Fitting Trees to Functional Data, With an Application to Time of Day Patterns

Yan Yu and Diane Lambert*

Abstract

Decision trees often give simple descriptions of complex, nonlinear relationships between several predictors and a univariate or multivariate response. But if the response is a high dimensional vector that can be thought of as a discretized function, then fitting a multivariate regression tree may be unsuccessful. This paper explores two ways to fit trees to functional data. Both first reduce the dimensionality of the data and then fit a standard multivariate tree to the reduced response. In the first approach, each individual's response curve is represented as a linear combination of spline basis functions, penalizing for roughness, and then a multivariate regression tree is fit to the coefficients of the basis functions. In the second, a multivariate regression tree is fit to the first several principal component scores for the responses. The two methods are illustrated with time of day patterns for customers who place international calls.

Key Words: Functional Response; Multivariate Regression Tree; Principal Component Analysis; Roughness Penalty; Smoothing; Splines.

1 Background

The problem we face is to predict a customer's time of day pattern for international calling from the information in the customer's first two international calls. For example, business customers on the East Coast who call France in their first two calls may be more likely to call only countries in Western Europe and only during business hours, so they may tend to place calls early in the business day, while business customers on the east coast who call only Japan might not tend to place calls early in the business day because of the difference in time zones. Some information in the first two calls, such as duration, is continuous, and some, such as country called, is categorical. The relationship between the first two calls and the subsequent time of day distribution might be complex, but the rule for predicting the time of day distribution from a customer's first two international calls needs to be simple enough to be understood by nontechnical people.

Because decision trees partition the values of the predictors (information in the first two calls in our case) into a limited number of sets, each of which corresponds to a different mean response

*Yan Yu is a Ph.D. student in statistics at Cornell University. Diane Lambert is in the Statistics Research Department at Bell Labs, Lucent Technologies. The research for this paper was completed during the summer of 1997 at Bell Labs. The authors thank Andreas Buja for his many perceptive comments on this paper, Linda Clark and Daryl Pregibon for the multivariate regression tree code and Mark Hansen, David Ruppert, Don Sun, Scott Vander Wiel and all the seminar participants at Bell Labs and Cornell University for valuable suggestions.

(mean time of day distribution for us), they are a natural tool for generating simple prediction rules. Breiman *et al* (1984) is the classic reference for building trees for a univariate response. Clark and Pregibon (1992) and Venables and Ripley (1994) describe building trees to univariate responses in the statistical language S (Chambers and Hastie 1992). Segal (1994) applies regression trees to longitudinal data. Linda Clark of Bell Labs and Daryl Pregibon of AT&T Labs have written S functions for fitting regression trees with a multivariate response; we used their functions to fit the multivariate regression trees in our paper.

This paper gives an example in which naively applying multivariate decision trees to long vector responses is not successful. Two procedures that reduce the dimension of the response and then fit a tree to a lower dimensional responses are presented. One approach represents each respondent's time of day distribution as a linear combination of spline basis functions and then fits a multivariate tree to the estimated coefficient vectors. The other approach uses the first several principal component scores as the response vector. Both approaches give sensible results in our application.

The remainder of the paper is organized as follows. Section 2 describes the data. Section 3 shows that fitting a standard multivariate tree to time of day distributions represented as histograms gives a poor fit and decision rules that are not sensible, when the number of bins in the histograms is so large that the bin counts for most customers are mainly zero. Section 4 then develops a spline tree that treats the responses as curves rather than as vectors, and Section 5 fits the proposed spline tree to our data. The decision rules based on the fitted tree are sensible, and bootstrapping can be used to assess the stability of the predictions (Section 6). Section 7 shows that fitting a tree to the first several principal component scores also gives sensible decision rules. The fit of the spline and principal components trees are similar; the spline tree may be preferable when a smooth curve is desired for prediction. Section 8 discusses some alternative ways to build trees for functional data.

2 The Data

Our data consist of the records for completed international calls for 1705 businesses and residences on the east coast of the United States. Each call record includes the caller's telephone number, start time of the call, called (termination) number, and duration in minutes. Here we use termination number to define a variable called "region of the world" that has eight categories that are intended to capture gross differences in time zones and communities of interest. The categories are Africa, Asia, Eastern Europe, India, Middle East, South America, Western Europe, and "other", which mainly covers small islands and calls to ships. Call records also include information that can be used to distinguish large businesses from residences, which gives a binary "business/other" predictor. A customer in the business category is known to be a business, but a customer in the other category could be either a residence or a small business.

The goal is to predict the time of day pattern for a customer's start of calls. To remove the effect of call volume, we take the fractions of calls in time intervals of fixed length to be the response, rather than the number of calls per time interval. Here we consider half hour and 15 minute intervals, giving a response vector of length 48 or 96 for each customer.

The predictors that we use for each customer are the mean duration D of the customer's first two international calls, the region of the world R for the first call, an indicator $R01$ that is one if the second call is to the same region of the world as the first call and zero otherwise, the waiting time W between the first two calls, the number WE of the first two calls placed on weekdays (instead of during the weekend) and the business/other indicator. In our set of data, the mean duration of the first two calls ranges from 0.06 minutes to 222.5 minutes across customers, with a median of 6.08 minutes and quartiles of 1.63 minutes and 14.8 minutes. The time between the first two calls ranges from less than one second to 18.7 weeks across customers, with a median of about 10 hours and quartiles of about .5 hours and 3 days. Time between first two calls is a crude measure of call rate, and call rates are as highly skewed across customers as times between first two calls are. The mean number of calls per week for customers in our data is 7, but one customer has a mean of 439 international calls per week.

