SLAC PUB-2336
May 1979
(M)

A TREE-STRUCTURED APPROACH TO NONPARAMETRIC MULTIPLE REGRESSION

Jerome H. Friedman*
Stanford Linear Accelerator Center
Stanford, California 94305/USA

Introduction

In the nonparametric regression problem, one is given a set of vector valued variables $\underline{X}$ (termed carriers) and with each an associated scalar quantity Y (termed the response). This set of carriers and associated responses $\{Y_i,\underline{X}_i\}$ $(1\leq i\leq N)$ is termed the training sample. In addition (usually at some later time), one is given another set of vector valued variables $\{\underline{Z}_j\}$ $(1\leq j\leq M)$ without corresponding responses and the problem is to estimate each corresponding response using the values of its carriers and the training sample. That is:

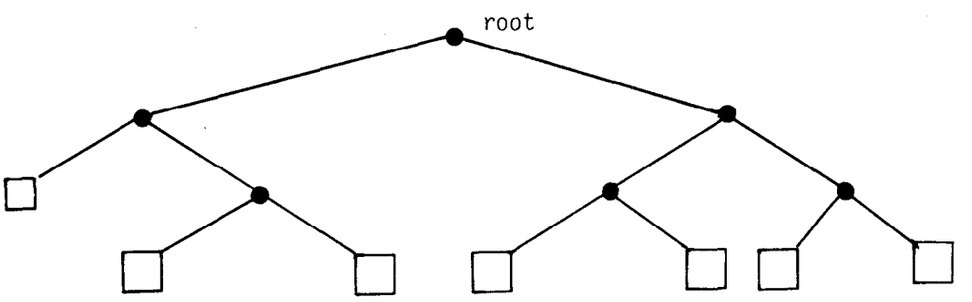
$$\hat{Y}(\underline{Z}_j) = \text{Rule}\,[\underline{Z}_j,\ \{Y_i,\underline{X}_i\}\ (1 \leq i \leq N)]\ (1 \leq j \leq M).$$

The rule for performing the estimation is usually referred to as the model or regression function.

In addition to this basic predictive role, there are usually other data analytic goals. One would like the model to reveal the nature of the dependence of the response on the respective carriers and lend itself to easy interpretation in a similar manner to the way parametric models often do via the fitted values of their parameters.

Binary Regression Tree

The nonparametric regression models discussed herein are based on binary trees. A binary tree is a rooted tree in which every node has either two sons (nonterminal nodes) or zero sons (terminal nodes). Figure 1 illustrates a simple binary tree.



● Nonterminal node

□ Terminal node

Figure 1

*This work is part of a joint research effort by Leo Breiman, Jerome Friedman, Lawrence Rafksy and Charles Stone. Work partially supported by the Department of Energy under contract number EY-76-C-03-0515.

(Presented at Smoothing Techniques for Curve Estimation Workshop, University of Heidelberg, Heidelberg, West Germany, April 1-4, 1979)

For these models, each node t represents:

1) a subsample S_t of the training sample,
2) a subregion R_t of the carrier data space,
3) a linear model $L_t(\underline{X}) = \underline{A}_t \cdot \underline{X} + B_t$ to be applied to $\underline{X} \in R_t$.

(For the models discussed in this report, the subsample S_t, represented by node t, is just the set of training vectors that lie in its corresponding subregion R_t.)

In addition, each nonterminal node represents:

4) a partitioning or splitting of R_t into two disjoint subregions $R_{l(t)}$ and $R_{r(t)}$

$(R_{l(t)} \cup R_{r(t)} = R_t$ and $R_{l(t)} \cap R_{r(t)} = 0)$

and a corresponding partitioning of S_t into two disjoint subsets $S_{l(t)}$ and $S_{r(t)}$.

The binary regression tree is defined recursively: let t_o be the root node and

S_{t_o} = entire training sample

R_{t_o} = entire carrier data space

$L_{t_o}(\underline{X})$ = linear (least squares fit) of Y on $\underline{X}$ using S_{t_o}.

Let t be a nonterminal node with left and right sons l(t) and r(t) respectively. Then

$R_{l(t)}$ and $R_{r(t)}$ are the subregions defined by the partitioning of t,

$S_{l(t)}$ and $S_{r(t)}$ are the subsamples defined by the partitioning of t.

The linear models associated with the left and right sons are derived from the parent model by modifying the dependence on one of the carriers J_t:

$$L_{l(t)} = L_t + a_{l(t)} X(J_t) + b_{l(t)}$$
$$L_{r(t)} = L_t + a_{r(t)} X(J_t) + b_{r(t)}. \qquad (1)$$

To construct the model one then needs:

1) a training sample $\{Y_i, \underline{X}_i\}$ $(1 \le i \le N)$
[This allows the definition of the root node R_{t_o}, S_{t_o}, $L_{t_o}(\underline{X})$],

2) a splitting rule which consists of
   a) a prescription for partitioning R_t into $R_{l(t)}$ and $R_{r(t)}$ ($S_{l(t)}$ and $S_{r(t)}$),
   b) a prescription for updating the model (choosing values for J_t, $a_{l(t)}$, $a_{r(t)}$, $b_{l(t)}$, $b_{r(t)}$, to get $L_{l(t)}$ and $L_{r(t)}$ (thereby defining the two son nodes of t),

3) stopping (termination) rule for deciding when not to split a node, thereby making it a terminal node.

Splitting Rule

The situation at a node that is to be split is depicted in Figure 2.

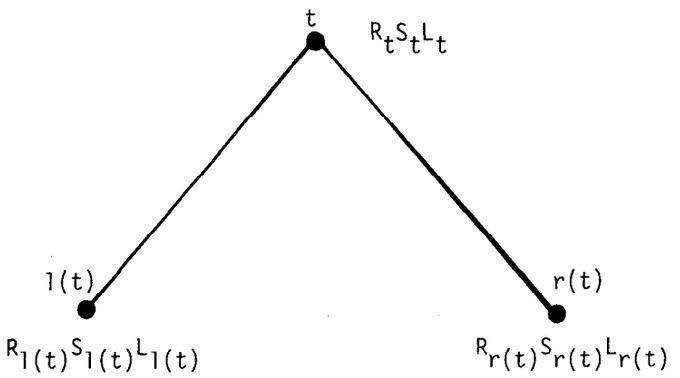


Figure 2

One has the subregion (subsample) and model associated with the parent [R_t (S_t) and L_t] and one would like to define the corresponding quantities for the two sons so as to best improve the fit of the model to the training sample. Let

$$\hat{Q}_t = \sum_{i \in S_t} [Y_i - L_t(\underline{X}_i)]^2 \tag{2}$$

be the empirical residual sum of squares associated with the parent and $\hat{Q}_{l(t)}$ and $\hat{Q}_{r(t)}$ be the corresponding quantities for the two sons. Then

$$\hat{I}_t = \hat{Q}_t - \hat{Q}_{l(t)} - \hat{Q}_{r(t)} \tag{3}$$

is an estimate of the improvement as a result of splitting node t. A reasonable goal is then to choose the partitioning so as to maximize $\hat{I}_t$ subject to possible limitations such as continuity and computability.

