Project III – Decision Tree [6 points]

The goal of this project is to implement a Decision Tree DT based on the approach presented in the class. You will also learn how to put into practice the evaluation techniques and to think about the performance improvement procedures for DT such as pruning.

1. Buillding the DT (3 pts)

Building the DT learning algorithm with some basic functionalities for testing. The interface needs to be functional but you are free to choose what you like. Your DT should deal with data with the following features:

- an arbitrary number of continuous numeric-valued features,
- an arbitrary number of classes, and
- an arbitrary number of example training instances.

Your program should be able to handle data files such as the ones provided for this project and should provide basic evaluation metrics as well as general information such as the depth, the size and the shape for the DT it builds. The shape can be a print of the nodes at every layer (you may choose an appropriate format). In your implementation, you are only allowed to use basic programming tools such as math libraries but no specialized libraries for machine learning or data mining are allowed.

2. Car case mpg base [1 pts]

This case base is about the prediction of mpg of a car given some of its features such as cylinder and model year. Preparing the case base consists of coding numerically all the values. For example, you may encode the maker values as follows: *Europe* may be converted into 1, *America* into 2, and Asia into 3. You also need to propose and try **two** different encodings for the continuous values in the case base and report if these encodings affect the classification of the resulting DT (you need to think of different thresholds). Select randomly 10% of the cases as a test case base for your evaluation. What is the result (recall) on the training case base? What it he result on the test case base? Discuss your results with the regards with the quantity, quality of the data as well as possible limits of the DT learning algorithm.

3. Wisconsin Breast Cancer (WBC) data [2 pts]

This is a larger scale case base where you can experiment with a more real world situation. It is a medical database representing the classification of 569 patients into those with and without malignant breast tumors. You should provide training set accuracy, test set accuracy, and the size of the decision tree (number of nodes, max depth).

What is the DT accuracy on the training and testing case bases? Can you provide an explanation of the observed performance?

Would pruning help improve the performance of you DT with this case base? Describe through an example how you would prune and what pruning method you would use.

4. Possible extensions to your work [no credit, completely optional but recommended]

Here are some ideas for those who wish to go further with their work:

- Implement and test one or more pruning algorithms and compare the results. Implement a pre-pruning and a post pruning and try to see which one performs better. You may also explore if there are differences in terms of running time.
- Implement k-fold cross validation algorithm. Using the WBC case base compare the results of k-fold with the results of your static evaluation that you carried on fixed portions for test and training data.
- During the coming week we will see how to use scikit-learn to quickly implement some learning algorithms. Compare the performance of your DT with other learning algorithm such as the DT implemented in scikit-learn, Bayes net, and SVM. You may also link the measured performance differences with the theoretical divergences between the considered approaches to try to find a justification.