CONSISTENCY FOR A SIMPLE MODEL OF RANDOM FORESTS

Leo Breiman
Technical Report 670
STATISTICS DEPARTMENT
UNIVERSITY OF CALIFORNIA AT BERKELEY

September 9, 2004

0) Introduction

Random Forests is a classification algorithm with a simple structure--a forest of trees are grown as follows:

> *1) The training set is a bootstrap sample from the original training set.*
>
> *2) An integer mtry is set by the user, where m mtry is less than the total number of variables. At each node, mtry variables are selected at random and the node is split on the best split among the selected mtry. The tree is grown to its maximal depth.*
>
> *3) In regression, as a test vector* ***x*** *is put down each tree it is assigned the average values of the y-values at the node it stops at. The average of these over all trees in the forest is the predicted value for* ***x.*** *The predicted value for classification is the class getting the plurality of the forest votes.*

Random Forests is an accurate algorithm having the unusual ability to handle thousands of variables without deletion or deterioration of accuracy. The difficulty is that although the mechanism appears simple, it is difficult to analyze.

A heuristic analysis is presented in this paper based on a simplified version of RF denoted RF0. The results from RF0 support the empirical results from RF. RF0 regression is consistent using a value of mtry that does not depend on the number of cases N The rate of convergence to the Bayes rule depends only on the number of strong variables and not on how many noise variables are also present.. This also implies consistency for the two class RF0 classification. The analysis also illuminates why RF is able to handle large numbers of input variables and what the role of *mtry* is.

Unlike single trees, where consistency is proved letting the number of cases in each terminal node become large (Breiman et.al [1984]) RF trees are built to

have a small number of cases in each terminal node. The driving force behind consistency is completely different. As the reader will see below it mostly resembles an adaptive nearest neighbor method that uses a smart distance measure. This concept appeared first in a novel and interesting technical report by Jin and Jeon[2002].

In this report, first the simple model RF0 is described. Then the computations are done from which consistency follows. The role of *ntry* is seen. Then remarks follow which point up the adaptive nature of the RF metric.

1. ***The RF0 Model***

A. *Some empirical results*

i) Omitting the bootstrapping of the training set has very little effect on the error rate.

ii) RF can have multiple cases in a single terminal node--i.e. all cases are of the same class. Using the medians of randomly selected variables to get to one case per terminal node has no effect on the error rate.

iii) Splitting without using the class labels i.e. splitting at the medians of the values of a randomly selected variable at a node significantly increases the error rate.

These empirical results will be built into our simple model. Assume that the training data $T=\{(y_n,\mathbf{x}_n),n=1,...,N\}$ is i.i.d and the input vectors **x** have M coordinates. The M variables in RF0 are assumed uniform and independent. (this is not a serious restriction and could be replaced by the assumption that the joint density is bounded above and below)

Each variable x_m is associated with a probability $p(m)$ which sum to one. The conditional probability of each class, given **x** is linear in **x** and depends equally on the "strong" variables only. If the sample size is large, then for strong variables, the Gini criterion is minimized by a split at the center of the node. The splits by weak variables are random.

The trees are constructed like this:

> *i) At each node, a single variable is selected with the mth variable having probability $p(m)$ of being selected.*
>
> *ii) If the variable is strong, the split is at the midpoint of the of values if the selected variable at the node.*

iii) If the variable is weak, the split is at a random point along its values in the node.

Assume the resulting tree is balanced, with each terminal node resulting from $L = \log_2 N$ splits. Assume also that there is only one case per terminal node (this can be inforced by splitting on medians in the lower branches of the tree) Bootstrapping the training set is not done. The analysis will show that the essential ingredient in consistency is the randomized selection of splitting variables at the nodes.

The iid random vector Θ used to construct each tree is a collection of independent variables which select one item out of M items with probability $p(m)$. Denote this collection for one tree by θ, other trees will be constructed using independent copies of θ. For a single tree denote the rectangle containing $\mathbf{x}_n$ by $R(\mathbf{x}_n, \theta)$.

