**CS3481 Fundamentals of Data Science**

**Assignment 1**

**Decision Tree Analysis Report**

**Student Name: LUO Peiyuan**

**SID: 56642728**

Table of Contents

[Question A 3](#_Toc129549587)

[Part 1 Decision Tree Building 3](#_Toc129549588)

[Question A 5](#_Toc129549589)

[Part 2 Built Decision Trees 5](#_Toc129549590)

[**Decision Tree 1** 5](#_Toc129549591)

[**Decision Tree 2** 5](#_Toc129549592)

[Decision Tree 3 6](#_Toc129549593)

[**Decision Tree 4** 6](#_Toc129549594)

[Decision Tree 5 7](#_Toc129549595)

[**Decision Tree 6** 7](#_Toc129549596)

[**Decision Tree 7** 8](#_Toc129549597)

[Decision Tree 8 8](#_Toc129549598)

[**Decision Tree 9** 9](#_Toc129549599)

[**Decision Tree 10** 9](#_Toc129549600)

[**Decision Tree 11** 10](#_Toc129549601)

[**Decision Tree 12** 10](#_Toc129549602)

[**Decision Tree 13** 11](#_Toc129549603)

[**Decision Tree 14** 11](#_Toc129549604)

[**Decision Tree 15** 12](#_Toc129549605)

[**Decision Tree 16** 13](#_Toc129549606)

[**Decision Tree 17** 14](#_Toc129549607)

[**Decision Tree 18** 14](#_Toc129549608)

[Question B 15](#_Toc129549609)

[Question B Analysis 1(Impact of Criterion and Test Set Size on Max Scores) 15](#_Toc129549610)

[Question B Analysis 2(Impact of Criterion and Test Size on Max Depth) 16](#_Toc129549611)

[Question B Analysis 3(Impact of Node Count on Max Scores) 17](#_Toc129549612)

[Question C Analysis 18](#_Toc129549613)

[Question D Analysis 20](#_Toc129549614)

[Example 1: Decision Tree 4 20](#_Toc129549615)

[Example 2: Decision Tree 1 21](#_Toc129549616)

[APPENDIX 22](#_Toc129549617)

# **Question A**

## **Part 1 Decision Tree Building**

**Step 1:**

1. Load the data
2. Generate the info of these data and load into corresponding field
3. Construct the train split function with input of test size, shuffle, and random state.
4. Construct the predict function with max depth, criterion, training data, training labels, testing data, and testing labels.
5. Construct the save tree file function to save the decision tree result into PNG files.

