CS3481 Fundamentals of Data Science

Assignment 1

Decision Tree Analysis Report

Student Name: LUO Peiyuan
SID: 56642728

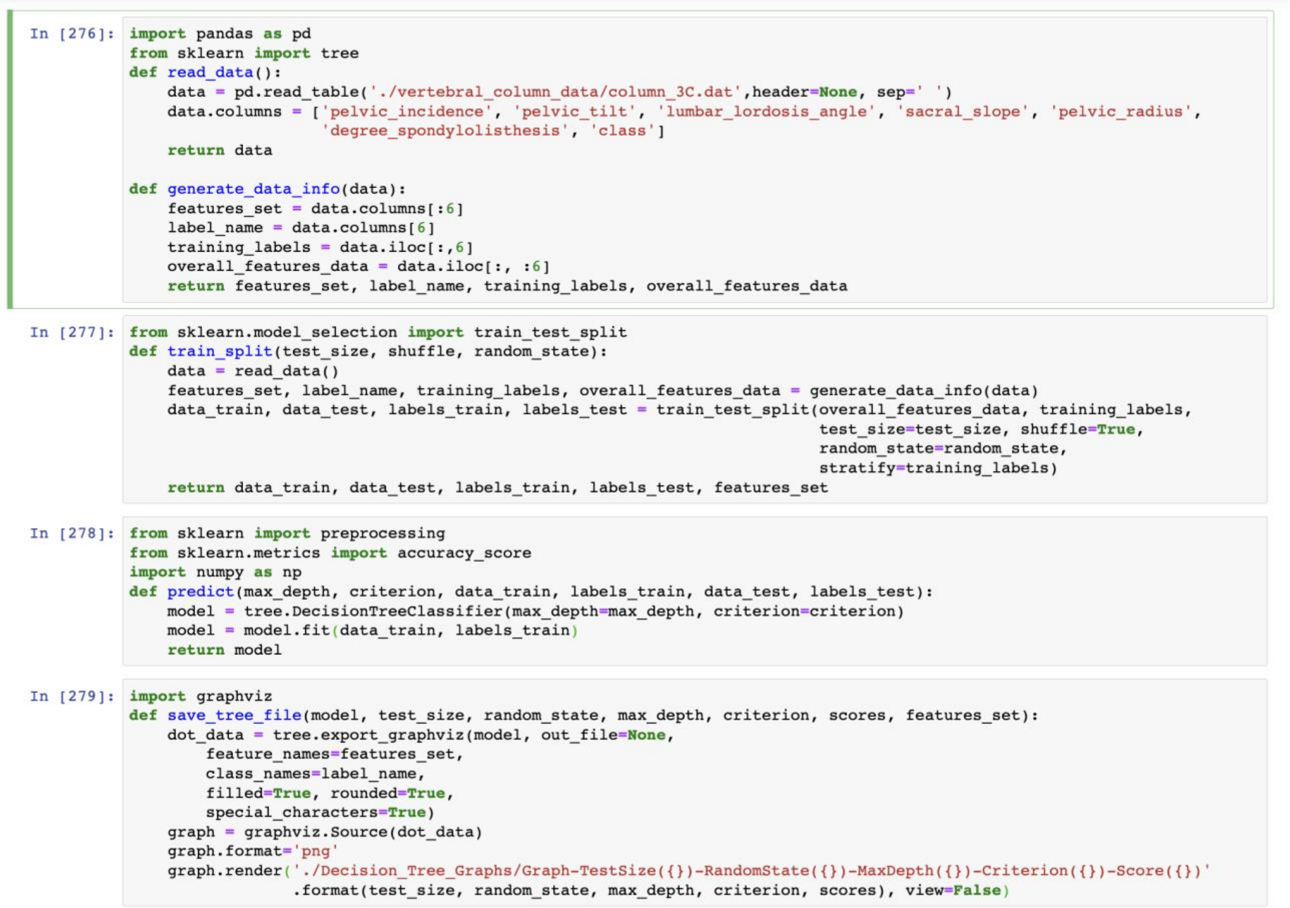
Table of Contents

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.



```
In [276]: import pandas as pd
          from sklearn import tree
          def read_data():
              data = pd.read_table('./vertebral_column_data/column_3C.dat',header=None, sep=' ')
              data.columns = ['pelvic_incidence', 'pelvic_tilt', 'lumbar_lordosis_angle', 'sacral_slope', 'pelvic_radius',
                              'degree_spondylolisthesis', 'class']
              return data

          def generate_data_info(data):
              features_set = data.columns[:6]
              label_name = data.columns[6]
              training_labels = data.iloc[:,6]
              overall_features_data = data.iloc[:, :6]
              return features_set, label_name, training_labels, overall_features_data
```

```
In [277]: from sklearn.model_selection import train_test_split
          def train_split(test_size, shuffle, random_state):
              data = read_data()
              features_set, label_name, training_labels, overall_features_data = generate_data_info(data)
              data_train, data_test, labels_train, labels_test = train_test_split(overall_features_data, training_labels,
                                                                                  test_size=test_size, shuffle=True,
                                                                                  random_state=random_state,
                                                                                  stratify=training_labels)
              return data_train, data_test, labels_train, labels_test, features_set
```

```
In [278]: from sklearn import preprocessing
          from sklearn.metrics import accuracy_score
          import numpy as np
          def predict(max_depth, criterion, data_train, labels_train, data_test, labels_test):
              model = tree.DecisionTreeClassifier(max_depth=max_depth, criterion=criterion)
              model = model.fit(data_train, labels_train)
              return model
```

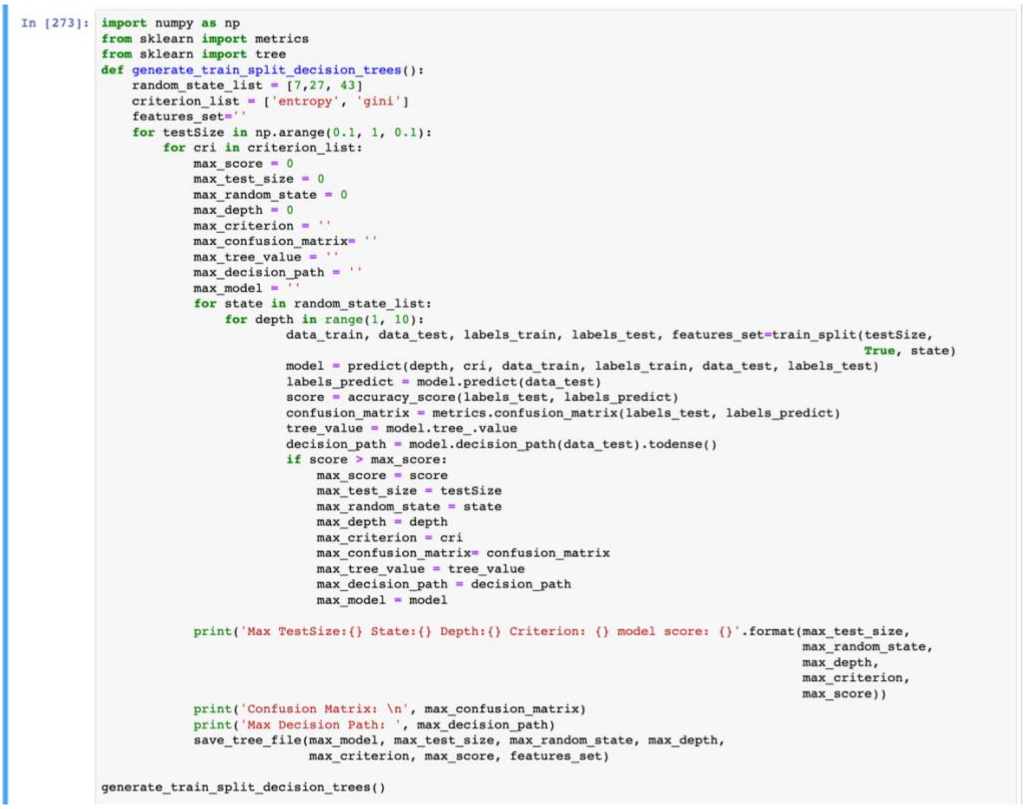
```
In [279]: import graphviz
          def save_tree_file(model, test_size, random_state, max_depth, criterion, scores, features_set):
              dot_data = tree.export_graphviz(model, out_file=None,
                  feature_names=features_set,
                  class_names=label_name,
                  filled=True, rounded=True,
                  special_characters=True)
              graph = graphviz.Source(dot_data)
              graph.format='png'
              graph.render('./Decision_Tree_Graphs/Graph-TestSize({})-RandomState({})-MaxDepth({})-Criterion({})-Score({})'
                           .format(test_size, random_state, max_depth, criterion, scores), view=False)
```

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).

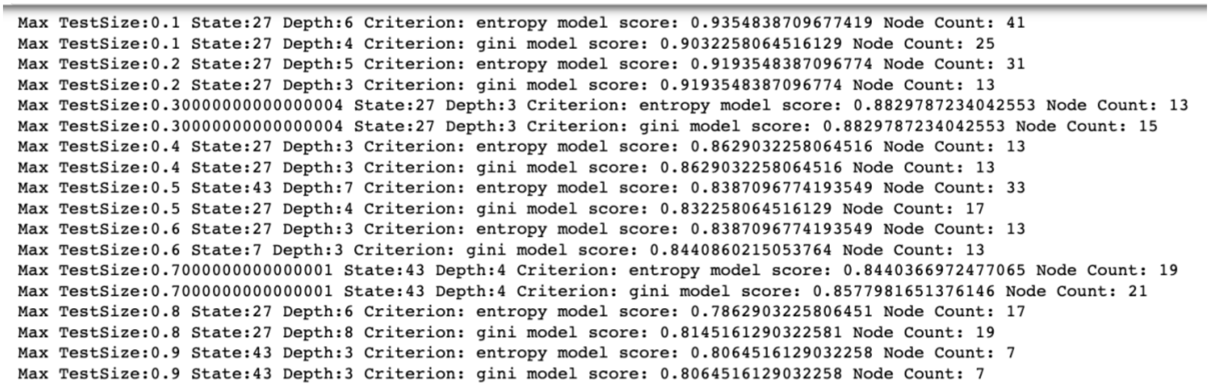
(3) Generate the training and testing data and labels using for loop with incremental test size, two kinds of criterion, random states, and max depth range from 1 to 9, then output the decision trees with max scores for each test size and different criterion.



```
In [273]: import numpy as np
          from sklearn import metrics
          from sklearn import tree
          def generate_train_split_decision_trees():
              random_state_list = [7,27, 43]
              criterion_list = ['entropy', 'gini']
              features_set=''
              for testSize in np.arange(0.1, 1, 0.1):
                  for cri in criterion_list:
                      max_score = 0
                      max_test_size = 0
                      max_random_state = 0
                      max_depth = 0
                      max_criterion = ''
                      max_confusion_matrix= ''
                      max_tree_value = ''
                      max_decision_path = ''
                      max_model = ''
                      for state in random_state_list:
                          for depth in range(1, 10):
                              data_train, data_test, labels_train, labels_test, features_set=train_split(testSize,
                                                                                                        True, state)
                              model = predict(depth, cri, data_train, labels_train, data_test, labels_test)
                              labels_predict = model.predict(data_test)
                              score = accuracy_score(labels_test, labels_predict)
                              confusion_matrix = metrics.confusion_matrix(labels_test, labels_predict)
                              tree_value = model.tree_.value
                              decision_path = model.decision_path(data_test).todense()
                              if score > max_score:
                                  max_score = score
                                  max_test_size = testSize
                                  max_random_state = state
                                  max_depth = depth
                                  max_criterion = cri
                                  max_confusion_matrix= confusion_matrix
                                  max_tree_value = tree_value
                                  max_decision_path = decision_path
                                  max_model = model

                      print('Max TestSize:{} State:{} Depth:{} Criterion: {} model score: {}'.format(max_test_size,
                                                                                                     max_random_state,
                                                                                                     max_depth,
                                                                                                     max_criterion,
                                                                                                     max_score))

                      print('Confusion Matrix: \n', max_confusion_matrix)
                      print('Max Decision Path: ', max_decision_path)
                      save_tree_file(max_model, max_test_size, max_random_state, max_depth,
                                     max_criterion, max_score, features_set)

          generate_train_split_decision_trees()
```

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.



