TREE-STRUCTURED STATISTICAL METHODS

BY

HEPING ZHANG, JOHN CROWLEY, HAROLD C. SOX, JR., RICHARD A. OLSHEN

TECHNICAL REPORT NO. 187

JANUARY 1997

PREPARED UNDER THE AUSPICES OF

PUBLIC HEALTH SERVICE GRANT

2 R01 CA59039-21

DIVISION OF BIOSTATISTICS

STANFORD UNIVERSITY

STANFORD, CALIFORNIA
Tree-structured Statistical Methods

By

Heping Zhang, John Crowley, Harold C. Sox, Jr., Richard A. Olshen

Technical Report No. 187
January 1997

Prepared Under the Auspices
Of
Public Health Service Grant
2 R01 CA59039-21

Division of Biostatistics
Stanford University
Stanford, California
Tree-structured Statistical Methods

Heping Zhang
Yale University

John Crowley
University of Washington

Harold C. Sox, Jr.
Dartmouth-Hitchcock Medical Center
and
Richard A. Olshen
Stanford University

January 20, 1997

Overview

This article is about binary tree-structured methods for biostatistics. The techniques, sometimes called “recursive partitioning,” can facilitate the automation of diagnoses and prognoses in clinical contexts. A brief account of the larger topic of rules for clinical prediction provides context for the more specific discussion that follows. Wasson et al. [42] give a broad view of such rules.

Classifying patients is a central element of the physician’s work. Typically, the physician asks questions like these. “Is this patient with chest pain suffering a heart attack, or does he simply have a strained muscle? What is the best diagnostic test for this patient with chest pain? During the next year, is this survivor of a heart attack likely enough to die that I should
do a costly test that might detect a life-threatening, correctable problem?" Answering questions helps in crafting good care, but it also is essential for matching health care resources to the patients who need them the most.

Until recent decades physicians had no choice but to answer these questions in a subjective, intuitive, idiosyncratic manner. Skill probably depended on clinical experience, but equally experienced physicians varied in their levels of skill. Physicians seldom wrote down the clinical findings that they used to estimate probabilities or their rules for combining findings. As a result, there was, and is, inter-physician variability and intra-physician variability in estimating probabilities and in making prognoses. Data-based rules are now available for these purposes. They enable the physician to interpret a patient’s findings in quantitative terms by reference to a large number of patients with similar findings and a known diagnosis or clinical outcome. The goal of using a clinical prediction rule is to use clinical findings to place the patient in a subgroup whose disease prevalence, outcome rate, or survival rate is known and to use that placement to infer some aspect of the patient’s course.

Clinical prediction rules are, as we have indicated, empirical. Their basis is a cohort of patients with known clinical findings and a known diagnosis or outcome. This cohort, the training set (also called the learning sample), is the set of patients from whom the key clinical predictors and the rule for combining them are discovered. Applying the rule to a separate cohort, the test set, can provide the subgroup-specific disease prevalences or outcome rates that are the basis for estimating probability or prognosis in an individual patient.

There are several steps involved in developing a clinical prediction rule.

Assemble the cohort. The first step is to decide upon the criteria for including the patient in the cohort that will form the training set. The investigator must ask, “What is the problem to be solved? Estimating a probability of coronary artery disease in patients with chest pain? Estimating the one-year death rate in heart attack survivors?” The answers to these questions determine the clinical criteria for admitting a patient to the cohort. One must also pay attention to the generalizability of the findings when defining the cohort. Will the clinical prediction rule apply to all hospitals or clinics? If so, the study must enroll patients from a variety of care settings. Will the findings apply to all patients with the cohort-defining problem (e.g., chest pain) or just those of a certain age, gender, or socioeconomic standing?
The second step in assembling the cohort is to decide on the size of the training set, though circumstances frequently limit the choice. A large cohort maximizes the chance that the clinical prediction rule will be optimal for other populations of patients. One empirical rule is to include five patients in the smallest outcome category for every clinical predictor in the rule.

Decide upon the outcome measure. One must answer this question, "What is the outcome to be predicted?" Then, one must state the criteria for deciding if the outcome has occurred, being sure that it is possible to collect the outcome criteria on all patients. Ideally, the measure is an unequivocal feature of the outcome, such as the result on a reliable indicator that disease is present (the "gold standard test") or death from the disease. The outcome should be useful to a clinician, such as an intermediate point on the path to a decision.

Decide which predictors to obtain. The predictors should be pertinent to the clinical problem. Obtaining the information should be feasible. Precise instructions on how to obtain the information are important to the clinician who wants to classify a patient accurately. The list of predictors should include all the clinical findings that could be pertinent, so that the study does not overlook an important predictor.