3 Fitting Trees to Raw Time of Day Fractions

The naive way to build a tree model for time of day data is to bin the start times of each customer's calls into fixed intervals. We assume that the n response vectors $\underline{Y}_1, \ldots, \underline{Y}_n$ across customers are independent, and fit a multivariate regression tree using the predictor vectors $\underline{X}_1, \ldots, \underline{X}_n$ as follows.

First, given a tree, the prediction $\underline{\hat{\mu}}$ at a node is the mean of the response vectors that fall in that node of the tree. Prediction error for one response in the node is measured by squared error loss, or the Euclidean distance

$$D(\underline{Y}_i, \underline{\hat{\mu}}) = \| \underline{Y}_i - \underline{\hat{\mu}} \|_2^2$$

of the observed response $\underline{Y}_i$ from the prediction $\underline{\hat{\mu}}$ for the node. The prediction error for a node is defined to be the sum of the squared error losses for all responses in the node. (Other error functions, such as L_1 error or entropy, can be used instead.) The prediction error of the tree is the sum of the prediction error over all nodes, $D(\hat{\mu}) = \sum_{nodes} \sum_i D(\underline{Y}_i, \underline{\hat{\mu}})$.

At each step of tree building, a node of the tree is split into a right and left branch so that more homogeneous subgroups are formed. The first split in a tree is found by comparing the prediction error obtained from the null tree that has no subgrouping to the prediction error obtained by partitioning the responses into two subgroups using each of the predictor variables in turn. If the predictor is ordered, the two branches are defined by whether the values of splitting variable in one node are all greater than those in the complementary node. Therefore the allowable splits preserve ordering. If the predictor is an unordered categorical variable, any split of the predictor into two disjoint subsets of the categories is permitted. The split that leads to the largest decrease in prediction error from the null tree is chosen as the first split of the tree. Later splits are found analogously by comparing the prediction error of the current tree to the prediction error after a node is split into a right branch and a left branch and choosing the split that gives the largest decrease in prediction error. (See Breiman *et al* (1984) and Clark and Pregibon (1992) for more information about fitting trees.)

Figure 1 shows the top piece of the fitted multivariate tree for our data when start times are binned in half hour intervals, so the response is a vector of length 48. The fitted tree is not sensible. First, the time between the customer's first two calls is the only variable that is chosen to split responses. While time between first two calls might be the best predictor, it is difficult to argue that other variables, such as the region of the world called, are irrelevant. Second, each split separates the response of only five customers, the minimum node size we used to fit the tree. Trees using minimum node sizes of ten and twenty have similar shapes. This behavior is not caused by outliers in predictor space. Trimming the extreme observations in the predictor space and taking square root or log transformations of waiting time between the first two calls and the mean duration of the first two calls to tame outliers gives the same fitted tree. Taking arcsine of the responses also has no effect on the fitted tree. Finally, fitting the tree is slow and gets slower as finer time intervals are used.

Here we have given the same weight to each response, even though some customers have many more calls than others. This is intentional; we do not want to pull predicted distributions towards heavy volume users. In other applications, however, it might be better to weight responses by the total number of observations used to compute the response.

4 Spline Trees for Functional Data

In our example, if the binning intervals are small, it is reasonable to associate the fraction of calls in an interval with the midpoint of the interval and treat each customer's response as a curve. To emphasize that the response is now treated as a curve, denote the response for customer i by $\underline{Y}_i(\underline{t})$ where $\underline{t} = (t_1, \ldots, t_m)$ are the midpoints of the time intervals.

If $\underline{Y}_i(\underline{t})$ is smooth, then it can be approximated by a linear combination of basis functions $\{\beta_1, \ldots, \beta_q\}$, and the coefficients of the linear combination for each individual can be used as the response for a multivariate tree. If a roughness penalty is imposed on the approximation, then each response is approximated by only a few basis functions, and the response vector is low dimensional. Generally, the lower the dimension of the response vector, the faster multivariate trees can be fit. Note that with this approach, it is possible to consider much finer time intervals, because the length of the response need not grow as fast as the number of time intervals.

There are many possible basis functions, such as B-splines. We prefer natural splines, though, because they have linear tails, which in our experience gives better behaved fits. (See De Boor (1978) for a discussion of different basis sets for smoothing.) Both B-splines and natural splines are considered in our application in Section 5. The knots for natural splines or B-splines can be chosen to satisfy a criterion on the fit of the approximation, or knots can be placed on a uniform grid or another fixed grid based on information about the behavior of the response curve. For example, most customers do not make calls between midnight and 7 a.m. and many start to make calls after 7 a.m. so we assign a knot at 7 a.m. and no knots before 7 a.m. in our example.

