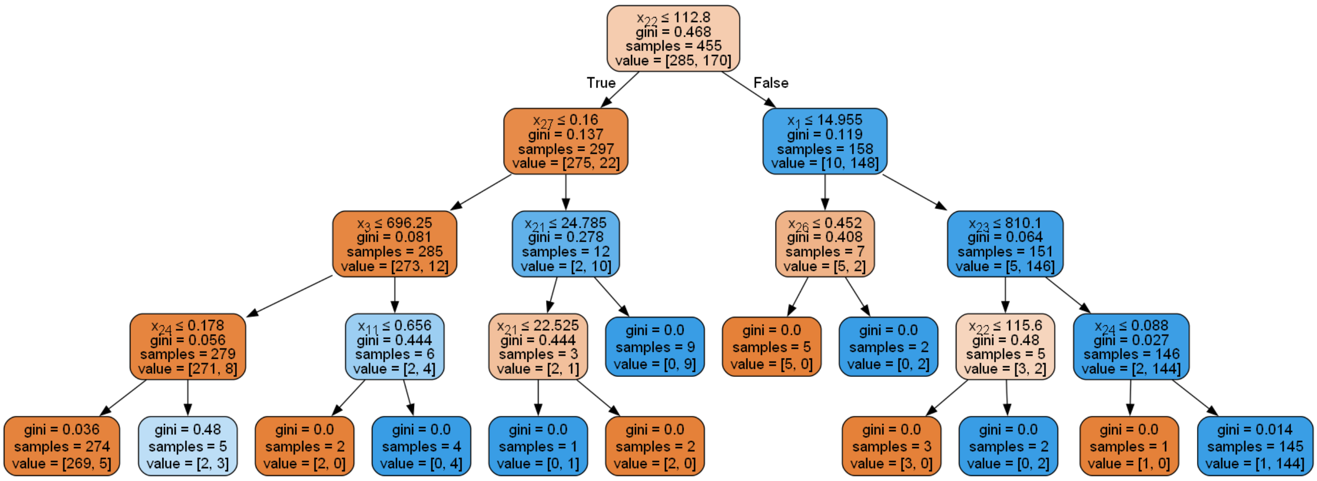

$x_{22} \leq 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \leq 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_{1} \leq 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_{3} \leq 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \leq 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_{26} \leq 0.452$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \leq 810.1$ gini = 0.064 samples = 151 value = [5, 146]
$x_{24} \leq 0.178$ gini = 0.056 samples = 279 value = [271, 8]
$x_{11} \leq 0.656$ gini = 0.444 samples = 6 value = [2, 4]
$x_{21} \leq 22.525$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 2 value = [0, 2]
$x_{22} \leq 115.6$ gini = 0.48 samples = 5 value = [3, 2]
$x_{24} \leq 0.088$ gini = 0.027 samples = 146 value = [2, 144]
gini = 0.036 samples = 274 value = [269, 5]
gini = 0.48 samples = 5 value = [2, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 4 value = [0, 4]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
gini = 0.014 samples = 145 value = [1, 144]

