**VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY UNIVERSITY OF SCIENCE**

**FACULTY OF INFORMATION TECHNOLOGY**



**LAB02 REPORT**

**SUBJECT: ARTIFICIAL INTELLIGENCE**

Decision Tree with scikit-learn

**Theory lecturers & instructors:** Mrs. Nguyen Ngoc Thao Mrs. Ho Thi Thanh Tuyen Mr. Nguyen Hai Dang

Mr. Le Ngoc Thanh

**Students:**

Dang Tien Dat 20127458

Table of Contents

[**I.** **Preparing the datasets:** 2](#_Toc102480884)

[II. Building the decision tree classifiers: 2](#_Toc102480885)

[1. Train/test – 40/60: 2](#_Toc102480886)

[2. Train/test – 60/40: 3](#_Toc102480887)

[3. Train/test – 80/20: 4](#_Toc102480888)

[4. Train/test – 90/10: 5](#_Toc102480889)

[**III.** **Evaluating the decision tree classifier:** 5](#_Toc102480890)

[1.Train/test – 60/40: 5](#_Toc102480891)

[2. Train/test – 40/60: 6](#_Toc102480892)

[3. Train/test – 80/20: 7](#_Toc102480893)

[4. Train/test – 90/10: 8](#_Toc102480894)

[**IV.** **The depth and accuracy of a decision tree:** 9](#_Toc102480895)

[**V.** **References:** 12](#_Toc102480896)

**Note:** *Before we proceed to write the functions to do the below requirements we need to import the necessary libraries (scikit-learn, numpy, matplotlib.pyplot, pandas, seaborn, ...) and data needed (connect-4.data) for this lab .*

*we will compile these libraries using the command:*

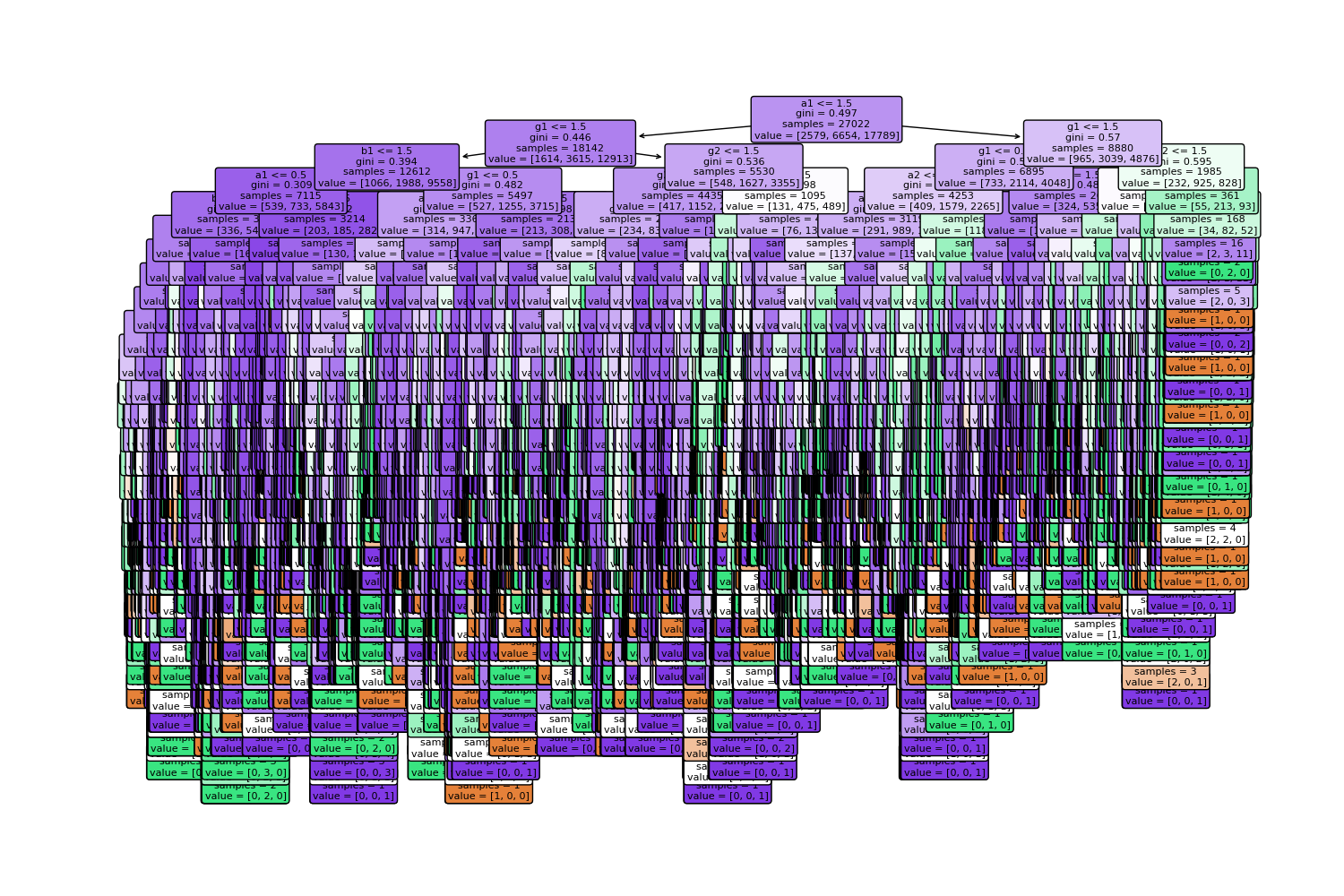
**pip install ‘library\_you\_need’ ex: pip install scikit-learn**