To fit a spline tree, take

$$\underline{Y}_i(\underline{t}) = \underline{f}_i(\underline{t}) + \underline{\epsilon}_i(\underline{t}),$$

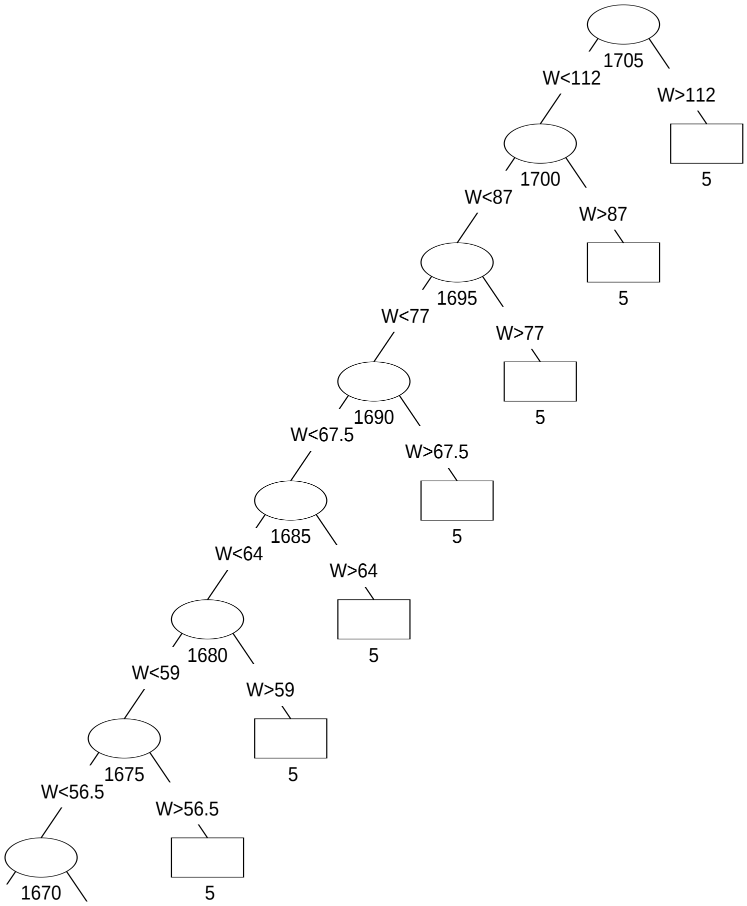

Figure 1: The root node of the fitted multivariate tree and the first several branches for the raw time of day fractions when start times are binned in half hour intervals. Below each node is the sample size. The fitted tree is unsuccessful, because each split pulls off only five responses, which is the minimum node size used to fit this tree. Time (in days) between the first two calls (W) is the only variable used to split nodes. The shape of the fitted tree does not change if the minimum node size is increased to 10 or 20.

where

$$\underline{f}_i(\underline{t}) = \sum_{j=1}^{q} \delta_{ij}\beta_j(\underline{t}) \tag{1}$$

for a set of basis functions $\beta = (\beta_{\mathbf{1}}(\underline{\mathbf{t}}), \ldots, \beta_{\mathbf{q}}(\underline{\mathbf{t}}))$ and a coefficient vector $\underline{\delta}_i = (\delta_{i1}, \ldots, \delta_{iq})^T$, where $\epsilon_i(\underline{t})$ is white noise with mean zero and constant variance. Then the $q \times n$ estimated coefficient matrix $\hat{\delta} = (\underline{\hat{\delta}_1}, \ldots, \underline{\hat{\delta}_n})$ minimizes the penalized sum of squares

$$S(\delta) = \sum_{i=1}^{n}\sum_{j=1}^{m}(Y_i(t_j) - f_i(t_j))^2 + \alpha J(f_i),$$

where $J(f) = \int f''(t)^2 dt$. Green and Silverman (1994) show that $\hat{\delta}$ is defined by

$$\hat{\delta} = (\beta^T\beta + \alpha K)^{-1}\beta^T \mathbf{Y}(\underline{t}), \tag{2}$$

where β is the $m \times q$ basis matrix, $\mathbf{Y}(\underline{t})$ is the $m \times n$ matrix of responses, α is a smoothing parameter that is the same for all customers, and $K_{jk} = \int \beta_j''(t)\beta_k''(t)dt$. The smoothing parameter can be chosen by cross-validation (see Green and Silverman, 1994). Or, α can be chosen to minimize the generalized cross-validation score

$$GCV(\alpha) = (mn)^{-1}\frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_i(t_j) - \hat{f}_i(t_j))^2}{(1 - m^{-1}trA(\alpha))^2},$$

where

$$A(\alpha) = \beta(\beta^T\beta + \alpha K)^{-1}\beta^T.$$

A spline is fit to each response curve using the Splus function `n.spline(t, knots, degree, nderiv)`, where $\underline{t}$ is the vector of midpoints for the relative frequency intervals, `knots` is the vector of pre-specified interior knots, `degree` specifies the degree of the piecewise polynomial fit, and `nderiv` specifies the number of derivatives evaluated for the roughness penalty. We use `degree` =3 and `nderiv`=2 in our example. The function `n.spline` returns the $m \times df$ basis matrix β, where df is the number of interior knots plus one when each basis function is centered so that the sum of each basis function evaluated at all the knots is zero. The matrix of coefficients $\underline{\hat{\delta}}_1, \ldots, \underline{\hat{\delta}}_n$ is obtained by a penalized regression of $\mathbf{Y}(\underline{t})$ on the basis matrix $\beta(\underline{\mathbf{t}})$ with roughness penalty J.