Figure 5.5: Decision tree for depth 4

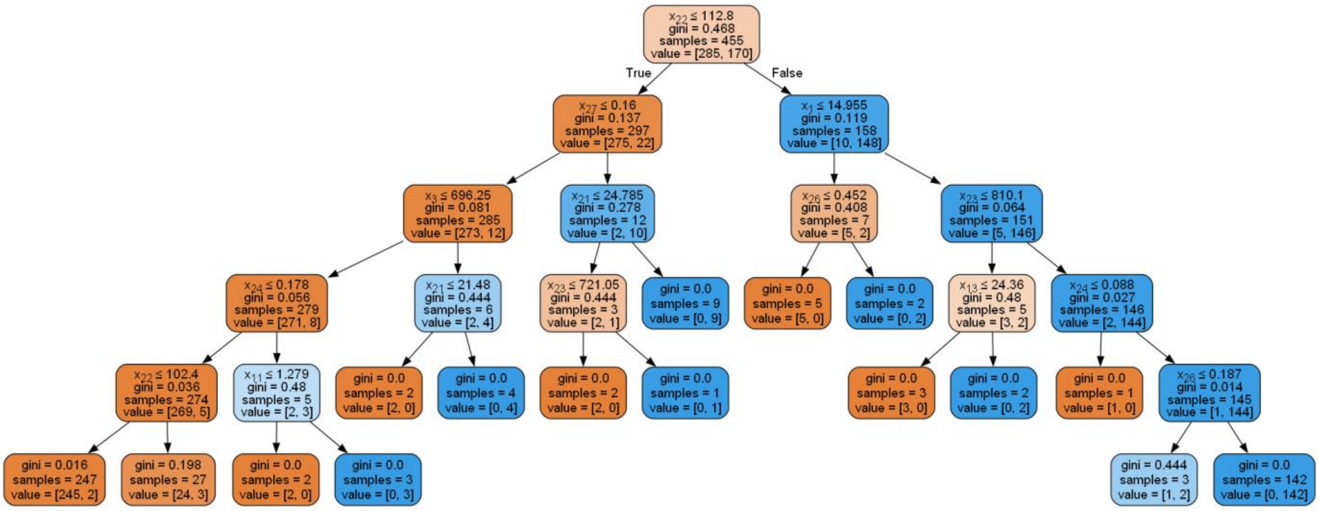

$x_{22} \leq 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \leq 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_{1} \leq 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_{3} \leq 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \leq 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_{26} \leq 0.452$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \leq 810.1$ gini = 0.064 samples = 151 value = [5, 146]
$x_{24} \leq 0.178$ gini = 0.056 samples = 279 value = [271, 8]
$x_{21} \leq 21.48$ gini = 0.444 samples = 6 value = [2, 4]
$x_{23} \leq 721.05$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 2 value = [0, 2]
$x_{13} \leq 24.36$ gini = 0.48 samples = 5 value = [3, 2]
$x_{24} \leq 0.088$ gini = 0.027 samples = 146 value = [2, 144]
$x_{22} \leq 102.4$ gini = 0.036 samples = 274 value = [269, 5]
$x_{11} \leq 1.279$ gini = 0.48 samples = 5 value = [2, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 4 value = [0, 4]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
$x_{26} \leq 0.187$ gini = 0.014 samples = 145 value = [1, 144]
gini = 0.016 samples = 247 value = [245, 2]
gini = 0.198 samples = 27 value = [24, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [0, 3]
gini = 0.444 samples = 3 value = [1, 2]
gini = 0.0 samples = 142 value = [0, 142]