Let $\mathbf{x}$ be a test point and drop it down the tree. If it comes to rest in $R(\mathbf{x}_n, \theta)$ assign it the response value y_n. As trees are built using other values of θ, let

$$Q(\mathbf{x},\mathbf{x}_n)=E_\theta(I(\mathbf{x}\in R(\mathbf{x}_n,\theta)$$

Note that the sum of $Q(\mathbf{x},\mathbf{x}_n)$ over n is one for all $\mathbf{x}$.

The forest estimate for y($\mathbf{x}$) is

$$\hat{y}(\mathbf{x})=\sum_n Q(\mathbf{x},\mathbf{x}_n)y_n \qquad (1)$$

It can be shown that 1-$Q(\mathbf{x},\mathbf{z})$ is a Euclidean distance between $\mathbf{x}$ and $\mathbf{z}$. Therefore, (1) shows that RF is a classical nearest neighbor algorithm. As will be seen, 1-$Q(\mathbf{x},\mathbf{z})$ is a smart and adaptive metric. It can be shown that the classical conditions for consistency are met. More will be done to get an estimate of how fast the Bayes rate is approached.

Assume there are S strong variables and W weak variables, corresponding to S "large" values of $p(m)$ equal to p_S ,W small values equal to p_W, and that $E(y|\mathbf{x})$ depends only on the S strong variables (S and W are unknown).

The Bayes estimate for y($\mathbf{x}$) is $E(y|\mathbf{x})$. We will show that

$$E_T \int (\hat{y}(\mathbf{x}) - E(y|\mathbf{x}))^2 \, \mathbf{dx} \qquad 1)$$

goes to zero as the sample size increases, where E_T is the expectation with respect to the training set., and that the rate of convergence depends only on S and not on W.

This proof is for regression. But , as Devroye et.al. show, the rate of convergence to the Bayes rate for the two class classification problem is faster than the square root of 1)

2 *Decomposition into Variance and Bias*

$$\hat{y}(\mathbf{x}) = \sum_n Q(\mathbf{x}, \mathbf{x}_n)(y_n - E(y|\mathbf{x}_n)) + \sum_n Q(\mathbf{x}, \mathbf{x}_n) E(y|\mathbf{x}_n)$$

Thus, with the E_T expectation implicit in all terms,

$$\int (\hat{y}(\mathbf{x}) - E(y|\mathbf{x}))^2 \, \mathbf{dx} \le 2 \int [\sum_n Q^2(\mathbf{x}, \mathbf{x}_n)(y_n - E(y|\mathbf{x}_n))]^2 + 2 \int [\sum_n Q(\mathbf{x}, \mathbf{x}_n)(E(y|\mathbf{x}) - E(y|\mathbf{x}_n))]^2$$

Using familiar terminology the first term is referred to as variance V(N),, the second as bias B(N). Assume that $E((y_n - E(y|\mathbf{x}_n))^2 \le v$. Then the variance term is initially bounded by

$$v \int \sum_n Q(\mathbf{x}, \mathbf{x}_n)^2 d\mathbf{x} \qquad 2)$$

3. *Bounding the Variance*

The bound on the variance is made possible by the randomization. Recall that $Q(\mathbf{x}, \mathbf{x}_n) = E_\theta(I(\mathbf{x} \in R(\mathbf{x}_n, \theta))$**.** so

$$Q(\mathbf{x}, \mathbf{x}_n)^2 \le E_\theta E_{\theta'} I(\mathbf{x} \in R(\mathbf{x}_n, \theta) \cap R(\mathbf{x}_n, \theta'))$$

where θ, θ' are independent outcomes of Θ. Then V(N) is bounded by:

$$\sum_n E_\theta E_{\theta'} P(R(\mathbf{x}_n,\theta)\cap R(\mathbf{x}_n,\theta')).$$

The rectangles are the product of m intervals $J(m,\theta), J(m,\theta')$ with

$$\prod_m |J(m,\theta)|=1/N \quad \prod_m |J(m,\theta')|=1/N.$$

The problem is to get a bound on V(θ,θ')= $\prod_m |J(m,\theta)\cap J(m,\theta')|$.