The estimated coefficients $\underline{\hat{\delta}}_i, i = 1, \ldots, n$ are used as responses in a multivariate regression tree. Fitting proceeds as in Section 3, except now the "responses" are estimated coefficient vectors instead of time of day relative frequency vectors and the "prediction" at a node is the mean estimated coefficient vector for responses in the node. The prediction error for one response in the node is measured by the standardized squared error loss (Mahalanobis distance)

$$D(\underline{\hat{\delta}}_i, \bar{\underline{\hat{\delta}}}) = (\underline{\hat{\delta}}_i - \bar{\underline{\hat{\delta}}})^T\beta^T\beta(\underline{\hat{\delta}}_i - \bar{\underline{\hat{\delta}}})$$

of the estimated coefficient $\underline{\hat{\delta}}_i$ from the prediction $\bar{\underline{\hat{\delta}}}$ for the node, where β is the basis function. As before, splitting proceeds by comparing the distance before splitting to the distance after splitting, choosing the split that gives the largest decrease in the total distance.

The predicted curve $\underline{\hat{f}}(\underline{t})$ at each node is computed as

$$\underline{\hat{f}}(\underline{t}) = \sum_{j=1}^{q} \bar{\underline{\hat{\delta}}}_j \beta_j(\underline{t}) \tag{3}$$

where $\bar{\underline{\hat{\delta}}}$ is the mean coefficient vector for the node. A 95% approximate pointwise confidence interval for $\underline{\hat{f}}(\underline{t})$ is computed as

$$\underline{\hat{f}}(\underline{t}) \pm 1.96 \frac{s_{node}(\underline{t})}{{n_{node}}^{1/2}}, \tag{4}$$

where $s_{node}(\underline{t})$ is the vector of pointwise sample standard deviations of $\underline{\hat{f}}(\underline{t})$ at the node and n_{node} is the number of responses at the node. As always, these standard errors are only appropriate if the fitted model represents the data well.

In our example and many others, it is important to control the number of final nodes in the tree. "Pruning" a tree both reduces the tendency to overfit and produces simpler prediction rules (see Breiman *et al* 1984 for a discussion of pruning). In practice, we prune using the Splus functions `prune.tree` and `snip.tree` (see Chambers and Hastie, 1992, chapter 9). The Splus function `polygon.tree` is used to plot the fitted tree.

5 An Example: Spline Trees Fit to the International Call Data

The start times for each customer were binned into half hour intervals, and natural splines with interior knots at (10a.m., 1p.m., 5p.m., 7p.m., 10p.m.) were used to approximate each customer's curve. A tree was fit to the 1705×6 matrix of fitted coefficients and pruned to eight nodes. Figure 2 shows the fitted tree after pruning. The fitted tree is interpretable and can be computed quickly. The same fitted tree is obtained, and just as quickly, when start times are binned into fifteen minute intervals instead of half hour intervals. If start times are binned into five minute or one minute interval, very similar trees are produced nearly as quickly.

The first split in Figure 2 separates Western Europe, Eastern Europe and Middle East from other regions. The second split separates short waiting time calls from longer ones, and South America from Africa, Asia and India. The tree then splits short duration calls from longer calls. The business/other indicator does not appear at any of the splits in the pruned tree, but the split on duration may be a surrogate for the business indicator, because businesses tend to make shorter calls than residences. We were surprised that day of week is not used at any of the splits, but trees fit separately to weekday and weekend time of day distributions, are similar, supporting the absence of day of week effects. The major difference between the weekday and weekend fitted trees is that the estimated terminal node variance is larger for the tree for weekend calls, because more customers in our data have weekday calls than have weekend calls.

Figure 3 gives the mean time of day curve and the number of customers for each of the terminal nodes in the pruned tree. Approximate 95% pointwise confidence bands for the mean are also shown. The panels are arranged from left to right and from bottom to top, with the leaf nodes from the most left to right in Figure 2. The bottom left panel, for example, shows the mean fitted

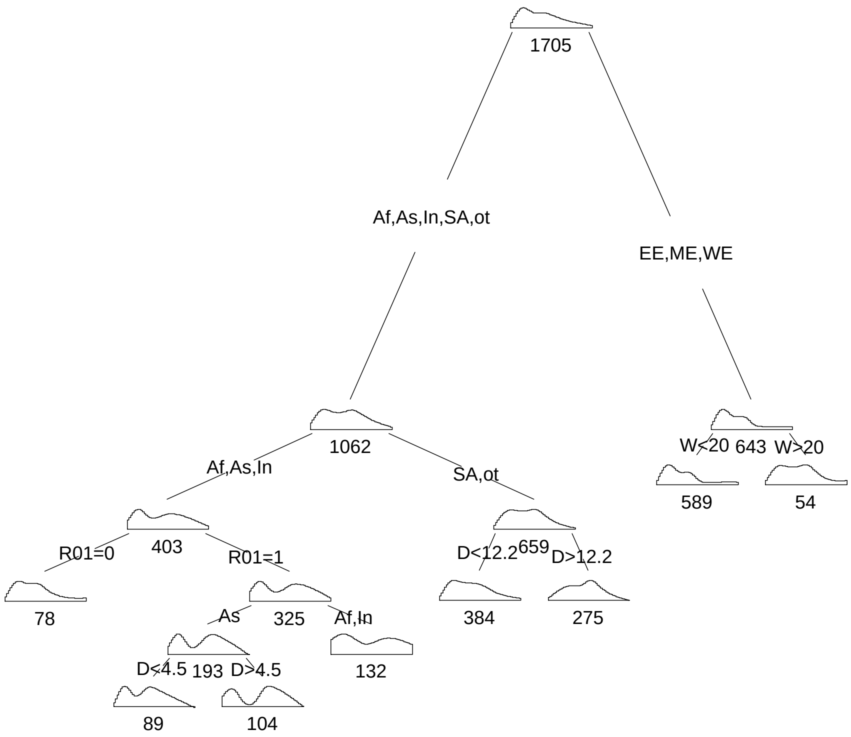

Figure 2: The fitted tree was obtained by binning the time of day data to half hour intervals and taking the natural spline coefficients as response vectors. The tree was pruned to have eight terminal nodes. The curve at each node displays the mean prediction for the node. Below each node is the sample size. The abbreviations Af, As, EE, In, ME, SA, WE and ot denote that the customer's first international call went to Africa, Asia, Eastern Europe, India, Middle East, South America, Western Europe or "other" respectively. R01 indicates whether the first two calls went to the same region of the world (R01=1) or different regions of the world. (R01=0). The other variables are duration (D, in minutes) of the first two calls and waiting time (W, in days) between the first two calls.