Since $L_{l(t)}$ and $L_{r(t)}$ are linear models (on $R_{l(t)}$ and $R_{r(t)}$) and $R_{l(t)} \cup R_{r(t)} = R_t$, one can think of $[L_{l(t)}, L_{r(t)}]$ as a piecewise-linear model on R_t. From (1)

$$\begin{aligned} L_{l(t)} - L_t &= a_{l(t)} X(J_t) + b_{l(t)} \\ L_{r(t)} - L_t &= a_{r(t)} X(J_t) + b_{r(t)} \end{aligned} \tag{4}$$

so that we want to choose the parameters on the RHS of (4) to best fit the residuals

$$r_i = Y_i - L_t(\underline{X}_i) \qquad (i \in S_i) \tag{5}$$

to the model associated with the parent node.

Consider the residuals (5) as a function of each of the carriers X(j) in turn. If $L_t(\underline{X})$ provides an adequate description of the dependence of the response on X(j), then there should be little structure in the values of the residuals when ordered on X(j). That is, a plot of r versus X(j) would resemble that of Figure 3a.

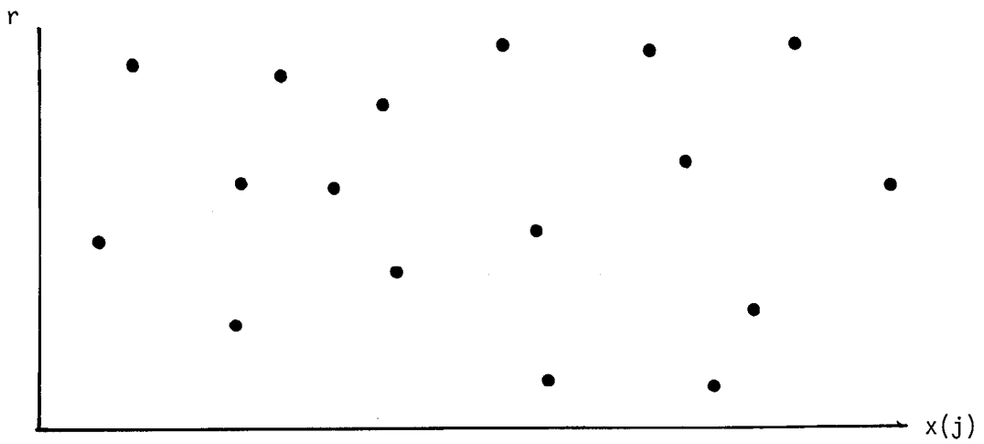


Figure 3a

On the other hand, considerable structure in the residuals (e.g., Figure 3b) would indicate that $L_t(\underline{X})$ does not provide an adequate description of the dependence of the response on X(j).

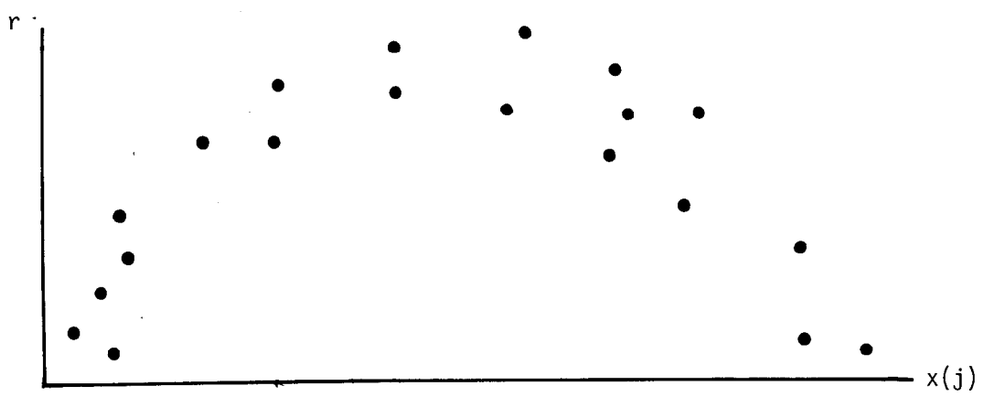


Figure 3b

The example of Figure 3b indicates a possible quadratic dependence of the residuals (and hence the response) on carrier X(j).

These observations motivate our splitting procedure. Each carrier X(j) is considered in turn. For each, a (univariate) continuous piecewise linear model is fit to the residuals from $L_t(\underline{X})$. That is, the model

$$\begin{aligned} r &= a_{lj}\,[X(j) - s_j] + b_j \qquad X(j) \le s_j \\ r &= a_{rj}\,[X(j) - s_j] + b_j \qquad X(j) > s_j \end{aligned} \tag{6}$$

is fit to $\{r_i, X_i(j)\}$ $(i \in S_t)$ by minimizing

$$\begin{aligned} Q_j = &\sum_{i=1}^{k} [r_i - a_{lj}\,(X_i(j) - s_j) - b_j]^2 \\ &+ \sum_{i=k+1}^{\#S_t} [r_i - a_{rj}\,(X_i(j) - s_j) - b_j]^2 \end{aligned} \tag{7}$$

with respect to j, a_{lj}, a_{rj}, b_j, and s_j. Here the $X_i(j)$ are ordered in ascending value and $X_k(j) \le s_j$ and $X_{k+1}(j) > s_j$. That is, the best (in the least squares sense) continuous piecewise linear fit (with s_j as the knot) is made to the residuals versus each carrier X(j) and the best fit (over the carriers) is chosen.