**Text

Description automatically generated**

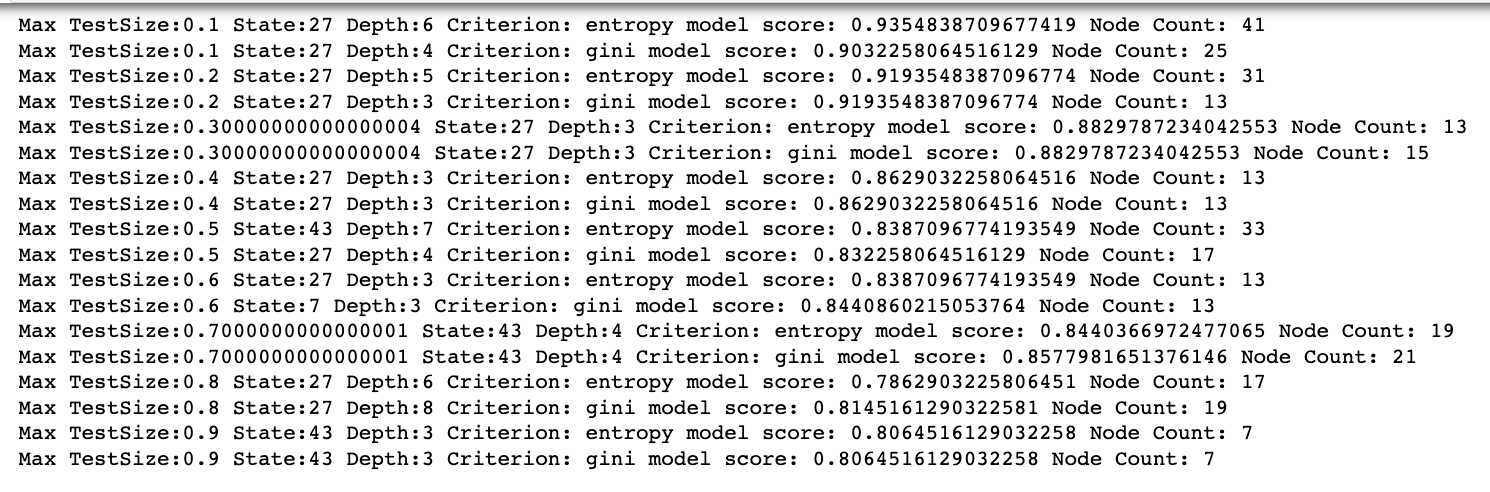
**Step 2:**

1. Choose three random states (7, 27, 43) to be the random state list.
2. Choose ‘entropy’ and ‘gini’ to be the criterion list (impurity measures).
3. Generate the training and testing data and labels using for loop with incremental test size, two kinds of criterion, random states, and max depth range from 1 to 9, then output the decision trees with max scores for each test size and different criterion.

****

**Step 3:**

I constructed 18 trees in total, and the general info of these trees is listed as below, and the detailed info is listed in the later parts.