curve for the 483 customers whose first call did not go to Western Europe and whose first two calls had a mean duration less than 5 minutes. The middle panel shows the mean fitted curve for the biggest group, which contains the 495 customers whose first call went to Middle East or South America and first two calls had a mean duration longer than five minutes.

Finally, Figure 4 shows the fitted tree obtained using the estimated coefficients for a cubic B-spline basis with the same fixed knots at (10a.m., 1p.m., 5p.m., 7p.m., 10p.m.). The initial nodes obtained with B-splines and natural splines are identical, but lower nodes are different. Also, the mean fitted time of day curves for the B-spline tree have modes in the tails, while the natural spline trees do not. This is not surprising, because natural splines have linear rather than cubic tails. In our example, the B-spline modes at midnight are not desirable. Matching B-splines to natural splines in terms of degrees of freedom rather than knots does not remove the modes at midnight.

6 Stability of A Spline Tree for Functional Data

An obvious question is how reliable is a spline tree? More refined questions are how good is a tree under some probability model and loss function and how can we judge the goodness of fit for a fitted tree in a particular application? Because we advocate using trees for exploratory analysis, we focus on only the second question of goodness-of-fit.

We assess the stability of the tree by straightforward bootstrapping. That is, we re-sample, with replacement, the 1705 customers in our data to form a new sample of size 1705, each observation consisting of two initial calls and a vector of binned time of day fractions. A spline tree was fit to each of 200 bootstrap samples. The entire fitting procedure, from natural spline fitting to pruning and computing mean response curves at each terminal node, was repeated for each bootstrap sample. For each tree, goodness-of-fit was measured by taking the total squared prediction loss $\sum_i \parallel \underline{Y}_i(\underline{t}) - \underline{\hat{f}}(\underline{t}) \parallel^2$, where $\underline{\hat{f}}(\underline{t})$ is the mean fitted response at the node containing $\underline{Y}_i(\underline{t})$. Three kinds of basis functions were considered for the trees: natural splines with fixed knots, B-splines with the same fixed knots (but, as a consequence, more degrees of freedom), B-splines with fewer fixed knots but the same degrees of freedom as natural splines, and B-splines with automatic knot selection. The same automatic knots were used for all curve responses and were only varied among bootstrap samples. The knots were chosen by using the Splus function `bs`, where $df - degree - 1$ number of interior knots are chosen at uniformly spaced quantiles of $\underline{t}$.

The results of the bootstrapping are given in Table 1. As would be expected, automatic knot selection gives substantially larger prediction errors, and more degrees of freedom give smaller prediction errors. Matching the natural spline and B-spline fits for degrees of freedom gives similar prediction error for the two kinds of trees, with the smaller error going to natural spline fits. Judging from the bootstrapped standard deviation of the prediction error, the difference between the trees fit to the natural and B-spline estimated coefficients matched on degrees of freedom is barely statistically significant.