Let the optimum values found for j, a_{lj}, a_{rj}, b_j, s_j be represented by J, a_l, a_r, b and s respectively. These solution values are used to both define the partitioning and update the model:

$$\begin{aligned} &\text{For } \underline{X} \in R_t\text{:} \\ &\quad \text{If } X(J) \le s, \text{ then } \underline{X} \in R_{l(t)} \\ &\quad \text{If } X(j) > s, \text{ then } \underline{X} \in R_{r(t)} \\ &\quad L_{l(t)}\,(\underline{X}) = L_t(X) + a_l\,[X(J) - s] + b \\ &\quad L_{r(t)}\,(\underline{X}) = L_t(\underline{X}) + a_r\,[X(J) - s] + b\,. \end{aligned} \tag{8}$$

If the model associated with the parent node is

$$L_t\,(\underline{X}) = \sum_{j=1}^{p} A_t(j)\,X(j) + B_t$$

then from (8), the corresponding quantities for the son nodes are:

$$\begin{aligned} A_{l(t)}(j) &= A_{r(t)}(j) = A_t(j) \quad j \ne J \\ A_{l(t)}(J) &= A_t(J) + a_l \\ A_{r(t)}(J) &= A_t(J) + a_r \\ B_{l(t)} &= B_t - a_l\,s + b \\ B_{r(t)} &= B_t - a_r\,s + b. \end{aligned} \tag{9}$$

Thus, the models associated with the left and right sons differ from the parent and each other only in their dependence on carrier J, and the constant terms are adjusted for continuity at the split point s.

After the split is made and the model updated for the two son nodes, the above procedure is applied recursively to $l(t)$ and $r(t)$ and their sons and so on until the nodes meet a terminal condition. This stops the splitting making terminal nodes. Starting with the root, this recursive procedure then defines the entire regression tree.

Stopping (Termination) Rule

The recursive splitting described above cannot continue indefinitely. At some point, the cardinality of the subsample $\#(S_t)$ will be too small to reliably estimate the parameters for defining the splitting and updating the model. Thus, a sufficient condition for making a node terminal is that the size of its subsample is too small to continue splitting.

Using this condition as the sole one for termination, however, can cause serious overfitting. Basically, a split should not be made if it is not worthwhile. That is, it does not improve the model fit. The quantity $\hat{I}_t$ (3) is an estimate of the improvement in the fit as a result of splitting node t. This quantity is always positive, indicating that the empirical residual sum of squares will always improve as a result of choosing the optimum splitting. However, since the empirical residual sum of squares is an optimistically biased estimate of the true residual sum of squares from the model, a positive value for $\hat{I}_t$ does not guarantee a positive value for the true improvement I_t. A more reasonable criterion would be:

If $\hat{I}_t > k$ accept split at t and continue, otherwise make t a terminal node.

The quantity k is a parameter of the procedure, the interpretation of which is discussed below. Although lack of sufficient fit improvement (as estimated by $\hat{I}_t$) is a necessary condition for making t a terminal node, it is not sufficient. It is possible that a particular split, although not yielding much improvement itself, can make it possible for further splitting to make dramatic improvements. This would be the case, for example, if there were substantial interaction effects between pairs or sets of carriers. A sufficient condition for making a node terminal would be if its split and all further splits of its descendants yield insufficient empirical improvement. This is illustrated in Figure 4.

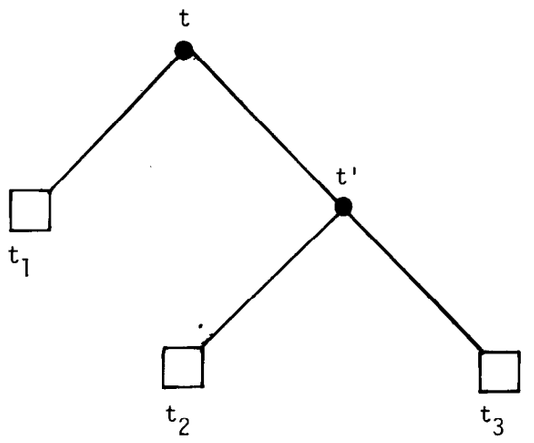


Figure 4

Here node t is split forming son nodes t_1 (which subsequently becomes terminal) and t'. Right son t' is further split forming nodes t_2 and t_3 which become terminal. The improvement associated with node t (and all further splits) is then defined to be

$$\hat{I}_t = \hat{Q}_t - \hat{Q}_{t_1} - \hat{Q}_{t_2} - \hat{Q}_{t_3} \tag{10}$$

That is the difference between the empirical residual sum of squares at node t and the sum of those associated with all terminal descendants of t. A reasonable condition for making t a terminal node is then

$$\text{If } \hat{I}_t \leq 2\ k \text{ make t terminal, otherwise accept split at t.} \tag{11}$$

The factor of two on the RHS of the inequality comes from the fact that two splits were required to form these three terminal nodes and this introduces even more optimistic bias than just one split. The condition (11) can be rewritten

$$\text{If } \hat{Q}_t + k \leq \hat{Q}_{t_1} + k + \hat{Q}_{t_2} + k + \hat{Q}_{t_3} + k \tag{12}$$

make t terminal,

Otherwise, accept split at t.

This suggests associating a cost C_t with each node t of the tree as follows:

$$\begin{aligned} &\text{If t is terminal} \quad C_t = \hat{Q}_t + k \\ &\text{If t is nonterminal} \quad C_t = \sum_{i \in t} C_{t_i} \end{aligned} \tag{13}$$

where the summation is over all terminal descendants of t. The decision to make a node terminal or not is then taken so as to minimize this cost. Note that if both sons of t [l(t) and r(t)] are terminated according to this prescription, then

$$\sum_{i \in t} C_{t_i} = C_{l(t)} + C_{r(t)} \,. \tag{14}$$

This suggests the following "bottom-up" recombination procedure for terminating the regression tree. First, the splitting procedure is applied as far as possible, terminating only for insufficient subsample cardinality. The nonterminal nodes of the resulting tree are then each considered in inverse order of depth. (The depth of a node is the number of nodes in the path from it to the root.) At each such node, the following termination rule is applied:

$$\underline{\text{If}}\ \hat{Q}_t + k \leq C_{l(t)} + C_{r(t)}$$

$$\underline{\text{then}}\ \text{make t terminal and } C_t = \hat{Q}_t + k \qquad (15)$$

Otherwise accept split at t and $C_t = C_{l(t)} + C_{r(t)}$.

