Biostatistics (2006), **7**, 1, *pp.* 71–84
doi:10.1093/biostatistics/kxi041
Advance Access publication on June 22, 2005

Extreme regression

MICHAEL LEBLANC*, JAMES MOON, CHARLES KOOPERBERG

Fred Hutchinson Cancer Research Center, 1100 Fairview Avenue North, M3-C102, Seattle, WA 98109, USA
mleblanc@fhcrc.org

SUMMARY

We develop a new method for describing patient characteristics associated with extreme good or poor outcome. We address the problem with a regression model composed of extrema (maximum and minimum) functions of the predictor variables. This class of models allows for simple regression function inversion and results in level sets of the regression function which can be expressed as interpretable Boolean combinations of decisions based on individual predictors. We develop an estimation algorithm and present clinical applications to symptoms data for patients with Hodgkin's disease and survival data for patients with multiple myeloma.

Keywords: Decision rules; HARE; Non-linear MARS; Prognostic groups; Regression; Survival; Tree-based models.

1. INTRODUCTION

Describing clinical, laboratory, and genomics values associated with good or poor patient outcome, such as drug toxicity, quality of life (QOL), response, or survival, is frequently of interest to clinical researchers. For instance, one may want to describe a group of patients with a particular type of cancer who have an estimated probability of 1-year survival of less than 0.30. Similarly, one may be interested in describing the 25% of patients in the study population with the worst survival, or investigate a sequence of rules as the fraction of patients in the worst prognosis group varies.

In this paper, we consider two specific applications that motivate a new regression method for describing extreme outcome groups. First, we consider patient symptoms in Hodgkin's disease (HD) patients. Health status and QOL were evaluated prospectively in 227 patients with early stage HD treated on Southwest Oncology Group (SWOG) Study S9133, comparing subtotal lymphoid irradiation with combined modality treatment (Press *et al.*, 2001; Ganz *et al.*, 2003). We construct simple rules to describe patients with poor symptom scores based on two other QOL measures. Second, we consider finding subsets of multiple myeloma patients with very poor prognosis. It is clinically interesting to describe a subgroup of patients with sufficiently poor prognosis for whom more aggressive therapy is appropriate. The data are based on several SWOG clinical trials which investigated multidrug combinations and schedules for treating myeloma. We construct a rule corresponding to approximately 30% of patients with poorest prognosis and then investigate the sequence of rules as the fraction of patients in the extreme subgroup varies.

We will utilize a new regression strategy for describing values of predictors associated with a target outcome. The model uses nonlinear low-dimensional expansions of the predictor variables consisting

*To whom correspondence should be addressed.

of combinations of 'minimum' and 'maximum' operations on univariate linear functions of predictors. Unlike commonly used linear and generalized additive models, the new model directly leads to simple decision rules based on intersections and unions of simple statements involving single covariate, e.g. ($x_1 > 3.5$ and $x_2 \leqslant 6.1$), or $x_3 > 1$.

2. BACKGROUND

Assume a continuous response variable depends on p predictor variables through a target function, $h(x) = h(x_1, \ldots, x_p)$, modulated by noise

$$y = h(x_1, \ldots, x_p) + \epsilon.$$

Based on a sample of observations $\{(y_i, x_{1i}, \ldots, x_{pi}); i = 1, \ldots, n\}$, an approximation function $\widehat{h}(x_1, \ldots, x_p)$ can be constructed. Typical uses for the fitted regression function include prediction of outcomes on future values of the predictor variables or characterizing the association between the predictors and the response. There are many commonly used statistical procedures for arriving at estimates of $h(x_1, \ldots, x_p)$, including linear models, neural networks, tree-based models, kernel methods, and models using splines. If the goal is primarily to understand the relationship of a small number of predictors to the response, models specified as a low-dimensional expansion of the predictors such as linear models, more general additive models, or models with at most simple interactions are particularly useful. Interpretation of these regression models is facilitated by estimates or plots conditional on subsets of the variables in the models. However, the level sets of these functions can be difficult to use. For instance, even the linear regression model $h(x) = x'\beta$ leads to complex-level set rules to describe the cases for which $\{x\colon x'\beta \geqslant q\}$. For nonlinear additive models, the situation becomes even more complex. See, for example, Figure 1, which represents two additive functions and a contour plot of their sum. On the other hand, tree-based methods

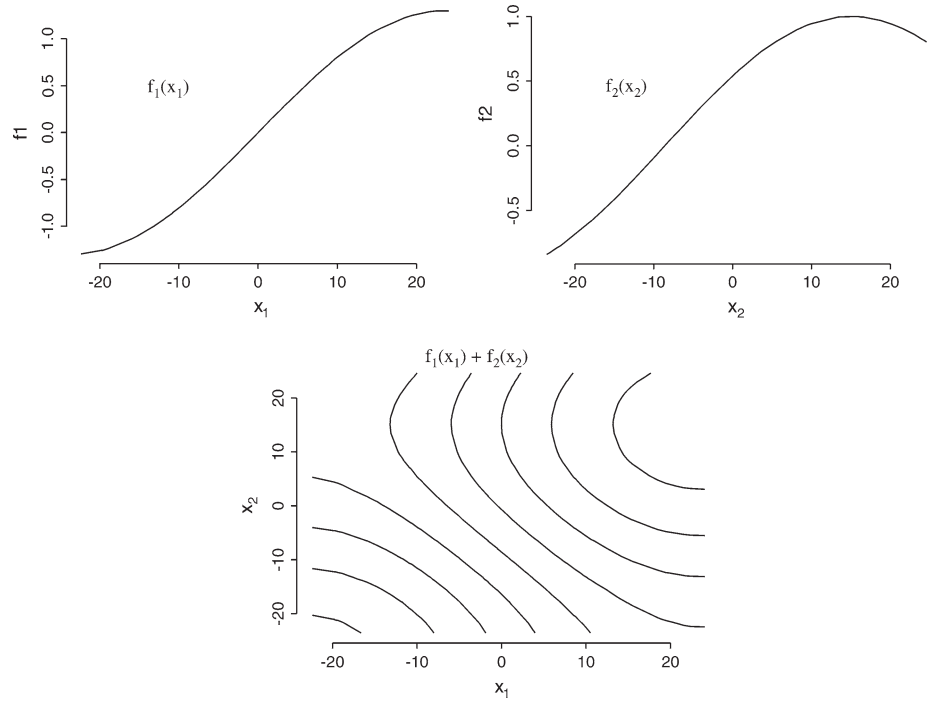

Fig. 1. Contour plot of an additive function of two variables.

