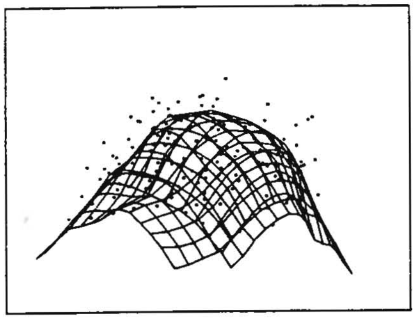
Estimating Functions of Mixed Ordinal and Categorical Variables Using Adaptive Splines

Jerome H. Friedman

Technical Report No. 108
June, 1991

Laboratory for Computational Statistics



Department of Statistics
Stanford University

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUN 1991**	2. REPORT TYPE	3. DATES COVERED **00-00-1991 to 00-00-1991**

4. TITLE AND SUBTITLE
Estimating Functions of Mixed Ordinal and Categorical Variables Using Adaptive Splines

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Stanford University,Department of Statistics,Stanford,CA,94309

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Multivariate a~aptive regression splines (MARS) is a methodology for nonparametrically estimating (and interpreting) general functions of a high-dimensional argument given (usually noisy) data. Its basic underlying assumption is that the function to be estimated is locally relatively smooth where smoothness is adaptively defined depending on the local characteristics of the function. The usual definitions of smoothness do not apply to variables that assume unorderable categorical values. After a brief review of the MARS strategy for estimating functions of ordinal variables, alternative concepts of smoothness appropriate for categorical variables are introduced. These concepts lead to procedures that can estimate and interpret functions of many categorical variables, as well as those involving (many) mixed ordinal and categorical variables. They also provide a natural mechanism for modeling and predicting in the presence of missing predictor values (ordinal or categorical).

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**51**	

ESTIMATING FUNCTIONS OF MIXED ORDINAL AND CATEGORICAL VARIABLES USING ADAPTIVE SPLINES

by

JEROME H. FRIEDMAN

STANFORD LINEAR ACCELERATOR CENTER

and

STANFORD UNIVERSITY

TECHNICAL REPORT NO. 108

JUNE 1991

PREPARED UNDER THE AUSPICES

OF

NATIONAL SECURITY AGENCY

GRANT MDA904-88-H-2029

DEPARTMENT OF STATISTICS

STANFORD UNIVERSITY

STANFORD, CALIFORNIA

Estimating Functions of Mixed Ordinal and Categorical Variables Using Adaptive Splines*

Jerome H. Friedman
Department of Statistics
and
Stanford Linear Accelerator Center
Stanford University

Abstract

Multivariate adaptive regression splines (MARS) is a methodology for nonparametrically estimating (and interpreting) general functions of a high-dimensional argument given (usually noisy) data. Its basic underlying assumption is that the function to be estimated is locally relatively smooth where smoothness is adaptively defined depending on the local characteristics of the function. The usual definitions of smoothness do not apply to variables that assume unorderable categorical values. After a brief review of the MARS strategy for estimating functions of ordinal variables, alternative concepts of smoothness appropriate for categorical variables are introduced. These concepts lead to procedures that can estimate and interpret functions of many categorical variables, as well as those involving (many) mixed ordinal and categorical variables. They also provide a natural mechanism for modeling and predicting in the presence of missing predictor values (ordinal or categorical).

1.0. Introduction. The problem of modeling and interpreting a general predictive relationship between a "response" variable y and a large number of simultaneously measured "predictor" variables $\mathbf{x} = (x_1, \cdots, x_n)$ is a challenging one studied in many disciplines. The objective is to use a given sample of "training" data $\{y_i, \mathbf{x}_i\}_1^N$ to derive a rule for estimating (missing) response values in future observations given only the values of the predictor variables. Another goal often present is that of trying to gain an understanding of the nature of the predictive relationship through an examination of the structure of the derived rule. This may reveal insight into the properties of the system that generated the data.

* Research supported jointly by the U.S. National Security Agency under Grant MDA 904-88-M-2029 and the U.S. Department of Energy under contract AC03-76SF00515.

This problem can be usefully cast as one of function estimation or approximation. The relationship between y and $\mathbf{x}$ is assumed to take the form

$$y = f(\mathbf{x}) + \epsilon(\mathbf{x}) \tag{1}$$

where f is a single valued deterministic function of the n predictor variables and ϵ is a random component reflecting the fact that the chosen predictor variables may not completely specify y; it may depend on other quantities that vary, but are not observed. This "error" term is defined to have zero expected value for all $\mathbf{x}$

$$E(\epsilon \mid \mathbf{x}) \equiv 0$$

so that the assumed true underlying ("target") function f can be defined by

$$f(\mathbf{x}) \equiv E(y \mid \mathbf{x})$$

with the expected values taken over the population from which the training and future data are presumed to be random samples. In this framework the goal of the training procedure is to use the training sample of size N, $\{y_i, \mathbf{x}_i\}_1^N$ to derive a function $\hat{f}(\mathbf{x})$ that can serve as a useful approximation to $f(\mathbf{x})$. Here usefulness is usually defined in terms of accuracy and often, in addition, interpretability.

For finite training samples the definition (1) for the true underlying function $f(\mathbf{x})$ is incomplete. (Any quantity can be expressed as the sum of two other quantities.) This identifiability problem must be resolved by defining those characteristics that distinguish the "signal" $f(\mathbf{x})$ from the "noise" $\epsilon(\mathbf{x})$. In parametric fitting $f(\mathbf{x})$ is assumed to be a member of a parametric family of functions whereas the noise is assumed to lie mainly outside that family. This is generally the case because the chosen parametric functions usually vary smoothly with changing $\mathbf{x}$ while the noise does not. The function estimation problem in this case then reduces to that of estimating the corresponding parameters from the training data. In nonparametric modeling the distinction between signal and noise is based solely on the notion of smoothness; $f(\mathbf{x})$ is taken to be that component of y that varies smoothly with changing values of $\mathbf{x}$, whereas the noise is taken to be the leftover part that does not. The effectiveness of a nonparametric procedure is determined by how well it can gauge the (local) smoothness properties of $f(\mathbf{x})$ and exploit them so as to filter out most of the noise without altering too much of the signal.

When the predictor variables all take on values in an ordered set there are many natural and exploitable definitions of smoothness, giving rise to a vast literature on nonparametric smoothing and function estimation. In high dimensional settings this exploitation has proven far more difficult, but some successes have been achieved [see for example Friedman (1991) along with the discussions and the many references therein.] When some (or all) of the predictor variables assume values for which there is no natural order relation (in the context of the problem) the notion of smoothness of the dependence of y on such variables is less readily apparent. In this paper a notion of smoothness of the dependence (of an ordinal variable) on (unorderable) categorical variables is introduced and then exploited (in the context of an adaptive algorithm) to model functions of (many) categorical variables, along with perhaps (many) ordinal predictor variables as well. When all of the predictor variables happen to be categorical this approach gives rise to a new method for analyzing large (sparse) many-dimensional contingency tables.

The technique presented in this paper is based on a modification of the multivariate adaptive regression spline (MARS) strategy (Friedman, 1991) for modeling functions of (many) ordinal variables. In order to be somewhat self contained, the paper begins with a brief overview of the MARS procedure. It next turns to a general discussion of smoothing on categorical variables introducing the basic ideas, followed by a description of the modifications to the MARS method necessary to implement them. The complete MARS algorithm for modeling functions with arguments of mixed ordinal and categorical variables is then described. Methods for interpreting models involving interactions between categorical and ordinal variables are presented. This is followed by an extension of the procedure to incorporate nested variables, which in turn leads to a natural and very general method for dealing with missing values among the predictor variables. Finally, some simulation studies and illustrative examples are presented.

