CS 180: Artificial Intelligence
Machine Problem 2: Decision Trees (Individual work OR by pairs)
Date Due: Wednesday, May 10, 2017

Problem Description

In this machine problem, you will experiment with the implementation of ID3 decision tree algorithm discussed in class. You can use C or Python to implement your program. Your code should be able to read in a dataset as a text file and produce training and test sets. Your decision tree program should be able to work on any dataset (do not hardcode attributes or values). As a sample dataset, your program should be able to build the decision tree for the PlayTennis example discussed in class.

Dataset

In evaluating your program for the report, you will be using the Tic-Tac-Toe-Endgame dataset that can be downloaded from the following website:

https://archive.ics.uci.edu/ml/datasets/Tic-Tac-Toe+Endgame

This dataset encodes the complete set of possible board configurations at the end of tic-tac-toe games, where "x" is assumed to have played first. The target concept is "win for x" (i.e., true when "x" has one of 8 possible ways to create a "three-in-a-row"). Below is a portion of the dataset. In each line, the first 9 values are 9 attribute (which are the board positions) values (x,o,b) and the last value is the classification (positive or negative).

x,x,x,x,o,o,x,o,o,positive

x,x,x,x,o,o,o,x,o,positive

x,x,x,x,o,o,o,o,x,positive

x,x,x,x,o,o,o,b,b,positive

x,x,x,x,o,o,b,o,b,positive

The program must print/display a trace of how the decision tree is generated. For each node of the decision tree, the trace should include information about each attribute tested and its information gain and the attribute that was chosen.

When the final decision tree is generated, print the tree in a way that is readable to the user. The visualization of your final decision tree output need not be part of your program. You can use another program to draw the decision tree. You will perform two experiments using your program. The details of the experiment are in discussed in the content of your report.

Written Report

You are expected to submit a written report which should include the following:

A brief discussion of what a decision tree learning is about as well as a description of your implementation of the algorithm. You should also describe the dataset together with the attributes and their possible values.

Experiment 1: Decision Tree Learning

For the first experiment, you will create a train and test set from the dataset. You will need to randomly separate the dataset into training and test sets. You will use this to perform n-fold cross-validation, where n=5. In other words, repeat this five times and average the results:

1. Create a random training and test set.

2. Use the training set to construct a decision tree.
3. Measure the performance of the tree on the test set.

Show the confusion matrix and accuracy of your decision tree on the dataset.

Experiment 2: Noisy training set

Create a new training set into which you introduce some noise. To do this, use again the first training and test date from Experiment 1 and randomly change the classification of 20 examples of the training set. Repeat the analysis of Experiment 1 for the new results (using only 1 train and test data) and compare the results to those for the original training set. Show the confusion matrix and accuracy of your decision tree on the dataset.

Analysis and Conclusion

Write an analysis of the performance of the decision tree of the dataset with regards to the two experiments along with appropriate conclusion.

Individual Contributions (Only for those who worked by pairs for this MP)

If you have worked by pair, please provide a brief summary of each individual group members contributions to the project.

Deliverables

Submit the source code of your program, input (train and test) and output files and your written report (in pdf format) via email. Please e-mail them as an attachment to crraquel@up.edu.ph. Use the following e-mail subject:

Subject: CS 180 MP2 – Raquel, Carlo (sample only, change this to your name)

The machine problem is due on Wednesday, May 10, 2017 at 12 midnight. You are only allowed to submit once. Put appropriate and complete documentation in your code. Late work will receive a deduction of 10% per day for a maximum of 1 week.

No form of academic dishonesty will be tolerated.

Your MP will be graded as follows:

Program	40%
Written Report	60%