

Supplemental Data

Boosted Decision Tree Analysis of SELDI Mass Spectral Serum Profiles

Discriminates Prostate Cancer from Non-Cancer Patients

by Qu et al.

The regular decision tree provides insights into the underlying data structure, which may lead to important discovery. However, the predictive power may not be as good as other learning approaches, such as the neural networks and the support vector machines.

In recent years several authors have indicated that using voting methods with decision trees can improve the predicting power tremendously. One example of voting methods is Breiman’s bagging method (Breiman, 1996), that is fitting the decision tree model many times on randomly resampled observations (bootstrap subsamples) and combine the decision trees using simple voting. Another example of voting methods is Freund and Schapire’s boosting method (Freund and Schapire, 1997), the AdaBoost algorithm, that is fitting the decision tree model many times on weighted observations and combine the decision trees using weighted voting. In both bagging and boosting, the combined classifier has better performance than each of the individual base decision trees. The boosting approach is generally more accurate in the test samples than the bagging approach (Schapire and Freund, 1998).

1. Boosted decision stump classifier

The i th training sample is denoted by $(y_i, Z_{i1}, \dots, Z_{ip})$, where y_i stands for the class (label) of observation i ($i = 1, \dots, N$) and Z_{ij} is the intensity of the j th peak of i th observation. In our case, we use $p = 124$ peaks.

Assume that there are two classes, class one (e.g. noncancer) is denoted by $y_i = 1$, and class two (e.g. cancer), is denoted by $y_i = -1$. We use decision “stumps”, as the base classifiers: each of these trees has only one split, using one peak. A decision stump usually is

a weak classifier, with rather high error rate. However, the combined stumps using weighted vote is expected to be a very accurate classifier.

2. The decision stump

A decision stump can be denoted by (Z, c) where Z is a peak, selected from the $p = 124$ peaks, and c is a threshold. This stump has two leaves, the left one contains the training samples with intensity of peak Z less than or equal to the threshold c , and the right leaf contains all other samples. If most the samples in the left leaf is, say cancer, then the samples with $Z \leq c$ will be classied as cancer. The classifier is denoted as $f(x)$, where x is a Boolean variable, “ $Z \leq c$ ”, and $f(x)$ takes value from $\{-1, 1\}$: $f(x_i) = 1$ if the i th train sample is classified as class one; and $f(x_i) = -1$ if the i th sample is classified as class two. The sample is misclassified if $y_i f(x_i) = -1$.

For the left leaf, $Z \leq c$, or $x = \text{“true”}$, let n_{11} and n_{21} be the numbers of observations with $y_i = 1$ and $y_i = -1$, respectively, i.e.,

$$n_{11} = \sum_{i=1}^N I\{(y_i = 1) \& (Z \leq c)\}, \quad n_{21} = \sum_{i=1}^N I\{(y_i = -1) \& (Z \leq c)\}, \quad (1)$$

where $I\{\text{statement}\}$ is the indicator function, which equals 1 if the statement is true, 0 otherwise. Similarly, let n_{12} and n_{22} be the numbers of observations for $y_i = 1$ and $y_i = -1$, respectively, and $Z > c$, i.e.,

$$n_{12} = \sum_{i=1}^N I\{(y_i = 1) \& (Z > c)\}, \quad n_{22} = \sum_{i=1}^N I\{(y_i = -1) \& (Z > c)\}. \quad (2)$$

Then the log likelihood for this multinomial model is

$$\log L = \sum_{u=1}^2 \sum_{v=1}^2 n_{uv} \log(p_{uv}), \quad (3)$$

where p_{uv} is evaluated by $n_{uv}/(n_{1v} + n_{2v})$. The peak Z and its threshold c are obtained by maximizing the log likelihood.

3. The decision stump for weighted observations

In boosting method, one creates an ensemble of decision stumps on weighted observations. Assume that we assigned weight w_i to the i th observation y_i so that the sum of the weights is N . For these weighted data, we still use equation (3) to find the split (Z, c) , except the counts n_{11}, n_{21}, n_{12} and n_{22} should be modified to incorporate the weights. Equation (1) becomes

$$n_{11} = \sum_{i=1}^N w_i I\{(y_i = 1) \& (Z \leq c)\} \quad n_{21} = \sum_{i=1}^N w_i I\{(y_i = -1) \& (Z \leq c)\},$$

and equation (2) becomes

$$n_{12} = \sum_{i=1}^N w_i I\{(y_i = 1) \& (Z > c)\}, \quad n_{22} = \sum_{i=1}^N w_i I\{(y_i = -1) \& (Z > c)\}.$$

4. The AdaBoost algorithm

The basic idea of boosting methods is to construct an ensemble of base classifiers on weighted observations. For the first round, we use equal weights to all observations, i.e., $w_i = 1$ for $i = 1, \dots, N$. We classify all the samples, and in the next round we increase the weights for the misclassified observations in the previous round, while decreasing the weights for the correctly classified observations. Therefore the next decision stump will focus on the samples misclassified by the previous stump. We repeat this procedure again and again until a certain number of stumps have been created.

Here is the algorithm with modification (Freund and Schapire, 1997; Hastie, Tibshirani, and Friedman, J., 2001. pp 301, and Friedman, Hastie, and Tibshirani, 2000).

1. For the first round, using equal weights, $w_i = 1$ ($i = 1, \dots, N$).
2. For $m = 1$ to M :
 - Construct a decision stump $f_m(x)$ for the training data with weights w_i ,

- Compute error rate

$$err_m = \frac{1}{N} \sum_{i=1}^N w_i I(y_i f_m(x_i) = -1)$$

- Compute confidence $\alpha_m = \log\{(1 - err_m)/err_m\}$.
- Update weights, set

$$w_i \leftarrow w_i \cdot \exp\{\alpha_m \cdot I(y_i f_m(x_i) = -1)\}, \quad i = 1, \dots, N,$$

and normalize the weights: $\sum_i w_i = N$.

3. Use the linear combination

$$f(x) = \sum_{m=1}^M \alpha_m f_m(x)$$

as the final combined classifier.

The combined classifier $f(x)$ is a weighted majority vote of the M base classifiers. For the i th sample, the m th base classifier $f_m(x_i) = 1$ if it is classified as class one, and $f_m(x_i) = -1$ if it is classified as class two. The contribution of the m th decision stump to the final vote is either α_m , if it votes for class 1, or $-\alpha_m$, if it votes for class 2. Therefore, if the total vote is positive, i.e.,

$$f(x_i) = \sum_{m=1}^M \alpha_m f_m(x_i) > 0, \quad i = 1, \dots, N, \quad (4)$$

that means the majority votes for class 1, then the sample is classified as class 1, and it is classified as class 2, otherwise ($f(x_i) \leq 0$). This is a weighted majority vote because α_m are not equal.