****

# **Question A**

## **Part 2 Built Decision Trees**

### **Decision Tree 1**

Test Set Size: 0.1

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 4

Impurity Measures: gini

Scores: 0.9032258064516129

**Diagram

Description automatically generated**

### **Decision Tree 2**

Test Set Size: 0.1

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 6

Impurity Measures: entropy

Scores: 0.9354838709677419

**Diagram

Description automatically generated**

### **Decision Tree 3**

Test Set Size: 0.2

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 3

Impurity Measures: gini

Scores: 0.9193548387096774

Diagram

Description automatically generated

### **Decision Tree 4**

Test Set Size: 0.2

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 4

Impurity Measures: entropy

Scores: 0.9032258064516129

Diagram

Description automatically generated

### **Decision Tree 5**

Test Set Size: 0.30000000000000004

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 3

Impurity Measures: gini

Scores: 0.8829787234042553

Diagram

Description automatically generated

### **Decision Tree 6**

Test Set Size: 0.30000000000000004

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 3

Impurity Measures: entropy

Scores: 0.8829787234042553

Diagram

Description automatically generated

### **Decision Tree 7**

Test Set Size: 0.4

Stratify = trainining\_labels

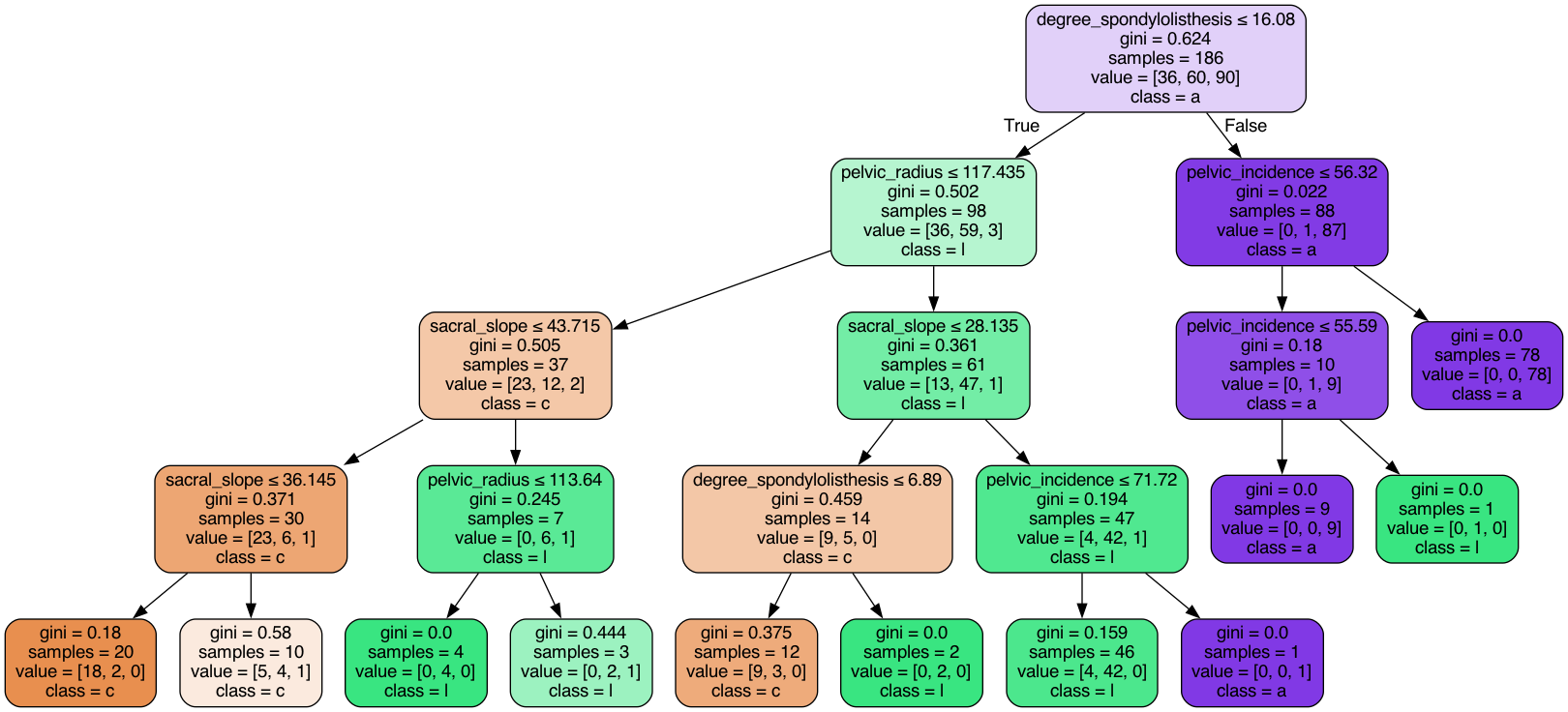
Shuffle: True

Random State: 27

Max Depth: 4

Impurity Measures: gini

Scores: 0.8709677419354839



### **Decision Tree 8**

Test Set Size: 0.4

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 3