This bottom-up recombination procedure insures that a node is made terminal only if its splitting and all possible further splitting yields insufficient improvement to the fit of the model, as determined by the improvement threshold parameter k.

This bottom-up recombination algorithm can be more easily understood intuitively by considering the following optimization problem. Let $\mathcal{J}$ be the set of all possible trees obtained by <u>arbitrarily</u> terminating the splitting procedure of the previous section. Let $T \in \mathcal{J}$ be one such tree and define its size $|T|$ to be the number of its terminal nodes. Let $\hat{Q}(T)$ be the empirical residual sum of squares associated with the regression model defined by T. The optimization problem is to choose that tree $T_k \in \mathcal{J}$, such that $\hat{Q}(T_k) + k|T|$ is minimum (breaking ties by minimizing $|T|$). The quantity k is a positive constant called the complexity parameter and T_k is said to be the optimally terminated tree for complexity parameter k. The complexity parameter is the analogue for this procedure to the smoothness parameter associated with smoothing splines or the bandwidth parameter associated with kernal estimates. Since $\hat{Q}(T_k)$ is monotone decreasing with increasing $|T_k|$, the value of k limits the size of the resulting optimally terminated tree T_k. Larger values of k result in smaller trees.

It can be shown (Breiman and Stone, 1977) that the bottom-up recombination procedure described above is an algorithm for solving this optimization problem where the complexity parameter k is just the improvement threshold parameter of that procedure. Thus, although motivated heuristically, that procedure is seen to have a natural interpretation in terms of generating optimally terminated trees T_k.

The complexity parameter k is the only parameter associated with this model. Ideally, its value should be chosen to minimize the true residual sum of squares $Q(T_k)$ associated with the model. This quantity is, of course, unavailable since only the training sample is provided. One could apply crossvalidation (e.g., see Breiman and Stone, 1977) or bootstrapping (Efron, 1977) techniques to obtain a less biased estimate of $Q(T_k)$ than $\hat{Q}(T_k)$. These estimates could be performed for various values of k and the best one chosen based on those estimates. However, this procedure is quite expensive computationally and not always reliable. Fortunately, a simple graphical procedure

allows one to obtain a reasonable estimate for a good value of the complexity parameter.

It can be shown (see Breiman and Stone, 1977) that for $k' > k$, $T_{k'}$ is a subtree of T_k (i.e., $T_{k'} \subset T_k$). To obtain $T_{k'}$, one simply applies the bottom up recombination procedure to T_k using the value k'. For $k' >> k$, $T_{k'}$ will likely be much smaller than T_k, while for k' only slightly larger than k the two trees will probably be identical. One can determine the smallest value of k' that will cause $T_{k'}$ to be smaller than T_k. For each nonterminal node $t \in T_k$, one has from (15)

$$\hat{Q}_t + k > \sum_{i \in t} (\hat{Q}_{t_i} + k) \tag{16}$$

where the summation is over all terminal descendants of t. If this were not the case, the node t would have been terminal in T_k. One can associate with each nonterminal node the complexity parameter value $\bar{k}_t$ that would cause it to become terminal. From (16) one has

$$\bar{k}_t = \frac{\hat{Q}_t - \sum_{i \in t} \hat{Q}_{t_i}}{|t| - 1} \tag{17}$$

where $|t|$ is the number of terminal descendants of t. The minimum value of $\bar{k}_t$ over all nonterminal nodes of T_k is the smallest complexity parameter value k' that reduces the size of the regression tree. That is,

$$k' = \min_{t \in T_k} \bar{k}_t. \tag{18}$$

Clearly, one can re-apply this procedure to $T_{k'}$ to determine the smallest complexity parameter value k'' (and the associated tree $T_{k''}$) that will cause $T_{k''}$ to be smaller than $T_{k'}$, and so on. Therefore, starting with T_k one can repeatedly apply this procedure to find all optimally terminated trees associated with complexity parameter values larger than k. Clearly, there are, at most, $|T_k|$ such trees. This entire series of trees can be obtained from T_k without re-applying the partitioning procedure and thus can be computed quite quickly. In particular, if one uses the partitioning procedure to obtain the regression tree for k=0, all optimally terminated trees for all possible complexity parameter values can be obtained with little additional effort.

Consider the collection of all such optimally terminated trees. As $|T_k|$ becomes larger $\hat{Q}(T_k)$ becomes smaller. A plot of $\hat{Q}(T_k)$ versus $|T_k|$ usually resembles that represented in Figure 5a.

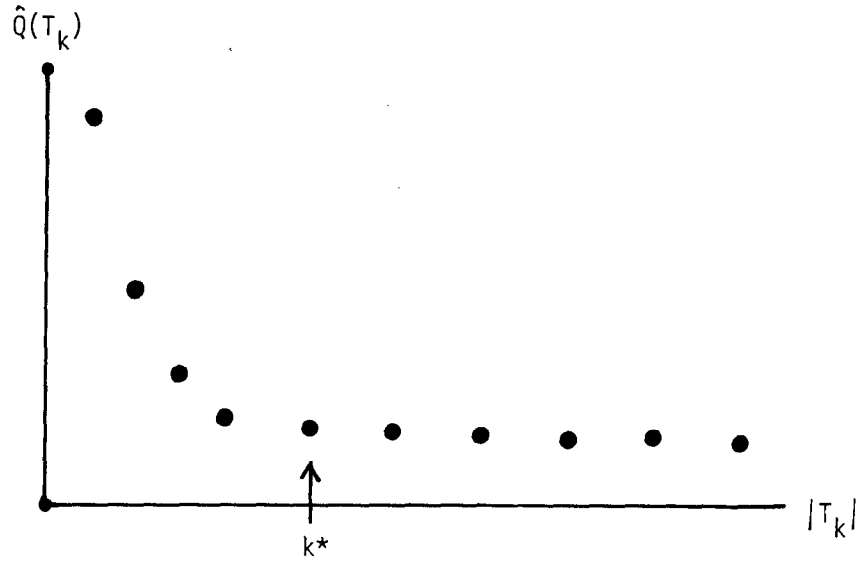


Figure 5a

There is usually a rapid decrease in the empirical residual sum of squares for the first few splits, followed by a very slow decrease with successive splits. The true residual sum of squares $Q(T_k)$ from the model tends also to decrease rapidly for the first few splits, followed by a slower decrease reaching a minimum, and then slightly increasing for even further splitting. This is illustrated in Figure 5b.