Figure 5.6: Decision tree for depth 5

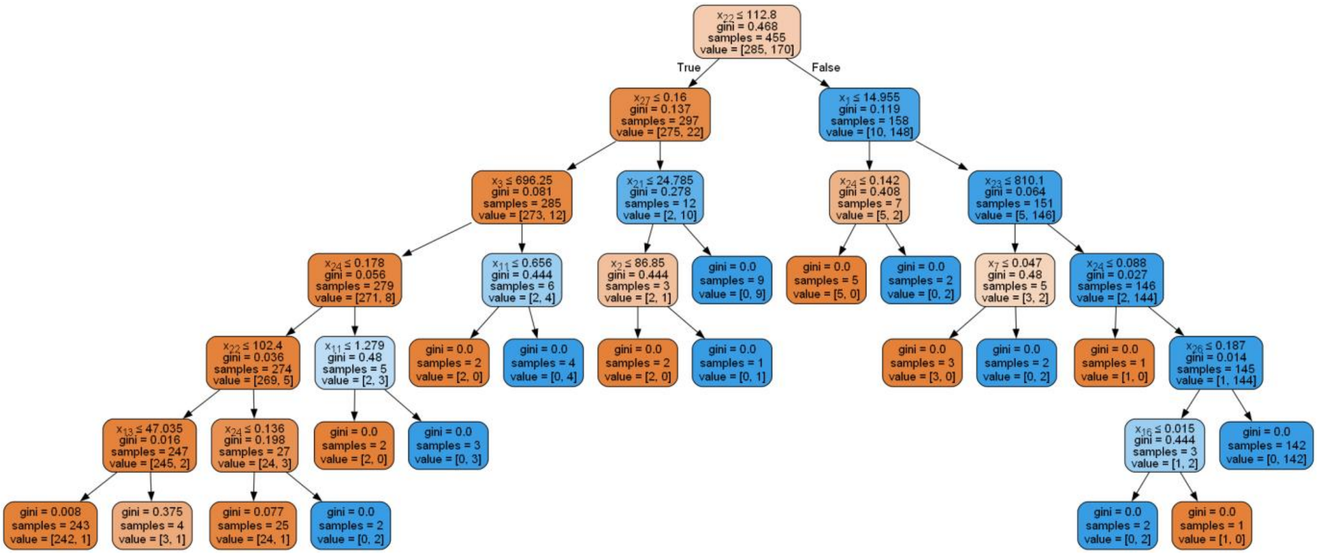

$x_{22} \leq 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \leq 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_1 \leq 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_3 \leq 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \leq 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_{24} \leq 0.142$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \leq 810.1$ gini = 0.064 samples = 151 value = [5, 146]
$x_{24} \leq 0.178$ gini = 0.056 samples = 279 value = [271, 8]
$x_{11} \leq 0.656$ gini = 0.444 samples = 6 value = [2, 4]
$x_2 \leq 86.85$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 5 value = [5, 0]
gini = 0.0 samples = 2 value = [0, 2]
$x_7 \leq 0.047$ gini = 0.48 samples = 5 value = [3, 2]
$x_{24} \leq 0.088$ gini = 0.027 samples = 146 value = [2, 144]
$x_{22} \leq 102.4$ gini = 0.036 samples = 274 value = [269, 5]
$x_{11} \leq 1.279$ gini = 0.48 samples = 5 value = [2, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 4 value = [0, 4]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
$x_{26} \leq 0.187$ gini = 0.014 samples = 145 value = [1, 144]
$x_{13} \leq 47.035$ gini = 0.016 samples = 247 value = [245, 2]
$x_{24} \leq 0.136$ gini = 0.198 samples = 27 value = [24, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [0, 3]
$x_{16} \leq 0.015$ gini = 0.444 samples = 3 value = [1, 2]
gini = 0.0 samples = 142 value = [0, 142]
gini = 0.008 samples = 243 value = [242, 1]
gini = 0.375 samples = 4 value = [3, 1]
gini = 0.077 samples = 25 value = [24, 1]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]