Impurity Measures: entropy

Scores: 0.8629032258064516

Diagram

Description automatically generated

### **Decision Tree 9**

Test Set Size: 0.5

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 4

Impurity Measures: gini

Scores: 0.832258064516129

Diagram

Description automatically generated

### **Decision Tree 10**

Test Set Size: 0.5

Stratify = trainining\_labels

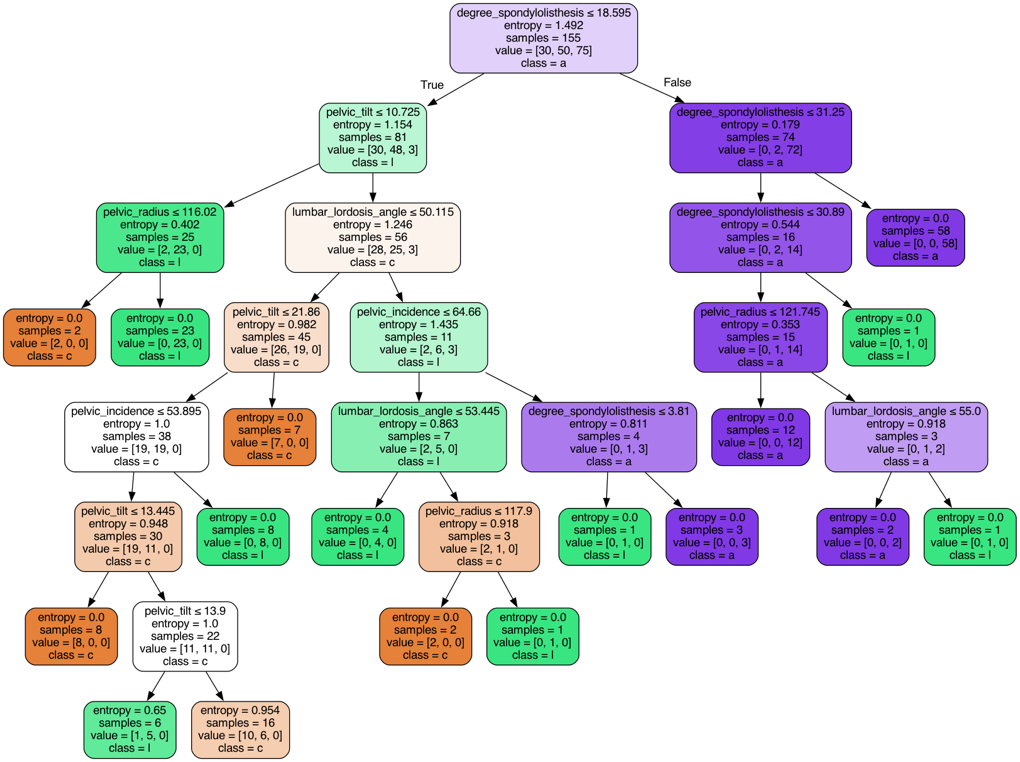
Shuffle: True

Random State: 43

Max Depth: 7

Impurity Measures: entropy

Scores: 0.8516129032258064



### **Decision Tree 11**

Test Set Size: 0.6

Stratify = trainining\_labels

Shuffle: True

Random State: 7

Max Depth: 3

Impurity Measures: gini

Scores: 0.8440860215053764

Diagram

Description automatically generated

### **Decision Tree 12**

Test Set Size: 0.6

Stratify = trainining\_labels

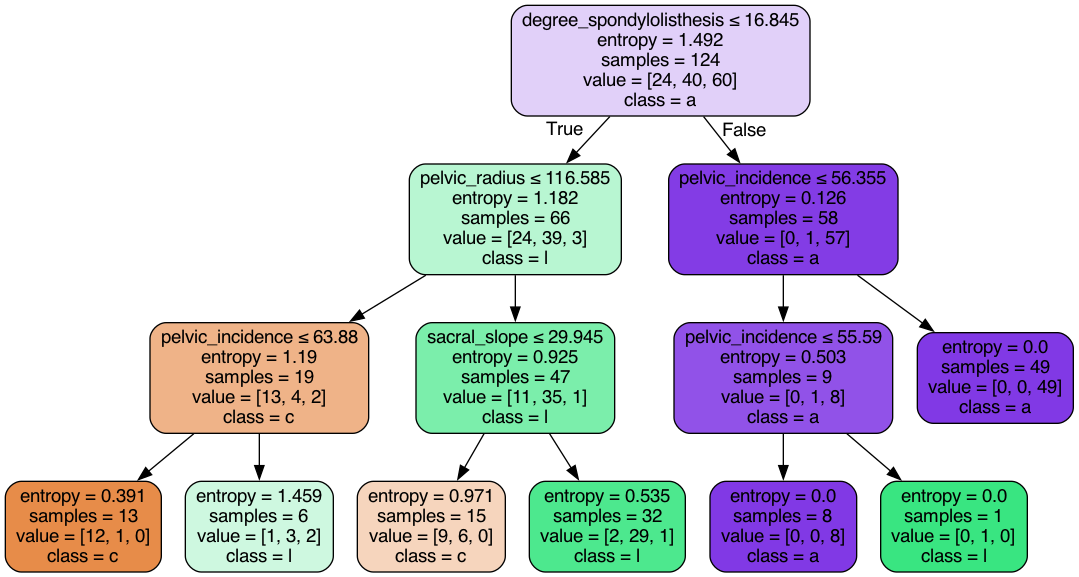
Shuffle: True

Random State: 27

Max Depth: 3

Impurity Measures: entropy

Scores: 0.8387096774193549



### **Decision Tree 13**

Test Set Size: 0.7000000000000001

Stratify = trainining\_labels

Shuffle: True

Random State: 43

Max Depth: 4

Impurity Measures: gini

Scores: 0.8302752293577982

Diagram

Description automatically generated

### **Decision Tree 14**

Test Set Size: 0.7000000000000001

Stratify = trainining\_labels

Shuffle: True

Random State: 43

Max Depth: 4

Impurity Measures: entropy

Scores: 0.8440366972477065

Diagram

Description automatically generated

### **Decision Tree 15**

Test Set Size: 0.8

Stratify = trainining\_labels

Shuffle: True

Random State: 27

Max Depth: 7

Impurity Measures: gini