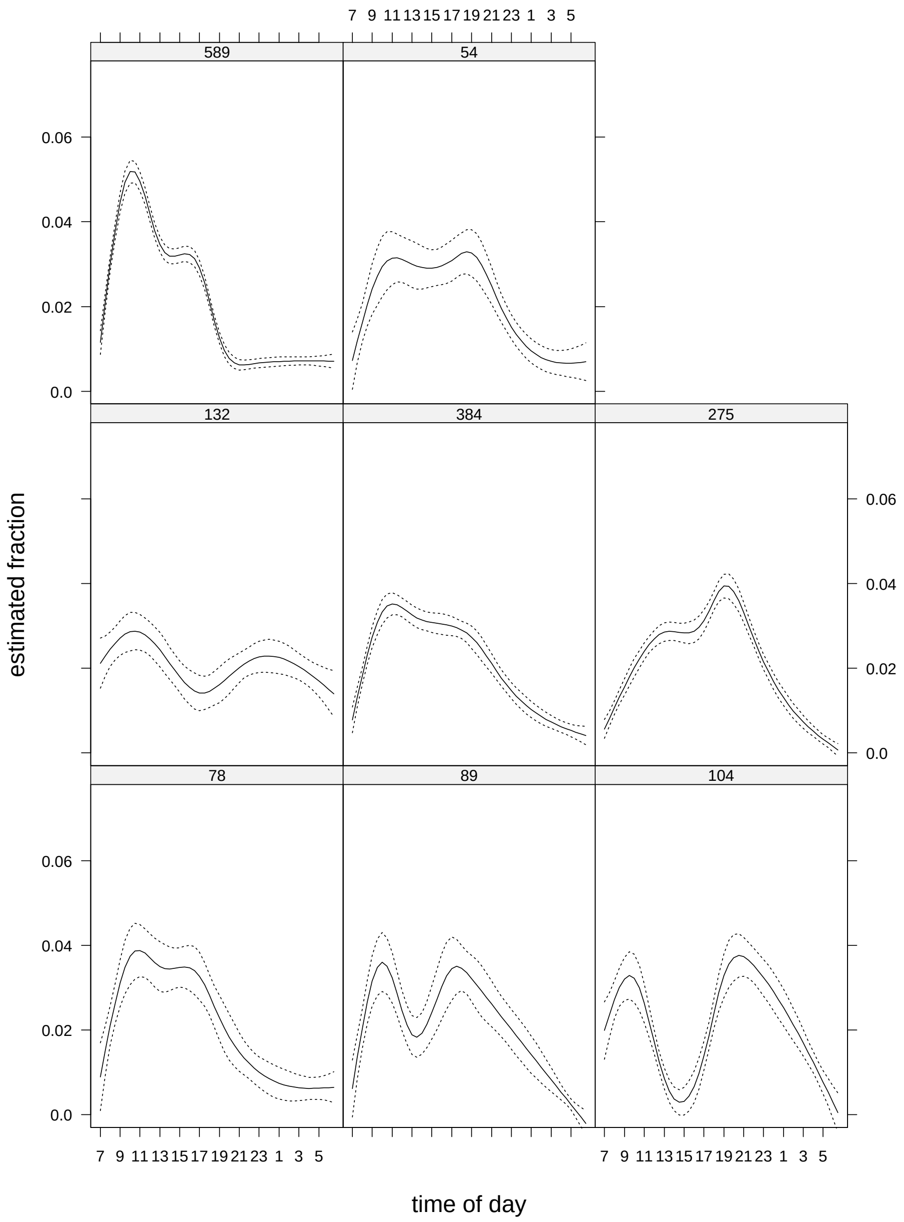

Figure 3: Mean time of day fractions and 95% confidence intervals for each leaf node of the fitted tree shown in Figure 2. The mean time of day fractions here are linear combinations of the spline basis functions with weights equal to the mean coefficient vector at the node. The panels from left to right and bottom to top correspond to the terminal nodes reading from left to right in Figure 2.

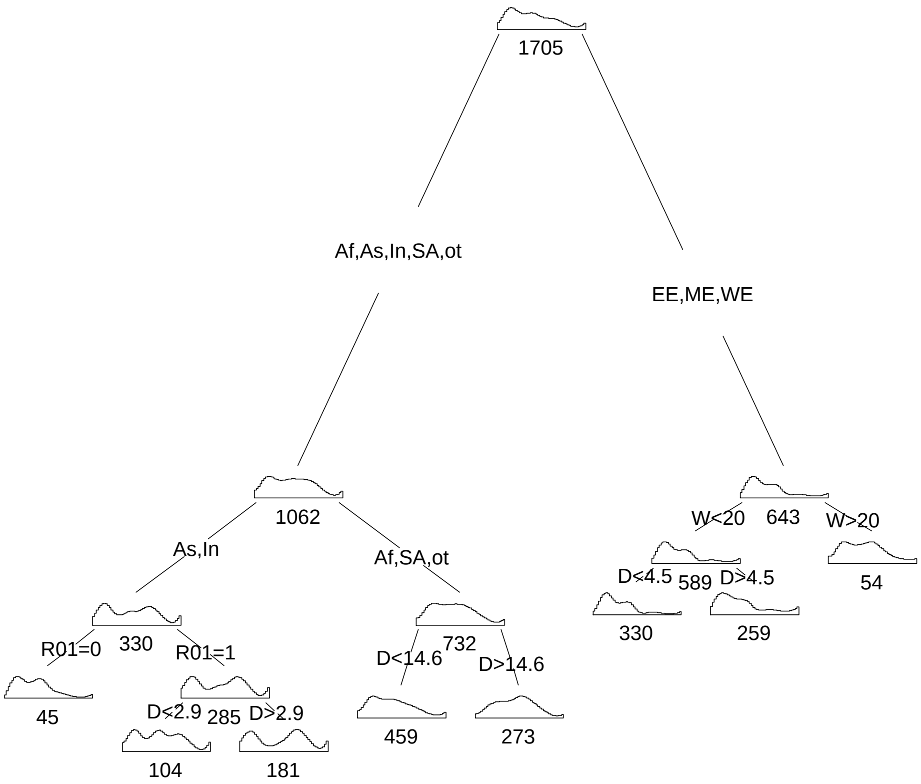

Figure 4: Tree fit to the binned half hour fractions using fitted coefficients for the B-spline basis functions and the interior knots used to fit the natural spline tree. Note that the mean response curves at the nodes have modes in their right tails.

Table 1: Results of bootstrapping natural and B-spline trees to the time of day data binned into half hour intervals. Prediction error is the total (over all responses) squared distance from the binned response to the mean fitted response curve for the node containing the response. The 50% confidence limits are the lower and upper quartiles of the prediction error over the 200 bootstrap samples. The 90% confidence limits are the lower and upper .05 quantiles of prediction error over the bootstrap samples. Bootstrap standard deviation is the sample standard deviation of the prediction error over the bootstrap samples.

