

CSCI 184 HW 1 -- Part 2: Programming

Due Date: Apr. 22, 11:59pm 2024

All assignments MUST have your name, student ID, course name/number at the beginning of your documents.

Your homework MUST be submitted via Camino with the file format and name convention as follows:

For Question Answering part, you can either write by hand or type your answers, but please ensure your submission is **pdf** file with name “**HW#_Name.pdf**”.

For programming questions, please upload your code and supporting files in “**HW#_Name.zip**”.

If you have any questions, please don't hesitate to contact me :)

This assignment is totally 70 points while 30 points are for filling in the needed function in the `decision_tree.py` file. **You will need to write a report for your outputs of the following three questions.**

Problem: Implement a **fixed-depth decision tree algorithm**, that is, the input to the ID3 algorithm will include the training data and **maximum depth** of the tree to be learned. The code skeleton as well as data sets for this assignment can be found on Camino.

Data Sets: The MONK's Problems were the basis of a first international comparison of learning algorithms. The training and test files for the three problems are named `monks-X.train` and `monks-X.test`. There are six attributes/features (columns 2–7), and binary class labels (column 1). See `monks.names` for more details.

Visualization: The code skeleton provided contains a function **`render_dot_file()`**, which can be used to generate .png images of the trees learned by both **scikit-learn** and your code. See the documentation for **`render_dot_file()`** for additional details on usage.

- a. (Learning Curves, 15 points) For **depth = 1, ..., 10**, learn decision trees and compute the **average training and test errors on each of the three MONK's problems**. Make **three plots**, one for each of the MONK's problem sets, plotting training and testing error curves together for each problem, with tree depth on the x-axis and error on the y-axis.

Note: You need to write your own function to learn the tree and cannot use **scikit-learn's `DecisionTreeClassifier`** for this question.

- b. (Weak Learners, 15 points) For **monks-1**, report the visualized learned decision tree and the confusion matrix on the test set for depth = 1, 3, 5. You may use **scikit-learn's confusion matrix()** function [1].

Note: You need to write your own function to learn the tree and cannot use **scikit-learn's DecisionTreeClassifier** for this question.

		Classifier Prediction	
		Positive	Negative
Actual Value	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

- c. (scikit-learn, 10 points) For **monks-1**, use **scikit-learn's DecisionTreeClassifier** [2] to learn a decision tree using criterion='entropy' for depth = 1, 3, 5. report the visualized learned decision tree and the confusion matrix on the test set for depth = 1, 3, 5. You may use **scikit-learn's confusion matrix()** function [1].

[1] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html

[2] <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html>