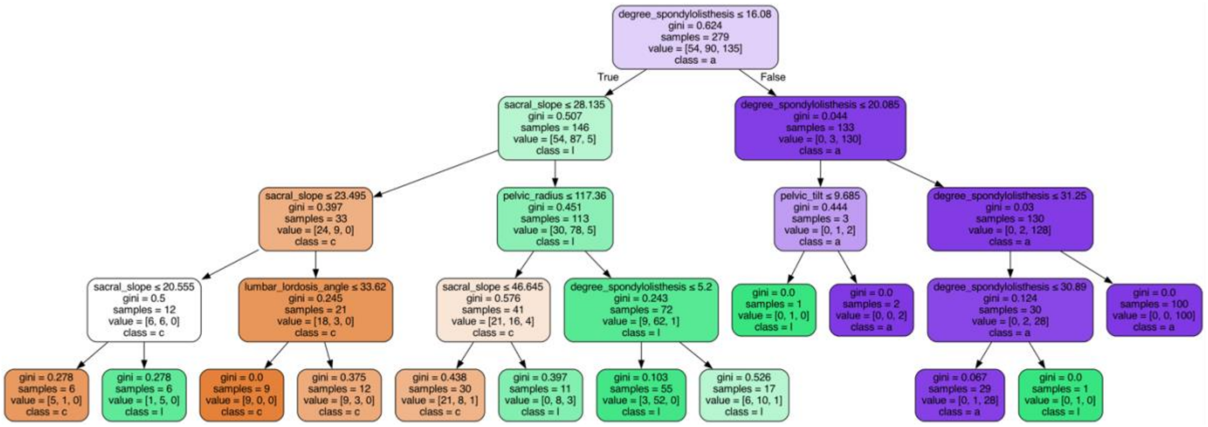
```
Max TestSize:0.1 State:27 Depth:6 Criterion: entropy model score: 0.9354838709677419 Node Count: 41
Max TestSize:0.1 State:27 Depth:4 Criterion: gini model score: 0.9032258064516129 Node Count: 25
Max TestSize:0.2 State:27 Depth:5 Criterion: entropy model score: 0.9193548387096774 Node Count: 31
Max TestSize:0.2 State:27 Depth:3 Criterion: gini model score: 0.9193548387096774 Node Count: 13
Max TestSize:0.30000000000000004 State:27 Depth:3 Criterion: entropy model score: 0.8829787234042553 Node Count: 13
Max TestSize:0.30000000000000004 State:27 Depth:3 Criterion: gini model score: 0.8829787234042553 Node Count: 15
Max TestSize:0.4 State:27 Depth:3 Criterion: entropy model score: 0.8629032258064516 Node Count: 13
Max TestSize:0.4 State:27 Depth:3 Criterion: gini model score: 0.8629032258064516 Node Count: 13
Max TestSize:0.5 State:43 Depth:7 Criterion: entropy model score: 0.8387096774193549 Node Count: 33
Max TestSize:0.5 State:27 Depth:4 Criterion: gini model score: 0.832258064516129 Node Count: 17
Max TestSize:0.6 State:27 Depth:3 Criterion: entropy model score: 0.8387096774193549 Node Count: 13
Max TestSize:0.6 State:7 Depth:3 Criterion: gini model score: 0.8440860215053764 Node Count: 13
Max TestSize:0.7000000000000001 State:43 Depth:4 Criterion: entropy model score: 0.8440366972477065 Node Count: 19
Max TestSize:0.7000000000000001 State:43 Depth:4 Criterion: gini model score: 0.8577981651376146 Node Count: 21
Max TestSize:0.8 State:27 Depth:6 Criterion: entropy model score: 0.7862903225806451 Node Count: 17
Max TestSize:0.8 State:27 Depth:8 Criterion: gini model score: 0.814516129032258 Node Count: 19
Max TestSize:0.9 State:43 Depth:3 Criterion: entropy model score: 0.8064516129032258 Node Count: 7
Max TestSize:0.9 State:43 Depth:3 Criterion: gini model score: 0.8064516129032258 Node Count: 7
```

Question A

Part 2 Built Decision Trees

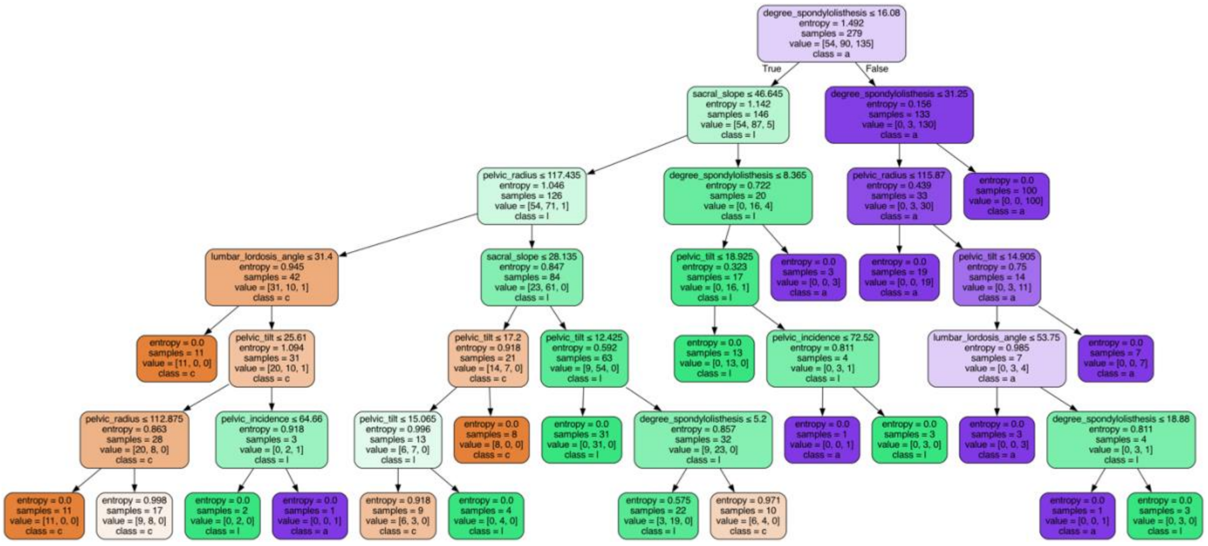
Decision Tree 1

Test Set Size: 0.1
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 4
Impurity Measures: gini
Scores: 0.9032258064516129



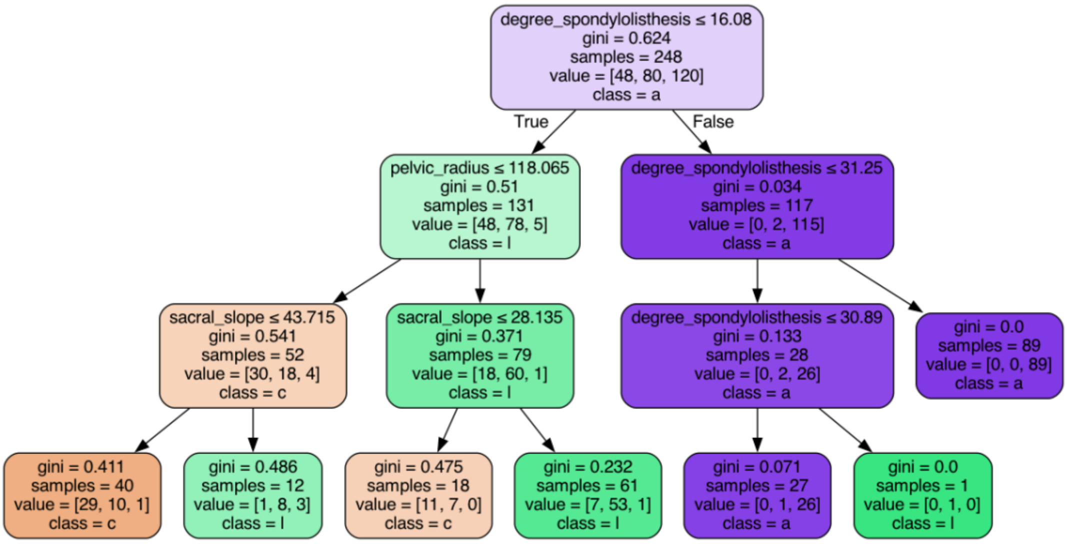
Decision Tree 2

Test Set Size: 0.1
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 6
Impurity Measures: entropy
Scores: 0.9354838709677419



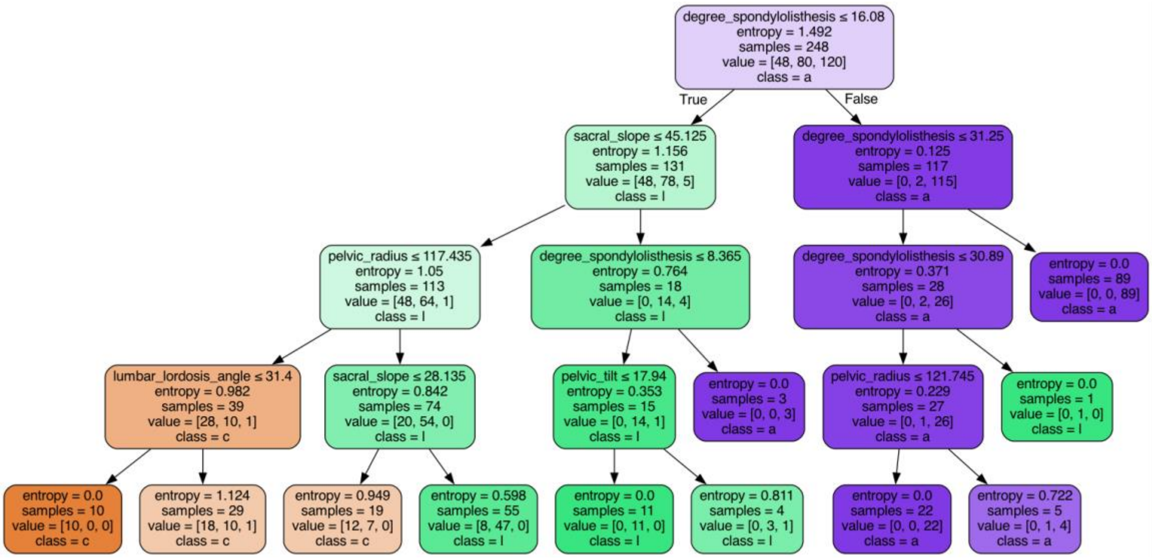
Decision Tree 3

Test Set Size: 0.2
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 3
Impurity Measures: gini
Scores: 0.9193548387096774