The length

$$|J(m,\theta)| = 2^{-n(m,\theta)}$$

where $n(m,\theta)$ is the number of cuts by the mth variable leading to the formation of the rectangle. This holds for the strong variables. It also holds for the expectation of the length for weak variables.

The sum of the $n(m,\theta)$ equals L. For the number of variables M large enough, assume that the $\{n(m,\theta)\}$ are independent and have a binomial distribution with L trials and probability $p(m)$ of success. Clearly:

$$|J(m,\theta)|\cap J(m,\theta')|\le 2^{-\max(n(m,\theta),n(m,\theta'))}$$

and

$$2^{-\max(n(m,\theta),n(m,\theta'))}=2^{-n(m,\theta)-[n(m,\theta')-n(m,\theta)]^+}$$

Multiplying gives:

$$V(\theta,\theta')\le \frac{1}{N}\prod_{,m} 2^{-[n(m,\theta')-n(m,\theta)]^+}$$

The right hand side can be evaluated exactly (see appendix). The relevant result is that if *p(m)L* is large, then

$$E_\theta E_{\theta'} 2^{-[n(m,\theta')-n(m,\theta)]^+} \approx 1/\sqrt{\pi p(m) L}$$

If $p(m)L$ is small, then

$$E_\theta E_{\theta'} 2^{-[n(m,\theta')-n(m,\theta)]^+} \approx e^{-p(m)L}$$

Recall that there are S strong variables with $p(m)=p_S$ and W weak with $p(m)=p_W$. Then

$$V(N) \leq (\pi p_S L)^{-S/2} \exp(-W p_W L) \qquad 3)$$

3) ***Bounding the Bias***

Bound the bias term (before integration) as

$$[E_\theta \sum I(\mathbf{x} \in R(\mathbf{x}_n,\theta)(E(y|\mathbf{x})-E(y|\mathbf{x}_n))]^2$$

$$\leq E_\theta [\sum I(\mathbf{x} \in R(\mathbf{x}_n,\theta)(E(y|\mathbf{x})-E(y|\mathbf{x}_n))]^2$$

$$\leq E_\theta [\sum I(\mathbf{x} \in R(\mathbf{x}_n,\theta)[(E(y|\mathbf{x})-E(y|\mathbf{x}_n)]^2$$

Use the approximation

$$E(y|\mathbf{x}_n)-E(y|\mathbf{x}) \approx (\mathbf{x}_n-\mathbf{x})\nabla E(y|\mathbf{x})$$

where ∇ is the gradient operator. By assumption $\nabla E(y|\mathbf{x})$ has zero components for all weak variables. Then

$$[(\mathbf{x}_n-\mathbf{x})\nabla E(y|\mathbf{x})]^2 \leq \|\mathbf{x}_n-\mathbf{x}\|_S^2 \|\nabla E(y|\mathbf{x})\|_S^2$$

where the subscript S indicates--sum over only those components corresponding to strong variables. Let $G=\sup_{\mathbf{x}} |\nabla E(y|\mathbf{x})\|_S^2$. The bias bound is now:

$$G E_\theta [\sum I(\mathbf{x} \in R(\mathbf{x}_n,\theta) \|\mathbf{x}_n-\mathbf{x}\|_S^2].$$

Consider integrating on $\mathbf{x}$. The mth component of $\mathbf{x}$ is constrained to be in an interval of length $|J(m,\theta)|$ containing the mth component of $\mathbf{x}_n$. Assume the mth component is a strong variable. Then

$$E(x_m - x_{m,n})^2 \le |J(m,\theta)|^3/3.$$

Recalling that the product of the lengths of the intervals is 1/N, gives the bias bound

$$B(N) \le \frac{G}{3} E_\theta \sum_S |J(m,\theta)|^2$$

Since $|J(m,\theta)|^2 = 2^{-2n(m,\theta)}$, taking expectations gives

$$B(N) \le \frac{G}{3} S \exp(-.75 p_S L)$$

4. ***Optimizing***

The upper bounds for $B(N)$ and *V(N)* contain the unknown parameters p_S and p_W. To see the effect of mtry the next step is the minimization of the sum of the bounds with respect to these parameters. Actually there is only one unspecified parameter since $Wp_W + Sp_S = 1$. Problem: find p_S to minimize

$$(\pi p_S L)^{-S/2} \exp(-W p_w L) + \frac{G}{3} S \exp(-.75 p_S L)$$

Skipping the algebraic details, an approximate solution is $p_S \approx L/S(S+.75)$ and the minimizing value is

$$\frac{1}{N^{.75/(S+.75)}}$$

NOTE The rate of approach to the Bayes risk depends only on the number of strong variables. This explains why RF does so well when there are many noise variables.

