Assignment 4:  
Decision Tree Learning

Deadline: Monday, May 7, 2012

This assignment will contribute 10 points to your final grade.

Description

In this assignment, you will continue with the exercise we performed in class on the spambase dataset\(^1\). The dataset consists of 57 attributes signifying word frequencies appearing in emails, along with a class label signifying whether the email was spam (1) or valid (0). During our class exercise, we saw how a decision tree can be created for spam detection using this data. You also became familiar with the pruning process (cost-complexity pruning) used by the CART algorithm. To continue, you will explore a different algorithm to do the pruning and compare the prediction performance with a few different methods.

Task 1: Data preparation

Begin this study by revisiting what we did in class. This would involve importing the data into R, splitting it into a training and test set, building a decision tree using rpart, and then pruning based on the 1SE error rates corresponding to the different pruned trees. Let \(T_{full}\) be the full tree (without pruning) and \(T_{pruned.CC}\) be the pruned tree.

Task 2: Pruning using estimated error

In this task, you will prune \(T_{full}\) using the method of estimated errors. To do so, first implement a function `confidence.interval` that returns the confidence interval for the true error rate, given a confidence level and observed error rate in a Bernoulli process.

```r
confidence.interval <- function(miss, samples, c) {
  #...
  f = miss / samples
  N = samples
  z such that Pr(X > z) = (1 - c)/2
  # return a vector with the extremes of the interval
}
```

The function takes the number of misclassified samples (\(miss\)), the total number of samples (\(samples\)) and a confidence level (\(c\)). The confidence interval is calculated using:

\[
p = \frac{f + \frac{z^2}{2N} \pm z\sqrt{\frac{f}{N} - \frac{f^2}{N^2} + \frac{z^2 N}{4N^2}}}{1 + \frac{z^2}{N}},
\]

where \(f = miss/samples\), \(N = samples\) and \(z\) is such that \(Pr(X > z) = (1 - c)/2\), \(X\) being a Normally distributed variable with zero mean and unit variance. Look into the `qnorm` function to compute the value of \(z\). The lower bound of the interval is obtained by using the minus operation, and the upper bound by using the plus operation.

All features of the tree required in this task are available in the `frame` dataframe of the tree object. If \(t\) is the tree object, then `t$frame` can be used to access this frame. The important pieces of information (for this task) in this data frame are:

\(^1\)http://archive.ics.uci.edu/ml/datasets/Spambase
1. **the row numbers**: rpart always creates a binary split. It numbers nodes using a simple binary tree indexing method: the two children of node $n$ are nodes $2n$ and $2n + 1$

2. **var**: the attribute used for splitting at the node; the value is <leaf> for leaf nodes

3. **n**: the number of training instances sitting at that node

4. **dev**: the number of misclassifications if the majority label (from the instances sitting at that node) is used for prediction

5. **yval**: the majority label of the training instances sitting at that node

Implement the pruning algorithm based on the pessimistic error estimation technique discussed in class. The pseudo code of a recursive implementation is as follows:

**Method name: EE.Pruning**

**Input:** Tree $t$, node number $nd$

**Output:** Pruned tree $t$

1. Call $EE.Pruning$ on left (node number $2 \times nd$) and right child (node number $2 \times nd + 1$) if they are not leaf nodes

2. For each leaf node in subtree rooted at $nd$
   
   (a) compute the confidence interval of the true error rate using confidence.interval (you will be using the dev and n attributes in $t$'s $frame$); use a value of 0.5 for the confidence level $c$

   (b) use the upper limit of the interval as an estimate of the true error rate

3. Combine all estimated error rates in the previous step in proportion to the size of the subset residing in the leaf nodes; call it $er_{subtree}$ (see lecture slides)

4. Compute the confidence interval of the true error rate for node $nd$ and use the upper limit as the estimated true error $er_{node}$

5. If $er_{node} - er_{subtree} \leq 0.03$ (we will consider a 3% difference to be negligible), then prune the tree $t$ by removing the entire subtree rooted at $nd$; this is relatively easy to do using the snip.rpart function

6. Return $t$

Call $EE.Pruning$ on $T_{full}$ and $nd = 1$. Lets call the resulting tree $T_{pruned.EE}$. Create plots of $T_{full}$, $T_{pruned.CC}$ and $T_{pruned.EE}$ and include in your report. Discuss your observations on how $T_{pruned.CC}$ differs from $T_{pruned.EE}$. Do the pruned trees help you understand how most spam emails are detected (at least in this data set)?

**Task 3: Evaluating predictions**

Perform predictions on the test data using the three tree models. In addition, perform prediction using a SVM classifier (use default parameter values) and a 5-nearest neighbor classifier. Create a table showing the test error, precision and recall in detecting spam, for the five methods (three trees, SVM, 5-NN). Do you observe higher precision for the pruned trees; if so, discuss why? Also, discuss the performance of the five methods based on the precision-recall values.

Next create ROC curves corresponding to each model. For this, you will require the predictions to be given as probabilities, instead of class labels. Read the documentation of the corresponding prediction functions (predict.svm, knn, and predict.rpart) to determine how probabilities on the class labels can be retrieved. The probabilities on the is_spam=1 class will be required for the task. The points for the ROC curve for a given method can be obtained as follows.
1. Sort the test instances (including the class labels) based on decreasing values of the probability on the is_spam=1 class

2. Set $r = 1$

3. For test instance 1 to $r$ (after the sorting)

   (a) determine $TP = \text{number of instances where true class label} = 1$

   (b) determine $FP = \text{number of instance where true class label} = 0$

   (c) add the point ($FP/Neg, TP/Pos$) to the ROC curve; here $Neg$ is the number of test instances (all of it) with class label = 0, and $Pos$ is that with class label = 1

4. Set $r = r + 1$ and repeat from Step 3; stop when $r$ is greater than number of test instances

Using the above method, plot ROC curves of each of the five methods. Include this in the report. Discuss your observations on the performance of the methods based on the ROC curve.

**Report**

Your report is **NOT** a collection of statements corresponding to the discussion asked from you. It must be complete in itself, primarily including the problem you are solving, description of the data, the method employed, relevant R code, the experiments performed, the results you observed, informative (with labels, captions, and the remaining glitter!) and readable plots, and most importantly, a discussion along the lines of what is asked in the tasks. You are strongly encouraged to include your own observations and discussions outside of what is asked. Tell me a story!

**Grading**

This assignment will be graded out of 100 points. The points split is as follows:

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>35</td>
<td>25</td>
<td>30</td>
</tr>
</tbody>
</table>

The points you receive in each task will depend on how accurately you complete the task. In addition, your R code should be clean and well-commented. Sticking to directions is **important**! Additional explorations (outside of what is asked in the tasks) is most welcome. The report will be graded based on content, structure, formatting (I do not want to see half of a page empty because the plot does not fit there — reorganize!!), spelling, grammar, and other usual characteristics of a good scholarly document.

**Submission**

Submit the pdf report (not the Latex or Lyx source) and the R code into Blackboard by **Monday, 7 May 2012, 11:59 PM**. Refer to the late policy on the course website.

You must work alone on this assignment.