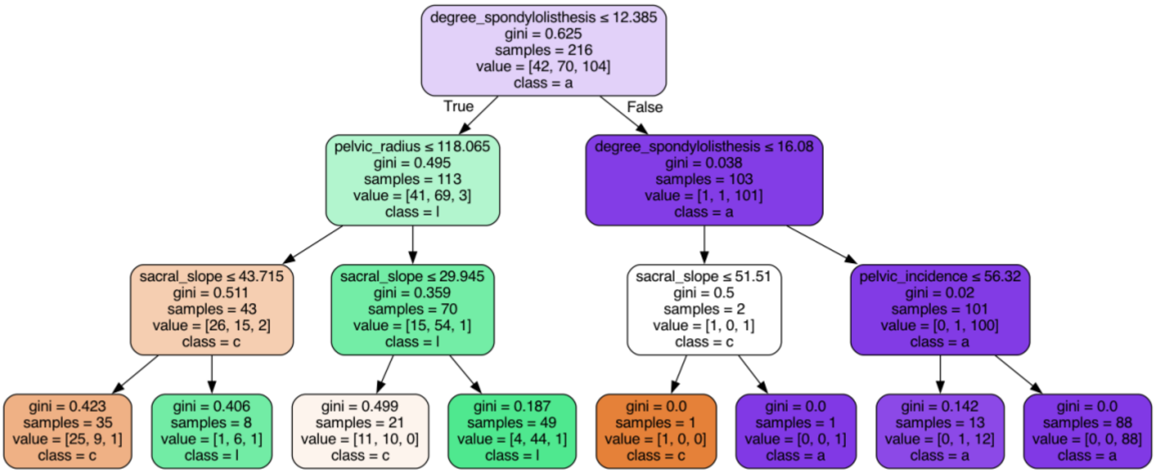
Decision Tree 4

Test Set Size: 0.2
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 4
Impurity Measures: entropy
Scores: 0.9032258064516129



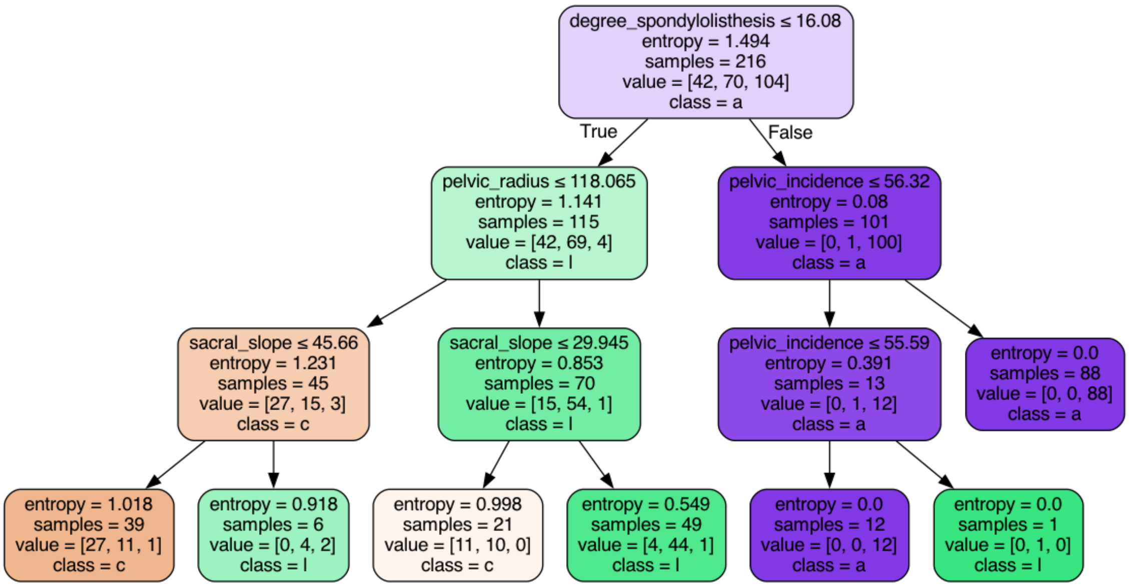
Decision Tree 5

Test Set Size: 0.30000000000000004
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 3
Impurity Measures: gini
Scores: 0.8829787234042553


degree_spondylolisthesis ≤ 12.385 gini = 0.625 samples = 216 value = [42, 70, 104] class = a
True
False
pelvic_radius ≤ 118.065 gini = 0.495 samples = 113 value = [41, 69, 3] class = l
degree_spondylolisthesis ≤ 16.08 gini = 0.038 samples = 103 value = [1, 1, 101] class = a
sacral_slope ≤ 43.715 gini = 0.511 samples = 43 value = [26, 15, 2] class = c
sacral_slope ≤ 29.945 gini = 0.359 samples = 70 value = [15, 54, 1] class = l
sacral_slope ≤ 51.51 gini = 0.5 samples = 2 value = [1, 0, 1] class = c
pelvic_incidence ≤ 56.32 gini = 0.02 samples = 101 value = [0, 1, 100] class = a
gini = 0.423 samples = 35 value = [25, 9, 1] class = c
gini = 0.406 samples = 8 value = [1, 6, 1] class = l
gini = 0.499 samples = 21 value = [11, 10, 0] class = c
gini = 0.187 samples = 49 value = [4, 44, 1] class = l
gini = 0.0 samples = 1 value = [1, 0, 0] class = c
gini = 0.0 samples = 1 value = [0, 0, 1] class = a
gini = 0.142 samples = 13 value = [0, 1, 12] class = a
gini = 0.0 samples = 88 value = [0, 0, 88] class = a

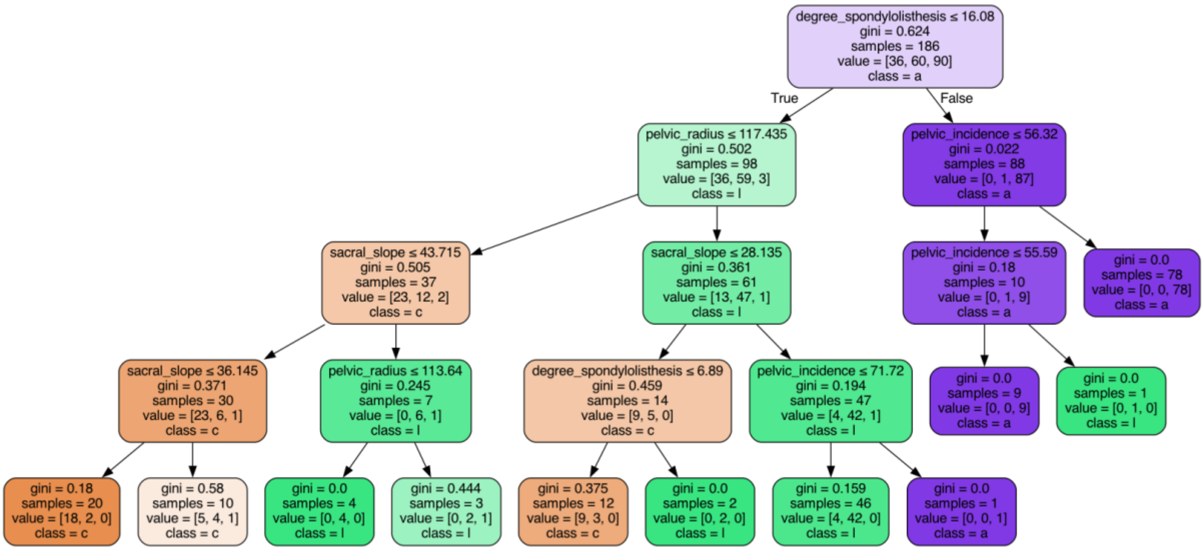
Decision Tree 6

Test Set Size: 0.30000000000000004
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 3
Impurity Measures: entropy
Scores: 0.8829787234042553


degree_spondylolisthesis ≤ 16.08 entropy = 1.494 samples = 216 value = [42, 70, 104] class = a
True
False
pelvic_radius ≤ 118.065 entropy = 1.141 samples = 115 value = [42, 69, 4] class = l
pelvic_incidence ≤ 56.32 entropy = 0.08 samples = 101 value = [0, 1, 100] class = a
sacral_slope ≤ 45.66 entropy = 1.231 samples = 45 value = [27, 15, 3] class = c
sacral_slope ≤ 29.945 entropy = 0.853 samples = 70 value = [15, 54, 1] class = l
pelvic_incidence ≤ 55.59 entropy = 0.391 samples = 13 value = [0, 1, 12] class = a
entropy = 0.0 samples = 88 value = [0, 0, 88] class = a
entropy = 1.018 samples = 39 value = [27, 11, 1] class = c
entropy = 0.918 samples = 6 value = [0, 4, 2] class = l
entropy = 0.998 samples = 21 value = [11, 10, 0] class = c
entropy = 0.549 samples = 49 value = [4, 44, 1] class = l
entropy = 0.0 samples = 12 value = [0, 0, 12] class = a
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l

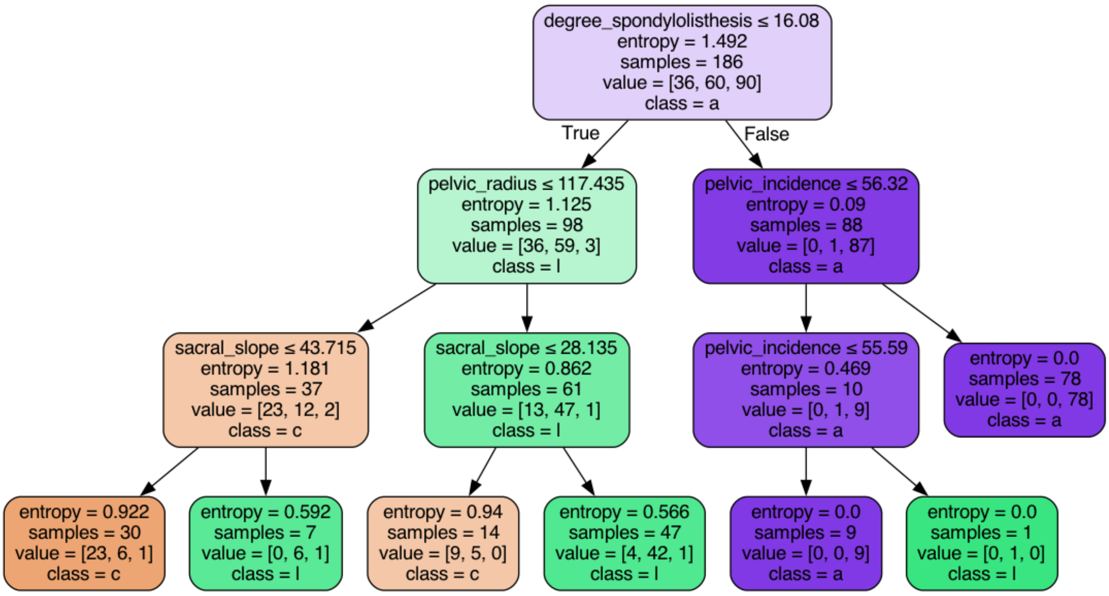
Decision Tree 7

Test Set Size: 0.4
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 4
Impurity Measures: gini
Scores: 0.8709677419354839



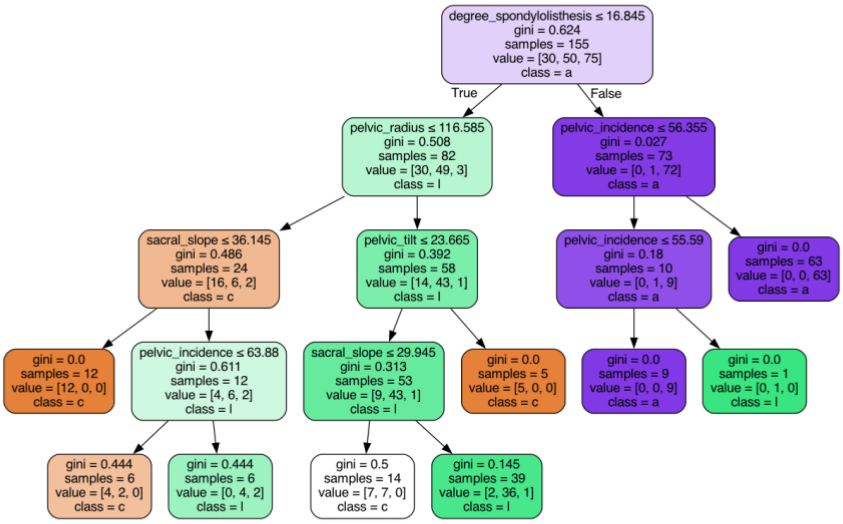
Decision Tree 8

Test Set Size: 0.4
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 3
Impurity Measures: entropy
Scores: 0.8629032258064516