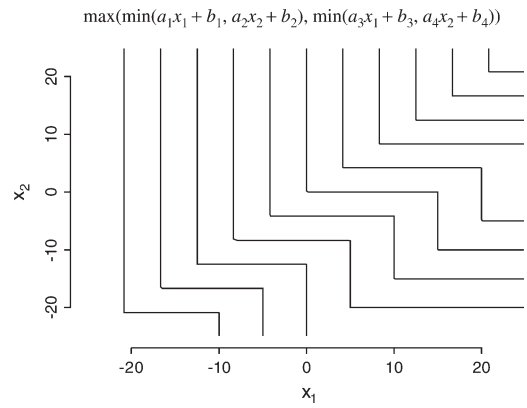

Fig. 2. Level sets for an extrema regression function.

(e.g. Breiman *et al.*, 1984) yield binary univariate decision rules that describe M terminal nodes that are disjoint regions $T_m, m = 1, \ldots, M$. The tree, $T(x)$, assigns the same value for all cases within each region, $x \in T_m \Rightarrow T(x) = \eta_m$. Therefore, the level set for a tree $\{x\colon T(x) \geqslant q\}$ will just be the union of those regions T_m for which $\eta_m \geqslant q$. It is this easily described inverse property that makes trees so desirable for constructing prognostic rules for clinical applications. One potential downside to inverting tree-based models relates to their discreteness. Consider a tree with only a few nodes representing significant fractions of the data. As one changes the threshold q, the percentage of the sample corresponding to $T(x) \geqslant q$ often changes in large jumps. Methods that construct ensembles of trees (e.g. via boosting) reduce the discreteness, but at the cost of losing the simple description of the level sets.

Our strategy is to use extrema functions which yield simple interpretations of level sets similar to those from tree-based models. But, akin to linear or additive models, these models are a smooth function of the predictors. We display resulting contours of a simple hypothetical extrema model in two dimensions in Figure 2.

We note that an alternative strategy to describe extreme outcome groups that does not model the underlying regression function is the patient rule induction method (PRIM) of Friedman and Fisher (1999). The PRIM method allows calibration of the group in terms of mean of the outcome y to describe box-shaped regions, $B = \{x\colon \bar{y}_B \geqslant q\}$ for box mean, $\bar{y}_B$. That method adaptively refines boxes in the predictor space and removes the data corresponding to one small region at a time in a stepwise fashion to construct rules. It has the advantage of being nonparametric and works well at estimating a single decision boundary for problems with large data sets.

By contrast, some advantages to using our underlying function approximation approach are that it should work well in situations with a smaller number of cases due to its more parametric form and it allows construction of a nested sequence of rules as the threshold q varies.

3. Model

Our proposal is to replace the usual additive main effect and multiplicative interaction regression model with extreme operators of maximum and minimum. An example of such a model is

$$f(x) = \max(\min(\beta_{01} + \beta_{11}x_1, \beta_{02} + \beta_{12}x_2), \min(\beta_{03} + \beta_{13}x_3, \beta_{04} + \beta_{14}x_4)).$$

More generally, we can write

$$f(x) = \max_j(f_1(x),\ f_2(x), \ldots,\ f_J(x)), \tag{3.1}$$

where each minimum term $f_j(x) = \min(g_{j,1}(x), \ldots, g_{j,K(j)}(x))$. Each of the component functions, $g_{j,k}(x)$, depends only on a single predictor. Label k denotes the component model of the jth 'min' term, where $j = 1, \ldots, J$ and $k = 1, \ldots, K(j)$, and $K(j)$ is the number of linear component functions in each 'min' term j. We use a simple univariate linear predictor

$$g_{j,k}(x) = \beta_{0,k} + \beta_{1,k} x_{l(k)},$$

where $l(k)$ is the label of the predictor in the kth term of component model j (formally we should write $l(k, j)$, $\beta_{0,j,k}$, and $\beta_{1,j,k}$ but we suppress the j for readability). The following development for the linear model could easily be extended to $g_{j,k}(x)$ functions that are more general smooth univariate functions of predictors.

The overall model is a continuous piecewise linear model that locally depends only on a single predictor variable on each partition $R_{j,k}$ defined as $\{x\colon g_{j,k}(x) = f(x)\}$. We denote the data points which fall into region $R_{j,k}$ as the active set of points associated with function $g_{j,k}(x)$. Adaptive function approximation methods using piecewise linear component functions have been successfully and widely used [e.g. multivariate adaptive regression splines (MARS, Friedman, 1991) and for survival analysis hazard regression (HARE, Kooperberg *et al.*, 1995)]. However, the extrema function formulation places different constraints on the nature of the piecewise linear model which are useful for describing the inverse of the regression function.

It is easily seen that the description of any q-level set $\Omega = \{x\colon f(x) \geqslant q\}$ is just

$$\Omega = \{x\colon \mathrm{OR}_j(g_{j,1}(x) \geqslant q \text{ AND} \cdots \text{AND } g_{j,K(j)}(x) \geqslant q)\}.$$

This is referred to as a Boolean expression in disjunctive normal form (a union of intersections of simple terms). It is critical that each component function $g_{j,k}(x)$ is a function of a single predictor for Ω to retain its form as an interpretable decision rule.

3.1 *A simulated example*

Prior to discussing the details of the estimation algorithm, we present an example of the fitted model where data were generated from the model

$$y = \max(\min(x_1, 0.5x_2), \min(-x_3, x_4)) + \epsilon,$$

where the covariates and the ϵ are all independent with a standard normal distribution. Figure 3 gives a graphical representation of the fitted model and shows the active points in each of the regions $R_{j,k}$ associated with each univariate function $g_{j,k}(x)$. The two panels on the left-hand side of the plot correspond to the first 'minimum' term and those on the right-hand side to the second 'minimum' term. Figure 4 gives a simple graphical representation of the inverse function plot. The shading indicates the directions of the decision rules. To read the plot, note that a level set rule is obtained by first taking the intersection within each column and then taking the union across columns. Therefore, the plot indicates a sequence of decision rules of the form $(\{x_1 \geqslant c_1(q)\} \text{ AND } \{x_2 \geqslant c_2(q)\}) \text{ OR } (\{x_3 < c_3(q)\} \text{ AND } \{x_4 \geqslant c_4(q)\})$ corresponding to any level set of the fitted model.