Collect the data and determine the outcome on a series of patients. It is fundamental here to avoid bias in collecting the data and deciding upon outcome. If the outcome is a clinical diagnosis, then it is all too easy to define it from the predictors for a patient for whom the outcome is not obvious. This can lead to excessive optimism regarding the worth of the predictors at hand. Thus, it is important that for purposes of the study, predictors and outcome are distinct. Moreover, a source of bias is avoided if the person who assigns the diagnosis is ignorant of any findings that comprise or inform the determination of predictors.

Identify the predictors and the rule for combining them. Most of what remains of the article is a detailed description of this step.

Determine the misclassification rate of the rule. The most important principle is to measure the misclassification rate in a new cohort of patients (the test set). If it is not possible to follow this principle, there are several cross-validatory techniques by which to estimate misclassification rates on new populations by using the training set patients. By far the best approach is to enroll a new cohort of patients, preferably by a new research group in a new clinical setting. The article by Wasson et al. [42] describes the
measurement of the misclassification rates. See especially [2] and [14].

The tree-structured statistical techniques we have mentioned are by now widely used in biostatistical inference, e.g., [1], [3], [4], [11], [17], [18], [24], [44], [46]. While there are many approaches, all have in common the successive partitioning of a "feature space" of predictors into subsets. The partitioning is done on the basis of a learning sample, and then, if one is fortunate, it is validated by a test sample. In this article, it is always the case that a non-terminal node of a tree has only two daughter nodes; thus the trees are binary trees. Each node of the trees corresponds uniquely to a subset of the feature space and thus to a unique set of constraints on the predictors of outcome. A decision rule or summary statistic or value of a regression, depending on the application, is the same within the region determined by the terminal node. Learning sample observations have (predictor, outcome) pairs. The hope is to partition so that regions are simple enough to be understandable in terms of the subject matter, yet homogeneous as to outcome. Prediction is made to future data for which predictors are known but outcomes are not. These techniques have been applied with success to classification, regression, survival analysis, and clustering. One popular way to form trees from data is that of "Classification and Regression Trees" (CART\textsuperscript{TM}). Depending on the nature of the response, the techniques may be referred to as classification trees (discrete response), regression trees (continuous response), survival trees (censored positive response), or tree-structured vector quantization (when the predictors and response are the same, univariate or vectorial, but there are constraints on the complexity of the prediction). In the literature, all of them are termed tree-based (or tree-structured) methods or recursive partitioning techniques. Those of the many ideas they have in common are well described by Breiman et al. [2], which has much historical background [8], [15], [16], [19], [32].

In what follows, we first present an early tree-based analysis that will serve to illustrate many aspects of tree-based methodology, not least the simplicity of the ultimate answers. Second, we lay out the key and common ground for all of the tree-based methods. Then, we fill in details to distinguish the major types of tree-based methods. In light of the recent surge of the development and use of survival trees, we devote particular attention to this area. Finally, we discuss some common tips, tricks, and traps one encounters in applying the tree-based methods.
An Example

One early notable application of binary classification trees was for the purpose of diagnosing patients who enter hospital emergency rooms with chief complaints of acute chest pain [18]. See also Goldman et al. [17]. Starting with about 100 initial variables that were thought to be predictors of a heart attack, Goldman and colleagues went through a preliminary screening of the predictors and selected forty of them for further consideration. Their goal was to construct a classification rule that can guide physicians in emergency rooms to decide in a timely manner (that is, before levels of fundamental enzymes are known) whether a patient has suffered or is suffering a myocardial infarction (heart attack). Although a definitive diagnosis of heart attack is typically done by testing the levels of these enzymes which tend to be released by damaged heart muscle, the importance of the computerized decision rule is that it is based on clinical measurements that are available almost immediately when a patient is admitted. By answering a maximum of 13 questions (Figure 1), any patient can be classified as being high or low risk for heart attack.

Outline of the Tree-based Methods

The Data and the Objective

Suppose that we have observed $p$ covariates $\mathbf{x}$ and a response $y$ for $n$ individuals. For the $i$th individual, the measurements are

$$\mathbf{x}_i = (x_{i1}, \cdots, x_{ip})'$$

and $y_i, i = 1, \cdots, n$.