Scores: 0.8185483870967742

Timeline

Description automatically generated

### **Decision Tree 16**

Test Set Size: 0.8

Stratify = trainining\_labels

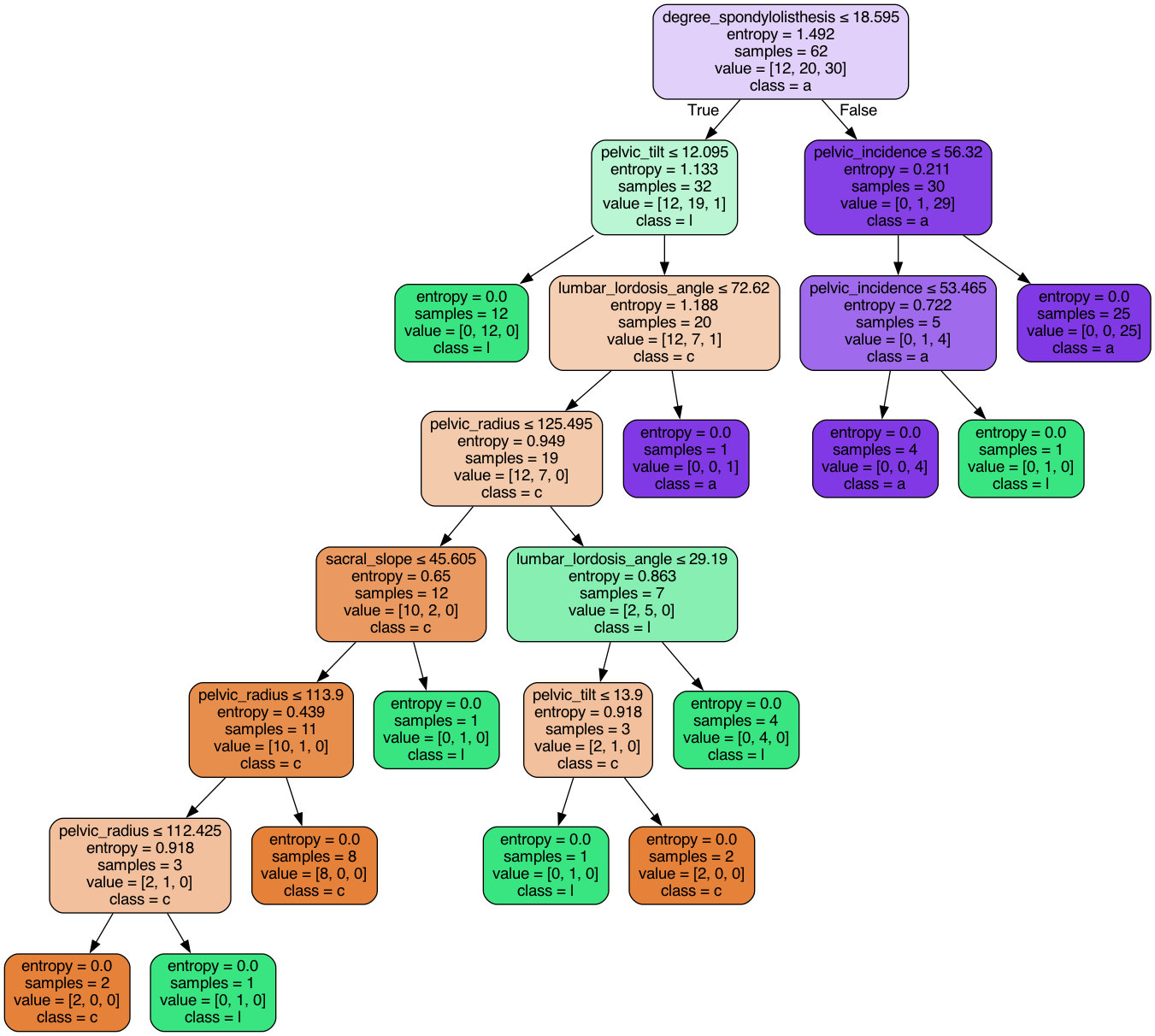
Shuffle: True

Random State: 43

Max Depth: 7

Impurity Measures: entropy

Scores: 0.7903225806451613



### **Decision Tree 17**

Test Set Size: 0.9

Stratify = trainining\_labels

Shuffle: True

Random State: 43

Max Depth: 3

Impurity Measures: gini

Scores: 0.8064516129032258

Diagram

Description automatically generated

### **Decision Tree 18**

Test Set Size: 0.8

Stratify = trainining\_labels

Shuffle: True

Random State: 43

Max Depth: 3

Impurity Measures: entropy

Scores: 0.8064516129032258

Diagram

Description automatically generated

# **Question B**

## **Question B Analysis 1(Impact of Criterion and Test Set Size on Max Scores)**

**Method:** First, I analyze the max score result of the trees with their different test sizes as x axis and their max scores as y axis. The result for the criterion of Gini and Entropy is shown as below.

**Annotation of the graph:** As elaborated in the tree building part, the result shown below is the best score result for each test size by choosing random state from [7, 27, 43] and choosing from max depth ranging from 1 to 9.

**Observations:**

1. As the below graphs shows, we may see the general image trend for the criterion of gini and entropy is quite similar. For both, the general tendency is that the larger the test size is, the lower the max scores are.
2. The maximum scores of all the results are 0.9193548387096774 for gini, and 0.9354838709677419 for entropy.