4. ESTIMATION

For squared error loss, the estimation problem can be represented as a minimization

$$L(\beta) = \sum (y_i - f(x_i))^2$$

subject to $g_{j,k}(x_i) \geqslant f_{j'}(x_i)$ and $g_{j,k}(x_i) \leqslant g_{j,k'}(x_i)$ for $x_i \in R_{j,k}$, where $f_j(x_i) = \min(g_{j,1}(x_i), \ldots, g_{j,K(j)}(x_i))$ for observations (y_i, x_i), $i = 1, \ldots, n$. Therefore, for a given partition the optimization is just

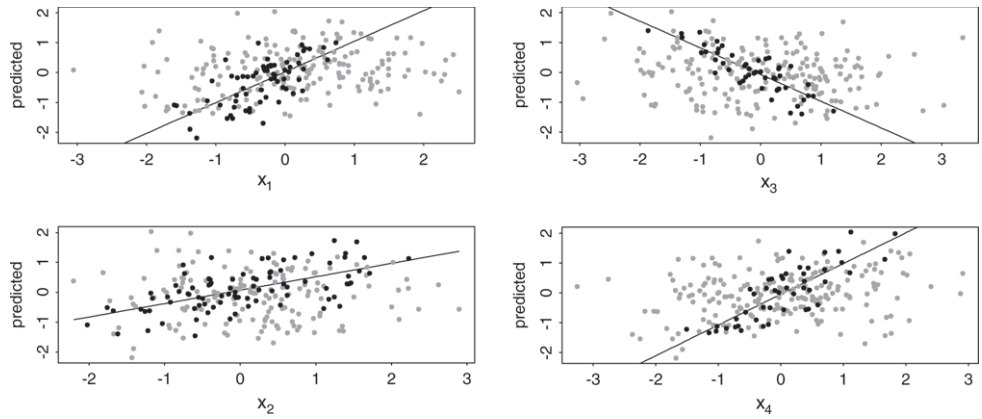

Fig. 3. Fitted functions to simulated data. Black dots refer to points associated with the given fitted function. Columns represent 'minimum' model terms.

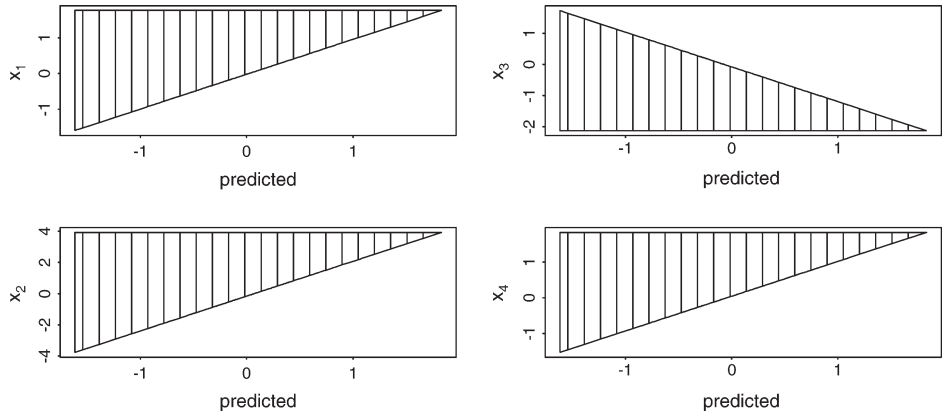

Fig. 4. Graphical representation of the inverse function decision rules $((\{x_1 \geqslant c_1(q)\} \text{ AND } \{x_2 \geqslant c_2(q)\})) \text{ OR } ((\{x_3 < c_3(q)\} \text{ AND } \{x_4 \geqslant c_4(q)\}))$ for simulated data.

a quadratic programing problem with linear inequality constraints. If there are $M = \sum_{j=1}^{J} K(j)$ component linear functions, then for each observation there are $M-1$ inequalities that must be satisfied, yielding a total of $n(M-1)$ inequality constraints. However, this optimization problem depends on a prespecified partition which assigns the data points to each one of the M linear component functions. For example, in Figure 3 the highlighted simulated points in each of the four panels show the assignment of cases for each of the four linear component functions. The much more challenging aspect of the optimization is finding an optimal or at least a good partition. As global searches would not be computationally feasible, we use an alternative algorithm for finding good estimated models.

For our estimation strategy, it is helpful to note that the objective function can also be represented as a weighted sum of squares of the M component linear models

$$L(\beta) = \sum h_{(j,k)}(x_{i,l(k)})(y_i - g_{j,k}(x_{i,l(k)}))^2,$$

where the weight function is an indicator $h_{(j,k)}(x_{i,l(k)}) = I\{g_{j,k}(x_{i,l(k)}) = f(x_i)\}$. For a fixed partition (or weight function), the ordinary least squares estimates of the coefficients are straightforward to

calculate. In addition, given current parameter estimates one can determine the observations for which $\{x_i: g_{j,k}(x_{i,l(k)}) = f(x_i)\}$ to update the partition. An intuitive algorithm can incorporate these two facts as two steps: (1) estimation and (2) partition (reassignment of observations to groups). This algorithm is similar to the *K*-means algorithm and the hinge selection component of the hinging hyperplane algorithm of Breiman (1993). However, there is no assurance of finding a global optimum. In addition, in our experience during early steps of that algorithm, if many observations are reassigned to other partitions/groups at a single step of the algorithm, the residual sums of squares may actually increase from one iteration to the next. This convergence issue can be addressed by introducing a step-size Δ to the updating.