2. Multivariate Adaptive Regression Splines. This section gives a brief overview of the multivariate adaptive regression spline (MARS) procedure described much more completely in Friedman (1991). It forms the basis for the techniques introduced in this paper. The MARS procedure is in turn based on a generalization of spline methods for function fitting. Splines have been extensively studied and have many desirable properties. [See for example, de Boor (1978), Shumaker (1976) (1984), Eubank (1988), and Wahba (1990).] We begin with a very brief review of traditional (fixed knot) regression spline fitting and then turn to the adaptive regression spline generalization.

2.1. A Micro-Introduction to Spline Fitting. First consider the case of only one predictor variable, x ($n = 1$). An approximating (qth order regression) spline function $\hat{f}_q(x)$ is obtained by dividing the range of x values into $K + 1$ disjoint regions separated by K points (called "knots"). The approximation takes the form of a separate qth degree polynomial in each region, constrained so that the function and its $q - 1$ lowest order derivatives are everywhere continuous. Generally the order of the spline is taken to be low ($q \leq 3$). Each qth degree polynomial is defined by $q + 1$ parameters so that there are a total of $(K + 1)(q + 1)$ parameters to be adjusted to best fit the data, usually by least squares. The continuity requirement however places q constraints at each knot location making a total of Kq constraints. The total number of free parameters is thus $K + q + 1$.

Regression spline fitting can be implemented by directly solving the constrained minimization problem described above. Usually, however, the problem is converted to an unconstrained optimization problem by choosing a set of basis functions $\{B_k^{(q)}(x)\}_0^{K+q}$ that span the space of all qth order spline functions (given the chosen knot locations) and performing a (linear) least-squares fit of the response on this basis function set. In this case the approximation takes the form

$$\hat{f}_q(x) = \sum_{k=0}^{K+q} a_k B_k^{(q)}(x) \tag{2}$$

where the values of the expansion coefficients $\{a_k\}_0^{K+q}$ are unconstrained, and the continuity constraints are intrinsically embodied in the basis functions $\{B_k^{(q)}(x)\}_0^{K+q}$. One such basis ("truncated power basis") is comprised of the functions

$$\{x^j\}_{j=0}^q, \quad \{(x - t_k)_+^q\}_1^K. \tag{3}$$

Here $\{t_k\}_1^K$ are the knot locations defining the $K+1$ regions and the truncated power functions are defined by

$$(x - t_k)_+^q = \begin{cases} 0 & x \le t_k \\ (x - t_k)^q & x > t_k. \end{cases}$$

The truncated power basis (3) is not the only basis appropriate for this application. Any set of $K+q+1$ linearly independent linear combinations of these basis functions (3) will also span the same space. The most popular basis is the (minimum support) "B-spline" basis owing to its superior numerical properties when used in conjunction with least-squares fitting. B-spline basis functions have support over, and are defined by, $K+2$ adjacent knot locations, whereas the truncated power functions have maximal support but are each defined by a single knot location. This latter property has important algorithmic consequences for adaptive regression spline strategies (see below).

Regression splines (of order q) are characterized by $K+1$ parameters: the number of knots K, and in addition their locations $\{t_k\}_1^K$. This provides the user with a great deal of flexibility in specifying the nature of the approximating function. This is in contrast to other techniques such as kernel methods (Parzen, 1962) and smoothing splines (Craven and Wahba, 1979) which are characterized by a single (smoothing) parameter. If the user has a good deal of knowledge about the nature of the true underlying function $f(x)$ (1) and sufficient intuition concerning the effect on the approximation of changes in the knot specification, this increased flexibility can be used to great advantage. On the other hand, lack of such knowledge can make choosing a good set of knots difficult.

The variance of the function estimate $\hat{f}(x)$ in any local region is proportional to the ratio of the local knot density to the local data (predictor variable) density. The bias is proportional to the local second derivative of the true underlying function $f''(x)$ divided by the local knot density. For any given $f(x)$ (1) and distribution of (abscissa) data points there is an optimal specification for the knots. This is however usually unknown. Standard defaults often involve placing the knots equispaced along the abscissa or at the $1/K(\times 100)$ percentiles of the (x) data distribution. The regression spline approximation is then characterized by a single parameter (number of knots K) as are kernel and smoothing-spline methods.

The flexibility of the regression spline approach can be enhanced by incorporating an automatic strategy for knot selection as part of the data fitting process. Many such strategies have been proposed, most of them involving a numerical minimization of the least squares criterion

$$\sum_{i=1}^{N} \left[y_i - \sum_{k=0}^{K+q} a_k B_k^{(q)}(x) \right]^2 \tag{4}$$

jointly with respect to the expansion coefficients $\{a_k\}_0^{K+q}$ and the knot locations $\{t_k\}_1^K$. Although sometimes effective, these approaches have many difficulties and can be computationally expensive. [See Eubank (1988) and references therein.]

An especially simple and effective strategy for automatically selecting both the number and locations for the knots was described by Smith (1982). She suggested using the truncated power basis (3) so that (4) becomes

$$\sum_{i=1}^{N}\left[y_i - \sum_{j=0}^{q} b_j x^j - \sum_{k=1}^{K} a_k (x - t_k)_+^q\right]^2 . \tag{5}$$

Here the coefficients $\{b_j\}_0^q$, $\{a_k\}_1^K$ can be regarded as the parameters associated with a multiple linear (least-squares) regression of the response y on the "variables" $\{x^j\}_0^q$ and $\{(x-t_k)_+^q\}_1^K$. Adding or deleting a knot t_k is viewed as adding or deleting the (corresponding) variable $(x-t_k)_+^q$. Smith's strategy consists of starting with a very large number of eligible knot locations $\{t_1, \cdots, t_{K_{\max}}\}$ (say one at every interior data point, $K_{\max} = N - 2$) and considering the corresponding "variables" $\{(x-t_k)_+^q\}_1^{K_{\max}}$ as candidates to be selected through a statistical variable subset selection procedure (Smith suggested a standard forward/backward stepwise approach).

Although quite simple, this approach to knot selection is both elegant and powerful. It automatically selects both the number of knots K and their locations $t_1, \cdots, t_K$. It thereby not only estimates the overall (global) amount of smoothing to be applied (controlled by K), but in addition it uses the data to estimate the separate relative amount of smoothing to be applied at different (abscissa) locations. In a large simulation study comparing many different smoothers over a wide variety of situations (Breiman and Peters, 1988), this method proved to be the best or among the best over the situations (true underlying function, abscissa design) considered. This approach has the additional virtue of being very simple to implement and fast to compute.

The adaptive regression spline strategy introduced by Smith (1982) was developed for the univariate ($n = 1$) smoothing problem. The real potential of this idea however is realized in the multivariate setting ($n >> 1$) where the function to be estimated can depend on many (measured) variables. The multivariate adaptive regression spline method (MARS, Friedman, 1991) can be viewed as a multivariate generalization of Smith's (1982) strategy.