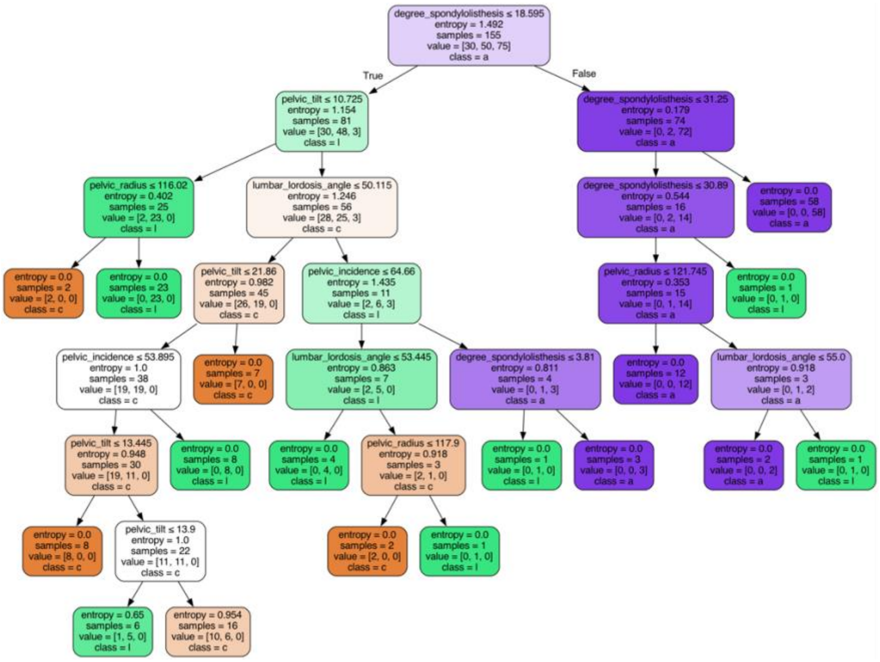
Decision Tree 9

Test Set Size: 0.5
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 4
Impurity Measures: gini
Scores: 0.832258064516129


degree_spondylolisthesis ≤ 16.845 gini = 0.624 samples = 155 value = [30, 50, 75] class = a
True
False
pelvic_radius ≤ 116.585 gini = 0.508 samples = 82 value = [30, 49, 3] class = l
pelvic_incidence ≤ 56.355 gini = 0.027 samples = 73 value = [0, 1, 72] class = a
sacral_slope ≤ 36.145 gini = 0.486 samples = 24 value = [16, 6, 2] class = c
pelvic_tilt ≤ 23.665 gini = 0.392 samples = 58 value = [14, 43, 1] class = l
pelvic_incidence ≤ 55.59 gini = 0.18 samples = 10 value = [0, 1, 9] class = a
gini = 0.0 samples = 63 value = [0, 0, 63] class = a
gini = 0.0 samples = 12 value = [12, 0, 0] class = c
pelvic_incidence ≤ 63.88 gini = 0.611 samples = 12 value = [4, 6, 2] class = l
sacral_slope ≤ 29.945 gini = 0.313 samples = 53 value = [9, 43, 1] class = l
gini = 0.0 samples = 5 value = [5, 0, 0] class = c
gini = 0.0 samples = 9 value = [0, 0, 9] class = a
gini = 0.0 samples = 1 value = [0, 1, 0] class = l
gini = 0.444 samples = 6 value = [4, 2, 0] class = c
gini = 0.444 samples = 6 value = [0, 4, 2] class = l
gini = 0.5 samples = 14 value = [7, 7, 0] class = c
gini = 0.145 samples = 39 value = [2, 36, 1] class = l

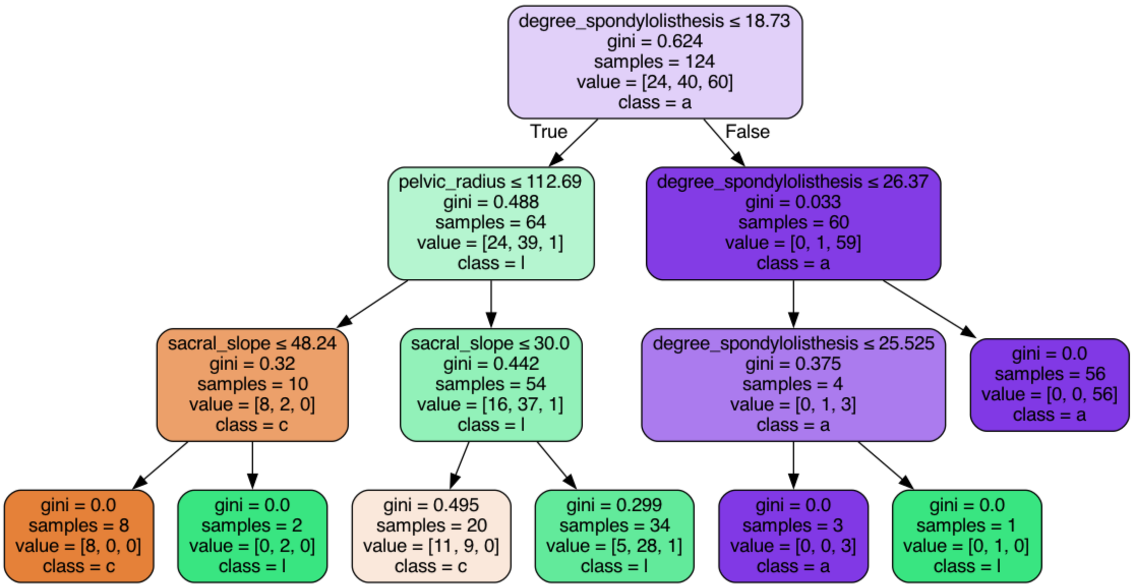
Decision Tree 10

Test Set Size: 0.5
Stratify = trainining_labels
Shuffle: True
Random State: 43
Max Depth: 7
Impurity Measures: entropy
Scores: 0.8516129032258064


degree_spondylolisthesis ≤ 18.595 entropy = 1.492 samples = 155 value = [30, 50, 75] class = a
True
False
pelvic_tilt ≤ 10.725 entropy = 1.154 samples = 81 value = [30, 48, 3] class = l
degree_spondylolisthesis ≤ 31.25 entropy = 0.179 samples = 74 value = [0, 2, 72] class = a
pelvic_radius ≤ 116.02 entropy = 0.402 samples = 25 value = [2, 23, 0] class = l
lumbar_lordosis_angle ≤ 50.115 entropy = 1.246 samples = 56 value = [28, 25, 3] class = c
degree_spondylolisthesis ≤ 30.89 entropy = 0.544 samples = 16 value = [0, 2, 14] class = a
entropy = 0.0 samples = 58 value = [0, 0, 58] class = a
entropy = 0.0 samples = 2 value = [2, 0, 0] class = c
entropy = 0.0 samples = 23 value = [0, 23, 0] class = l
pelvic_tilt ≤ 21.86 entropy = 0.982 samples = 45 value = [26, 19, 0] class = c
pelvic_incidence ≤ 64.66 entropy = 1.435 samples = 11 value = [2, 6, 3] class = l
pelvic_radius ≤ 121.745 entropy = 0.353 samples = 15 value = [0, 1, 14] class = a
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l
pelvic_incidence ≤ 53.895 entropy = 1.0 samples = 38 value = [19, 19, 0] class = c
entropy = 0.0 samples = 7 value = [7, 0, 0] class = c
lumbar_lordosis_angle ≤ 53.445 entropy = 0.863 samples = 7 value = [2, 5, 0] class = l
degree_spondylolisthesis ≤ 3.81 entropy = 0.811 samples = 4 value = [0, 1, 3] class = a
entropy = 0.0 samples = 12 value = [0, 0, 12] class = a
lumbar_lordosis_angle ≤ 55.0 entropy = 0.918 samples = 3 value = [0, 1, 2] class = a
pelvic_tilt ≤ 13.445 entropy = 0.948 samples = 30 value = [19, 11, 0] class = c
entropy = 0.0 samples = 8 value = [0, 8, 0] class = l
entropy = 0.0 samples = 4 value = [0, 4, 0] class = l
pelvic_radius ≤ 117.9 entropy = 0.918 samples = 3 value = [2, 1, 0] class = c
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l
entropy = 0.0 samples = 3 value = [0, 0, 3] class = a
entropy = 0.0 samples = 2 value = [0, 0, 2] class = a
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l
entropy = 0.0 samples = 8 value = [8, 0, 0] class = c
pelvic_tilt ≤ 13.9 entropy = 1.0 samples = 22 value = [11, 11, 0] class = c
entropy = 0.0 samples = 2 value = [2, 0, 0] class = c
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l
entropy = 0.65 samples = 6 value = [1, 5, 0] class = l
entropy = 0.954 samples = 16 value = [10, 6, 0] class = c

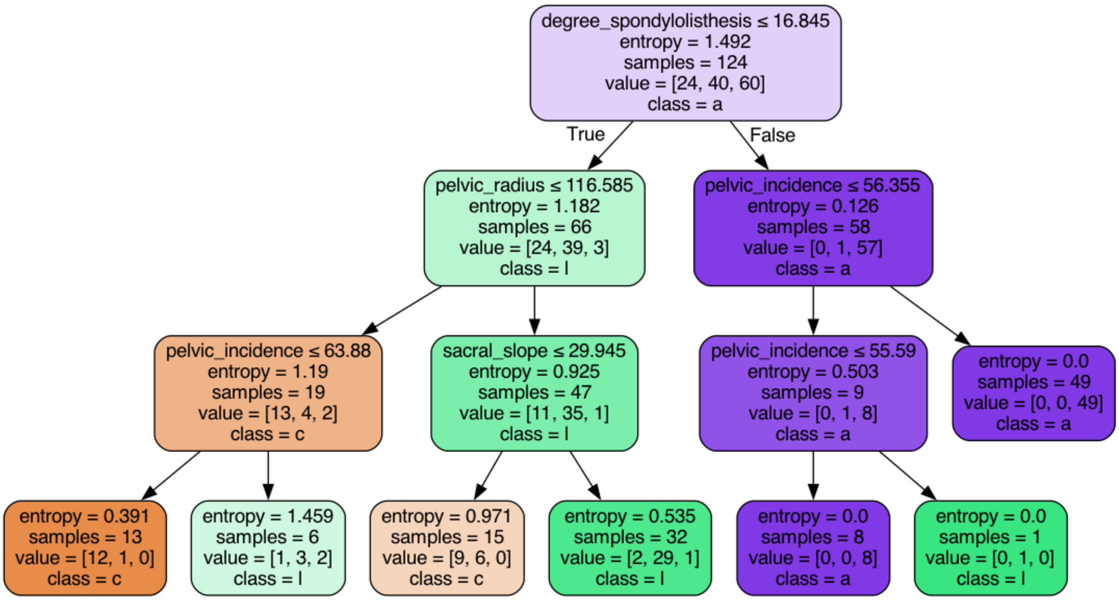
Decision Tree 11

Test Set Size: 0.6
Stratify = trainining_labels
Shuffle: True
Random State: 7
Max Depth: 3
Impurity Measures: gini
Scores: 0.8440860215053764