Algorithm: Local Linear Update

1. Set initial values for coefficients $\beta_{j,k0}$ and hence $g_{j,k0}(x)$.
2. Set $m = 1$ to loop over steps 3–4 until convergence. At convergence, no cases are reassigned groups.
3. **Partition:** Determine active sets $R_{j,km}$ corresponding to the kth model component in the jth term

$$\{x: g_{j,km}(x) = f(x)\}.$$

4. **Estimate:** Find unconstrained least squares estimates for $\widehat{\gamma}_{j,km}$ using data in all active sets, $R_{j,km}$. Update the coefficients

$$\beta_{j,k(m+1)} = \beta_{j,km} + \Delta \times (\widehat{\gamma}_{j,km} - \beta_{j,km}).$$

Let $m \leftarrow m + 1$.
The step-size Δ can be a fixed number, $\Delta \leqslant 1$. We also allow a step-size selection option, where step-size is successively reduced in magnitude (within each estimation step 4) until the resulting error sums of squares is no larger than for the previous iteration. We take $\Delta_s = \Delta \upsilon^{s-1}$, where $\upsilon = 0.75$ and $\Delta = 0.5$.

We note that a sufficiently small step-size will always lead to a decrease in the sum of squares. To see this, for the given partition, we calculate the least squares solution. Movement toward that solution will decrease the sum of squares. Continue until an observation is at the edge of two partitions. At that point, the observation can be reassigned to the other partition without any increase in sum of squares. However, now one can obtain least squares estimates given the new partition; then movement towards the new least squares estimates will again decrease the sum of squares. For computational reasons, such a small step-size is typically not practical.

We also constrain the number of observations used to estimate any component function. For instance, if $R_{j,km}$ at any step m identifies fewer than K_n observations, then the K_n observations with the smallest absolute values of $d = g_{j,k}(x) - f(x)$ are used for estimation. Our default is $K_n = \min\{50, \max\{0.05n, 25\}\}$.

Finally, while the algorithm can give good local minimum solutions, one can sometimes improve final residual error by restarting the algorithm a small number of times using the initial estimates with added noise. Our experience suggests that this is more an issue as the number of unique values of the covariates decreases. We have added this option to the software, but to simplify the description, we did not use it in the simulated and real data examples that follow.

Given the constrained nature of the estimation method, closed form estimates of the variance of the regression function are not readily available. For a given extreme regression (XR) model specification, we propose using nonparametric bootstrap estimates of the marginal standard errors or approximate confidence intervals. Approximate confidence bounds for rules for a single q-level set can be obtained by inverting the upper and lower bounds of the point-wise regression intervals.

4.1 *Model building*

Model building and variable selection can be implemented using greedy, semigreedy, or stochastic searches. We have implemented a simple stepwise algorithm where the model is expanded at each step by adding one additional linear component. This linear component can be added to an existing 'minimum' term or as a new linear component in the outer 'maximum' term. We rewrite the model at the mth iteration as $f^m(x) = \max_{j=1}^{J^m}(u_j^m(x))$, where $u_j^m(x) = \min(g_{j,1(j)}(x), \ldots, g_{j,K(j,m)}(x))$.

Algorithm: Stepwise Model Building

1. Start with a single univariate linear model. That is, $J^1 = 1$ and $K(1,1) = 1$. The linear term would correspond to the best single linear predictor.
2. Loop over predictor variables and positions in the model.
3. Consider adding a new linear component model $h(x) = a+bx$ for each of the p predictors. Potential models are as follows: A new univariate linear term to an existing 'min' term for a particular j, $1 \leqslant j \leqslant J^m$,
$$u_j^{m+1}(x) = \min(g_{j,1(j)}(x), \ldots, g_{j,K(j,m)}(x), h(x)),$$
so that $K(j, m+1) = K(j, m)+1$ and $J^{m+1} = J^m$, and models for which an additional simple term is added. Here, the minimum term would only be the univariate linear function $h(x)$, $u_{J^m+1}^{m+1}(x) = h(x)$, then $J^{m+1} = J^m + 1$ and $K(J^{m+1}, m+1) = 1$. All other components remain unchanged.
4. Refit all potential models using the local linear update algorithm.
5. The new model is the allowable model that reduces the error the most.

The model size can be chosen using a less biased estimate of the prediction error than the residual error on the training data set such as k-fold cross-validation (or resampled averaged k-fold cross-validation), a low-bias method for selecting complexity. We have used a computationally cheap generalized cross-validation (GCV) estimate with a penalty parameter to acknowledge selection of terms. We use the default penalty of 1.5 per parameter in the model similar in spirit to the default GCV penalty used in the MARS algorithm.

5. Patient symptoms in HD

In this section, we investigate the association between patient symptoms (measured by symptom distress scale) and two other QOL measures for patients with early stage HD. Data were gathered from 227 patients with early stage HD treated on an SWOG Study comparing subtotal lymphoid irradiation with combined modality treatment (Press *et al.*, 2001; Ganz *et al.*, 2003). Measures in this analysis were collected at baseline; a more detailed analysis of the QOL data was presented by Ganz *et al.* (2004, ASCO). The two predictor QOL measures included Cancer Rehabilitation Evaluation System (CARES) physical (phys) and CARES psychosocial (psyc). The outcome was patient symptoms (symptom distress scale, sds) scores. Our goal was to construct a simple model to characterize QOL attributes associated with higher patient symptom scores. We specified a model where it was sufficient to have either extreme CARES physical or CARES psychosocial scores to impact symptom score. Such a model can be expressed as a 'maximum' model
$$\max(g_1(\text{phys}), g_2(\text{psyc})).$$

Approximately two-thirds of the cases (150) were analyzed, with the remaining 77 cases retained as a test sample. The fitted XR model and data points corresponding to each of the submodels are represented in Figure 5. Patients with either an increased (worse) CARES psychosocial score or an increased (worse) CARES physical score had increased (worse) symptom scores. Rules associated with any given symptom

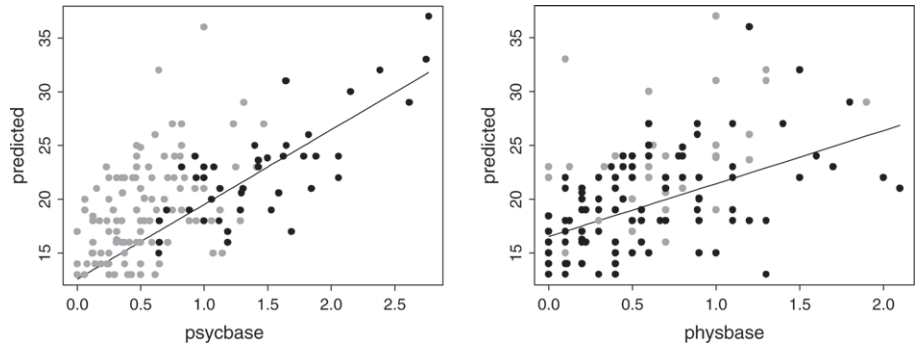

Fig. 5. Component regression functions for Hodgkin's symptom analysis. Black dots represent observations associated with given fitted function.