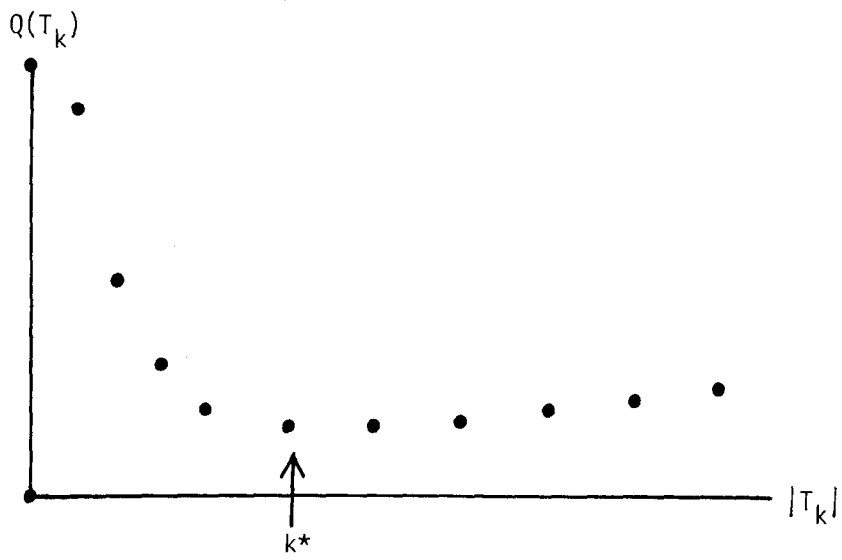


Figure 5b

The increase of $Q(T_k)$ for large $|T_k|$ is a result of oversplitting, which causes increased variance to be associated with the parameter estimates. The tree T_{k*} asso-

ciated with the value k* that minimizes $Q(T_k)$ is the desired regression tree. Comparing Figures 5a and 5b, one sees that a reasonable estimate of k* can be obtained from $\hat{Q}(T_k)$ versus $|T_k|$ by choosing that value at which the decrease in $\hat{Q}(T_k)$ for increased $|T_k|$ fails to be substantial as judged from earlier decreases. Since $Q(T_k)$ versus $|T_k|$ is highly asymmetric about $|T_{k*}|$, it is wise to choose a value slightly beyond this point since slight oversplitting is much less damaging (in terms of true residual sum of squares) than undersplitting. Since $Q(T_{k*})$ is at a minimum value, values of k reasonably close to k* will cause $Q(T_k)$ to differ little from $Q(T_{k*})$. Thus, the precise value obtained for the estimate is not crucial. A good estimate for the optimum complexity parameter can thus be obtained by simple inspection of a plot of $\hat{Q}(T_k)$ versus $|T_k|$ for the collection of optimally terminated trees T_k.

The Model

The model $L_T(\underline{X})$, represented by a binary regression tree T, can be represented as

$$L_T(\underline{X}) = \sum_{t' \in T} L_{t'}(\underline{X}) \quad 1(\underline{X} \in R_{t'}) \tag{19}$$

with the sum over all terminal nodes $t' \in T$. The submodel $L_{t'}(\underline{X})$ associated with each terminal node t' is linear, having the form

$$L_{t'}(\underline{X}) = \sum_{j=1}^{p} A_{t'}(j)\, X(j) + B_{t'}. \tag{20}$$

Although the parameters $A_{t'}$ and $B_{t'}$ appear linearly in (19), the global model is far from linear (unless the tree has only one node) since the regions $R_{t'}$ are determined adaptively from the training data.

By construction, the regions $R_{t'}$ associated with the terminal nodes are mutually exclusive so that for any set of carrier values $\underline{X}$ there is only one non-zero term in summation (19). Owing to the binary tree representation of the model, it is possible to determine which term will be non-zero for a given $\underline{X}$ without explicitly evaluating all of the terms in the summation. At each nonterminal node t of the tree, the split coordinate J_t and the split point s_t are stored. For any set of carrier values $\underline{X}$, the tree can be simply searched to find its corresponding terminal region. At each nonterminal node visited (starting with the root), $X(J_t)$ is compared to s_t to determine which son node to next visit:

If $X(J_t) \leq s_t$: visit left son

Otherwise: visit right son.

The region $R_{t'}$ associated with the first terminal node t' so visited is the one containing $\underline{X}$, and the value of its associated model $L_{t'}(\underline{X})$ is the estimated response of the global model [the non-zero term in (19)]. This search procedure is illustrated in Figure 6.

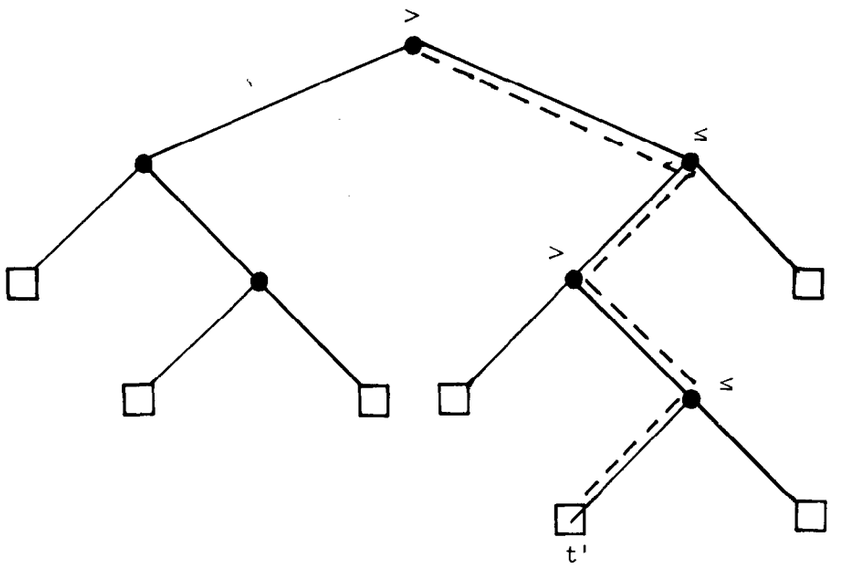


Figure 6

Speculative Splitting

The procedure described above for splitting each node, forming its two son nodes, is greedy in the sense that it tries to find that partitioning that maximizes the immediate improvement of the fit of the model to the training data represented by the node. This would be the best partitioning under the assumption that the son nodes are to be terminal and there will be no further splitting. However, for most nonterminal nodes, recursive application of the procedure accomplishes considerable further splitting.