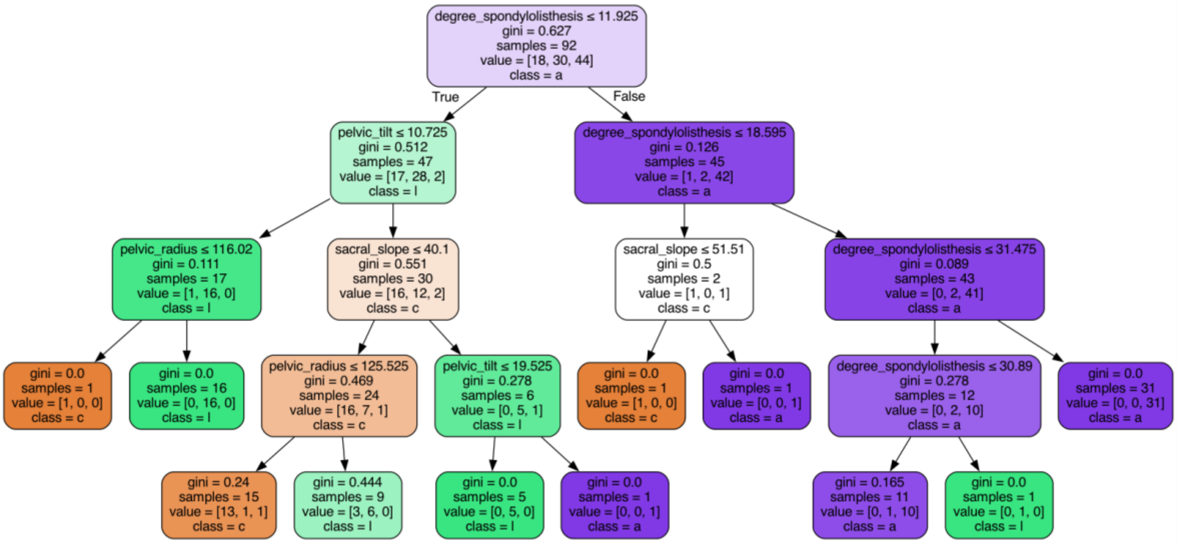
Decision Tree 12

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

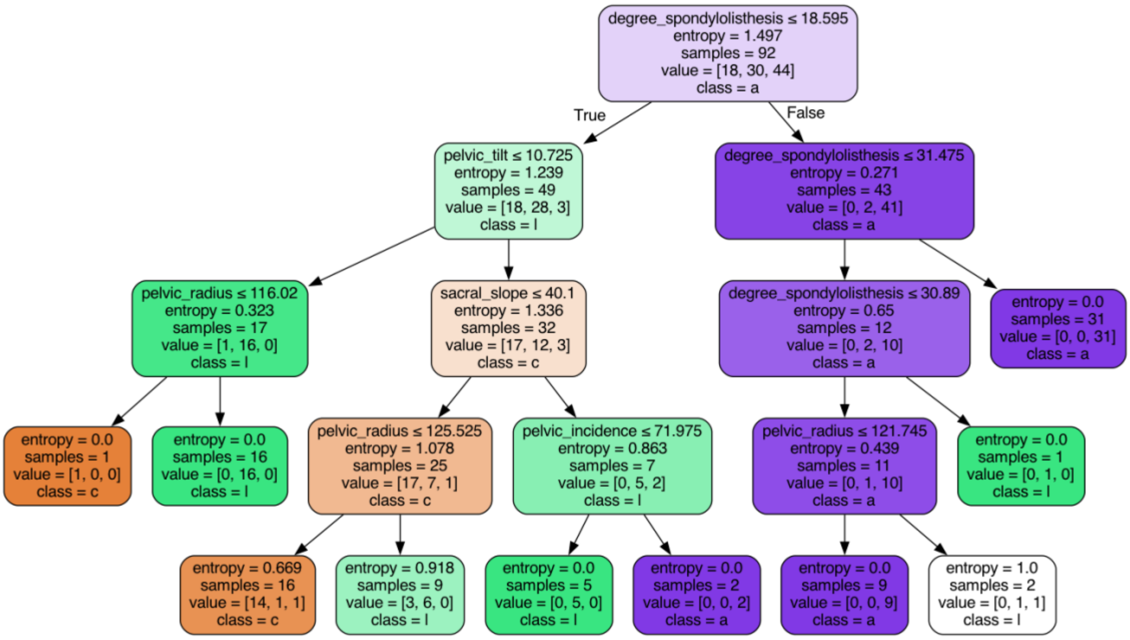
Test Set Size: 0.7000000000000001
Stratify = trainining_labels
Shuffle: True
Random State: 43
Max Depth: 4
Impurity Measures: gini
Scores: 0.8302752293577982



degree_spondylolisthesis ≤ 11.925 gini = 0.627 samples = 92 value = [18, 30, 44] class = a
True
False
pelvic_tilt ≤ 10.725 gini = 0.512 samples = 47 value = [17, 28, 2] class = l
degree_spondylolisthesis ≤ 18.595 gini = 0.126 samples = 45 value = [1, 2, 42] class = a
pelvic_radius ≤ 116.02 gini = 0.111 samples = 17 value = [1, 16, 0] class = l
sacral_slope ≤ 40.1 gini = 0.551 samples = 30 value = [16, 12, 2] class = c
sacral_slope ≤ 51.51 gini = 0.5 samples = 2 value = [1, 0, 1] class = c
degree_spondylolisthesis ≤ 31.475 gini = 0.089 samples = 43 value = [0, 2, 41] class = a
gini = 0.0 samples = 1 value = [1, 0, 0] class = c
gini = 0.0 samples = 16 value = [0, 16, 0] class = l
pelvic_radius ≤ 125.525 gini = 0.469 samples = 24 value = [16, 7, 1] class = c
pelvic_tilt ≤ 19.525 gini = 0.278 samples = 6 value = [0, 5, 1] class = l
gini = 0.0 samples = 1 value = [1, 0, 0] class = c
gini = 0.0 samples = 1 value = [0, 0, 1] class = a
degree_spondylolisthesis ≤ 30.89 gini = 0.278 samples = 12 value = [0, 2, 10] class = a
gini = 0.0 samples = 31 value = [0, 0, 31] class = a
gini = 0.24 samples = 15 value = [13, 1, 1] class = c
gini = 0.444 samples = 9 value = [3, 6, 0] class = l
gini = 0.0 samples = 5 value = [0, 5, 0] class = l
gini = 0.0 samples = 1 value = [0, 0, 1] class = a
gini = 0.165 samples = 11 value = [0, 1, 10] class = a
gini = 0.0 samples = 1 value = [0, 1, 0] class = l

Decision Tree 14

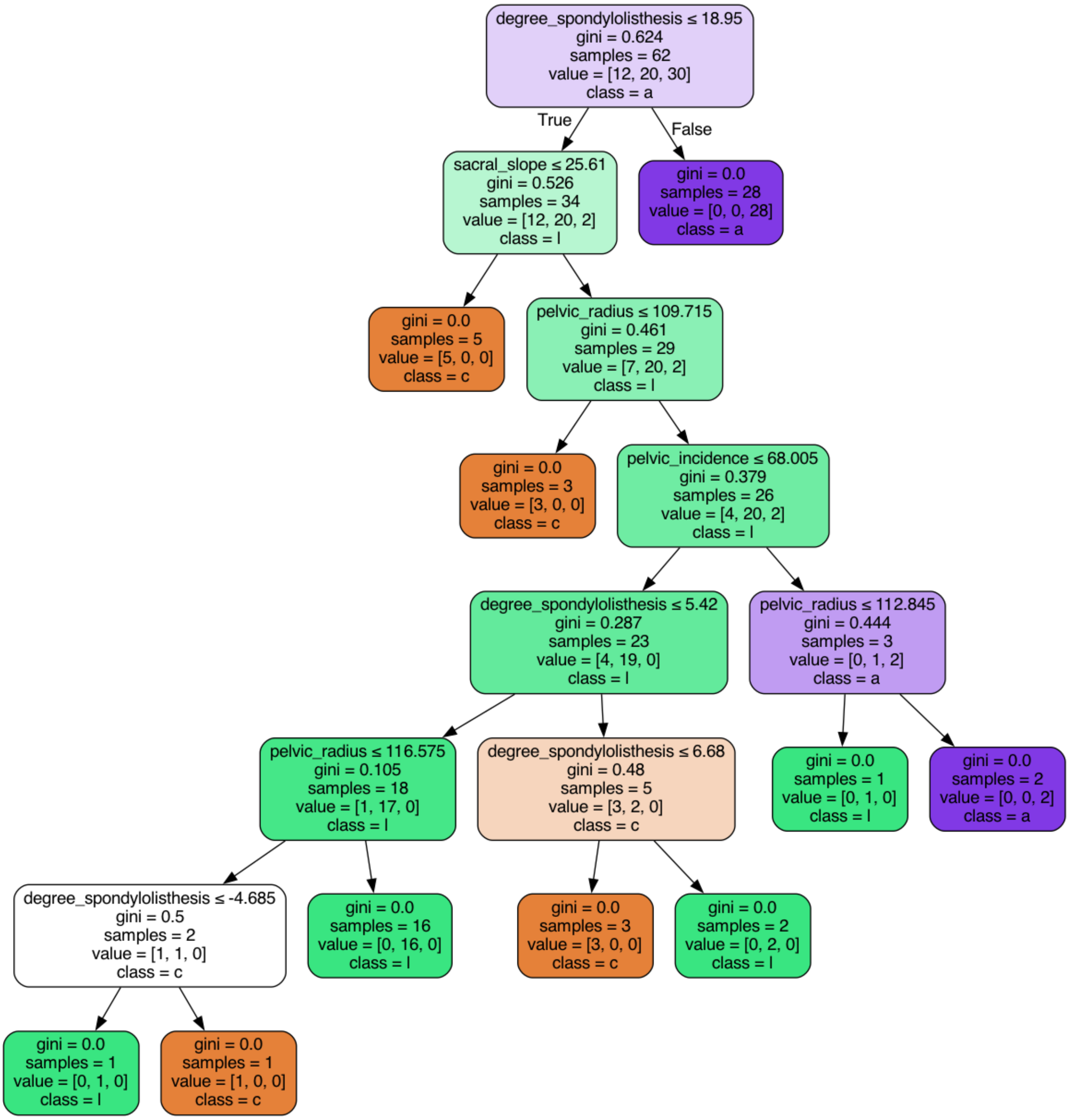
Test Set Size: 0.7000000000000001
Stratify = trainining_labels
Shuffle: True
Random State: 43
Max Depth: 4
Impurity Measures: entropy
Scores: 0.8440366972477065



degree_spondylolisthesis ≤ 18.595 entropy = 1.497 samples = 92 value = [18, 30, 44] class = a
True
False
pelvic_tilt ≤ 10.725 entropy = 1.239 samples = 49 value = [18, 28, 3] class = l
degree_spondylolisthesis ≤ 31.475 entropy = 0.271 samples = 43 value = [0, 2, 41] class = a
pelvic_radius ≤ 116.02 entropy = 0.323 samples = 17 value = [1, 16, 0] class = l
sacral_slope ≤ 40.1 entropy = 1.336 samples = 32 value = [17, 12, 3] class = c
degree_spondylolisthesis ≤ 30.89 entropy = 0.65 samples = 12 value = [0, 2, 10] class = a
entropy = 0.0 samples = 31 value = [0, 0, 31] class = a
entropy = 0.0 samples = 1 value = [1, 0, 0] class = c
entropy = 0.0 samples = 16 value = [0, 16, 0] class = l
pelvic_radius ≤ 125.525 entropy = 1.078 samples = 25 value = [17, 7, 1] class = c
pelvic_incidence ≤ 71.975 entropy = 0.863 samples = 7 value = [0, 5, 2] class = l
pelvic_radius ≤ 121.745 entropy = 0.439 samples = 11 value = [0, 1, 10] class = a
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l
entropy = 0.669 samples = 16 value = [14, 1, 1] class = c
entropy = 0.918 samples = 9 value = [3, 6, 0] class = l
entropy = 0.0 samples = 5 value = [0, 5, 0] class = l
entropy = 0.0 samples = 2 value = [0, 0, 2] class = a
entropy = 0.0 samples = 9 value = [0, 0, 9] class = a
entropy = 1.0 samples = 2 value = [0, 1, 1] class = l