**Summarization:**

1. The Criterion doesn’t make too much impact on the max scores of each test size under this condition.
2. When the test set size increase at a relative high level, the model may have over-fitting problem, resulting in the decrease of the model score.

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

## **Question B Analysis 2(Impact of Criterion and Test Size on Max Depth)**

**Method:** First, I analyze the max depth of the trees whose result is the max scores at that test set size with their different test sizes as x axis and their max depth as y axis. The result for the criterion of Gini and Entropy is shown as below.

**Annotation of the graph:** As elaborated in the tree building part, the result shown below is the max depth of which the max scores occur for each test size by choosing random state from [7, 27, 43] and choosing from max depth ranging from 1 to 9.

**Observations:**

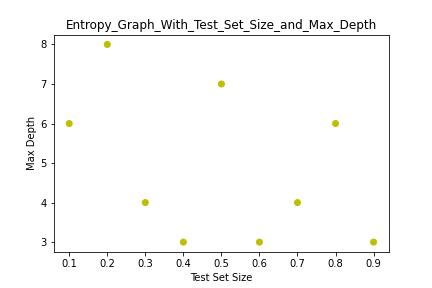
1. As the below graphs shows, we may see that the max depth when the max scores result occurs for each test quite differs between gini and entropy.
2. We can see that for both of gini and entropy, when the test size is 0.9, both has the minimum depth

**Summarization:**

We may see that the criterion will influence the max depth of which the max scores occur for each test set size.

Chart, scatter chart

Description automatically generated



## **Question B Analysis 3(Impact of Node Count on Max Scores)**

**Observation:**

The number of node count for the max scores for each test size mainly range from 10 to 20 for gini and 12.5 to 20 for entropy.