An approximating (qth order regression) spline function $\hat{f}_q(\mathbf{x})$ of n variables ($\mathbf{x} = \{x_1, \cdots, x_n\}$) is defined analogously to that for one variable. The n-dimensional space R^n is divided into a set of disjoint regions and within each one $\hat{f}_q(\mathbf{x})$ is taken to be a polynomial in n variables with the maximum degree of any single variable being q. The approximation $\hat{f}_q(\mathbf{x})$ is constrained so that it and all its derivatives to order $q - 1$ are everywhere continuous. This places constraints on the approximating polynomials in separate regions along the ($n-1$-dimensional) region boundaries. As in the univariate case, the approximation is most easily constructed by choosing a basis function set (of n-variables) that spans the space of all qth order n-dimensional spline functions given the particular set of chosen regions. The approximation is then obtained by fitting the coefficients of this expansion to the data.

For $n > 2$ (and usually for $n = 2$) the disjoint regions defining the spline approximation are taken to be tensor products of disjoint intervals on each of the variables, delineated by knot locations. Thus, placing K_j knots on each of the variables ($1 \le j \le n$) produces $\prod_{j=1}^{n}(K_j + 1)$ regions. A basis function set that spans the space of spline functions over this set of regions is the tensor product of the corresponding univariate spline bases associated with the knot locations on each of the variables

$$\hat{f}_q(\mathbf{x}) = \sum_{k_1=0}^{K_1+q} \cdots \sum_{k_n=0}^{K_n+q} a_{k_1,\cdots k_n} \prod_{j=1}^{n} B_{k_j}^{(q)}(x_j). \tag{6}$$

Here $\{B_{k_j}^{(q)}(x_j)\}_{k_j=0}^{K_j}$ is the basis function set for a qth order spline approximation given the locations of the K_j knots on x_j $(1 \le j \le n)$. The size of this tensor product basis (6) and thus the number of coefficients to be estimated in a (linear least squares) fit to the data is

$$\prod_{j=1}^{n}(K_j + q + 1). \tag{7}$$

For cubic splines $(q = 3)$ with $K_j = 5$ knots (only) on each variable there are 59,049 coefficients to be estimated in five dimensions. In six dimensions $(n = 6)$ that number is 531,441, while for $n = 10$ it is 3.5×10^9. This exponential increase in both estimation and computational complexity with increasing dimension (for the same level of refinement) is a reflection of the "curse-of-dimensionality" (Bellman, 1961). Gargantuan training samples are required for straightforward tensor product spline approximations in high dimensions.

The multivariate adaptive regression spline (MARS) strategy employs the tensor product representation (6) with the truncated power basis (3), and considers a very large number $[K_j \lesssim 0(N)]$ of eligible knot locations on each variable. In analogy with the Smith (1982) strategy, each of the $(K_j + q + 1)^n$ basis functions so derived is taken to be a candidate "variable" to be potentially selected through a statistical variable subset selection procedure.

As in the univariate $(n = 1)$ case, this multivariate adaptive spline strategy can be motivated from geometrical considerations. The goal is to choose a good set of regions to define the spline approximation for the problem at hand [target function $f(\mathbf{x})$ (1)]. Both statistical and computational considerations restrict their number to be very small relative to that generated by a complete tensor product of univariate intervals. Selecting a small subset of basis functions from those representing the complete tensor product has the effect of producing a spline approximation on a corresponding (small) set of (larger) regions, each of which is a selected union of regions from the original tensor product.

The attractive aspects of such a procedure are far more dramatic in the multivariate case than in univariate $(n = 1)$ settings (Smith, 1982). First (and foremost) its adaptability, which can be useful in univariate fitting, is absolutely crucial in approximating all but the simplest functions of high dimensional arguments. The procedure automatically chooses the approximating regions in the n-dimensional predictor variable space. As a consequence it chooses the number of (distinct) variables that enter into each corresponding basis function (interaction order). It also chooses which particular variables comprise the basis functions that enter the model, thereby providing automatic variable subset selection. Candidate basis functions involving predictor variables unrelated to the response are less likely to be selected. Moreover, this variable subset selection aspect is a local property; namely, in any local region of the predictor variable space, basis functions defining its subregions are most likely to involve only the variables most strongly associated with the response in that particular region. This local variable subset selection property, along with the ability to automatically adjust the relative amount of smoothing in each local region of the n-dimensional predictor space, provides considerable flexibility to parsimoniously approximate a wide range of functions.

A consequence of the basis function subset selection implementation is the ease with which constraints can be applied to the solution. Basis functions in the candidate tensor product pool that violate any (user supplied) constraints are simply made ineligible for selection. For example, if an additive model

$$\hat{f}(\mathbf{x}) = \sum_{j=1}^{n} f_j(x_j) \tag{8}$$

(no interactions among the variables) was potentially thought to be adequate, all candidate basis functions involving more than one variable would be made ineligible for inclusion in the model (Friedman and Silverman, 1989). Just as easily one could limit interactions to particular variables, and even limit the particular other variables with which they are permitted to interact.

A feature of approximations based on tensor product functions is the straightforward ability to handle predictor variables of different types. Predictor variables that assume values in different kinds of sets are easily incorporated into the same regression model. So far it has been assumed that they all have values on real intervals. For these types of variables ordinary spline functions (3) are appropriate. For periodic variables that assume values on a circle (direction, months of the year) periodic splines form an appropriate basis. (Periodic splines are spline functions that are constrained to have the same values at both ends of the interval.) The full multivariate tensor product basis over all the variables would then contain mixtures of both types of univariate basis functions in some of its candidates to be selected. The basis functions themselves take on real (interval) values, but their arguments can assume values on any index set. This ability to handle and/or mix variables of different types is at the heart of the approach proposed below for modeling with variables that assume unorderable categorical values. The task is to find appropriate (real valued) basis functions for such variables to be incorporated into a MARS strategy.

There are two basic problems that limit the straightforward application of the MARS strategy outlined above; they are computational feasibility and model selection. The total number of candidate basis functions in the full tensor product is $O(N^n)$ which, except for very small values for both quantities (n, N), would require prohibitive resources to compute and store. Implementing the procedure as it is described above would require $O(N^n)$ (partial) linear-least-squares fits to enter each new basis function. In order for the procedure to be practical, a computationally feasible algorithm is necessary. This is described in Section 2.2.

Model selection also presents a difficult problem. Like all variable selection procedures that use the data response values to choose a subset, MARS is a highly nonlinear fitting procedure. This provides it with its power and flexibility but causes all of the usual model selection criteria for linear procedures to be inappropriate (see Breiman, 1989). Of these only ordinary cross-validation implemented by explicitly refitting with observations removed (Stone, 1974) or (explicit) bootstrapping (Efron, 1983) survive as statistically viable alternatives. Model selection based on cross-validation, and an approximate criterion that is more rapidly computable, are described in Section 2.3.

In addition to these two basic problems, there are a large number of "engineering details" concerning the implementation that while having no direct bearing on the fundamental ideas, nonetheless have a substantial impact on performance. These are discussed in Friedman (1991).

2.2. MARS Algorithm. This section presents a brief overview of the MARS algorithm that is described in full detail in Friedman (1991). The goal is to provide a computationally feasible approach that approximates the basis function subset selection procedure outlined in the previous section. It chooses a (relatively small) subbasis, based on the data at hand, from the (very large)

n-variable complete tensor product spline basis (6) with knots at every distinct marginal data value. One representation for these basis functions is

$$B_m(\mathbf{x}) = \prod_{k=1}^{K_m} [s_{km}(x_{v(k,m)} - t_{km})]_+^q. \tag{9}$$

Here K_m is the number of factors (interaction order) in the mth basis function, s_{km} assumes only two values, $s_{km} = \pm 1$, and indicates the (left/right) sense of the truncation, $v(k,m)$ labels the predictor variables, $1 \leq v(k,m) \leq n$, and t_{km} is a knot location on each of the corresponding variables. The exponent q is the order of the spline approximation. This "two-sided" truncated power basis (9) is equivalent to the tensor product truncated power basis (3) (6) when the monomials $\{x_j^k\}_{k=1}^{q-1}{}_{j=1}^{n}$ on each variable, and an overall constant $B_0(\mathbf{x}) = 1$, are included.