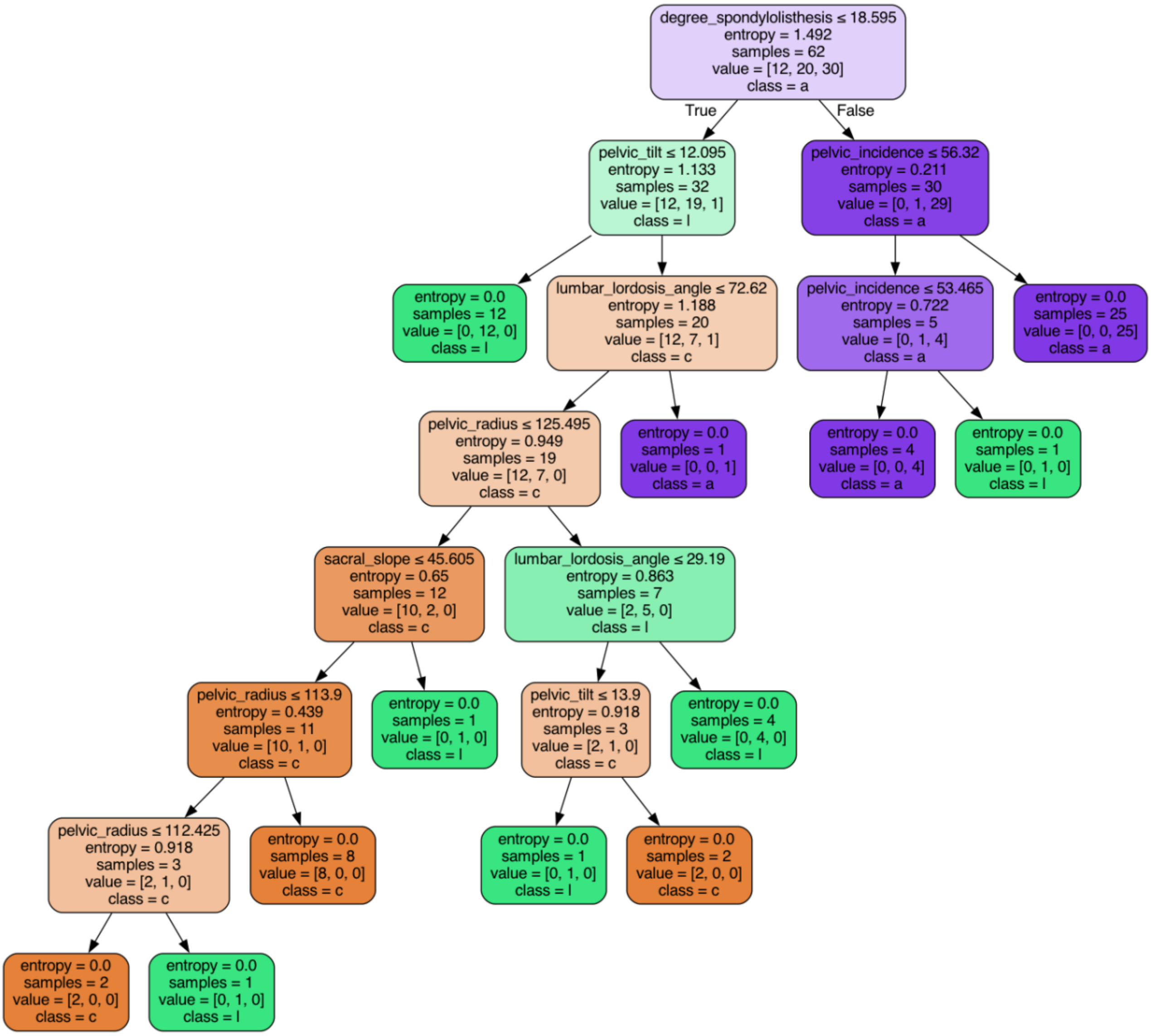
Decision Tree 15

Test Set Size: 0.8
Stratify = trainining_labels
Shuffle: True
Random State: 27
Max Depth: 7
Impurity Measures: gini
Scores: 0.8185483870967742



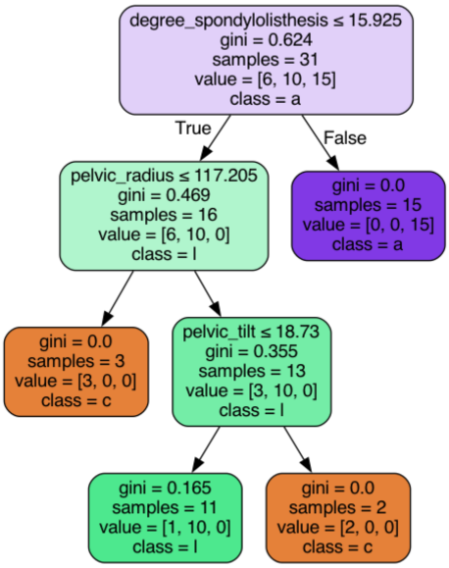
Decision Tree 16

Test Set Size: 0.8
Stratify = traininining_labels
Shuffle: True
Random State: 43
Max Depth: 7
Impurity Measures: entropy
Scores: 0.7903225806451613



Decision Tree 17

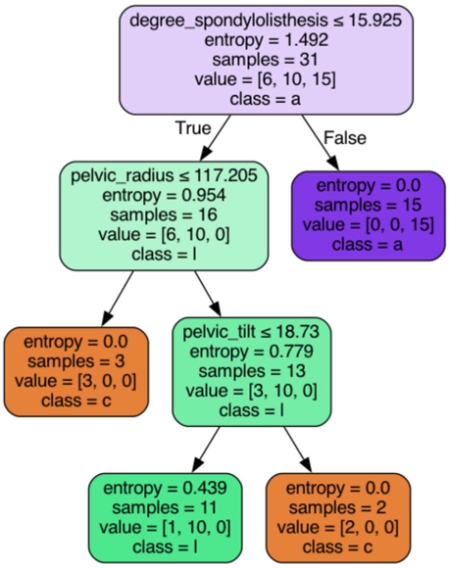
Test Set Size: 0.9
Stratify = trainining_labels
Shuffle: True
Random State: 43
Max Depth: 3
Impurity Measures: gini
Scores: 0.8064516129032258



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



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.

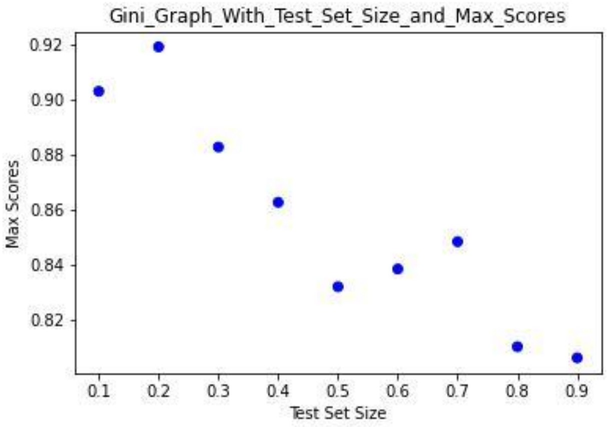
Annotation of the graph: As elaborated in the tree building part, the result shown below is the best score result for each test size by choosing random state from [7, 27, 43] and choosing from max depth ranging from 1 to 9.

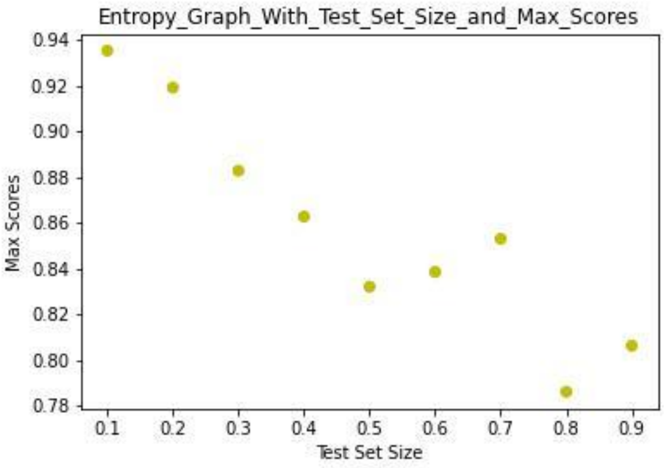
Observations:

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.





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.

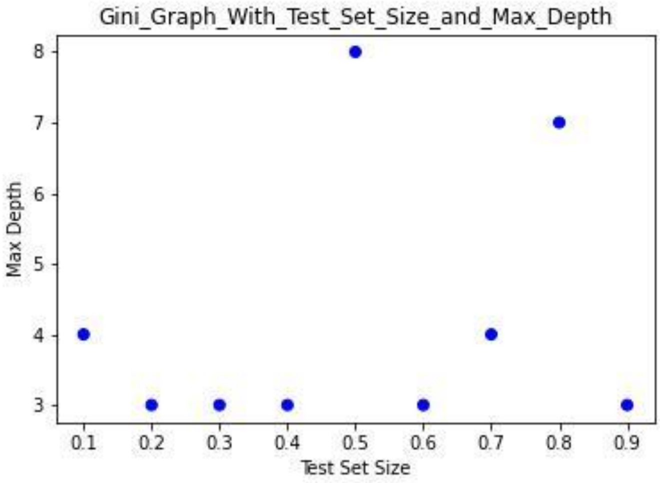
Annotation of the graph: As elaborated in the tree building part, the result shown below is the max depth of which the max scores occur for each test size by choosing random state from [7, 27, 43] and choosing from max depth ranging from 1 to 9.

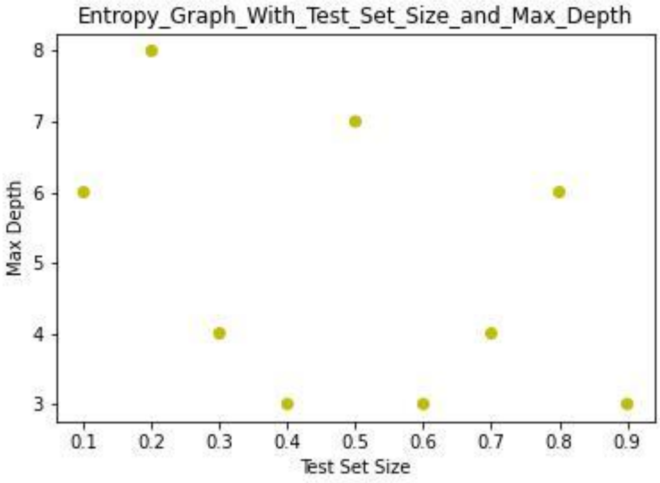
Observations:

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.