Ideally, one would like to find the optimum sequence of cuts for improving the model fit. It is not always the case that the split that yields the best immediate improvement is the first in the best sequence of splits. Greedy strategies usually produce good solutions to optimization problems but seldom produce the optimum solutions. Finding the optimum regression tree is equivalent to binary tree optimization which is known to be NP-complete, thus requiring super polonomial computation time.

In the regression tree context, this situation arises when there are interaction effects between carriers. For example, if there is a strong interaction between carriers I and J, a single split on the Ith or Jth carrier will not significantly improve the model fit, but a split on I followed by one on J (or vise versa) will result in substantial improvement. However, a totally greedy strategy will probably fail to make that first cut (on I or J), preferring instead to cut on another carrier that yields more immediate improvement. Ultimately, the interaction will be detected by the procedure unless both the interactions and carrier designs are completely symmetric. However, the power of the procedure will be enhanced if these situations can

be detected and the proper sequence of splits is made immediately.

To this end, we augment the splitting procedure at each nonterminal node, described earlier, by the following procedure. For each coordinate j, provisionally divide the subsample represented by the node at the median of X(j). The two subsamples so created are each independently fit to a complete p-variate linear model. The empirical residual sum of squares resulting from this independent piecewise linear fit is then minimized over all coordinates and the result $\hat{Q}(2p+2)$ is compared to that obtained by the univariate continuous piecewise linear procedure $\hat{Q}(3)$, described earlier. If the optimum coordinate is the same for both procedures or if

$$\hat{Q}(3) \leq \frac{\#(S_t) - 3}{\#(S_t) - 2(p+1)} \hat{Q}(2p+2), \qquad \#(S_t) > 2(p+1) \tag{21}$$

then the split is made and the model updated in the usual manner, as described earlier. Here $\#(S_t)$ is the cardinality of the subsample represented by the node. If these conditions are both not met, then the following splitting strategy is employed. The coordinate that yielded the minimum $\hat{Q}(2p+2)$ is chosen as the split coordinate for the node. The split point is determined by provisionally splitting this coordinate at several (~10) equally spaced quantiles and finding the point that yields the best independent p-variate piecewise linear fits. However, the model augmentation parameters a_l, a_r, and b (9) are all set to zero so that there is no change in the model and, thus, no improvement in the model fit as a result of this split. This split is thus purely speculative in that by itself it results in no model improvement, but it should help to define good subsequent splits on each of its sons.

Example

In order to gain insight into the application of the partitioning procedure and the resulting regression tree model, we apply it to a computer generated training sample. Artificial rather than actual data is used so that the resulting regression tree model can be evaluated in the light of the known true underlying model. The training sample was created by generating 200 random carrier points $\{\underline{X}_i\}$ $(1 \leq i \leq 200)$ in the unit six-dimensional hypercube, $\underline{X}_i \in (0,1)^6$. Associated with each such vector valued carrier was a response value Y_i evaluated as

$$Y_i = 10 \sin [\pi X_i(1) X_i(2)] + 20 [X_i(3) - 1/2]^2 \tag{22}$$
$$+ 10 X_i(4) + 5 X_i(5) + 0 X_i(6) + \epsilon_i .$$

The set $\{\epsilon_i\}$ $(1 \leq i \leq 200)$ were generated as iid standard normal. For this example, the response has no dependence on one of the carriers [X(6)], a purely linear dependence on two others [X(4) and X(5)], an additive quadratic dependence on one [X(3)], and a nonlinear interaction dependence on two more [X(1) and X(2)].

The results of applying the regression tree analysis to this training sample, $\{Y_i, \underline{X}_i\}$ $(1 \leq i \leq 200)$, are summarized in Figures 7 and 8. The average response value is 14.3 with variance 27.1. The true mean squared error (MSE) of the best global linear least squares fit is $\sigma^2 = 7.25$, while for the regression tree this value is $\sigma^2 = 2.35$. The true intrinsic variance resulting from the noise term (ϵ) is, of course, $\sigma_I^2 = 1.0$. Figure 7 plots both the empirical MSE (from the training sample itself, solid circles) and the true MSE (open squares) as a function of tree size $|T_k|$, for all of the optimally terminated trees T_k. The value of the complexity parameter k associated with each tree is indicated above its corresponding solid circle. Inspection of Figure 7 shows, for example, that a complexity parameter value of k=0 yields a tree with 30 terminal nodes, an empirical MSE of 0.8, and a true MSE of 2.4. A value of k = 20, on the other hand, yields a tree of 10 terminal nodes, with empirical MSE 1.7, and a true MSE of 2.6. A value of $k \geq 433$ causes the tree to degenerate to solely the root node and the corresponding model is then just the global linear least squares fit. The general behavior of both the empirical and true MSE's as a function of $|T_k|$ is seen to generally correspond to that depicted in Figures 5a and 5b.

By inspecting the plot of the empirical MSE's versus $|T_k|$ (open circles) before calculating the corresponding true values (open squares), the 14 terminal node tree corresponding to k = 14 was chosen as an estimate for the optimum tree. After calculating the true MSE's, one sees that the best tree would have been the 12 terminal node tree associated with k = 16. However, any choice in the range $8 \leq |T_k| \leq 17$ is seen to be nearly as good from the point of view of true MSE.

Figure 8 depicts the regression tree associated with our choice of k = 14. Above the tree are shown the coefficients associated with the respective carriers [X(1) through X(6)] and the constant term, for the global linear least squares fit. This is the model L_{t_o} associated with the root node. Above each node t is the empirical residual sum of squares $\hat{Q}_t$ associated with it. Below each nonterminal node (solid circles) are shown its split coordinate J_t and split point s_t. Below each terminal node (open squares) are shown the coefficients of the first three carriers and the constant term for the model L_t associated with that node, as well as the number of training observations (circled) S_t. The values of the coefficients for the last three carriers are the same for the models associated with all nodes of the tree, as given by the global linear least squares fit.