The MARS algorithm uses a forward/backward stepwise strategy to produce a set of basis functions (9). The forward part is an iterative (recursive) procedure. Each iteration simultaneously constructs an expanded list of basis functions to be considered and then decides which ones to enter at that step. Each iteration adds two new basis functions to the current model. This forward stepwise procedure is continued until a relatively large number of basis functions are included, in a deliberate attempt to overfit the data (Breiman, Friedman, Olshen, and Stone, 1984). A final appropriately sized basis function set is then selected through a backward stepwise variable subset selection procedure using the basis functions produced by the forward algorithm as candidate "variables." The model selection criterion used with the backward stepwise procedure is described in Section 2.3.

The forward stepwise procedure begins with one basis function in the model

$$B_0(\mathbf{x}) = 1. \tag{10a}$$

After the Mth iteration there are $2M + 1$ functions

$$\{B_m(\mathbf{x})\}_0^{2M} \tag{10b}$$

in the model, each of the form (9). The $(M + 1)$st iteration adds two new basis functions

$$\begin{aligned} B_{2M+1}(\mathbf{x}) &= B_{\ell(M+1)}(\mathbf{x})[+(x_{v(M+1)} - t_{M+1}]_+^q \\ B_{2M+2}(\mathbf{x}) &= B_{\ell(M+1)}(\mathbf{x})[-(x_{v(M+1)} - t_{M+1})]_+^q \end{aligned} \; . \tag{10c}$$

Here $B_{\ell(M+1)}(\mathbf{x})$ is one of the $2M+1$ basis functions already chosen (9) (10b), $0 \leq \ell(M+1) \leq 2M$, $v(M + 1)$ is one of the predictor variables (not represented in $B_{\ell(M+1)}(\mathbf{x})$), and t_{M+1} is a knot location on that variable. The three parameters $\ell(M + 1)$, $v(M + 1)$, and t_{M+1} defining the two new basis functions are chosen to be those that provide the most improvement in the fit of the (new) model to the data

$$\begin{aligned} (\ell(M+1), v(M+1), t_{M+1}) = \underset{\substack{\ell,v,t \\ \{a_m\}_0^{2M+2}}}{\operatorname{argmin}} \sum_{i=1}^{N} \Big\{ & y_i - \sum_{m=0}^{2M} a_m B_m(\mathbf{x}) \\ & - a_{2M+1} B_\ell(\mathbf{x})[+(x_v - t)]_+^q - a_{2M+2} B_\ell(\mathbf{x})[-(x_v - t)]_+^q \Big\}^2 . \end{aligned} \tag{10d}$$

Since $B_{\ell(M+1)}(\mathbf{x})$ has the form given by (9) the two new basis functions $B_{2M+1}(\mathbf{x})$ and $B_{2M+2}(\mathbf{x})$ will also have that form. Their interaction levels K_{2M+1} and K_{2M+2} will be one higher than $K_{\ell(M+1)}$, the interaction level o: $B_{\ell(M+1)}(\mathbf{x})$. For example, if $\ell(M+1) = 0$ (10a) then two additive (main effect) terms are entered into the model. If $\ell(M+1) = 0$ (10a) happens to be chosen at every iteration, then the result will be an additive model (8) (sum of functions each of a single variable). Interaction effects are produced by choosing $\ell(M+1) > 0$.

Although the forward/backward stepwise MARS algorithm produces a basis function subset of the form given by (9), and was motivated by the basis function (variable) subset selection strategy described in the previous section, it is not equivalent to that strategy. The MARS algorithm must enter basis functions of low interaction order before it can (construct and) enter basis functions of higher interaction level. It can, of course, later delete the low order interaction terms through the backward stepwise part of the procedure. A faithful implementation of the multivariate adaptive regression spline strategy (Section 2.1) would however allow any basis function in the complete tensor product basis to enter at any stage. Especially with small to moderate training samples and a large number of variables, the MARS algorithm is likely to favor the entering of lower order interaction terms compared to a faithful rendering of the adaptive spline strategy. This bias toward producing models with relatively low order interactions can represent a strong statistical advantage in those cases where the true underlying function $f(\mathbf{x})$ (1) is not dominated by interactions of the very highest order. The strength of this bias is inversely proportional to the training sample size. For small samples the MARS algorithm will try to produce models involving lower order interactions, whereas for larger sample sizes, it will more favorably entertain higher order interactions as potential candidates.

2.3. Model Selection. The forward stepwise MARS algorithm is iterated until $M_{\max}$ (tensor product spline) basis functions are synthesized. An important aspect of the MARS strategy is to choose this number to be substantially larger than would be optimal, and then to delete excess basis functions. The deletion strategy is a standard linear regression backward subset selection procedure with the $M_{\max}$ basis functions representing the stock of "variables" to be potentially selected/deleted. The motivation for this strategy lies in the (suboptimal) greedy nature of the forward stepwise algorithm. At each iteration it produces two new basis functions using only those that have already been produced in earlier iterations. Thus, the simpler basis functions synthesized early may tend to be highly suboptimal and not very useful when used in conjunction with more complex ones produced in later iterations. Their main contribution in this case is to serve as ingredients (factors) for developing the later basis functions. In order to provide adequate opportunity for the possible synthesis of these more complex (higher interaction order) basis functions, the forward stepwise procedure is allowed to produce an excess number of basis functions, which then compete (on an equal basis) with the earlier ones for inclusion in the final model.

In order to implement this type of model selection, a criterion is required that estimates (future) lack-of-fit on representative data not part of the training sample. The model that minimizes this criterion, when used with the deletion strategy described above, is taken to be the final function estimate. Since the MARS procedure is highly nonlinear, only criteria based on sample reuse such as cross-validation (Stone, 1974) or bootstrapping (Efron, 1983) can be (strictly) justified. The cross-validation criterion is

$$CV(M) = \frac{1}{N} \sum_{i=1}^{N} [y_i - \hat{f}_{M\setminus i}(\mathbf{x}_i)]^2 \tag{11}$$

where the dependence of the criterion (and model) on the number of basis functions M is explicitly indicated. Here (11) $\hat{f}_{M\setminus i}$ is the M basis function model considered in the backward stepwise deletion process, estimated with the ith (training) observation removed. Due to the hierarchical structure of the set of models considered with the stepwise strategy, this criterion (11) can be evaluated for all ($0 \leq M \leq M_{\max}$) models with the same computation required for the evaluation of just one of them (the largest).