The objective is to model the probability distribution of $P(y | \mathbf{x})$ or some functional of this conditional distribution. Here, $\mathbf{x}$ can be an array of mixed categorical (nominal or ordinal) and continuous variables. Some components may have missing values. It is the nature of $y$ that mandates the choice of methodology. In most applications, the outcome, $y$, is either a continuous (with or without censoring) or categorical variable. Recently, the tree-based methods have been developed to allow for vectorial $y$ [16], [37], [47]. Here, our discussion focuses on binary and censored continuous $y$ for (a) these are where the tree-based methods are applied for the most part in medicine, and
Figure 1: Classification Tree for Diagnosing Heart Attack.
Table 1 provides the questions (Q1 to Q13) used in this tree. This figure is based on Figure 1 of Goldman et al. [18].
Table 1: Questions Used in Figure 1

<table>
<thead>
<tr>
<th>Label</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>Does the emergency room EKG show ST-segment elevation or a Q wave that is suggestive of infarction and is not known to be old?</td>
</tr>
<tr>
<td>Q2</td>
<td>Did the present pair or episodes of recurrent pain begin 42 or more hours ago?</td>
</tr>
<tr>
<td>Q3</td>
<td>Is the pain primarily in the chest but radiating to the shoulder, neck, or arms?</td>
</tr>
<tr>
<td>Q4</td>
<td>Does the emergency room EKG show ST-segment elevation or a Q wave that is suggestive of ischemia or strain and not known to be old?</td>
</tr>
<tr>
<td>Q5</td>
<td>Is the present pain (A) similar to but somehow worse than prior pain diagnosis as angina or (B) the same as pain previously diagnosed as an MI?</td>
</tr>
<tr>
<td>Q6</td>
<td>Does local pressure reproduce the pain?</td>
</tr>
<tr>
<td>Q7</td>
<td>Has the chest pain associated with diaphoresis?</td>
</tr>
<tr>
<td>Q8</td>
<td>Is the patient 40 years or older?</td>
</tr>
<tr>
<td>Q9</td>
<td>Is the patient 70 years or older?</td>
</tr>
<tr>
<td>Q10</td>
<td>Was this pain diagnosed as angina (and not an MI) the last time the patient had it?</td>
</tr>
<tr>
<td>Q11</td>
<td>Is the pain primarily in the chest but radiating to the left shoulder?</td>
</tr>
<tr>
<td>Q12</td>
<td>Did the present pain or episodes of recurrent pain begin 10 or more hours ago?</td>
</tr>
<tr>
<td>Q13</td>
<td>Is the patient 50 years or older?</td>
</tr>
</tbody>
</table>

These questions are taken from Figure 1 of Goldman et al. [18].
(b) *logistic regression* and *linear discriminant function analysis* (binary case), and Cox proportional hazard modeling (in the case of survival analysis) are standard approaches to analyzing such outcomes; it is worthwhile to understand the strengths and limitations of both the more classical and the tree-based methods.

**Basics of the Tree-based Technique**

Look again at the tree in Figure 1. This tree has eight layers of nodes. In general, the number of layers varies from case to case. The first layer is always the unique root node, namely, the circle on the top. There are 13 internal (the circle) and 14 terminal (the box) nodes that are scattered among the various layers. The root and the internal nodes are connected to two nodes in the next layer that are called left and right daughter nodes, but terminal nodes do not have “children.” Moreover, the tree is not necessarily “balanced” in that not all nodes in the same layer have daughter nodes. The thrust of the tree-based technique is to answer these questions:

- What are the contents of the nodes and how do we split a node?
- How do we declare a node terminal?
- What inferences do we make for the various terminal nodes?
- What have we learned about our data and the possibly complex relationships among the predictors and outcome as a result of studying the tree?

The subsection below, Splitting a Node, addresses the first item, and it is followed by a subsection on Terminal Nodes. There, we discuss how to determine terminal nodes. The last question is best answered on a case-by-case basis.

**Splitting a Node**

The root node contains the learning sample. The learning sample summarizes the information from the past experience and allows us to learn the underlying data structure. In Goldman et al. [18], it contains 482 patients.
The terminal nodes correspond, as was indicated, to disjoint subgroups of this learning sample. The union of two subgroups in the daughter nodes comprises the subgroup of their parent node. For example, the root node in Figure 1 has 482 patients who are divided into two subgroups: one with 443 patients and the other with 39. So, a node is merely a subgroup of the learning sample.

A critical step of the tree-based technique is to determine the split from one parent node to the two daughter nodes. Since splitting the root node is identical in terms of criterion to that for other nodes, it suffices to explain how to split the root node. Thus, we consider how the 482 patients in the study of Goldman et al. [18] might be divided into two subgroups.