Inspection of the binary regression tree (Figure 8) shows that the partitioning procedure behaved reasonably. It made no splits on coordinates five and six and one split on coordinate four. It made no change to the coefficients associated with these carriers from that given by the global linear least squares fit. It made substantial changes to the coefficients associated with the (first) three carriers for which the response has a highly nonlinear dependence. The first split deals with the additive nonlinear dependence by splitting the third coordinate near its central value and

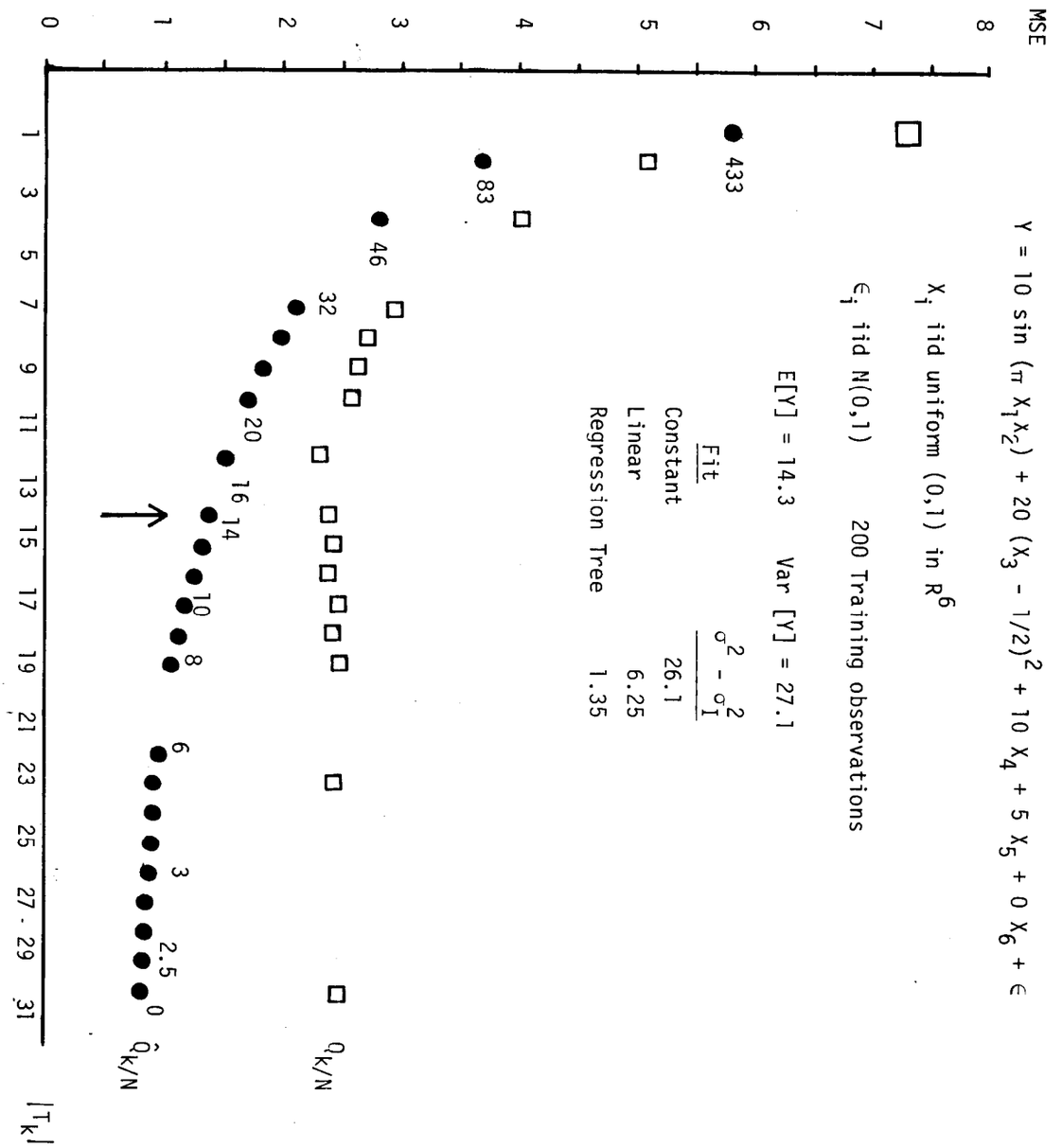
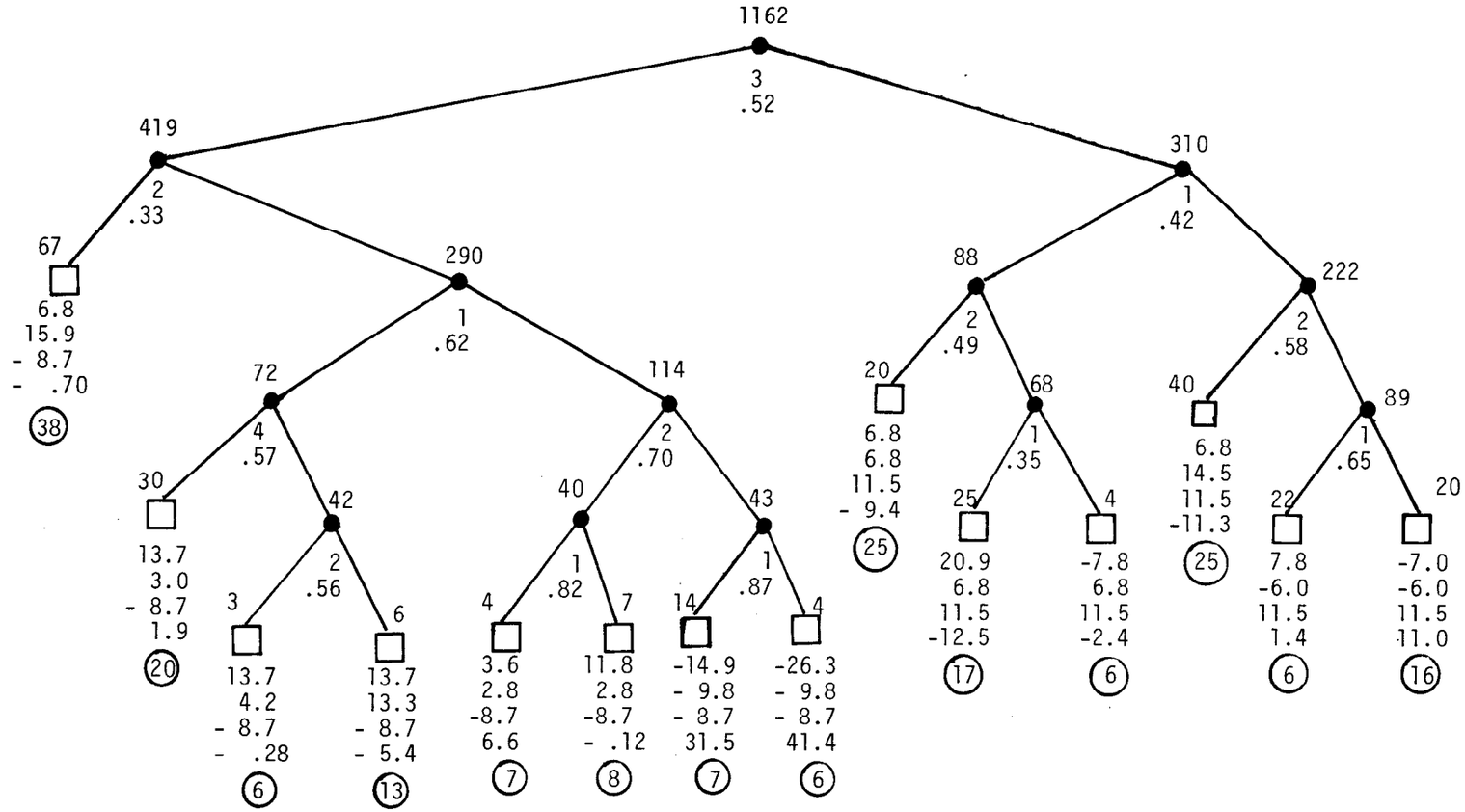