The cross-validation criterion (11) requires the entire modeling procedure to be reapplied N times, each with one of the observations removed. It is often approximated by an analogous procedure (F-fold cross-validation), that reapplies the modeling $F < N$ times with (approximately) N/F different observations being removed each time. [$F = 10$ is often used – see Breiman et al. (1984).] Friedman (1991) proposed an approximation to (11) that requires only one evaluation of the model. It is a modification of the generalized cross-validation (GCV) criterion proposed by Craven and Wahba (1979) for use in conjunction with linear fitting methods

$$GCV(M) = \frac{1}{N} \sum_{i=1}^{N} [y_i - \hat{f}_M(\mathbf{x}_i)]^2 / \left[1 - \frac{C(M)}{N}\right]^2 . \tag{12}$$

The numerator of (12) is the lack-of-fit on the training data and the denominator represents an (inverse) penalty for increasing model complexity $C(M)$. This criterion can be (strictly) motivated for linear fitting where the basis function expansion is prespecified and only the (linear) expansion coefficients are adjusted to best fit the data. In this case $C(M) = M$, the number of parameters being fitted. The proposed modification (Friedman, 1991) for the more general case, where both the basis function set and the expansion coefficients are data determined, is to increase the "cost-complexity" $C(M)$ to reflect the additional degree to which the model is being fit to the data;

$$C(M) = M \cdot (d/2 + 1) + 1 \tag{13}$$

where here (13) M is the number of nonconstant basis functions in the model $\hat{f}_M(\mathbf{x})$ (12) being considered. The quantity d in (13) represents an additional contribution by each basis function to the overall model complexity resulting from the (nonlinear) fitting of the basis function parameters ℓ, v, an t (10d) to the data at each iterative step. Its contribution for each basis function is $d/2$ since each such nonlinear fit gives rise to two basis functions.

The quantity d in (13) can be regarded as a smoothing parameter of the procedure. Larger values result in fewer basis functions being retained thereby producing smoother estimates. An optimal value can be estimated through cross-validation. This is equivalent to cross-validating the number of basis functions M (11) since there is a one-to-one correspondence between a value for d and the size of the corresponding model produced in any particular situation. A possible advantage to using d is that its value should be more stable across situations involving differing sample sizes since N is explicitly accounted for in the penalty (12).

The modified GCV criterion (12) (13) is motivated by ad hoc heuristics and can only be justified to the extent that it performs well in model selection. Simulation results (Friedman, 1991) indicate that this is the case over a wide variety of situations using $d = 3$. The advantage over cross-validation is computational; the MARS algorithm need only be applied once. In many situations (problem size-computing platform) cross-validation is routinely feasible. In those cases for which it is not, the modified GCV criterion (12) (13) represents a computationally feasible alternative, especially for initial exploratory work.

2.4. Interpretation. Applying the MARS procedure produces a model in the form of an expansion in (two-sided) tensor product basis functions (9)

$$\hat{f}(\mathbf{x}) = a_0 + \sum_{m=1}^{M} a_m \prod_{k=1}^{K_m} [s_{km}(x_{v(k,m)} - t_{km})]_+^q. \tag{14}$$

It can be directly used to estimate missing response values y given a set of predictor variables $\mathbf{x} = (x_1, \cdots, x_n)$. In this form however it is of little interpretive value. One can increase its value for interpreting the nature of the target function $f(\mathbf{x})$ (1) by a simple rearrangement of terms:

$$\hat{f}(\mathbf{x}) = a_0 + \sum_{K_m=1} f_i(x_i) + \sum_{K_m=2} f_{ij}(x_i, x_j) + \sum_{K_m=3} f_{ijk}(x_i, x_j, x_k), \cdots. \tag{15}$$

The first sum in (15) collects together all basis functions that involve only one variable ($K_m = 1$). Each function $f_i(x_i)$ in that sum is itself a weighted sum of spline basis functions, namely those that involve x_i (and only x_i). Thus each $f_i(x_i)$ is a spline representation of a univariate function (2) (3). If its argument, x_i, does not appear in any higher order products ($K_m > 1$), then the contribution of x_i to the model is additive (main effect) and can be viewed by simply plotting $f_i(x_i)$ versus x_i.

The second sum analogously collects together all basis functions involving two (and only two) variables ($K_m = 2$). Each $f_{ij}(x_i, x_j)$ is the weighted sum of those basis functions involving both x_i and x_j, but no other variables. These functions (if present) represent two-variable interactions between x_i and x_j, and when added to the corresponding main effect functions (if any)

$$f_{ij}^*(x_i, x_j) = f_i(x_i) + f_j(x_j) + f_{ij}(x_i, x_j) \tag{16}$$

yield a tensor product spline representation of a bivariate function. If neither x_i nor x_j appear in higher order interactions, then (16) represents their joint contribution to the model that can be visually interpreted by viewing a contour or perspective mesh plot of $f_{ij}^*(x_i, x_j)$ against its arguments. Joint contributions from variables involved in higher (than two) variable interactions (if any) are constructed in an analogous manner by combining their highest order interaction terms with the corresponding lower order ones that are present in the model (14) (15). These contributions however are not readily viewable through standard graphical techniques.

The representation of the MARS model given by (15) is called the ANOVA decomposition since it breaks up the model into main (additive) effects and interaction effects of various orders. Each individual function in (15) is called an "ANOVA function" and is an expansion in tensor product spline functions involving identical predictor variable sets. (Since the locations of each of the ANOVA functions can be arbitrarily defined, they are each individually translated to have zero minimum value, and the additive constant a_0 (15) is adjusted appropriately.) The ANOVA decomposition identifies the variables that enter the model, whether they contribute additively or are involved in interactions, the order of the interaction effects and the particular variables that participate in them.

In many situations the best fitting MARS model is additive (8) or involves at most two variable interactions ($K_m \leq 2$). In these cases the model (ANOVA decomposition (15)) can be fully viewed graphically as described above. When interactions involving more than two variables are required, their contributions are not readily amenable to straightforward graphical representation, and the entire model cannot be simultaneously graphically viewed. It is still possible however to construct a sequence of views of the MARS model that collectively provides insight into the (intrinsically) high dimensional dependence. The idea is to (judiciously) choose a subset of the variables so that when their values are simultaneously fixed, the functional dependence of the MARS model on the complement variables involves at most two-variable interactions which can be viewed graphically. By examining the changing nature of these graphs as the values of the selected (conditioning) variables are changed, one can often gain some insight into the multivariate functional relationship. Due to the simple tensor product representation of the MARS model (14) such a strategy is especially straightforward to implement.

Let $\mathbf{z}$ be a d-dimensional vector ($d < n$) in the predictor variable space whose components are a subset of $\{x_1, \cdots, x_n\}$. A d-dimensional "slice" of the predictor space (Friedman, 1991) is defined by assigning (simultaneous) values to the components of $\mathbf{z}$. Let $\tilde{\mathbf{z}}$ be the $(n-d)$-dimensional vector whose components are the variables complement to those defining $\mathbf{z}$. The MARS model along the slice is a function of $\tilde{\mathbf{z}}$:

$$\tilde{f}(\tilde{\mathbf{z}}) = a_0 + \sum_{m=1}^{M} a_m \prod_{k=1}^{K_m} b_{km}(x_{v(k,m)} \mid \mathbf{z}) \tag{17}$$

where the factors b_{km} are the truncated power spline functions in (14) conditioned on the values in $\mathbf{z}$. If $v(k,m) \notin \mathbf{z}$, b_{km} is unaffected by conditioning on $\mathbf{z}$; otherwise it evaluates to a constant multiplying the coefficient a_m. Thus, the sliced model (17) has the same (tensor product) form as any MARS model; it has a corresponding ANOVA decomposition that can be interpreted and graphically visualized as discussed above.

For maximal interpretive value the particular variables defining the slice $\mathbf{z}$ must be chosen with care. They should simultaneously meet two goals: their number should be as small as possible, and the resulting sliced model $\tilde{f}(\tilde{\mathbf{z}})$, on the complement set $\tilde{\mathbf{z}}$, should be as simple as possible. In any case, it must involve no more than two-variable interactions for convenient viewing. This requires (at a minimum) that all the slicing variables each be involved in three or more variable interactions and preferably not with each other. This information is directly available from the ANOVA decomposition of the full (unsliced) MARS model so that the choice for the best slicing variable subset is usually readily apparent (see Section 4.5).