Type of splines	natural	B-splines	B-splines	B-splines
Degrees of freedom	6	6	8	6
Choice of knots	fixed	fixed	fixed	automatic
prediction error	56.3	59.0	51.1	74.3
50% confidence limits	(55.4,57.2)	(58.2,59.7)	(50.2,51.9)	(73.2,75.4)
90% confidence limits	(54.2,58.6)	(56.7,61.0)	(49.1,53.1)	(71.7,76.7)
bootstrap standard deviation	1.33	1.28	1.20	1.53

7 Principal Component Trees

Instead of reducing the dimension of the response by treating it as a curve, we now reduce the dimension by treating it as a vector and applying principal component analysis, retaining only the first several principal components. That is, take

$$\underline{\delta}_i = \sum_{j=1}^{m} \beta_j \underline{Y}_i(t_j),$$

where β_j is the weighting coefficient and the principal component scores $\underline{\delta}_i$ are the uncorrelated linear combinations $\underline{Y}(\underline{t})$ of the response with variances that are as large as possible. See Anderson (1984) for more details.

Figure 5 gives the first six principal component loadings of the time of day fractions for our data. The first component contrasts calls placed during peak business hours with calls placed during the evening and early morning, which can be considered as a contrast between business and residential customers. This component explains 15.8% of the total variance. The first six principal components together explain 45% of the total variance, while the natural spline basis used in section 4 explains 42% of the total variance. Figure 6 shows the tree fit to the first six principal component scores of the time of day fractions $\underline{Y}_i(\underline{t})$. The splits are sensible. They are also stable, in the sense that the same fitted tree is obtained if the data are binned in five minute intervals. The computations are slow for finer binning, however.

The predicted curve $\underline{\hat{f}}(\underline{t})$ at each node is again computed by the average of the $\underline{\hat{f}}_i(\underline{t})$ in the node as in equation 3. The prediction errors for the spline tree and principal component tree for our data are similar.

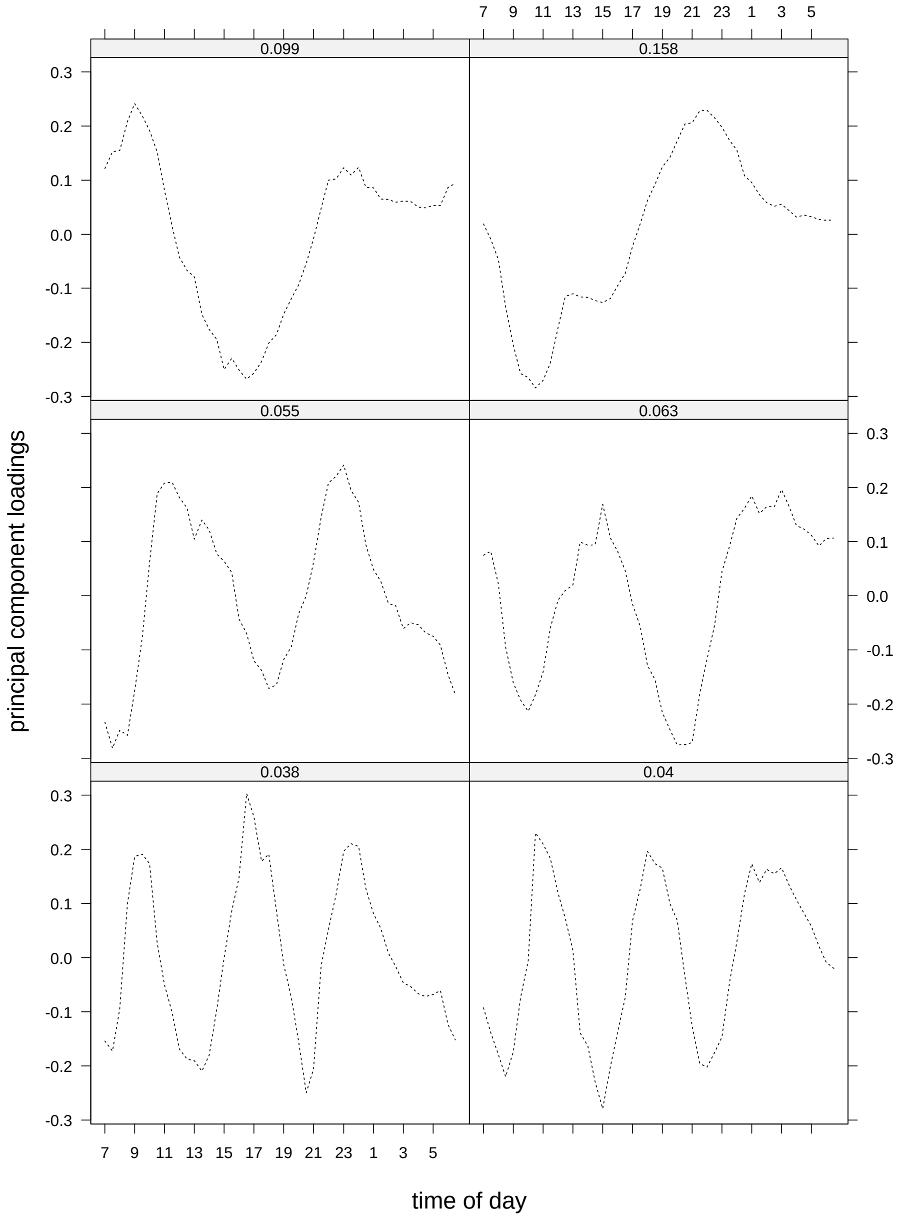

Figure 5: The first six principal component loadings of the time of day fractions for half hour intervals are displayed from the top right to the bottom left panels. The numbers in the top margins indicate the amount of total variation accounted for by each principal component.