MSE
$Y = 10 \sin(\pi X_1 X_2) + 20 (X_3 - 1/2)^2 + 10 X_4 + 5 X_5 + 0 X_6 + \epsilon$
X_i iid uniform (0,1) in R^6
ϵ_i iid N(0,1)
200 Training observations
E[Y] = 14.3
Var [Y] = 27.1
Fit
$\sigma^2 - \sigma_I^2$
Constant
26.1
Linear
6.25
Regression Tree
1.35
433
83
46
32
20
16
14
10
8
6
3
2.5
0
$Q_{k/N}$
$\hat{Q}_{k/N}$
$|T_k|$

Figure 7

L_{t_o} = (6.8, 6.8, 1.1, 12.0, 5.0, 0.19, -1.6)



$$Y = 10 \sin(\pi X_1 X_2) + 20 (X_3 - 1/2)^2 + 10 X_4 + 5 X_5 = 0 X_6 + \epsilon$$

Figure 8

augmenting the corresponding coefficient with roughly equal and opposite slopes on each side. The remaining splits tend to alternate between the first two carriers trying to deal with their interaction. The procedure made three speculative splits ($\hat{Q}_t = \hat{Q}_{l(t)} + \hat{Q}_{r(t)}$) which tended to be well rewarded with later splits. There was only one noise split (on carrier four) for which the true underlying model had a purely linear dependence. As shown in Figure 7, the resulting regression tree model provides a much more accurate description of the training sample (in terms of true MSE) than a simple linear least squares fit.

Discussion

The example of the previous section illustrates the advantages and, to some extent, the disadvantages of the regression tree approach to nonparametric multiple regression. The basic strength of the procedure is that it is practical and seems to work. From the computational point of view, the procedure is quite fast. The time required to construct the model grows with the number of carriers p and training set cardinality N as p N log N. The time required to estimate the response value for a test observation $\underline{X}$ grows as log $|T|$ independently of p.

The statistical strength of this procedure lies with its property of being locally adaptive. It tries to partition the carrier data space into convex regions that are as large as possible subject to the constraint that a (different) linear model is a reasonable approximation to the response dependence within the region. It treats the problem (locally) in the subspace of smallest possible dimension; that is, the subspace associated with the carriers for which the response has a nonlinear dependence. In the example above, after the first cut, the procedure treated the problem mainly in the subspace of the first two carriers even through globally the problem is six-dimensional. This tends to reduce the bias associated with lack of model fit by making the most effective use of each splitting possibility. On the other hand, variance is reduced by estimating each coefficient with the largest possible training subsample. Each coefficient is estimated using all of the data associated with the largest subtree in which its corresponding carrier does not appear as a split coordinate. In the example, the entire training data set was used to estimate the coefficients of carriers 4, 5 and 6. The two coefficients associated with carrier three were each estimated using approximately one-half of the sample. The several coefficients each associated with the first two carriers are estimated using correspondingly smaller subsamples. In this way, the procedure tries to make a good trade-off between the conflicting goals of reducing both bias and variance of the model.

From the data analytic point of view, the regression tree can be interpreted as representing the nonlinear aspects of the model. The linear aspects are represented by the global linear least squares fit associated with the root node. If a global linear model adequately describes the data, then the tree will tend to collapse to only the root node. Thus, the procedure provides a loose goodness-of-fit for linear models.

Carriers that appear as split coordinates and significantly augment the model are ones for which the response dependence is highly nonlinear. (However, the converse is not necessarily true. A carrier that is highly correleated with another for which there is a highly nonlinear response dependence, will also be one for which the response has a nonlinear dependence. This carrier may never appear as a split coordinate because the splitting procedure may always prefer the carrier to which it is highly correlated). By inspecting the details of the regression tree, these inferences can be made locally in each region of the carrier data space.

Possible limitations of this regression tree approach center around its lack of continuity and robustness. The resulting model is not strictly continuous at every terminal cell boundary. It is strictly continuous at "brother" cell boundaries (those with a common parent node) but not at "cousin" cell boundaries (those for which the common ancestor is once or several times removed). However, the model is still relatively smooth at these boundaries (especially for close cousins) since the models associated with these terminal nodes share all but a few common coefficients. Still, the regression tree approach will not be appropriate in those situations for which an absolutely continuous approximation is required.

The lack of robustness associated with the procedure follows directly from its use of least squares fitting. This is easily overcome (at the expense of some computation) by simply substituting the robust/resistant analogs for the least squares routines. It is interesting to note that extreme outliers do not cause the regression tree procedure to break down (as is the case for linear least squares regression) even when least squares fitting is used. The splitting procedure tends to isolate outliers into unique terminal nodes and then proceeds with the rest of the data. Thus, although extreme outliers can seriously weaken the procedure by wasting cuts, they do not cause it to totally break down.

References

Breiman, L. and Stone, C.J. (1977). Parsimonious Binary Classification Trees. Technology Service Corporation, Santa Monica, Ca., Technical Report.

Efron, B. (1977). Bootstrap Methods: Another Look at the Jackknife. Stanford University Statistics Dept., Technical Report No. 32.