Figure 5.7: Decision tree for depth 6

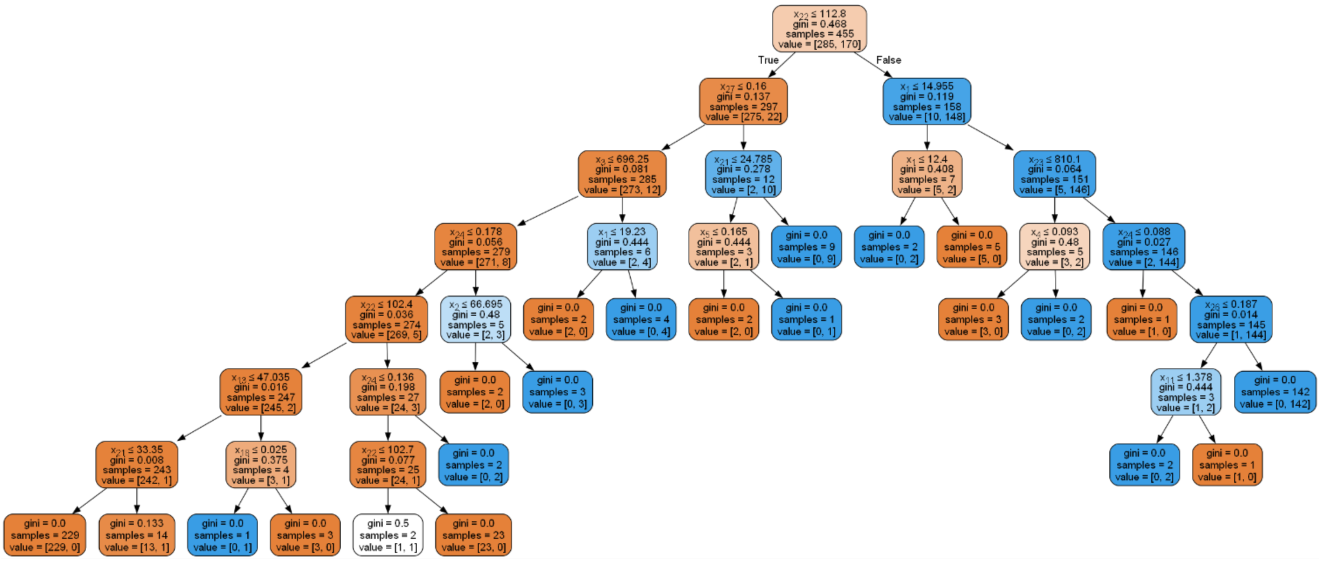

$x_{22} \leq 112.8$ gini = 0.468 samples = 455 value = [285, 170]
True
False
$x_{27} \leq 0.16$ gini = 0.137 samples = 297 value = [275, 22]
$x_1 \leq 14.955$ gini = 0.119 samples = 158 value = [10, 148]
$x_3 \leq 696.25$ gini = 0.081 samples = 285 value = [273, 12]
$x_{21} \leq 24.785$ gini = 0.278 samples = 12 value = [2, 10]
$x_1 \leq 12.4$ gini = 0.408 samples = 7 value = [5, 2]
$x_{23} \leq 810.1$ gini = 0.064 samples = 151 value = [5, 146]
$x_{24} \leq 0.178$ gini = 0.056 samples = 279 value = [271, 8]
$x_1 \leq 19.23$ gini = 0.444 samples = 6 value = [2, 4]
$x_5 \leq 0.165$ gini = 0.444 samples = 3 value = [2, 1]
gini = 0.0 samples = 9 value = [0, 9]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 5 value = [5, 0]
$x_4 \leq 0.093$ gini = 0.48 samples = 5 value = [3, 2]
$x_{24} \leq 0.088$ gini = 0.027 samples = 146 value = [2, 144]
$x_{22} \leq 102.4$ gini = 0.036 samples = 274 value = [269, 5]
$x_2 \leq 66.695$ gini = 0.48 samples = 5 value = [2, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 4 value = [0, 4]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
$x_{26} \leq 0.187$ gini = 0.014 samples = 145 value = [1, 144]
$x_{13} \leq 47.035$ gini = 0.016 samples = 247 value = [245, 2]
$x_{24} \leq 0.136$ gini = 0.198 samples = 27 value = [24, 3]
gini = 0.0 samples = 2 value = [2, 0]
gini = 0.0 samples = 3 value = [0, 3]
$x_{11} \leq 1.378$ gini = 0.444 samples = 3 value = [1, 2]
gini = 0.0 samples = 142 value = [0, 142]
$x_{21} \leq 33.35$ gini = 0.008 samples = 243 value = [242, 1]
$x_{18} \leq 0.025$ gini = 0.375 samples = 4 value = [3, 1]
$x_{22} \leq 102.7$ gini = 0.077 samples = 25 value = [24, 1]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 2 value = [0, 2]
gini = 0.0 samples = 1 value = [1, 0]
gini = 0.0 samples = 229 value = [229, 0]
gini = 0.133 samples = 14 value = [13, 1]
gini = 0.0 samples = 1 value = [0, 1]
gini = 0.0 samples = 3 value = [3, 0]
gini = 0.5 samples = 2 value = [1, 1]
gini = 0.0 samples = 23 value = [23, 0]

Figure 5.8: Decision tree for depth 7

Comments on the Results

The analysis of the Decision Tree classifiers with varying depths provides insights into how the complexity of the model influences its performance, particularly in terms of accuracy. The following comments reflect the observations derived from the table, graph, and visualizations of the decision trees at different depths.

1. Accuracy vs. Max Depth

- **General Trend:** The accuracy of the Decision Tree classifier fluctuated as the max depth increased, ranging from 0.904 at depth 3 to 0.939 at depth 7. Notably, the accuracy remained relatively stable across several depths (2, 5, and 6), where it was around 0.921, before increasing significantly at depth 7.
- **Shallow Trees (Depth 2 to 4):**
    - At a depth of 2, the tree achieved an accuracy of 0.921, demonstrating that even a shallow tree can capture the essential patterns in the data.
    - However, at depth 3, the accuracy dropped to 0.904, suggesting that the model might have been underfitting, failing to capture more complex relationships in the data.
    - At depth 4, the accuracy improved slightly to 0.912, indicating that increasing depth helped to capture more nuanced patterns without overfitting.
- **Deeper Trees (Depth 5 to 7):**
    - At depths 5 and 6, the accuracy plateaued at 0.921, similar to the performance at depth 2. This suggests that the additional splits did not contribute significantly to improving model performance.
    - Interestingly, at depth 7, the accuracy increased to 0.939, the highest observed. This indicates that the model was able to leverage the additional complexity to improve its classification performance, potentially by capturing more detailed interactions between features.