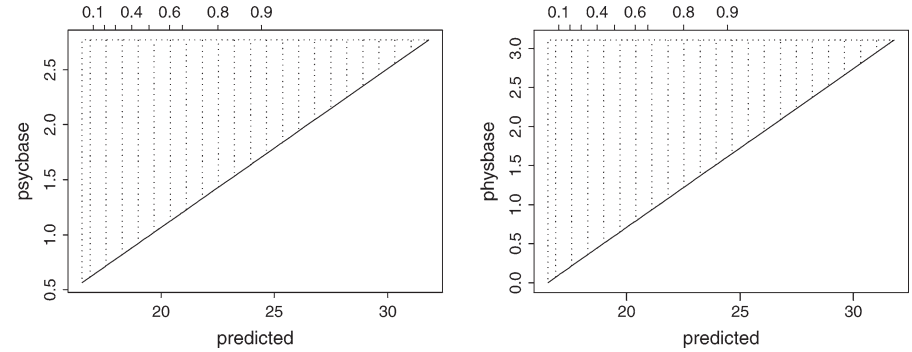

Fig. 6. Inverse function for Hodgkin's data. The top horizontal axis indicates quantiles of the predicted values from the model.

score are presented in Figure 6. The top axis of this figure gives the same predictions as a function of the quantile of the model predicted values.

For instance, the rule

$$(\text{psyc} \geqslant 1.64 \text{ OR phys} \geqslant 1.52)$$

corresponds to a symptom score of >24 or approximately the worst 10% of patients with respect to symptom score. We note that a score of 25 reflects moderate-to-severe symptom problems (Ganz *et al.*, 2003). We compare the form of this rule to ones constructed using tree-based regression (with 10-fold cross-validation to pick tree size), linear regression and MARS using a GCV with a default penalty of 1.5 per term parameter. Our implementation of the MARS algorithm was based on code written by Trevor Hastie and Rob Tibshirani. The tree-based rule for the group with a score of at least 24 is (psyc $\geqslant$ 1.31 AND phys $\geqslant$ 0.90) and represents 9% of the sample. Using linear regression, the rule is $14.8 + 3.7\text{psyc} + 5.1\text{phys} > 24$. The MARS rule is somewhat more complicated in terms of piecewise linear splines

$$21.8 - 7.5(\text{psyc} - 0.8)^- + 6.4(\text{phys} - 1.23)^+ + 5.0(\text{psyc} - 0.57)^+(\text{phys} - 0.8)^- > 24,$$

where $(x - c)^+ = (x - c)$ if $(x - c) > 0$ else 0 and $(x - c)^- = -(x - c)$ if $(x - c) < 0$ else 0. While the tree-based regression yields a simple decision rule, its flexibility suffers due to the discrete nature of the tree. Adding the next node to the rule changes the fraction of the group from 9% to 28%. The linear, MARS, and XR models are smooth functions of the covariates, so rules corresponding to other values of the outcome such as 20 or 23 are easily constructed. While simple rule construction, not model prediction

was the primary goal of this analysis, we also evaluated mean absolute prediction error averaged over 25 repeated divisions of the data in training and test sets of sizes (150/77). The average errors for the three smooth modeling methods were quite close (3.09 for XR, 2.96 for linear regression, and 3.05 for MARS). The average error was slightly larger for tree-based regression, 3.36.

We note that an alternative strategy for describing a specific outcome group would be first to categorize the outcome variable at a threshold, for instance (sds $>$ 24), then use a modeling method to construct rules to describe patients with a high probability of being in the poor (sds $>$ 24) group. If simple Boolean rules were of interest, XR or tree-based regression modified to binomial outcome could be used.

6. Patient survival in multiple myeloma

We use data from several recent SWOG multiple myeloma clinical trials to demonstrate XR models with survival data. SWOG S8624 tested different multidrug combinations and schedules for myeloma therapy. Two additional studies (SWOG S9028 and S9210) were used as validation for the prognostic rules developed from the first study (Crowley *et al.*, 2001). Patients eligible for these studies had untreated, newly diagnosed multiple myeloma of any stage.

There were 479 patients available with complete data for these four variables in the training data set from study S8624. The survival data consist of t_i, time under observation, δ_i, an indicator of failure ($1 =$ death, $0 =$ censored) and x_i a vector of covariates for individual i. We considered four potential predictors of survival, serum β_2 microglobulin (sb2m), serum calcium (calcium), serum albumin (albumin), and creatinine (creatinine). To facilitate computation, we constructed models by regressing on the counting process or martingale residuals from the null model. These residuals are defined as $M_i = \delta_i - \widehat{\Lambda}(t_i)$, where $\widehat{\Lambda}(t_i)$ is the empirical cumulative hazard estimator. Using martingale residuals (or weighted versions of martingale residuals) has previously been proposed for adaptive model-building procedures with censored survival data (e.g. LeBlanc and Crowley, 1999). We constructed a sequence of XR models using the forward stepwise method described previously and picked a model for further investigation based on GCV. At the final stage, the models were refitted using an extension of the XR to exponential survival data as described in Section 8. The stepwise model-building method resulted in a model with four component terms,

$$\max(\min(g_{11}(\text{sb2m}), g_{12}(\text{calcium})), \min(g_{21}(\text{creatinine}), g_{22}(\text{albumin}))).$$

The inverse model is represented in Figure 7 in terms of the quantiles of the observations in the training sample. Each column corresponds to a 'minimum' term. Therefore, a level set rule is obtained by first taking the intersection within each column and then taking the union across columns.

For multiple myeloma, there is considerable interest in identifying a single extreme outcome group of patients with worst survival appropriate for more aggressive dose-intensive therapy. In addition, the poor outcome group must be sufficiently large to conduct future clinical trials.

