Lab 1: Decision Trees

Each section has equal weight. Please discuss coding issues and workarounds in the discussion board, and post your code. Post your code on the class discussion board or agree with someone else's code. Alternatively, suggest fixes to other posted code. You are not graded on your coding ability, but on your analysis.

1) Run the example on the Indian diabetes dataset. Plot the decision tree for the gini measure with a max depth of 3.

2) Now apply the online example to data from class. Use the insurance fraud dataset we discussed in class. Split the data into a 10% train and 90% test set using random_state = 1. Create a decision tree with a max depth of 3 using a gini measure. Print the accuracy on the test set and the tree. Is this a good approach? Why or why not?

3) Create a decision tree on the same data with max depth of 3 and an entropy measure. Does the accuracy change? Does the tree change? Discuss which measure you think is better.

4) Now split the data into 70% train and 30% test using random_state = 1. Redo 2 and 3. Have the trees and accuracy changed? Are the trees more or less similar now? Discuss which split you think is better and why.

5) Evaluate how the accuracy changes with the depth of the tree with the 70-30 data. Look at the accuracy for a max depth of 1, 2, 3, … 10, 15, 20. Do you see underfitting? Do you see overfitting?

6) What variable provides the most information gain in the insurance fraud data (for the 70-30 split)?

7) Decision trees are a "white box" method. What do you observe about the insurance fraud data using decision trees?

8) We talked about 'clamping' in class. Should we have done clamping on the data before using a decision tree?

9) Read the paper Perner, 2013, on comparing decision trees. Does the proposed similarity measure find that your previous analysis is correct?

10) Are there problems with Perner, 2013, that you could improve upon? How about if there are different amounts of data in one tree? While Perner focuses on rules, what other measures could you use?