1. **Preparing the datasets:**

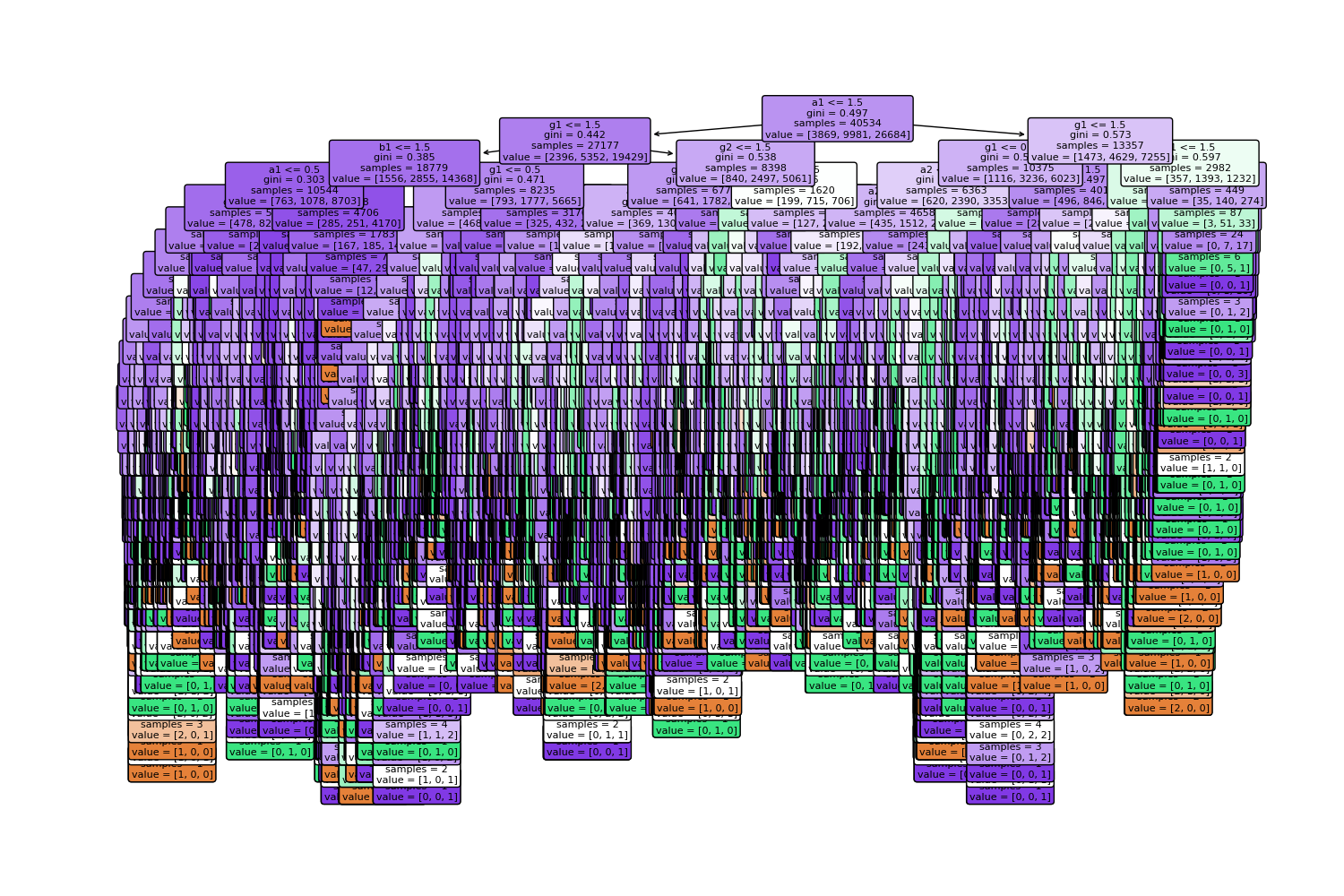
Using libraries to split data train\_test\_split(..,test\_size=testSize\*0.01, shuffle=True, stratify=y)

