MP11

Added by Javari, Amin, last edited by Javari, Amin on Jul 15, 2016

Machine Problem 11 (**optional** - *30* points)

Purpose:

- Get deeper understanding and working experience of decision trees through implementation.

Instructions:

- This is an individual assignment, which means it is OK to discuss with your classmates and TAs regarding the methods, but it is not OK to work together or share code.
- Similar libraries or programs of frequent pattern mining algorithms can be found on-line, but you are prohibited from using these resources directly. This means you can not include external libraries, or modify existing programs, since the purpose of this MP is to help you go through frequent pattern mining step by step.
- You can use Java/C++/Python/Matlab as your programming language. No other languages are allowed.
- Copying source code from other students will give you 0 grade. We will run plagiarism detection software.

Requirements:

- For your answers in the Answer Document, you should include 1) the outputs 2) a brief explanation about the outputs; 3) answers to all questions in Question to ponder
- Put all your codes in a separate folder with the name NetId_MP11_codes. Do not use sub-folders inside this folder. All of your codes should have been successfully compiled before submission. Do not include files other than the codes you write. Put a single readme.txt file in the code folder to briefly describe the functionalities of your codes and how to run them.
- Your **PDF** submission file should be at the same level as your code folder. Compress these two together into a zip file, and name it MP11.netid.zip. Submit this zip file through Compass2g.

NOTE: This is an optional MP. You should submit your MP11.netid.zip file to MP11 submission link on Compass2G. The deadline for this MP is (8/4/16).

Input:

- In this MP, we use the car evaluation dataset from the UCI Machine Learning Repository. The car.names file enumerates the class categories as well as the non-class attributes. You should use car.data file to create your test and training sets. Each car has 4 classes : unacc, acc, good, vgood and is judged using 6 attributes: buying, maint, doors, persons, lug_boot, safety.

Required Outputs:

(a) [13, L3] (Step 1) Implement decision tree induction following the algorithm described in the slides. Use Information gain as the attribute selection method. Try out your implementation on the car evaluation dataset. Build your decision tree based on your training data and apply it to your test dataset to get predicted output classes. Use 10 fold cross validation as your validation method. Report the overall accuracy and true positive rate for each class.

Question to ponder A: Let's assume your input data has a small number of unknown values. Propose a method to handle those values; any reasonable choice will be good.

Question to ponder B: Let's assume your input data has some numerical attributes. How can numerical attributes be used in building decision trees?

(b) [6, L3] (Step 2) Use different attribute selection methods (Information gain, Gain Ration and Gini Index) to build your decision trees and compare the obtained decision trees with each other in terms of accuracy.

(c) [6, L3] (Step 3) Investigate the effect of decreasing the size of the training data. Use 10%, 40%, 70% and 100% of the training data in the experiments.