First, the division of the root node is described and implemented by means of a predictor. The purpose of splitting is to generate two offspring whose union is preferred to the root node in some sense. As was mentioned earlier, there were 40 selected potential predictors of a heart attack, denoted by $x$, that entered into the tree-based analysis. If $x_j$ is an ordered covariate such as age, two subgroups result from the question of the form "Is $x_j > c$?" Here the cutoff point $c$ is in the range of the observed values of $x_j$. The $i$-th subject goes to the right or left node according to whether or not $x_{ij} > c$. Q2, 8, 9, 12, and 13 in Figure 1 and Table 1 are precisely this type of question. On the other hand, many medical studies involve nominal covariates. For example, the body sites of pain in the present example include the chest, shoulder, and neck. We can send a patient to the left or right node by asking questions such as "is the pain in the neck only?" and "is the pain in the neck and shoulder?" Given the number of covariates (here, it is 40) and the number of possible cutoff points for every covariate, there are many of possibilities to split the root node into two nodes. Therefore, we must be specific in what we mean by a desirable split.

If we take age as a tentative splitting covariate and consider its cutoff at 40, as a result of the question "Is $x_j$ (age) > c (40)?", we have the following table:

<table>
<thead>
<tr>
<th>Left Node ($t_L$)</th>
<th>$x_j \leq c$</th>
<th>$n_{11}$</th>
<th>$n_{12}$</th>
<th>$n_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right Node ($t_R$)</td>
<td>$x_j &gt; c$</td>
<td>$n_{21}$</td>
<td>$n_{22}$</td>
<td>$n_2$</td>
</tr>
</tbody>
</table>

What would be desirable in this case? Obviously, we would want to choose
a split so that the distributions of \( y \) in the daughter nodes are homogeneous. To reflect this idea in the table above, a desirable left (right) node \( t_L(t_R) \) should have the property that either \( n_{11}(n_{21}) \) is much greater than \( n_{12}(n_{22}) \) or vice versa. In other words, we force most of the MI cases to either the left or right node. In a perfect situation where \( n_{11} = n_{22} = 0 \), the two nodes are pure (or completely homogeneous) because each of them contains only one value of the outcome. In contrast, their parent node includes a mixture of \( n_{11} \) non-MI and \( n_{22} \) MI patients. This is what we mean by “more desirable” here. Mathematically, one frequently used measure of node homogeneity is defined through the entropy function as follows:

\[
h(t_L) = \frac{n_{11}}{n_1} \log\left(\frac{n_{11}}{n_1}\right) + \frac{n_{12}}{n_1} \log\left(\frac{n_{12}}{n_1}\right).
\]

(1)

Then, we select a split that maximizes the weighted node homogeneity:

\[
\frac{n_1}{n} h(t_L) + \frac{n_2}{n} h(t_R).
\]

(2)

It is also interesting to view the criterion (1) from other points of view. Thus, suppose that \( y \) in node \( t_L \) has a binomial distribution with a frequency of \( \theta \) so that

\[
P(y = 1|t_L) = \theta.
\]

Then, the log-likelihood function from the \( n_1 \) observation in node \( t_L \) is

\[
n_{11} \log(\theta) + n_{12} \log(1 - \theta).
\]

The maximum of this log-likelihood function is proportional to (1). Not surprisingly, many criteria of “more desirable” are couched as maxima of certain likelihood functions. See section on use of Likelihood Functions below.

**Terminal Nodes**

After the node splitting procedure described above is applied to the root node, the resulting daughter nodes can also be split in the same way, followed by the granddaughter nodes and so on. This splitting process always terminates because the number of study subjects is finite. For example, the number of possible splits for the data of Goldman et al. [18] cannot exceed 481. Of course, we can force the process to stop at any point. In usual
practice, we end up with a large tree, which is generally too large to be useful. In order that we end up with a useful tree, a rigorous rule for pruning some over-fitting nodes is required. Goldman and colleagues did not know a priori the 14 terminal nodes in Figure 1. Initially, these terminal nodes had offspring.