A rule representing the worst 30% of patients in this group is

$$(\text{sb2m} \geqslant 5.8 \text{ AND calcium} \geqslant 9.4) \text{ OR } (\text{creatinine} \geqslant 1.25 \text{ AND albumin} < 2.7).$$

Other rules for other fractions of the sample can be determined from Figure 7.

Tree-based methods are also popular methods for describing prognostic groups with survival data. We used a method which finds optimal variables and split points based on the log rank test statistic, then uses bootstrap resampling methods to select tree size (e.g. LeBlanc and Crowley, 1993; Crowley *et al.*, 1997). The pruned tree that was selected had five terminal nodes. The group of patients corresponding to the worst survival was described by sb2m $\geqslant$ 9.4, which identified 22% of patients in the learning sample. Due to the discreteness of the tree the next larger group was described by

$$\text{sb2m} \geqslant 9.4 \text{ OR } (\text{sb2m} < 9.4 \text{ AND creatinine} \geqslant 1.7),$$

but described 37% of the training sample. Therefore, while tree-based methods yield simple Boolean prognostic rules, controlling the fraction of patients in the group is not facilitated.

The extreme model can also be directly used to construct multiple prognostic groups as is frequently done using tree-based methods for survival data. In Figure 8, we plot groups corresponding to the rules defined by the quartiles of the distribution of the regression estimates, $q_{0.25}$, $q_{0.50}$, and $q_{0.75}$, both for the new method and for the tree-based model collapsed to four groups. The tree-based prognostic groups vary quite substantially in the fraction of patients in each group (0.13–0.37).

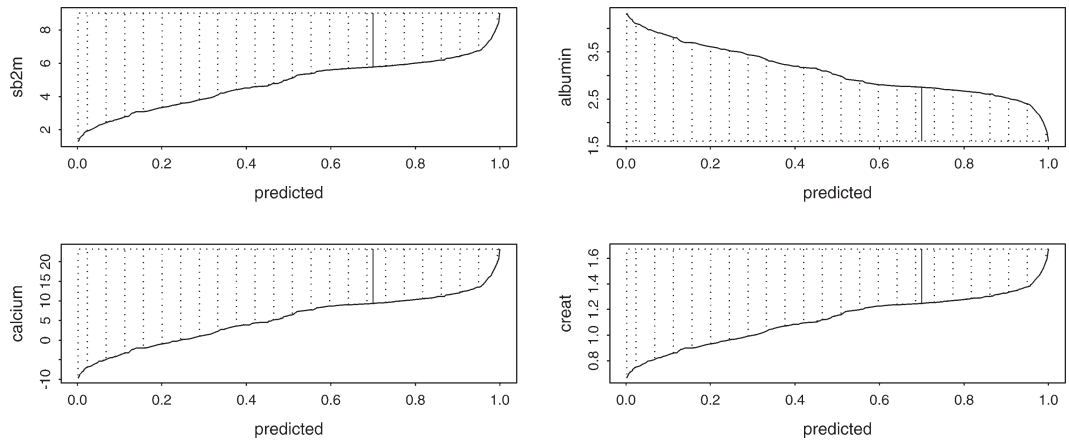

Fig. 7. Rules corresponding to level sets of the myeloma extreme model. The single vertical line corresponds to the rule for the worst 30% patients.

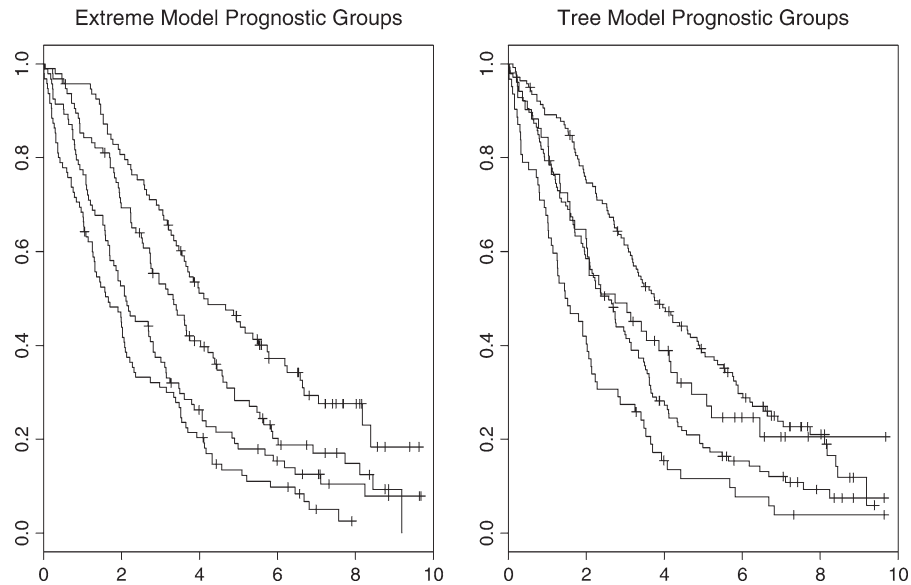

Fig. 8. Prognostic rules defined by quartiles of the estimated extreme model and groups from tree model evaluated on the test sample data.

If simple decision rules are not required, other modeling methods are well suited for studying the association of predictors to the outcome. We used the proportional hazards model (Cox, 1972) and the adaptive modeling method HARE due to Kooperberg *et al.* (1995) on the training sample. HARE builds spline models which can include nonlinear terms and interactions among predictors and between predictors in time. The linear Cox model involved all four predictor variables and if the resulting index is treated as a single predictor, the logarithm of the partial likelihood on the test sample was -1599.3. For the HARE method, we chose to limit the model to proportional hazards models, use a penalty analogous to Akaike Information Criterion (AIC) of 4 per term and linear spline basis functions. The resulting model included a nonlinear term for serum β_2 microglobulin and a linear term for albumin. The partial likelihood of the model index evaluated on the test sample was -1594.3, larger (better) than the linear model. As a comparison, the partial log likelihood for the extreme model was -1596.7 for the XR method and lowest (worst) for the tree-based model, -1604.5. Again, while the tree model yields simple rules, the predictive performance is limited to the discreteness of the model.