# Building the decision tree classifiers:

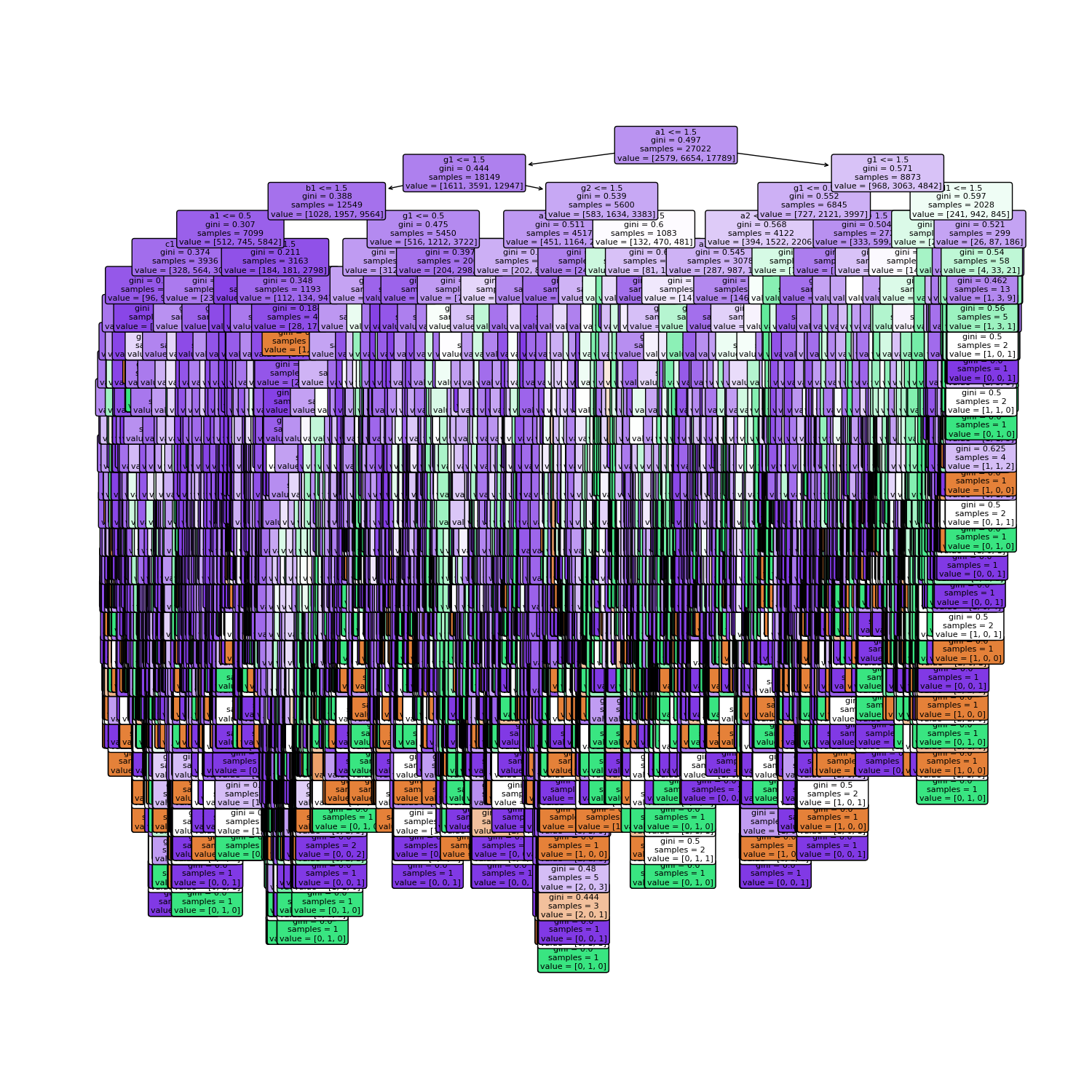
1. Train/test – 40/60:



