Data Mining and Machine Learning Homework #4: Tree based methods, bagging and boosting

Due on Monday, Apr 25, 2011, on paper. (if I am not in my office, you can slide it in under the door).

Format of the report: Explain clearly your methods and your results. Point out what you find interesting and try to make sense of it. Use tables and plots if they better represent your results and explain them. Please put your code and less important output in the appendix.

1. Design a simulation example to demonstrate that bootstrap aggregating (bagging) can improve the performance of tree method. The example can be concerned either with a classification problem or a regression problem. You can follow the computation procedure in Breiman's bagging paper (Sec 2.2 or 3.2) as you implement the bagging idea and do the comparison with tree.

2. Explore "The Insurance Company Benchmark" data, which is available from UCI KDD Archive website http://kdd.ics.uci.edu/databases/tic/tic.html. There was a competition for best prediction of this dataset in 2000! "The data consists of 86 variables and it includes product usage data and socio-demographic data derived from zip area codes. The data was collected to answer the following question: Can you predict who would be interested in buying a caravan insurance policy and give an explanation why?". The information about the variables is available at the data website.

   We now have learned a collection of learning methods which can be applied to this data set. For this exercise, you are required to experiment with tree, random forest and boosted tree. You are encouraged to compare with other methods but it is optional. Standard R packages for tree and random forest are rpart and randomForest. There are at least three packages available for booting: ada, gbm and mboost. You need to do a bit research and find out which one you want to use.

   The dataset is particularly challenging because the number of customers who bought the insurance was extremely small compared to the size of the customer pool. One could blindly predict everyone not buying insurance and still achieve a high prediction accuracy. But that is not a useful model to answer our question. How much has your model improve the prediction comparing with the "blind prediction"? You are welcome to try any idea to further improve your results.