2. Visualizations of Decision Trees at Different Depths

- **Full Decision Tree (No Depth Limit):**
    - The full decision tree, without any depth limit, is highly complex with multiple branches. While this model likely captures intricate patterns in the training data, it risks overfitting, which is why it’s essential to evaluate the effect of depth on generalization to unseen data.
- **Decision Tree at Depth 2:**

  - The tree at depth 2 is notably simple, with only a few splits. It captures the most prominent patterns but lacks the complexity to accurately classify more challenging cases. This is reflected in the accuracy, which, while decent, is not the highest.

- **Decision Trees at Depth 4 and 5:**
  - These trees show increased complexity compared to depth 2, with more branches and decision nodes. The additional splits at these depths allow the model to capture more detailed relationships between features, which explains the slight increase in accuracy from depth 3 to 4.
  - However, the accuracy plateau suggests that while these trees are more complex, they do not necessarily translate into better performance, likely because the added complexity does not correspond to better generalization to the test data.
- **Decision Tree at Depth 7:**
  - The tree at depth 7 shows a significant increase in both complexity and accuracy. This depth likely strikes a balance between capturing complex patterns and maintaining generalization to the test set, as evidenced by the highest accuracy achieved.
  - The increased depth allows the model to make more precise distinctions between classes, reducing misclassifications and leading to improved performance.

3. Analysis of the Trade-off Between Depth and Accuracy

- **Overfitting vs. Underfitting:**
  - The results illustrate the classic trade-off between model complexity and accuracy. Shallow trees (e.g., depth 2 and 3) risk underfitting, as they do not capture enough detail from the data. On the other hand, extremely deep trees (beyond depth 7) could potentially overfit, memorizing training data rather than learning generalizable patterns.
  - Depth 7 appears to be the sweet spot where the tree is complex enough to capture necessary details without overfitting, as indicated by the peak in accuracy.
- **Implications for Model Selection:**
  - When choosing the optimal tree depth, it’s essential to consider not only the accuracy but also the interpretability and computational efficiency. While a depth of 7 offers the best

accuracy, it comes at the cost of increased complexity, which may not always be desirable depending on the application.

- For practical purposes, a tree depth that balances simplicity and performance, such as depth 4 or 5, might be preferable, especially when considering interpretability in a clinical context.

Summary

In conclusion, varying the depth of the Decision Tree reveals how model complexity impacts accuracy. While deeper trees generally offer better performance, there is a point beyond which additional complexity may not provide significant benefits. The tree at depth 7 provided the highest accuracy, suggesting that for this specific dataset and split, a more complex model was beneficial. However, the decision on the optimal depth should be guided by a balance between accuracy, interpretability, and the risk of overfitting.

6. Conclusion

6.1 Summary of Findings

In this report, Decision Tree classifiers were built to classify tumors as malignant or benign based on imaging features from the UCI Breast Cancer Wisconsin dataset. The models were evaluated across various train/test splits, and the impact of tree depth on accuracy was analyzed.

6.2 Final Remarks

The decision trees performed well across different splits, with accuracies varying depending on the depth of the tree. However, deeper trees sometimes led to overfitting, demonstrating the importance of choosing an optimal tree depth.

7. References

[1]. Wolberg,William, Mangasarian,Olvi, Street,Nick, and Street,W.. (1995). Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository. https://doi.org/10.24432/C5DW2B.

[2]. Uci-Ml-Repo. (n.d.). *GitHub - uci-ml-repo/ucimlrepo: Python package for dataset imports from UCI ML Repository*. GitHub. https://github.com/uci-ml-repo/ucimlrepo

[3]. *3.4. Metrics and scoring: quantifying the quality of predictions*. (n.d.). Scikit-learn. https://scikit-learn.org/stable/modules/model_evaluation.html

[4]. *DecisionTreeClassifier*. (n.d.). Scikit-learn. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

[5]. *seaborn.countplot — seaborn 0.13.2 documentation*. (n.d.). https://seaborn.pydata.org/generated/seaborn.countplot.html

[6]. *User Guide — graphviz 0.20.3 documentation*. (n.d.). https://graphviz.readthedocs.io/en/stable/manual.html

[7]. *Choosing Colormaps in Matplotlib — Matplotlib 3.9.2 documentation*. (n.d.). https://matplotlib.org/stable/users/explain/colors/colormaps.html