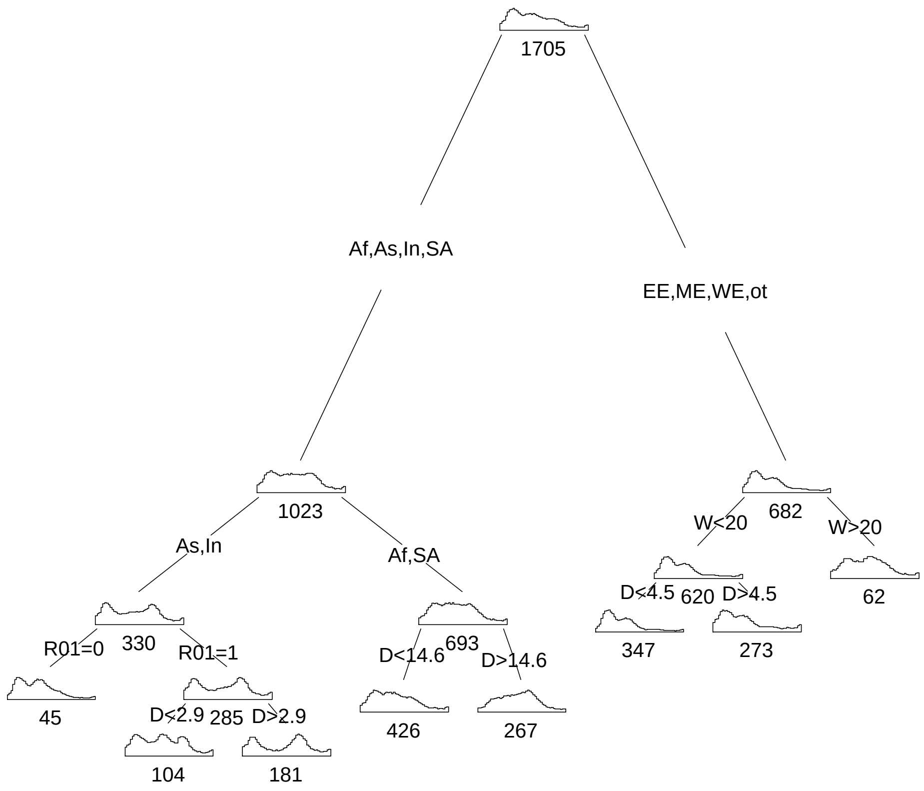

Figure 6: The fitted principal component tree for the half hour time of day data. Six principal component scores were used and The tree was pruned to have eight terminal nodes. The predicted time of day fractions for the nodes were computed by averaging the fitted values in the node.

8 Discussion

We have explored two ways to fit trees to high dimensional responses that can be thought of as curves and applied both to time of day data. The basic idea is to reduce each response curve to a low dimensional vector. In the first approach, we use the coefficients of a spline basis function to represent the curves. We advocate using natural splines, because their tails are less "floppy" than B-splines, but other basis functions may be preferable in other applications. For example, Fourier basis functions can be used for periodic data. Wavelet bases do not fit into our framework, because they do not always produce a fixed dimensional representation of a curve. Loess and kernel smoothers do not fit into our framework, because they are not fixed dimensional representations of curves.

In the second approach, the dimension of the response is reduced by principal components analysis, and a tree is fit to the first several principal components scores. The principal component tree for our data is similar to the spline tree and has nearly the same prediction error. Both spline trees and principal component trees can be easily generalized to include weights, transformations of the response, or more general loss functions, for example.

There are other ways to reduce a curve to a low dimensional vector that might be more effective in some settings. For example, the values of the curve at a set of fixed points might be useful, or a set of features of the curve such as its average, maximum and minimum values and its average derivative might be useful ways to summarize the response. These kinds of reductions may need to be tailored to the application, however, and have not been explored here.

References

Anderson, T.W. (1984), *An Introduction to Multivariate Statistical Analysis* (2nd ed.), New York: John Wiley.

Becker, R.A. and Cleveland, W.S. (1996), *S-PLUS Trellis Graphics*, Seattle, WA: MathSoft, Inc.

Becker, R.A., Chambers, J.M., and Wilks, A.R. (1988), *The New S Language*, Belmont MA: Wadsworth.

Breiman, L., Friedman, J.H., Olshen R.A. and Stone, C.J. (1984), *Classification and Regression Trees*, Belmont MA: Wadsworth.

Clark, L.A., and Pregibon, D. (1992), "Tree-based Models," Chapter 9 in Chambers and Hastie (1992).

Chambers, J.M. and Hastie, T.J. (1992), *Statistical Models in S*, London: Chapman and Hall.

De Boor, C. (1978), *A Practical Guide to Splines*, New York: Springer-Verlag.

Green, P.J., and Silverman, B.W. (1994), *Nonparametric Regression and Generalized Linear Models*, London: Chapman & Hall.

Ramsay, J.O. and Silverman, B.W. (1997), *Functional Data Analysis*, New York: Springer.

Rice, J.A., and Silverman, B.W. (1991), "Estimating the Mean and Covariance Structure Nonparametrically When the Data are Curves", *Journal of the Royal Statistical Society,* Ser.B, 53, 233-243.

Segal, M.R. (1994), "Representative Curves for Longitudinal Data via Regression Trees", *Journal of Computational and Graphical Statistics*, vol.3,2,214-233.

Venables, W.N., and Ripley, B.D. (1994), *Modern Applied Statistics with S-Plus*, New York: Springer-Verlag.