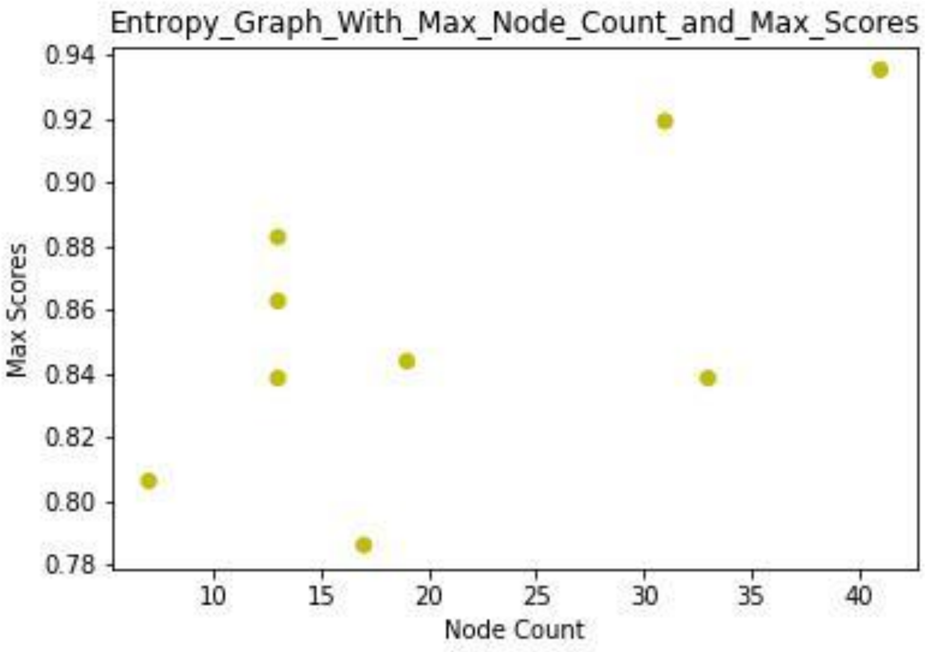
Question B Analysis 3(Impact of Node Count on Max Scores)

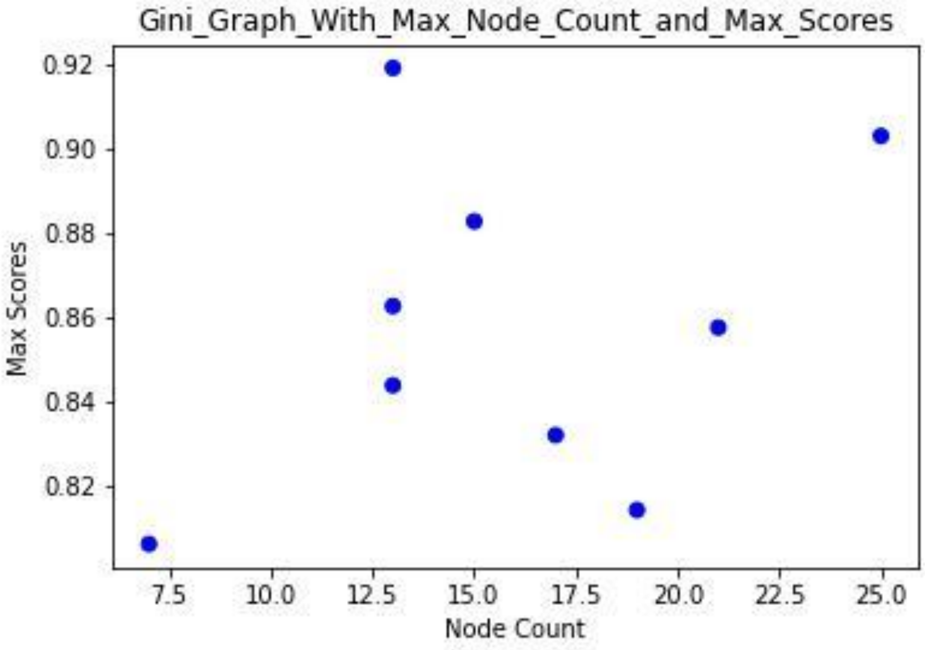
Observation:

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.

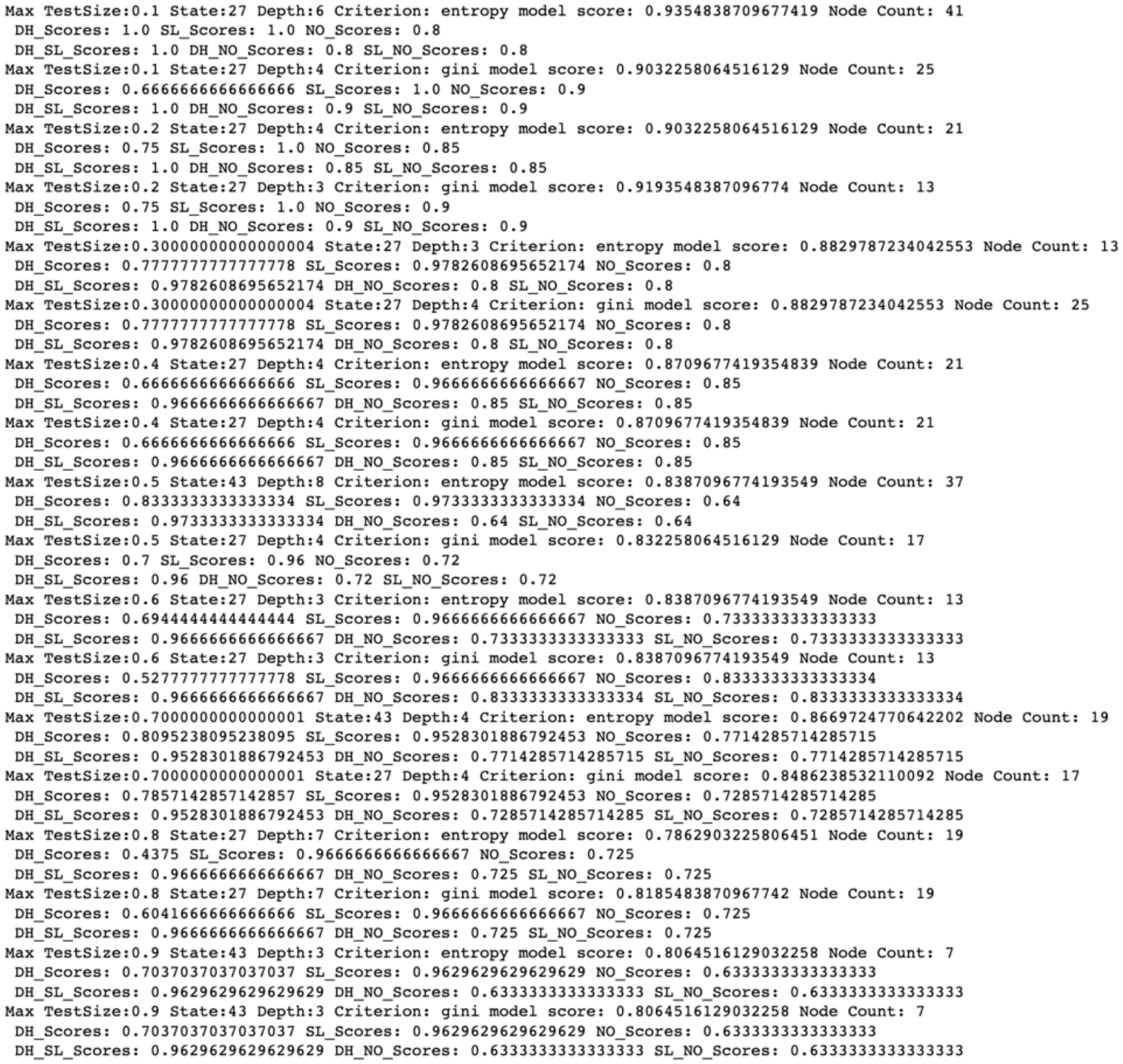




Question C Analysis

Methods:

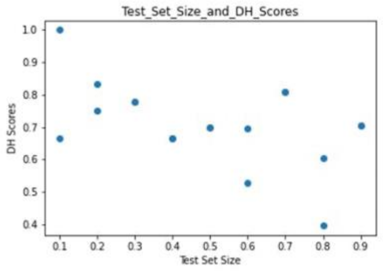
I calculated the accuracy scores for DH, SL, NO, DH or SL, DH or NO, SL or NO for each constructed decision trees. The figure is shown below.