For the purpose of illustration, take a part of Figure 1 as displayed in Figure 2, and use it as if it is an entire initial tree. The question is: “can we prune away some of the nodes?” If we can answer this question in a general way, then we will know how to prune any tree. To this end, we introduce a measure of the quality of a tree. Recall that the objective of the tree-based method is to extract homogeneous subgroups of the study sample. Whether we have achieved it depends on whether the terminal nodes are indeed sufficiently homogeneous. Hence, the quality of a tree, denoted by $T$, is really the quality of its terminal nodes and we have

$$R(T) = \sum_{t \in \bar{T}} p(t)r(t),$$  \hspace{1cm} (3)$$

where $\bar{T}$ is the set of terminal nodes of tree $T$; $r(t)$ summarizes the quality of node $t$; and $p(t)$ is the proportion of subjects falling into node $t$. For binary outcomes, $r(t)$ is usually taken to be the within-node misclassification cost.
The size of a tree is another important aspect, which here is the fundamental measure of its complexity. Note that the total number of nodes in a tree, $T$, is $2|\hat{T}| - 1$, where $|\hat{T}|$ is the number of the terminal nodes of $T$. Hence, the complexity of $T$ can be defined directly as $|\hat{T}|$. Usually, a unit cost, called a complexity parameter, is assigned to each terminal node, and the sum of all costs becomes the penalty for the tree complexity. Therefore, the final quality measure of a tree is the following cost-complexity:

$$R_\alpha(T) = R(T) + \alpha|\hat{T}|,$$

(4)

where $\alpha(>0)$ is the complexity parameter.

For a given complexity parameter and an initial tree such as the one in Figure 2, there is a unique smallest subtree of the initial tree that minimizes the cost-complexity measure (4). Importantly, if $\alpha_1 > \alpha_2$ the optimally pruned subtree corresponding to $\alpha_1$ turns out to be a subtree of the one corresponding to $\alpha_2$. So, as we increase the complexity parameter, we have a sequence of nested optimally pruned subtrees. This sequence has to have finite length, and the last one is the root node. That the successive optimally pruned subtrees are nested can entail important savings in computation [2].

Here is how pruning works for the tree in Figure 2. Before we start, we must specify a misclassification cost that reflects the severity of the mistake that results when an MI patient is classified to non-MI or vice versa. Let $C(i|j)$ be the misclassification costs that a class $j$ patient is classified as a class $i$ patient. Here, we there are two classes of patients: 0 for non-MI and 1 for MI patients. For medical reasons, it is natural to choose $C(0|1) > C(1|0)$ because the consequence is potentially more severe when an MI patient is wrongly diagnosed than when a non-MI patient is. As did the authors, we take $C(1|0) = 1$ and $C(0|1) = 15$, which means that the a false positive diagnosis costs as much as 15 false negative ones. Table 2 gives the misclassification costs for all nodes and their designated classes. The third and fourth columns list the misclassification costs as a result of classifying the node as MI and non-MI, respectively. The minimum of these two types of cost determines the final class membership (column 5) of a node. The expected node cost, $r(t)$, is the minimum of the two costs divided by the node size; for instance, $r(D) = \min(31, 60)/46 = 31/46 = 0.67$. The final within-node cost (the last column) is obtained by weighting $r(t)$ by $p(t)$. For example, within node 3, $r(3) = 0.59$ and $p(3) = 37/194 = 0.19$. Hence, the final cost equals $0.59 \times 0.19 = 0.11$. 

12
Table 2: Misclassification Costs

<table>
<thead>
<tr>
<th>Node label</th>
<th>Node size</th>
<th>Misclassification Costs</th>
<th>Designated class</th>
<th>$r(t)$</th>
<th>Weighted cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>194</td>
<td>185</td>
<td>9*15=135</td>
<td>Non-MI</td>
<td>0.70</td>
</tr>
<tr>
<td>2</td>
<td>169</td>
<td>154</td>
<td>5*15=75</td>
<td>Non-MI</td>
<td>0.44</td>
</tr>
<tr>
<td>3</td>
<td>37</td>
<td>22</td>
<td>5*15=75</td>
<td>MI</td>
<td>0.59</td>
</tr>
<tr>
<td>A</td>
<td>132</td>
<td>132</td>
<td>0</td>
<td>Non-MI</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>20</td>
<td>20</td>
<td>0</td>
<td>Non-MI</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>17</td>
<td>2</td>
<td>5*15=75</td>
<td>MI</td>
<td>0.12</td>
</tr>
<tr>
<td>D</td>
<td>46</td>
<td>31</td>
<td>4*15=60</td>
<td>MI</td>
<td>0.67</td>
</tr>
</tbody>
</table>

* Number of misclassified subjects multiplied by the cost unit.
† Misclassification cost divided by the node sample size.
‡ $r(t)$ multiplied by $p(t)$, the proportion of subjects in the node.