2.5. Degree-of-Continuity. One of the properties that characterizes a spline approximation is its order q (14). The approximation and its derivatives to order $q-1$ are constrained to be continuous. There are important statistical and computational considerations involved with this choice in the context of an adaptive spline strategy. These are discussed in detail in Friedman

(1991). The strategy outlined there is to use $q = 1$ (piecewise-linear) splines to construct an initial model (10c) (10d) (14). The discontinuous (first) derivatives thereby produced are then smoothed by using the initial model to derive an analogous piecewise-cubic basis with continuous first derivatives. An important aspect of this strategy is that derivatives are smoothed separately within each ANOVA function (15) [see Friedman (1991), Section 3.7].

3.0. Categorical Variables. The MARS procedure described above and in Friedman (1991) assumes that all predictor variables are ordinal; that is, there is an order relation among and a notion of distance between their possible values. The definition of a spline function (3) considers its argument to be ordinal. Not all predictor variables of interest are of this type. For example, periodic variables do not take on values that are orderable, but there is a distance relation between them. After ordinal variables, the most commonly occurring type of variable is nominal or categorical. Such variables assume a discrete set of values

$$x \in \{c_1, \cdots, c_K\} \tag{18}$$

that are neither orderable nor possess a distance relation; two categorical values are either equal or they are not equal. In some situations all predictor variables are of this type, while in others both ordinal and categorical variables are present. In either case, it is important to be able to model predictive relationships involving categorical variables.

Consider first the case of a single variable x that is categorical (18) and one would like to estimate $f(x) = E(y \mid x)$ (1). The simplest and unbiased estimate is

$$\hat{f}(x = c_k) = a_k = \text{ ave}(y \mid x = c_k) \tag{19}$$

with the average in (19) taken over the training data. These values (19) are the least-squares estimates of the coefficients in the basis function expansion

$$\hat{f}(x) = \sum_{k=1}^{K} a_k I(x = c_k), \tag{20}$$

where the basis functions are indicator ("dummy") variables of the categorical variable taking on each of its values. This (function) estimate will be accurate (low variance) to the extent that all categorical values are represented adequately with sufficient number of counts in the training data. If not, then accuracy (mean squared error on future data) may be improved by using biased estimates in the hope that the increased bias-squared will be more than offset by reduced variance. One such class of biased estimators regularizes the least-squares estimates (19) by shrinking them toward the global response mean [James and Stein (1961); see also Gu and Wahba (1991)]. This reduces the global variability of the function estimate.

In estimating functions of an ordinal variable, regularization is generally introduced through smoothing rather than global shrinking; estimates in local neighborhoods are shrunk towards each other to reduce local variability. This will be successful to the extent that the target function $f(x)$ (1) is itself smooth in the sense that its value is relatively stable (compared to the noise $\epsilon(x)$) in local regions. The goal of an adaptive smoother such as MARS is to choose the size of these regions to include the largest number of counts for given variation of the target function. In this way the smoothness of $f(x)$ is exploited to achieve maximal variance reduction for a given increase in bias-squared.

A local neighborhood represents a particular (contiguous) subset of values of an ordinal variable. Smoothness is defined as relatively low variability of the target function $f(x)$ when values of x are restricted to lie in this subset. Smoothness of the dependence of a function on a categorical variable can be analogously defined, namely low variability of the target function when its argument is restricted to particular subsets of its values. (The notion of a contiguous subset however has no meaning in this case.) This defines a smooth function $f(x)$ on a categorical variable x as one whose values tend to cluster about a relatively small number of different values, as x ranges over its complete set of values (18). This definition of smoothness depends on the variability of $f(x)$ within such clusters but not between them. A categorical variable "smoothing" procedure would attempt to discover the particular subsets of x values corresponding to each of the clusters and then produce as its function estimate the mean response value within each one.

Let $A_1, \cdots, A_L$ be subsets of the set of values (18) realized by a categorical variable x

$$A_\ell \subset \{c_1, \cdots, c_K\}, \qquad 1 \le \ell \le L, \tag{21}$$

and take as the function estimate the basis function expansion

$$\hat{f}(x) = \sum_{\ell=1}^{L} a_\ell I(x \in A_\ell), \qquad L \le K, \tag{22}$$

where the coefficients $\{a_\ell\}_1^L$ are estimated by least-squares. If $L = K$ then (22) is equivalent to the unbiased estimate (20) (provided the subsets span all values of x (18)), whereas for $L < K$ smoothing (bias) has been introduced. For a given L the goal is to choose the subsets $A_1, \cdots, A_L$ to best fit the training data. The value chosen for L is the one that minimizes future prediction error as estimated through some model selection criterion (see Section 2.3).

This procedure can be implemented in direct analogy to an adaptive spline strategy, with the basis functions (22) (indicator functions over subsets of categorical values (18) (21)) playing the role of the truncated power spline functions (2) (3). One considers all basis functions of the form

$$I(x \in A), \tag{23}$$

where A ranges over all possible subsets of (18), as candidate "variables" to be selected through a statistical variable subset selection procedure. The result of this variable (basis function) selection procedure will be a model of the form (22) with the (categorical value) subsets $A_1, \cdots, A_L$, and their number L, automatically estimated from the data.

This correspondence between spline basis functions (3) for ordinal variables and indicator functions over value subsets (23) for categorical variables forms the central idea leading to the generalizations described below. Both delineate subsets of values for their respective type of variable: indicator functions directly, and spline functions through the knot locations. Also, both restrict the form of the function estimate to be regular within each subset of values: a constant for the indicator functions, and low (qth) order polynomials for splines.

Consider now the case where there are n predictor variables $\mathbf{x} = (x_1, \cdots, x_n)$ all of which are categorical. (In this situation the data can be thought of as an n-way contingency table giving the average response value and number of counts in each cell.) Proceeding in direct analogy with the ordinal case, a set of basis functions can be derived by taking the tensor product over all of the variables of the univariate basis functions (23) defined on each one

$$\{I(x_j \in A_{\ell j})\}, \qquad 1 \leq j \leq n. \tag{24}$$

An adaptive strategy would consider all of the basis functions in this complete tensor product as candidate "variables" to be selected through a variable subset selection procedure. The MARS algorithm that approximates this strategy would be the same as that described in Section 2.2 with the replacement in (10c) (10d) of the truncated power spline basis functions by indicator functions over the categorical variable subsets

$$\begin{aligned} [+(x_v - t)]_+^q &\leftarrow I(x_v \in A) \\ [-(x_v - t)]_+^q &\leftarrow I(x_v \notin A). \end{aligned} \tag{25}$$

The lack-of-fit of the resulting model (10d) is minimized with respect to ℓ, v, and the subset A. Here (25) indicator functions take the place of spline functions and (categorical) value subsets take the place of knot locations on the respective predictor variables. The forward/backward stepwise procedure and model selection (Section 2.3) are the same. The resulting model has the form

$$\hat{f}(\mathbf{x}) = a_0 + \sum_{m=1}^{M} a_m \prod_{k=1}^{K_m} I(x_{v(k,m)} \in A_{km}) \tag{26}$$

in analogy with (14), which has a corresponding ANOVA decomposition (Section 2.4) that can be interpreted in the same manner as in the ordinal case. The corresponding curve and surface (16) plots would be replaced by one and two way tables (see examples below). Also, slicing can be implemented (Section 2.4) to explore higher order interaction effects in exactly the same manner as for ordinal variables.