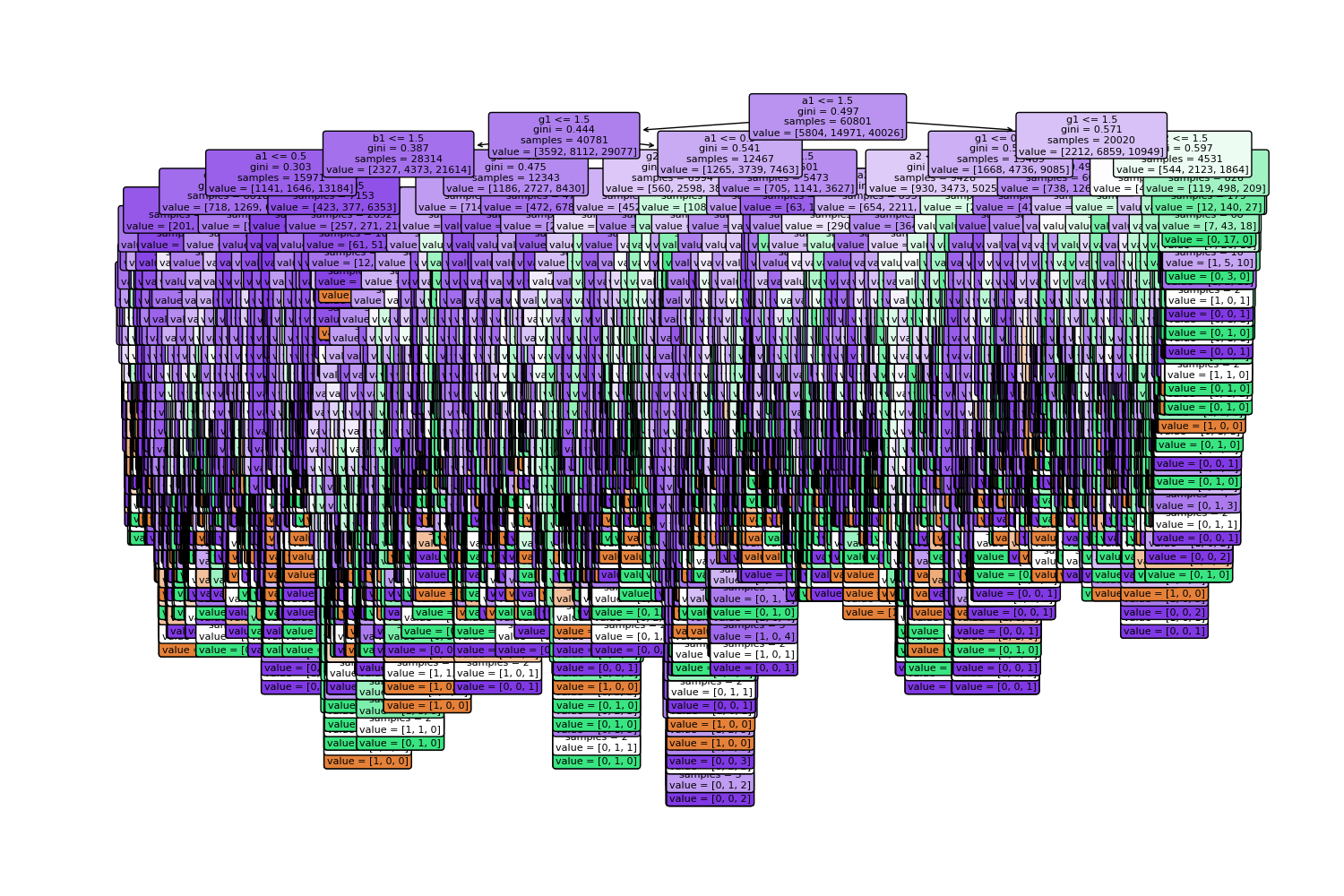
1. Train/test – 60/40:



1. Train/test – 80/20:



1. Train/test – 90/10:



1. **Evaluating the decision tree classifier:**

1.Train/test – 60/40:

* Decision tree classifier report:

Calendar

Description automatically generated

* Decision tree classifier confusion matrix:

Chart, treemap chart

Description automatically generated

2. Train/test – 40/60:

* Decision tree classifier report:

Calendar

Description automatically generated

* Decision tree classifier confusion matrix:

Chart, treemap chart

Description automatically generated

3. Train/test – 80/20:

* Decision tree classifier report:

Calendar

Description automatically generated

* Decision tree classifier confusion matrix:

Chart, treemap chart

Description automatically generated

4. Train/test – 90/10:

* Decision tree classifier report:

Calendar

Description automatically generated

* Decision tree classifier confusion matrix:

Chart, treemap chart

Description automatically generated

1. Interpret and comment:

* Interpret the classification report: Tell the accuracy of win, draw, loss (2,1,0), average macro, average weight
* Interpret the classification matrix: The rows and columns are true label and predicted label respectively (The prediction has the most accurate results when the true label and predicted label is at position [2,2] )
* Comment on the performances of those decision tree classifiers: If max\_deapth is None, the nodes will be expanded until there are only nodes and a balanced binary tree. If max\_deapth = 2, 3, 4, 5, 6, 7 then the tree will have a height corresponding to max\_deapth. Labels 2 will be higher than labels 0, 1 because the train sets are mainly label 2 (label win).

1. **The depth and accuracy of a decision tree:**

This task works on the 80/20 training set and test set:

* The decision tree for each max\_depth:

The max\_deapths correspond from left to right and top to bottom, respectively: None, 2, 3, 4, 5, 6, 7

** Diagram

Description automatically generated**

**Diagram

Description automatically generated Diagram, schematic

Description automatically generated**

**Diagram

Description automatically generated A picture containing diagram

Description automatically generated**

**Diagram

Description automatically generated**

* **Relationship of depth and accuracy by chart:**

**Chart, line chart

Description automatically generated**

* **Relationship of depth and accuracy by table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **max\_depth** | **None** | **2** | **3** | **4** | **5** | **6** | **7** |
| **Accuracy** | **0.77** | **0.660672** | **0.663558** | **0.674289** | **0.681024** | **0.695752** | **0.70752** |

* **Comment on the above statistics:**
* Accuracy only increases to a limit point and will no longer increase proportionally with max\_deapth
* Accuracy increases with the depth of the decision tree classifiers. ( As depth increases, accuracy increases).

1. **References:**
2. [**https://www.kaggle.com/code/haimfeld87/analysis-and-classification-of-mushrooms/notebook**](https://www.kaggle.com/code/haimfeld87/analysis-and-classification-of-mushrooms/notebook)
3. [**https://www.geeksforgeeks.org/graph-plotting-in-python-set-1/**](https://www.geeksforgeeks.org/graph-plotting-in-python-set-1/)
4. [**https://scikit-learn.org/stable/modules/tree.html**](https://scikit-learn.org/stable/modules/tree.html)
5. [**https://scikit-learn.org/stable/modules/generated/sklearn.tree.plot\_tree.html**](https://scikit-learn.org/stable/modules/generated/sklearn.tree.plot_tree.html)
6. [**https://scikit-learn.org/stable/modules/generated/sklearn.model\_selection.train\_test\_split.html?fbclid=IwAR0F1PpGJvVxvcb7jLiLbQB9y3GCNZcnsa2TAQUMh3lti6tWyO8IMe9Q9m4**](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html?fbclid=IwAR0F1PpGJvVxvcb7jLiLbQB9y3GCNZcnsa2TAQUMh3lti6tWyO8IMe9Q9m4)