5. ***Optimizing Using Mtry.***

In practice, with an largely unknown data set, we have only vague ideas about the size of S and what to take for the $\{p(m)\}$ that will give near

optimal performance . The advice given to users is to try several values of *mtry* and use the one that gives lowest cross-validated (oob) error rate.

We examine the performance of this procedure when there are S strong variables and W weak ones. Choose *mtry* variables from among the M at random with replacement--then choose the one that gives the best split. Assume all strong variables give equal results on the split. The same for all weak variables. If the selection is all weak, then choose one at random to split on. If there is more than one strong variable selected in *mtry*, select one at random to split on.

The probability that a variable in the W group will be chosen for a split is:

$$\frac{1}{W}(W/M)^{mtry} \qquad 5)$$

In the S group, the probability is

$$\frac{1}{S}(1-(W/M)^{mtry}) \qquad 6)$$

At the optimal *mtry* the optimal p_S will be equal to 6). Assume W large compared to S, then

$$mtry \approx M/S(1+(4/3)S)$$

Thus looking for an optimal mtry will give values of the probabilities for using the variables that are close to optimal computed as without knowing the size of S and W. Note also that the optimal value of *mtry* does not depend on N. Unlike kNN nearest neighbor where k must be increased with N to get optimal convergence, one value of *mtry* does for all sample sizes N.

6 Remarks

As simplistic as the above model is, it goes far to clarify the behavior of Random Forests. We see that it is an adaptive nearest neighbor azlgorithm where

i) The randomization works to reduce the variance.
ii) It adapts to the loss function by having the narrowest widths in the terminal nodes corresponding to the largest components of the loss function.
iii) It automatically adapts to the sample size.
iv) The optimal value of *mtry* does not depend on the sample size.

A questionable part of RF0's definition is the assumption that with a linear loss function, the best cut by a strong variable will be at the center of the node. This is true only for nodes containing a large sample. Otherwise the cut will depend more strongly on the individual values of the $\{y_n\}$ and the independence property used to derive 2) is not applicable. Removing this assumption makes a consistency proof at least an order of magnitude more difficult and involves ideas centered around the VC dimension.

I am indebted to the Lin-Jeon technical report for suggestive insights that led to some of the above conclusions.

References

Breiman, L. [2001] Random Forests, *Machine Learning* **45** 5-32.
Breiman, L. Friedman, J. Olshen, R. Stone, C[1984] "Classification and Regression Trees"
Devroye, L, Gyorfi, L. Lugosi, G. "A Probabilistic Theory of Pattern Recognition"
Lin, Y. and Jeon, Y. [2002] Random forests and Adaptive Nearest Neighbors. Technical Report 1005, Dept. Statistics, Univ. Wisconsin

Appendix ***(****surely this must be known somewhere)*

Set

$$\varphi(z)=Ez^{n(m,\theta')-n(m,\theta)}$$
$$=(p(z+1/z)+1-p)^L$$

where $p=2p(m)$

$$P(n(m,\theta')-n(m,\theta)=k)=\frac{1}{2\pi i}\oint\varphi(z)z^{-k-1}dz$$

so

$$\sum_{k\geq 0}2^{-k}P(n(m,\theta')-n(m,\theta)=k)=\frac{1}{2\pi i}\oint\varphi(z)z^{-1}/(1-(1/2z))dz$$

Substituting $z=e^{i\theta}$ in the right hand side and doing simplification gives

$$\frac{1}{\pi}\int\frac{(2-\cos\theta)}{(5-4\cos\theta)}(1-2p\sin^2(\theta/2))^L\,d\theta$$

The results stated above for large and small p can be derived from this integral.