**Summarization:**

When the number of nodes is too low, it may encounter the under-fitting problem, and when the number of nodes is too high, it may encounter the over-fitting problems.

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generated

# **Question C Analysis**

**Methods:**

I calculated the accuracy scores for DH, SL, NO, DH or SL, DH or NO, SL or NO for each constructed decision trees. The figure is shown below.

Text

Description automatically generated

**Observations:**

1. The accuracy scores of “DH” or “NO” are much lower than the accuracy scores of “SL”.
2. However, the accuracy scores of the combination of “DH or NO” are much higher than the scores of single “DH” or single “NO”.

Chart, scatter chart

Description automatically generated

1. From the below figure and the samples number that our decision trees used, we can see that the “a” means “SL”, “l” means “NO”, and “c” means “DH”.

Diagram

Description automatically generated with low confidence

1. As we can see from many previously generated decision trees (i.e. decision tree 12, decision tree 13, etc), and some of their leaf node’s values [0] and values[1] are similar, which means they were not split clearly from each other, so they were confused with each other.

**Question C Summarization:**

Given the accuracy scores of the “DH” and “SL” pair of classes are likely to be confused with each other.

# **Question D Analysis**

**Parameters:**

From Question C Analysis, we can know that in the decision graph:

“a” stands for “SL”

“l” stands for “NO”

“c” stands for “DH”

## **Example 1: Decision Tree 4**

Diagram

Description automatically generated

**The confused leaf nodes:** node 1, node 2, and node 3

**The confused decision paths are listed as below:**

Path 1: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 45.125 == True

-> pelvic radius <= 117.435 == True

-> lumbar\_lordosis\_angle <= 31.4 == False

Result: 10 “NO” are misclassified as “DH”

Path 2: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 45.125 == True

-> pelvic\_radius <= 117.435 == False

-> lumbar\_lordosis\_angle <= 31.4 == True

Result: 7 “NO” are misclassified as “DH”

Path 3: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 45.125 == True

-> pelvic\_radius <= 117.435 == False

-> lumbar\_lordosis\_angle <= 31.4 == False

Result: 8 “DH” are misclassified as “NO”

## **Example 2: Decision Tree 1**

Timeline

Description automatically generated with low confidence

**The confused leaf nodes:** node 1, node 2, node 3, node 4, node 5, and node 6

**The confused decision paths are as below:**

Path 1: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 28.135 == True

-> sacral\_slope <= 23.495 == True

-> sacral\_slope <= 20.555 == True

Result: 1 “NO” are misclassified as “DH”

Path 2: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 28.135 == True

-> sacral\_slope <= 23.495 == True

-> sacral\_slope <= 20.555 == False

Result: 1 “DH” are misclassified as “NO”

Path 3: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 28.135 == True

-> sacral\_slope <= 23.495 ==False

-> lumbar\_lordosis\_angle <= 33.62 == False

Result: 3 “NO” are misclassified as “DH”

Path 4: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 28.135 == False

-> pelvic\_radius <=117.36 ==True

-> sacral\_slope <= 44.465 ==True

Result: 8 “NO” are misclassified as “DH”

Path 5: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 28.135 == False

-> pelvic\_radius <=117.36 ==False

-> sacral\_slope <= 44.465 ==True

Result: 3 “DH” are misclassified as “NO”

Path 6: degree\_spondylolisthesis <=16.08 == True

-> sacral\_slope <= 28.135 == False

-> pelvic\_radius <=117.36 ==False

-> sacral\_slope <= 44.465 ==False

Result: 6 “DH” are misclassified as “NO”

# APPENDIX

**My original code for this assignment:**

[**https://github.com/alfreddLUO/CS3481-Decision-Tree-Analysis-HW1.git**](https://github.com/alfreddLUO/CS3481-Decision-Tree-Analysis-HW1.git)

**Vertebral Column dataset from the UCI Machine Learning Repository:**

[**https://archive.ics.uci.edu/**](https://archive.ics.uci.edu/)