

CMPT459: Data Mining, Spring 2021

Prof: Martin Ester

TAs: Arash Khoeini, Madana Krishnan Vadakandara Krishnan

Assignment 1 [total marks: 100]

The goal of this assignment is to implement a Decision Tree and to test it on a dataset to classify people into those who earn less than 50k and more than 50k based on their attributes. The adults dataset consists of 14 features (6 continuous and 8 categorical) and one class label. Provided data.zip file includes three files:

- *data.summary.txt*: information about the features
- *adult.data.csv*: training data
- *adult.test.csv*: testing data

- a) [25 marks] Present a pseudo-code for a simple Decision Tree with error reduction pruning:
- i) The information gain is used as a split criterion. [5 marks]
 - ii) The tree is grown deep, i.e. it is grown until all training examples corresponding to a leaf node belong to the same class. [5 marks]
 - iii) Works on categorical data. [5 marks]
 - iv) Works on numerical data. [5 marks]
 - v) Error reduction pruning using validation data. [5 marks]
- b) [75 marks] Implement your pseudo-code using Python. Your implementation should include the following functions (method signatures might be different based on the specific needs of your implementation):

- *grow(dataset) -> tree*: grows a deep tree on the given dataset and returns the tree object. [20 marks]
- *prune(dataset, tree) -> tree*: accepts a tree object and prunes it using the validation dataset. Returns the pruned tree. [15 marks]
- *test(dataset, tree) -> accuracy*: returns the accuracy of the given tree on the given dataset. [10 marks]

You should use the above methods to complete the following tasks:

1. Train and evaluate on the dataset *adult.data.csv* using 5-fold-cross-validation. Each time you grow a tree, you need to prune it before evaluation. You can leave 10% of the training data as validation data. Report the average accuracy. [15 marks]

2. Train one final tree on the dataset *adult.data.csv* and use it to predict samples in *adult.test.csv* and save outputs in a csv file. [10 marks]
3. You need to properly handle missing values. Explain briefly how you did that. [5 marks]

[IMPORTANT] Submit a file *[student-id].zip* which includes the following:

- A file *report.pdf* with the pseudocode and your answers for tasks 1. and 3.
- One and only one *.py* file which includes all your implemented functions and classes.
- A file *predictions.csv* with the predictions for the test data.
- A *requirements.txt* file, including all the required packages to run your code.
- *data/* directory with the datasets.

Running the *python* command on your *.py* file should reproduce all your reported results. You need to use relative paths for accessing your data through your code to make it runnable on all machines.

Deadline: The deadline is **23:59 pm on Feb 11th**. We accept late submissions up to 24 hours late but deduct 10% of the marks. You will lose all the marks for submissions after that.

Libraries: You can use libraries including math, numpy, scipy, random, etc. You **MUST** provide YOUR OWN implementation for the information gain calculation, decision tree growing, pruning and the methods to perform 5-fold-cross validation and predictions. These **MUST** be implemented from scratch i.e. not using scikit-learn libraries. You will be marked on the correctness of your implementation.