Finally, consider the case of n predictor variables, n_o of which are ordinal and n_c that are categorical. (The target function $f(\mathbf{x})$ (1) can be regarded as an n_c-way contingency table, each cell of which represents a (different) function of the n_o ordinal variables.) Spline basis functions (3) are defined for each of the ordinal variables and subset indicator functions (24) for each of the categorical variables. The tensor product of these respective functions over all of the variables forms a basis in the $n = n_o + n_c$ dimensional predictor space. These serve as candidate "variables" for a variable (basis function) subset selection strategy.

The MARS algorithm for mixed ordinal and categorical variables is a straightforward generalization of that for either all ordinal (10) or all categorical variables (10) (25). Optimization with respect to the previous basis function $B_\ell(\mathbf{x})$ (already in the model) is done in the same manner. The type of factor multiplying it (10c) (10d) will depend on the type of variable x_v that is being considered to serve as the factor argument: spline factor (10c) (10d) for an ordinal variable or subset indicator function (24) (25) for a categorical variable. For a spline factor (ordinal variable) optimization is performed with respect to the knot location t (10d) whereas for an indicator factor (categorical variable) it is with respect to the corresponding subset of categorical values. The resulting joint optimization with respect to the predictor variable x_v and the parameter of its corresponding factor will give rise to the best factor (of either type) to multiply $B_\ell(\mathbf{x})$ (10c) (10d), which itself may be a mixture of spline and indicator factors. The entire optimization (10d) over ℓ, v, and t or A produces the next pair of basis functions (10c) to include in the model at the $(M+1)$st iteration. As in the all ordinal (or all categorical) case, this forward stepwise procedure for synthesizing basis functions is continued until a relatively large number $M_{\max}$ are produced. Then a backward stepwise procedure is applied using a model selection criterion in exactly the same way as described in Section 2.3.

3.1. Computation. The principal computational issue in an implementation of the MARS algorithm centers on the minimization of the lack-of-fit criterion (10d) (25) jointly with respect to all expansion coefficients and the parameters associated with the two new basis functions (knot location or categorical value subset). Optimization with respect to the other parameters (ℓ and v) is done by repeated (nested) applications of this (interior) minimization procedure. An important concern is that the computation increase only linearly with the training sample size N since this is generally the largest parameter of the problem. For the case of optimizing with respect to a knot location (ordinal variable), Friedman (1991) presented least-squares updating formulae requiring computation of $O(M\ N)$, where $2M$ is the number of basis functions currently in the model.

For a categorical variable x_v the optimization is done jointly with respect to the expansion coefficients and subsets of its values (18) (21)

$$A^* = \underset{\substack{A\subset\{c_1,\cdots,c_K\} \\ \{a_m\}_0^{2M+1}}}{\operatorname{argmin}} \sum_{i=1}^{N}\left[y_i - \sum_{m=0}^{2M} a_m \widetilde{B}_m(\mathbf{x}_i) - a_{2M+1}B_\ell(\mathbf{x}_i)I(x_{vi}\in A)\right]^2 . \tag{27}$$

Here (27) $\{\widetilde{B}_m(\mathbf{x})\}_0^{2M}$ are an orthonormalized set of basis functions that span the same (function) space as $\{B_m(\mathbf{x})\}_0^{2M}$ (10b). For a given subset A, minimization of (27) with respect to the coefficients $\{a_m\}_0^{2M+1}$ requires computation $O(MN)$. Once this optimization has been performed for one subset, it can be computed rapidly for any other subset with computation proportional only to M. This is because the minimum (27) for any given subset A (21) can be computed directly from the quantities

$$\sum_{i=1}^{N} y_i B_\ell(\mathbf{x}_i) I(x_{vi}=c_j) \text{ and } \sum_{i=1}^{N} \widetilde{B}_m(\mathbf{x}_i) B_\ell(\mathbf{x}_i) I(x_{vi}=c_j),\ \ 0\le m\le 2M. \tag{28}$$

These quantities can be evaluated once and for all at the beginning. Calculation of A^* (27) by complete enumeration over all possible subsets would therefore require computation proportional to

$$M(N+2^{K-1}). \tag{29}$$

For $K \lesssim 10$ this does not present a serious computational burden. For substantially larger values of K, however, the associated exponential growth (29) reduces the viability of this approach. The categorical value subset optimization problem (27) is equivalent to a least-squares variable subset selection using $\{B_\ell(\mathbf{x})I(x_v = c_j)\}_{j=1}^{K}$ (18) as the set of candidate "variables" to be potentially entered. The complete enumeration strategy mentioned above is equivalent to an "all subsets" variable selection method, for which powerful branch and bound algorithms exist (see Hocking (1977)) that, while still requiring exponential time, dramatically reduce computation. Use of these algorithms can double or triple the size (K) of the candidate set that is computationally feasible. An alternative approach is to employ an (approximate) stepwise variable subset selection procedure. Such a procedure does not necessarily produce an optimal subset, but usually produces a reasonably good one. Also, it is not essential that an optimal subset be found for any particular basis function since basis functions entered later have the opportunity to (at least partially) compensate for suboptimalities present in earlier ones. In addition, given the (suboptimal) stepwise nature of the other aspects of the MARS algorithm there may be little gain in applying an exact procedure in this one part.

Using a stepwise strategy in (27) reduces the computation to $O[M(N + K^2)]$ so that the total computation associated with the MARS algorithm is in this case is proportional to

$$M_{\max}^3 \left[nN + \alpha \sum_{j=1}^{n_c} K_j^2 \right] \tag{30}$$

where N is the sample size, n is the total number of predictor variables, $M_{\max}$ is the (maximum) number of basis functions produced by the forward stepwise algorithm, $\{K_j\}_1^{n_c}$ are the number of values associated with each of the n_c categorical variables, and α is a proportionality constant. Since in the pure ordinal variable case the computation is proportional to $nNM_{\max}^3$ (Friedman, 1991), the additional computational burden associated with the introduction of categorical variables is small except for very large values of K_j (~ 100).

3.2. Interpretation. Applying the modified MARS procedure for mixed ordinal and categorical variables produces a model in the form of a tensor product expansion

$$\begin{aligned} \hat{f}(\mathbf{x}) = a_0 + \sum_{m=1}^{M} a_m \prod_{\ell=1}^{K_{cm}} I(x_{v(\ell,m)} \in A_{\ell m}) \\ \cdot \prod_{k=1}^{K_{om}} [s_{km}(x_{v(k,m)} - t_{km})]_+^q. \end{aligned} \tag{31}$$

Here (31) the categorical and ordinal factors for each basis function have been separately collected together. Each basis function may involve only ordinal variables ($K_{cm=0}$), only categorical variables ($K_{om} = 0$) or mixtures of both ($K_{cm} > 0$ and $K_{om} > 0$). An ANOVA decomposition (15) of such a model can be obtained in the same manner as in the pure ordinal case (Section 2.4). It provides information as to which predictor variables (ordinal and/or categorical) enter the model, whether their respective contributions are additive (main effects) or involve interactions with other (ordinal or categorical) variables, and which variables participate in the interactions. Because of the tensor product representation (31) of the model, "slicing" can be implemented and exploited in the same manner as when all variables are ordinal (Section 2.4).