For the smooth methods that allow for control of the group size (XR, linear Cox, and HARE), groups based on quartiles of the distribution of the regression estimates, $q_{0.25}$, $q_{0.50}$, and $q_{0.75}$, were compared. The assignments to prognostic groups (1–4) by all methods are generally close even though the qualitative description of the rules are very different. All three smooth methods agree to the same prognostic group in 59% of the test sample cases and are within ± 1 prognostic group for 96% of the test sample cases.

7. Simulations

We performed a small simulation study to investigate the performance of the XR algorithm. As a comparison, we considered estimation error and calibration using forward stepwise linear regression, MARS, and tree-based regression. Sample sizes of 500 observations were generated where the predictor variables are taken from a five-dimensional spherical normal distribution $N(0,1)^p$ and where the response is defined as $y = f(x) + \epsilon$, where ϵ has an independent $N(0, \sigma_k^2)$ distribution, and where σ_k^2 was chosen to control the signal to noise ratio. We consider four models:

Model A: $f(x) = 0$;

Model B: $f(x) = \min(x_1, x_2, x_3)$;

Model C: $f(x) = \max(\min(x_1, 0.5x_2), \min(x_3, x_4))$; and

Model D: $f(x) = x_1 + x_2 + x_3$.

The constant σ_k^2 for Model A was set to 1 and chosen to make the signal-to-noise ratio 1.5 for Models B–D. Model A is the null model where the response is unrelated to the predictor variables. Therefore, for this model we expect a measure of the amount of overfitting for each method. Models B and C are of the extrema model form so one would expect XR to perform well on those models. However, neither of these models should be well approximated by a linear regression model. Model D is a linear model which should be difficult for XR to yield good approximations.

In assessing method performance for Models B through D, we standardize estimation error by the mean absolute deviation for the model values $\tau = \text{ave}|f(x^*) - \overline{f(x^*)}|$ over the covariate values, x^*, from $N(0,1)^p$ calculated by a large separate simulated sample. We also consider the estimation of a level set chosen to be approximately the 80th percentile of the true model values $f(x^*)$, again calculated by the separate large simulated sample.

An additional 500 observations from the same model are used as a test data set $\mathcal{L}_{Tk}$ for each simulation to estimate model error. The test set was not used for model selection. We report the estimate of relative

absolute error for each method on the test sample

$$\frac{1}{K\tau}\sum_{k=1}^{K}|\widehat{f}(x_{Tk}) - f(x_{Tk})|,$$

where $\tau = 1$ for Model A. We calculated level set error as the average misclassification, on the test sample

$$\frac{1}{K}\sum_{k=1}^{K}(|I\{f(x_{Tk}) \geqslant q\} - I\{\widehat{f}(x_{Tk}) \geqslant q\}|).$$

In addition to estimating the level sets using the regression methods, we also use a data refinement (peeling) algorithm analogous to the PRIM method of Friedman and Fisher (1999). Since this method only calculates averages over regions (not the regression function), we calculate the level error by modifying the threshold and defining $q' = \text{ave}(f(x_{Lk})I\{x_{Lk} \geqslant q\})$ to make the results comparable to the regression-based strategies.

Since each of the methods has tuning parameter choices for estimation, we note important choices here. While different results could be obtained for different tuning parameters, we think the general conclusions would be supported for a range of implementations. For linear regression, backward selection was used and the model was selected by GCV. For the MARS algorithm, we limited interactions to fourth degree and chose the model by GCV using the suggested default penalty of 1.5 per term parameter. As in Hodgkin's example, our implementation of the MARS algorithm was using code written by Trevor Hastie and Rob Tibshirani. For the tree-based method, the minimum node size was set at 25 observations and the tree size was selected by 10-fold cross-validation, which tends to be standard for model selection for that method. For peeling, box-shaped regions were calculated by removing a fraction of 5% of the sample at each step (with a minimum of 10 observations). After reaching a mean response value for the region $\overline{y} \geqslant q'$, the resulting box was identified. Those data associated with that box were removed and the peeling was repeated until no extreme groups could be found. Our version of the peeling algorithm differed from the published PRIM algorithm in that we did not include a pasting option of regrowing boxes after peeling. For XR, we used adaptive step-size selection starting at a step-size of 0.5. We also restricted the model so that at least 5% of the sample was used in the estimation of any component model. The XR model was selected by GCV using a penalty of 1.5 per parameter. We did not place a limit on the maximum number of components in each 'minimum' function. We repeated the analysis for 25 simulated data sets.

The results of the simulation given in Table 1 show that all of the methods lead to relatively low model error in the case of no signal (Model A). For Models B and C, the stepwise linear method does not do well for model approximation. The XR method yields substantially lower model error for these two models that fall in the extreme model class. MARS, which builds piecewise linear approximation, is the second best methodology on all models. Therefore, if prediction were the only consideration, MARS would be a good overall choice for this group of models. Tree-based methods, due to their discrete nature,

Table 1. *Mean relative model error for simulated data (standard error) for stepwise linear (linear), MARS, tree-based models (trees), and stepwise selected XR (stepwise XR) techniques*

Model	Linear	MARS	Trees	Stepwise XR
A	0.061 (0.061)	0.101 (0.036)	0.042 (0.032)	0.087 (0.034)
B	0.613 (0.006)	0.204 (0.008)	0.293 (0.006)	0.058 (0.006)
C	0.671 (0.009)	0.454 (0.013)	0.516 (0.014)	0.062 (0.006)
D	0.061 (0.007)	0.110 (0.009)	0.553 (0.007)	0.518 (0.012)

Table 2. *Error in level set estimation at threshold corresponding to* 80*th percentile of model values (standard error)*

Model	Linear	MARS	Trees	Peeling	Stepwise XR
B	0.181 (0.006)	0.076 (0.009)	0.081 (0.011)	0.070 (0.008)	0.021 (0.005)
C	0.157 (0.005)	0.098 (0.006)	0.124 (0.013)	0.110 (0.013)	0.015 (0.003)
D	0.013 (0.002)	0.024 (0.003)	0.136 (0.007)	0.132 (0.007)	0.117 (0.005)

do not approximate the Models B–D well. For the linear Model D, XR and trees do not yield good approximations. For both methods, it takes a large number of terms to approximate additive models. We also investigated the impact of the adaptive stepwise component of algorithm for the two models of extreme form (Models B and C). If the correct model is fit, the absolute error (standard error) for Model B is 0.056 (0.006) and for Model C is 0.058 (0.005); both only slightly better than the stepwise XR method.