From Table 2, we calculate the misclassification cost for the tree in Figure 2 as follows. Note that it has four terminal nodes, labeled A, B, C, and D. As is defined in (3), the tree misclassification cost is the sum of the weighted misclassification costs of its terminal nodes. Based on the last column of Table 2, the weighted costs for terminal nodes A through D are respectively 0, 0, 0.01, and 0.16. Thus, the tree misclassification cost is

$$0 + 0 + 0.01 + 0.16 = 0.17.$$  

The complexity of this tree is 4 because it has 4 terminal nodes. If we choose a complexity parameter, $\alpha = 0.05$, it follows from (4) that the present tree cost-complexity equals $0.17 + 4 \times 0.05 = 0.19$. Table 3 provides three ranges of the complexity parameter that correspond to three nested subtrees. The cost-complexities are also given in this table when a complexity parameter is chosen in the range. The thresholds of the range are determined by these considerations. We prune off some terminal nodes only if the tree cost-complexity is improved after the pruning. This decision obviously depends on the choice of the complexity parameter, $\alpha$. For instance, if $\alpha = 0$ the initial tree, $T_0$, has a smaller cost-complexity than any of its subtrees. Therefore, we cannot prune off any terminal nodes with $\alpha = 0$. What is the smallest $\alpha$
Table 3: Nested Sequence of Subtrees

<table>
<thead>
<tr>
<th>Subtree</th>
<th>Range of α</th>
<th>Nodes in the Subtree</th>
<th>Cost-Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_0$</td>
<td>0 to 0.1</td>
<td>1,2,3,A,B,C,D</td>
<td>0.01+0.16+4+0.05=0.19</td>
</tr>
<tr>
<td>$T_1$</td>
<td>0.1 to 0.215</td>
<td>1,2,3,A,D</td>
<td>0.11+0.16+3+0.19=0.75</td>
</tr>
<tr>
<td>$T_2$</td>
<td>0.215+</td>
<td>1</td>
<td>0.70+0.28=0.98</td>
</tr>
</tbody>
</table>

such that some of the terminal nodes can be removed? It turns out to be

$$\min_{t \in T_0} \frac{r(t)p(t) - R(T(t))}{|\hat{T}(t)| - 1},$$

where $T(t)$ is a subtree rooted at node $t$ and the minimization is over all internal nodes of $T_0$. See [2]. Now, $T_0$ has three internal nodes 1, 2, and 3 and 0.1 is the minimum of the corresponding three numbers: $\frac{0.7-0.17}{4-1} = 0.18$, $\frac{0.39-0.01}{3-1} = 0.19$, and $\frac{0.11-0.1}{2-1} = 0.1$. When $\alpha = 0.1$ is applied, we can prune off terminal nodes B and C without loss of cost-complexity, leading to tree $T_1$ in Table 3. Next, we can ask the same question: what is the smallest $\alpha$ such that some of the terminal nodes of $T_1$ can be removed? This tree has two internal nodes labeled 1 and 2. It is easy to see that the smallest $\alpha$ equals 0.215 and it leads to the single node tree, $T_2$. In general, we repeat the same process until we reach the single node tree.

The next step is to select a subtree from the nested sequence. A special aspect of the study of Goldman et al. was that the tree was used to classify patients at another hospital [18]. These additional data constitute a validation data set, or also called test sample. The misclassification costs for the three nested trees were respectively $R^*\{T_0\} = 0.45$, $R^*\{T_1\} = 0.59$, and $R^*\{T_2\} = 0.52$. Because $R^*\{T_0\}$ is the smallest, $T_0$ is the best choice, implying that we cannot prune any nodes. When an independent test sample is not available, a cross-validation procedure is usually recommended. We refer to Breiman et al. [2] for details.
Figure 3: The $L^1$ Wasserstein Distance between Two Kaplan-Meier Curves. Note that one curve ($S_L$) is darker than the other ($S_R$).

**Survival Trees**

In this section, we explain how to use the ideas expressed above to analyze censored survival data. Censored survival data arise from many medical studies; see, e.g., [1], [3], [4] for some typical examples. We face the same basic issues. One is to define a splitting criterion by which to divide a node into two, and the other is to choose a "right-sized" tree for subsequent use. Many criteria have been proposed in the literature, but they differ primarily in the way of declaring what daughter nodes are desirable. Segal [38] and Intrator and Kooperberg [23] are two important and helpful reviews. See also LeBlanc and Crowley [27] and Crowley et al. [9].

**Gordon and Olshen's Rule**

One early proposal was made by Gordon and Olshen [22]. The idea is this: when a node is divided into two, we can compute the Kaplan-Meier curves (see, e.g., [31]) separately for each. A desirable split can be characterized as one that results in two very different survival functions in the daughter nodes. They used the so-called $L^p$ Wasserstein metrics, $d_p(\cdot, \cdot)$, as the measure of discrepancy between the two survival functions. Specifically, for $p = 1$, the Wasserstein distance, $d_1(S_L, S_R)$, between two Kaplan-Meier curves, $S_L$ and $S_R$, is the shaded area in Figure 3.