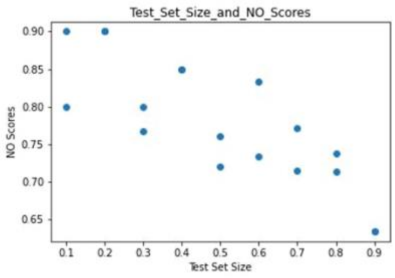


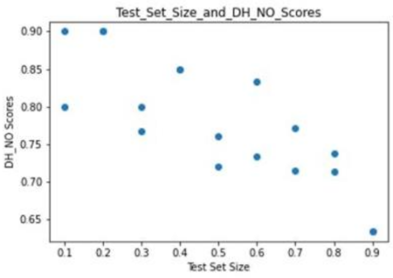
```
Max TestSize:0.1 State:27 Depth:6 Criterion: entropy model score: 0.9354838709677419 Node Count: 41
 DH_Scores: 1.0 SL_Scores: 1.0 NO_Scores: 0.8
 DH_SL_Scores: 1.0 DH_NO_Scores: 0.8 SL_NO_Scores: 0.8
Max TestSize:0.1 State:27 Depth:4 Criterion: gini model score: 0.9032258064516129 Node Count: 25
 DH_Scores: 0.6666666666666666 SL_Scores: 1.0 NO_Scores: 0.9
 DH_SL_Scores: 1.0 DH_NO_Scores: 0.9 SL_NO_Scores: 0.9
Max TestSize:0.2 State:27 Depth:4 Criterion: entropy model score: 0.9032258064516129 Node Count: 21
 DH_Scores: 0.75 SL_Scores: 1.0 NO_Scores: 0.85
 DH_SL_Scores: 1.0 DH_NO_Scores: 0.85 SL_NO_Scores: 0.85
Max TestSize:0.2 State:27 Depth:3 Criterion: gini model score: 0.9193548387096774 Node Count: 13
 DH_Scores: 0.75 SL_Scores: 1.0 NO_Scores: 0.9
 DH_SL_Scores: 1.0 DH_NO_Scores: 0.9 SL_NO_Scores: 0.9
Max TestSize:0.30000000000000004 State:27 Depth:3 Criterion: entropy model score: 0.8829787234042553 Node Count: 13
 DH_Scores: 0.7777777777777778 SL_Scores: 0.9782608695652174 NO_Scores: 0.8
 DH_SL_Scores: 0.9782608695652174 DH_NO_Scores: 0.8 SL_NO_Scores: 0.8
Max TestSize:0.30000000000000004 State:27 Depth:4 Criterion: gini model score: 0.8829787234042553 Node Count: 25
 DH_Scores: 0.7777777777777778 SL_Scores: 0.9782608695652174 NO_Scores: 0.8
 DH_SL_Scores: 0.9782608695652174 DH_NO_Scores: 0.8 SL_NO_Scores: 0.8
Max TestSize:0.4 State:27 Depth:4 Criterion: entropy model score: 0.8709677419354839 Node Count: 21
 DH_Scores: 0.6666666666666666 SL_Scores: 0.9666666666666667 NO_Scores: 0.85
 DH_SL_Scores: 0.9666666666666667 DH_NO_Scores: 0.85 SL_NO_Scores: 0.85
Max TestSize:0.4 State:27 Depth:4 Criterion: gini model score: 0.8709677419354839 Node Count: 21
 DH_Scores: 0.6666666666666666 SL_Scores: 0.9666666666666667 NO_Scores: 0.85
 DH_SL_Scores: 0.9666666666666667 DH_NO_Scores: 0.85 SL_NO_Scores: 0.85
Max TestSize:0.5 State:43 Depth:8 Criterion: entropy model score: 0.8387096774193549 Node Count: 37
 DH_Scores: 0.8333333333333334 SL_Scores: 0.9733333333333334 NO_Scores: 0.64
 DH_SL_Scores: 0.9733333333333334 DH_NO_Scores: 0.64 SL_NO_Scores: 0.64
Max TestSize:0.5 State:27 Depth:4 Criterion: gini model score: 0.832258064516129 Node Count: 17
 DH_Scores: 0.7 SL_Scores: 0.96 NO_Scores: 0.72
 DH_SL_Scores: 0.96 DH_NO_Scores: 0.72 SL_NO_Scores: 0.72
Max TestSize:0.6 State:27 Depth:3 Criterion: entropy model score: 0.8387096774193549 Node Count: 13
 DH_Scores: 0.6944444444444444 SL_Scores: 0.9666666666666667 NO_Scores: 0.7333333333333333
 DH_SL_Scores: 0.9666666666666667 DH_NO_Scores: 0.7333333333333333 SL_NO_Scores: 0.7333333333333333
Max TestSize:0.6 State:27 Depth:3 Criterion: gini model score: 0.8387096774193549 Node Count: 13
 DH_Scores: 0.5277777777777778 SL_Scores: 0.9666666666666667 NO_Scores: 0.8333333333333334
 DH_SL_Scores: 0.9666666666666667 DH_NO_Scores: 0.8333333333333334 SL_NO_Scores: 0.8333333333333334
Max TestSize:0.7000000000000001 State:43 Depth:4 Criterion: entropy model score: 0.8669724770642202 Node Count: 19
 DH_Scores: 0.8095238095238095 SL_Scores: 0.9528301886792453 NO_Scores: 0.7714285714285715
 DH_SL_Scores: 0.9528301886792453 DH_NO_Scores: 0.7714285714285715 SL_NO_Scores: 0.7714285714285715
Max TestSize:0.7000000000000001 State:27 Depth:4 Criterion: gini model score: 0.8486238532110092 Node Count: 17
 DH_Scores: 0.7857142857142857 SL_Scores: 0.9528301886792453 NO_Scores: 0.7285714285714285
 DH_SL_Scores: 0.9528301886792453 DH_NO_Scores: 0.7285714285714285 SL_NO_Scores: 0.7285714285714285
Max TestSize:0.8 State:27 Depth:7 Criterion: entropy model score: 0.7862903225806451 Node Count: 19
 DH_Scores: 0.4375 SL_Scores: 0.9666666666666667 NO_Scores: 0.725
 DH_SL_Scores: 0.9666666666666667 DH_NO_Scores: 0.725 SL_NO_Scores: 0.725
Max TestSize:0.8 State:27 Depth:7 Criterion: gini model score: 0.8185483870967742 Node Count: 19
 DH_Scores: 0.6041666666666666 SL_Scores: 0.9666666666666667 NO_Scores: 0.725
 DH_SL_Scores: 0.9666666666666667 DH_NO_Scores: 0.725 SL_NO_Scores: 0.725
Max TestSize:0.9 State:43 Depth:3 Criterion: entropy model score: 0.8064516129032258 Node Count: 7
 DH_Scores: 0.7037037037037037 SL_Scores: 0.9629629629629629 NO_Scores: 0.6333333333333333
 DH_SL_Scores: 0.9629629629629629 DH_NO_Scores: 0.6333333333333333 SL_NO_Scores: 0.6333333333333333
Max TestSize:0.9 State:43 Depth:3 Criterion: gini model score: 0.8064516129032258 Node Count: 7
 DH_Scores: 0.7037037037037037 SL_Scores: 0.9629629629629629 NO_Scores: 0.6333333333333333
 DH_SL_Scores: 0.9629629629629629 DH_NO_Scores: 0.6333333333333333 SL_NO_Scores: 0.6333333333333333
```

Observations:

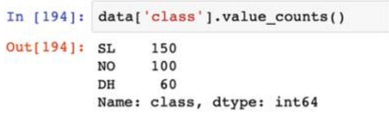
1. The accuracy scores of “DH” or “NO” are much lower than the accuracy scores of “SL”.
2. However, the accuracy scores of the combination of “DH or NO” are much higher than the scores of single “DH” or single “NO”.



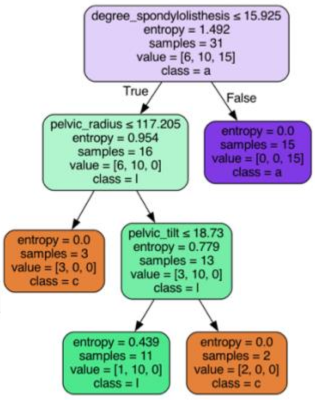




3. From the below figure and the samples number that our decision trees used, we can see that the “a” means “SL”, “l” means “NO”, and “c” means “DH”.



```
In [194]: data['class'].value_counts()

Out[194]: SL     150
          NO     100
          DH      60
          Name: class, dtype: int64
```



4. 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:

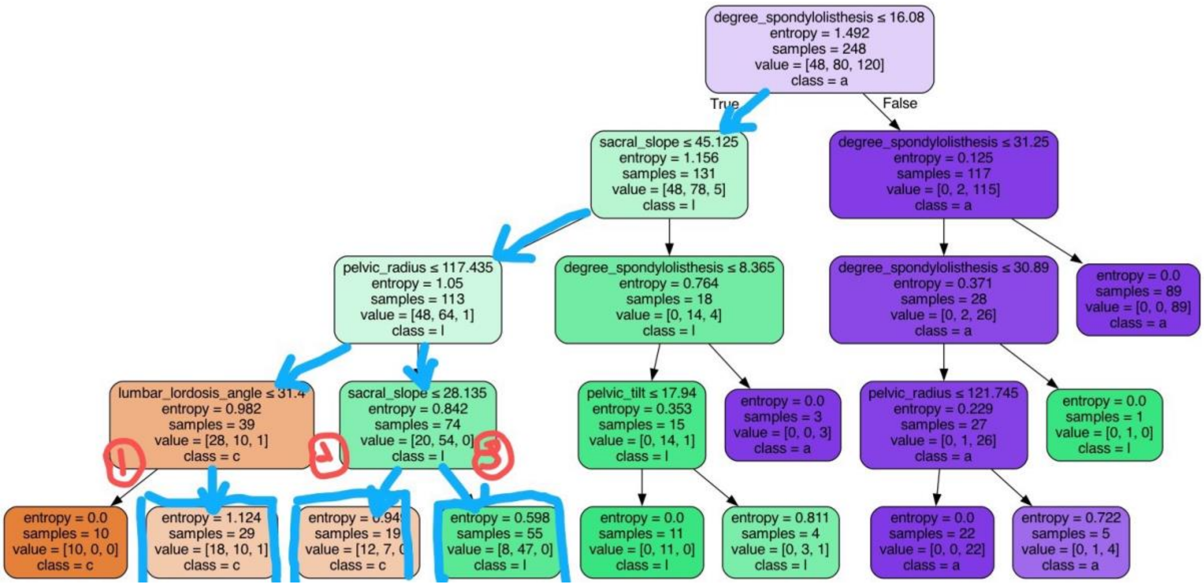
Given the accuracy scores of the “DH” and “SL” pair of classes are likely to be confused with each other.

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



degree_spondylolisthesis ≤ 16.08 entropy = 1.492 samples = 248 value = [48, 80, 120] class = a
True
False
sacral_slope ≤ 45.125 entropy = 1.156 samples = 131 value = [48, 78, 5] class = l
degree_spondylolisthesis ≤ 31.25 entropy = 0.125 samples = 117 value = [0, 2, 115] class = a
pelvic_radius ≤ 117.435 entropy = 1.05 samples = 113 value = [48, 64, 1] class = l
degree_spondylolisthesis ≤ 8.365 entropy = 0.764 samples = 18 value = [0, 14, 4] class = l
degree_spondylolisthesis ≤ 30.89 entropy = 0.371 samples = 28 value = [0, 2, 26] class = a
entropy = 0.0 samples = 89 value = [0, 0, 89] class = a
lumbar_lordosis_angle ≤ 31.4 entropy = 0.982 samples = 39 value = [28, 10, 1] class = c
sacral_slope ≤ 28.135 entropy = 0.842 samples = 74 value = [20, 54, 0] class = l
pelvic_tilt ≤ 17.94 entropy = 0.353 samples = 15 value = [0, 14, 1] class = l
entropy = 0.0 samples = 3 value = [0, 0, 3] class = a
pelvic_radius ≤ 121.745 entropy = 0.229 samples = 27 value = [0, 1, 26] class = a
entropy = 0.0 samples = 1 value = [0, 1, 0] class = l
1
2
3
entropy = 0.0 samples = 10 value = [10, 0, 0] class = c
entropy = 1.124 samples = 29 value = [18, 10, 1] class = c
entropy = 0.94 samples = 19 value = [12, 7, 0] class = c
entropy = 0.598 samples = 55 value = [8, 47, 0] class = l
entropy = 0.0 samples = 11 value = [0, 11, 0] class = l
entropy = 0.811 samples = 4 value = [0, 3, 1] class = l
entropy = 0.0 samples = 22 value = [0, 0, 22] class = a
entropy = 0.722 samples = 5 value = [0, 1, 4] class = a

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

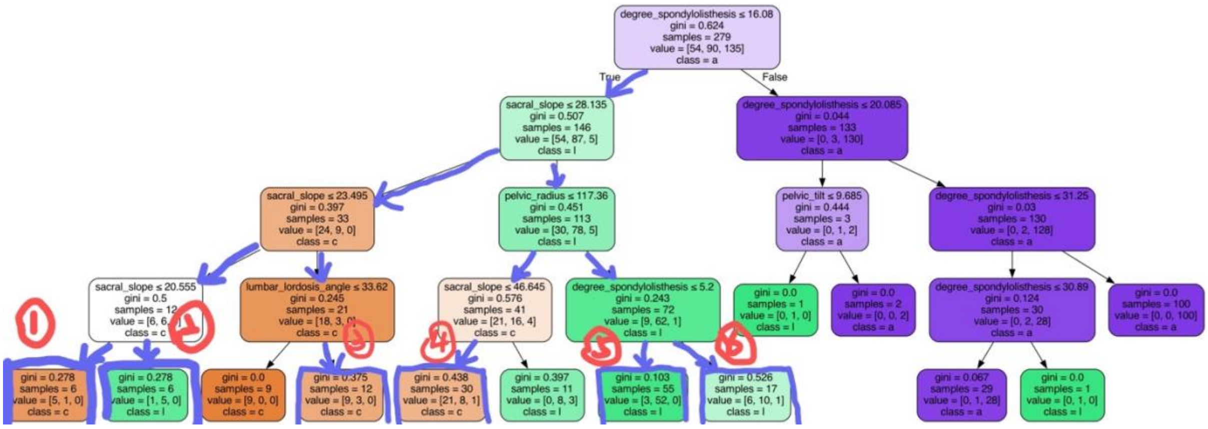
The confused decision paths are listed as below:

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



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

Vertebral Column dataset from the UCI Machine Learning Repository:
https://archive.ics.uci.edu/