The results for finding the 80th percentile level set for extreme outcome are presented in Table 2 and are consistent with the function approximation results. Other level sets were considered and the conclusions were similar. We consider only the non-null Models B–D. While linear models have difficulty with nonlinear function Models B and C, MARS generally does reasonably well. However, descriptions of the level sets are not easy to describe with either linear regression or MARS, while tree-based methods, peeling, and the XR method yield easy decision rules. Tree-based methods suffer because the discrete model form and significant number of observations in each node do not allow it to calibrate to a specific output value. The peeling method allows control of the group outcome, but appears to come with higher variability in this setting, thus limiting its performance.

8. Discussion

We have investigated a class of regression functions which yield a locally univariate regression surface. Inverting the model directly results in descriptions of the level sets in terms of decision rules. We believe that these rules, like those from tree-based models, are often interpretable for clinical applications describing covariate values associated with different levels of patient outcome. However, unlike tree-based models, the model form is smooth in the covariates, so one can control the rules with respect to specific outcome level.

While our simulations and examples suggest that the simple alternating estimation algorithm can perform well, there are other potential estimation strategies worthy of exploration. Improved algorithms could include much more computationally intensive stochastic searches for regions potentially followed by constrained linear optimization. These approaches may be more useful if more categorical predictors are used in the model, making the covariate distributions more discrete.

Estimation for XR can be relatively easily extended to other outcome types using parametric likelihood-based methods, albeit at the cost of additional computing time required to get maximum likelihood solutions for the component models. For example, for survival data, a model assuming exponential survival times would be $\log(\lambda_x) = \max_j(\min(g_{j1}(x), \ldots, g_{j,K(j)}(x)))$, where λ_x is the hazard function, and for binary outcomes, one can replace the squared error with binomial error and use local logistic regression. In that case the model would be $\log(p_x)/(1 - p_x) = \max_j(\min(g_{j1}(x), \ldots, g_{j,K(j)}(x)))$.

We note that the XR method has similarities to Logic Regression (Ruczinski *et al.*, 2003) which is a general regression method for generating Boolean rules. Note that the extrema functions ‘minimum’ and ‘maximum’ are continuous versions of ‘AND’ and ‘OR’ for binary prediction. However, while an important strength of Logic Regression is a complete stochastic model search and simple representation for a potentially large number of binary predictors, this new method is best suited for combining a small number of continuous predictors.

ACKNOWLEDGMENTS

The authors want to thank Carol Moinpour and John Crowley for helpful conversations and Mark Blitzer for his review of the manuscript. This work was supported by the National Institutes of Health through grants R01-CA090998, R01-CA074841, and R01-CA053996. Software is available from mleblanc@fhcrc.org.

REFERENCES

BREIMAN, L. (1993). Hinging hyperplanes for regression, classification and function approximation. *IEEE Transactions on Information Theory* **3**, 999–1013.

BREIMAN, L., FRIEDMAN, J. H., OLSHEN, R. A. AND STONE, C. J. (1984). *Classification and Regression Trees*. Belmont, CA: Wadsworth.

COX, D. R. (1972). Regression models and life-tables (with discussion). *Journal of the Royal Statistical Society, Series B* **34**, 187–220.

CROWLEY, J., LEBLANC, M., JACOBSON, J. AND SALMON, S. (1997). Some exploratory tools for the analysis of survival data. In Lin, D. and Fleming, T. (eds), *The First Seattle Symposium in Biostatistics: Survival Analysis*, Lecture Notes in Statistics. New York: Springer.

CROWLEY, J., JACOBSON, J. AND ALEXANIAN, R. (2001). Standard-dose therapy for multiple myeloma: the Southwest Oncology Group experience. *Seminars in Hematology* **38**, 203–208.

FRIEDMAN, J. (1991). Multivariate adaptive regression splines (with discussion). *Annals of Statistics* **19**, 1–141.

FRIEDMAN, J. AND FISHER, N. (1999). Bump hunting in high dimensional data (with discussion). *Computing and Statistics* **9**, 123–162.

GANZ, P. A., MOINPOUR, C. M., MCCOY, S., PAULER, D. K., PRESS, O. W. AND FISHER, R. I. (2004). Predictors of vitality (energy/fatigue) in early stage Hodgkin's disease (HD): results from Southwest Oncology Group (SWOG) Study 9133. *Journal of Clinical Oncology* **22**(14S), No. 6546.

GANZ, P. A., MOINPOUR, C. M., PAULER, D. K., KORNBLITH, A. B., GAYNOR, E. R., BALCERZAK, S. P., GATTI, G. S., ERBA, H. P., MCCOY, S., PRESS, O. W. AND FISHER, R. I. (2003). Health status and quality of life in patients with early-stage Hodgkin's disease treated on Southwest Oncology Group Study 9133. *Journal of Clinical Oncology* **21**, 3512–3519.

KOOPERBERG, C., STONE, C. J. AND TRUONG, Y. K. (1995). Hazard regression. *Journal of the American Statistical Association* **90**, 78–94.

LEBLANC, M. AND CROWLEY, J. (1993). Survival trees by goodness of split. *Journal of the American Statistical Association* **88**, 457–467.

LEBLANC, M. AND CROWLEY, J. (1999). Adaptive regression splines in the Cox model. *Biometrics* **55**, 204–213.

PRESS, O. W., LEBLANC, M., LICHTER, A. S., GROGAN, T. M., UNGER, J. M., WASSERMAN, T. H., GAYNOR, E. R., PETERSON, B. A., MILLER, T. P. AND FISHER, R. I. (2001). A phase III randomized intergroup trial of subtotal lymphoid irradiation versus doxorubicin, vinblastine and subtotal lymphoid irradiation for stage IA–IIA Hodgkin's disease (SWOG 9133, CALGB 9391). *Journal of Clinical Oncology* **19**, 4238–4244.

RUCZINSKI, I., KOOPERBERG, C. AND LEBLANC, M. (2003). Logic regression. *Journal of Graphical and Computational Statistics* **12**, 475–511.

[*Received October 15, 2004; revised June 10, 2005; accepted for publication June 13, 2005*]