An optimal split is chosen to maximize the distance, $d_1(S_L, S_R)$. Here,
$S_L$ and $S_R$ are respectively the Kaplan-Meier curves for the left and right daughter nodes. Replacing the quantity (2) with $d_1(S_L, S_R)$ we can produce an initial tree as described above in the section on Splitting a Node.

To prune an initial survival tree, $T$, Gordon and Olshen [22] suggested a tree cost-complexity as follows. Consider a terminal node, $t \in T$. First, estimate the Kaplan-Meier curve $S_t$. Second, find the closest $\delta_t$ to $S_t$ in terms of $d_1(S_t, \delta_t)$; here $\delta_t$ must be chosen from piecewise constant survival functions that have at most one point of discontinuity. That is, $\delta_t$ has at most two constant pieces. Then, define the within-node cost, $R(t)$, as $d_1(S_t, \delta_t)$. This can be viewed as the deviation of survival times about their median. Finally, applying the same formula (4) we have the tree cost-complexity. Obviously, the same principle applies as we use different Wasserstein metrics. It should be noted, however, that when censoring depends on the covariates, the $L^p$ Wasserstein metrics tend to produce splits (due to structure in the censoring) when in fact there is no dependence of survival upon covariates [9].

**Use of the Log-Rank Test**

In survival analysis, the log-rank test is a popular approach for testing the significance of differences between the survival times of two groups. Motivated by this fact, Ciampi et al. [6] and Segal [35] suggested selecting a split that results in the largest log-rank test statistic, which is defined as follows. A partition gives a sequence of $2 \times 2$ tables at times when failures occurred.

<table>
<thead>
<tr>
<th>Dead</th>
<th>Alive</th>
<th>$m_{11}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n_{11}$</td>
<td>$n_i$</td>
<td></td>
</tr>
</tbody>
</table>

$\text{Left Node}(t_L) \quad x_j \leq c$

$\text{Right Node}(t_R) \quad x_j > c$

The log-rank test statistic is

$$LR = \frac{\sum_{i=1}^{k} (a_i - E_i)}{\sqrt{\sum_{i=1}^{k} V_i}}$$

where $k$ is the number of distinct failure times,

$$E_i = \frac{m_{11} n_{11}}{n_i}$$
and

\[ V_i = \left( \frac{m_{i1}(n_i - m_{i1})n_{i1}}{n_i(n_i - 1)} \right) \left( 1 - \frac{n_{i1}}{n_i} \right). \]

The log-rank test (or any similar two sample test) is a measure of between node difference. However, with this approach a measure of cost for each node is not readily available for use in pruning. Segal [35] also recommended a practical bottom-up procedure. The basic idea is this. For each internal node (including the root node) of an initial tree, we assign it a value that equals the maximum of the log-rank statistics over all splits starting from the internal node of interest. Then, plot the values for all internal nodes in an increasing order and decide a threshold from the graph. If an internal node corresponds to a smaller value than the threshold, we prune all of its offspring.

LeBlanc and Crowley [26] introduced the notion of “goodness-of-split” complexity as a substitute for cost-complexity in pruning the tree. Let \( G(t) \) be the value of the log-rank test at node \( t \). Then the split-complexity measure is

\[ G(T) = \sum_{i \notin \tilde{T}} G(t) - \alpha(|\tilde{T}| - 1). \]

Note that the summation above is over the set of internal (non-terminal) nodes and \(|\tilde{T}| - 1\) is the number of internal nodes. The negative sign is a reflection of the fact that \( G \) is to be maximized, whereas the cost-complexity \( R \) is minimized. LeBlanc and Crowley [26] recommend choosing \( \alpha \) between 2 and 4 (when the log-rank test is expressed in the \( \chi^2 \) form) and using bootstrap techniques to deflate the value of \( G \). An alternative pruning method based on permutation p-values for the logrank test is described in LeBlanc and Crowley [27].

In some medical situations such as in cancer, the goal of a tree-based analysis is to arrive at a few (perhaps three or four) groups that define the “stages” of disease. Treatment strategies or randomization algorithms within a clinical trial can then be designed with these prognostic groups or stages in mind. Even an optimally pruned tree may have many terminal nodes, so nodes with similar survival need be combined in a final staging system. Ciampi et al. [6] termed this process “amalgamation” and suggested combining terminal nodes based on comparisons using the log-rank statistic.
LeBlanc and Crowley [26] define an ordered categorical variable (based for example on median survival) describing the terminal nodes and subject that single variable to a recursive partitioning scheme to amalgamate the nodes. Less formal techniques are described in LeBlanc and Crowley [27].

**Use of Likelihood Functions**

Several likelihood based splitting and pruning criteria have been proposed. Davis and Anderson [12] assume that the survival function within any given node is an exponential function with a constant hazard. The splitting criterion of LeBlanc and Crowley [25] and Ciampi et al. [7] are both based on the assumption that the hazard functions in two daughter nodes are proportional, but unknown. The difference between their two approaches is whether the full or partial likelihood function in the Cox proportional hazard model should be used. For the same logic, these authors defined various tree cost-complexities using the likelihood ratio statistic by comparing the survival times in a parent node with those in its daughter nodes. A related method due to Therneau, Grambsch, and Fleming [41] makes use of what are termed martingale residuals from the Cox model as the input to a cost-complexity scheme using least squares as the cost.

**A Straightforward Extension**

Zhang [43] examined a straightforward tree-based approach to censored survival data. Note that we observe a binary death indicator and the (failure or censored) time. If we treat these two outcomes separately, we can compute the within-node impurity, $i_\delta$, of the death indicator and the within-node quadratic loss function, $i_y$, of the time as already defined by Breiman et al. [2]. Then, the within-node impurity for both the death indicator and the time is a weighted combination, $w_\delta i_\delta + w_y i_y$. Some choices of weights $w_\delta$ and $w_y$ have been recommended by Zhang [43].

Several applications to real data have indicated that this approach and the use of the log-rank test produce very similar tree structures. Perhaps surprisingly, a preliminary simulation suggests that this simple extension outperforms the more sophisticated ones in discovering the underlying structures of data. More extensive simulations are still warranted to study the performance of the various splitting criteria.
Which Is Better?

This is still an open question, and perhaps it has no clear answer. Obviously, there is no shortage of splitting criteria for survival analysis. There is, however, very little evidence to suggest which approach is best under what circumstances. Some limited simulations comparing several of the methods have been reported in the literature [9], [10], [43]. Our recommendation is to construct survival trees using a number of approaches. Experts are likely to see on their own subject matter grounds which tree makes better sense than others.

Software

The best tested software is the commercial CART program as is distributed by Salford Systems in San Diego. It has various versions for Windows, DOS, Macintosh, and Unix systems. A tree function is also available in Splus [40]. Free software for survival trees is available, but it is less organized and tested. Four of the splitting criteria introduced above are implemented together in the C language and available upon request to heping.zhang@yale.edu. Specific programs are also available by sending email to various sites, such as dstein@scott.cts.com (Salford Systems), mark@biostat.ucsf.edu [35], rdaids@osdc.harvard.edu [12], and mikel@fhcrc.swog.org[26].

Other extensions of the tree-structured method have also been developed to analyze longitudinal data and clustered binary responses. Manuscripts and programs are available upon request to mark@biostat.ucsf.edu for continuous longitudinal data [37] and heping.zhang@yale.edu for multiple correlated binary responses [47].

Discussion

The application of the tree-structured methods to many areas of research is growing, e.g., [1], [3], [4], [5], [11], [24], [45], [46]. Nevertheless, logistic regression for binary data and Cox proportional hazard models for censored survival data still dominate applications. The main advantage of tree-based methods is their ability to produce intuitive and appealing tree structures without requiring the users to specify and select conventional models. This advantage
is more obvious when the classical, parametric models are not appropriate. See [23] for interesting examples. Several authors have compared the tree-structured methods with other methods [29], [36], [39]. Related programs are available upon request from wj@mit.edu in addition to the sites given above. The computational complexity was an issue, but no longer. To date, the application of the tree-based methods have been mostly for exploratory and secondary analyses. Recently, Zhang and Bracken [46] demonstrated the use of the tree-based methods as an intermediate step in hypothesis testing. Tree stability is another important concern. The tree is not a parameter, and it is not necessarily stable to small perturbations in the data. However, the resulting decision rules tend to be. Bayes theory may shed some light on this problem. Much work remains to strengthen the basis for statistical inference in this area. The theoretical properties of the tree-structured methods are largely unexplored, but exceptions include Donoho [13], Gordon and Olshen [19], [20], [21], LeBlanc and Crowley [26], Lugosi and Nobel [30], Nobel [34], and Nobel and Olshen [33].

References


23