Interpretational problems arise however when one tries to graphically visualize such a model (31). These problems occur when interactions between ordinal and categorical variables are present. In the absence of such interactions there is no problem. The categorical and/or ordinal parts can be visualized separately using curves and/or surfaces to view the respective ordinal contributions, and one and/or two way tables to study the contributions from the categorical variables. The problem is that cures and surfaces are not appropriate for representing functions of categorical variables, and tables are not very useful for viewing smooth relationships on ordinal variables. Thus neither type of representation is suitable for simultaneously representing model contributions that intrinsically depend on both (categorical-ordinal interactions).

The tensor product nature (31) of a MARS model permits it to be decomposed in a manner that can aid in interpreting categorical-ordinal interactions. This "categorical-ordinal decomposition" of a MARS model is achieved by rearranging the terms, and the factors within each term, in a manner similar to that of the ANOVA decomposition. The model (31) can be reexpressed as

$$\hat{f}(\mathbf{x}) = a_0 + \sum_{m=1}^{M} L_m(\mathbf{x}_c) a_m \prod_{k=1}^{K_{om}} [s_{km}(x_{v(k,m)} - t_{km}]_+^q \tag{32a}$$

with

$$L_m(\mathbf{x}_c) = \prod_{\ell=1}^{K_{cm}} I(x_{v(\ell,m)} \in A_{\ell m}). \tag{32b}$$

Here $L_m(\mathbf{x}_c)$ collects together the dependence of the mth term on its categorical variables $\mathbf{x}_c$ (if any, $K_{cm} > 0$). Each $L_m(\mathbf{x}_c)$ evaluates to either zero or one depending an a logical (and-or) condition on the values of the categorical variables comprising its argument. (For $K_{cm} = 0$, $L_m(\mathbf{x}_c) = 1$ by definition). The "or" ($\vee$) condition is within each variable $x_{v(\ell,m)}$ and is given by the explicit subset of values $A_{\ell m}$. The "and" condition ($\wedge$) is between the variables. Let

$$A_{\ell m} = \{c_{j\ell m}\}_{j=1}^{J_{\ell m}}, \tag{33}$$

then the logical condition is

$$\bigwedge_{\ell=1}^{K_{cm}} \bigvee_{j=1}^{J_{\ell m}} (x_{v(\ell,m)} = c_{j\ell m}), \tag{34}$$

and $L_m(\mathbf{x}_c)$ is an indicator function of the truth of (34). This can be easily interpreted by listing the variables and the corresponding subset values

$$\{v_{(\ell,m)}, \{c_{j\ell m}\}_{j=1}^{J_{\ell m}}\}_{\ell=1}^{K_{cm}}. \tag{35}$$

Owing to the hierarchical nature of models produced by the MARS algorithm several of the $L_m(\mathbf{x})$ (32b) are likely to be identical. Let

$$\{\widetilde{L}_\ell(\mathbf{x}_c)\}_{\ell=1}^{M_c} = \{\text{unique } L_m(\mathbf{x}_c)\} \tag{36}$$

be the set of unique categorical factors appearing in the MARS model (31) (32). Then the model can be recast as

$$\hat{f}(\mathbf{x}) = a_0 + f_o(\mathbf{x}_o) + f_c(\mathbf{x}_c) + \sum_{\ell=1}^{M_c} \widetilde{L}_\ell(\mathbf{x}_c) f_\ell(\mathbf{x}_o). \tag{37}$$

Here (37)

$$f_o(\mathbf{x}_o) = \sum_{K_{cm}=0} a_m \prod_{k=1}^{K_{om}} [s_{km}(x_{v(k,m)} - t_{km})]_+^q \tag{38a}$$

is the contribution (if any) involving purely ordinal variables,

$$f_c(\mathbf{x}_c) = \sum_{K_{om}=0} a_m L_m(\mathbf{x}_c) \tag{38b}$$

is the contribution (if any) involving purely categorical variables, and the sum in (37) characterizes the categorical-ordinal interactions (if any). Each $f_\ell(\mathbf{x}_o)$ (37) is a function of ordinal variables only and is the (weighted) sum of the ordinal factors multiplying each $\widetilde{L}_\ell(\mathbf{x}_c)$ (36) in the MARS model (32a)

$$f_\ell(\mathbf{x}_o) = \sum_{m\in\{\widetilde{L}_\ell\}} a_m \prod_{k=1}^{K_{om}} [s_{km}(x_{v(k,m)} - t_{km})]_+^q. \tag{38c}$$

Equation 37 is the "categorical-ordinal decomposition" of the MARS model (31). It is comprised of a pure ordinal contribution $f_o(\mathbf{x}_o)$, a pure categorical contribution $f_c(\mathbf{x}_c)$, and interactions between both types of variables (sum in (37)). Each of the functions appearing in this decomposition ($f_o(\mathbf{x}_o)$, $f_c(\mathbf{x}_c)$, $\{f_\ell(\mathbf{x}_o)\}_1^{M_c}$) is an expansion in tensor product basis functions of the form produced by MARS. They each have their own individual ANOVA decompositions that can be interpreted and visualized as described above, $f_o(\mathbf{x}_o)$ and each $f_\ell(\mathbf{x}_o)$ by viewing curves and surface plots, and $f_c(\mathbf{x}_c)$ through one- and two-way tables. One can regard $f_o(\mathbf{x}_o)$ and $f_c(\mathbf{x}_c)$ (if either or both are present) as the "base" or pure contributions of the ordinal and/or categorical variables (respectively), and the (ordinal) functions $\{f_\ell(\mathbf{x}_o)\}_1^{M_c}$ as being conditionally added to the model based on the combination of values of certain categorical variables (34–37). The logical condition $\widetilde{L}_\ell(\mathbf{x}_c)$ (34) leading to the conditional inclusion of $f_\ell(\mathbf{x}_o)$ is easily interpreted from the corresponding list of categorical variables and corresponding value subsets (35).

The categorical-ordinal decomposition (37) is most useful as an interpretational tool when the number of terms M_c in the sum in (37) is not large, and the $f_\ell(\mathbf{x}_o)$ appearing there tend to involve (ordinal) variables different from each other ($\{f_{\ell'}(\mathbf{x}_o)\}_{\ell'\neq\ell}$) and from $f_o(\mathbf{x}_o)$. When this is not the case, (37) can often be too unwieldy to provide a great deal of insight into the nature of $\hat{f}(\mathbf{x})$; the model is intrinsically too high dimensional to be easily visualized. When this happens, slicing (Section 2.4) can often be of value. As discussed in Section 2.4, the goal of slicing is to reduce the dimensionality of the model by conditioning on a judiciously chosen subset of the variables so that the resulting model on the complement variables is easier to interpret. In the presence of interactions between categorical and ordinal variables interpretability includes the requirement that the complement variables all be of one type, either all categorical or all ordinal. This requirement is in addition to those discussed in Section 2.4. All the necessary information for choosing the smallest variable subset for slicing that meets these requirements is contained in the ANOVA decomposition (15) of the full MARS model (31). (See illustration in Section 4.5.)

3.3. Nested Variables. In some problems one has predictor variables that are meaningful only when some other (categorical) predictor variable takes on values within a particular subset. For example, a treatment variable x_j may have three possible values: medication, therapy, or surgery. Associated with each of these values is a distinct set of other variables (ordinal or categorical) that characterize each corresponding treatment, and only have meaning if that particular treatment was applied. These latter variables are said to be nested within the treatment variable, each to the corresponding value for which it has meaning. Formally let x_j ("nestor") take on K_j categorical values

$$x_j \in \{c_{1j} \cdots c_{K_j j}\}$$

and consider another variable x_v ("nestie") that has meaningful values only when $x_j \in A_{vj}$, where

$$A_{vj} \subset \{c_{1j} \cdots c_{K_j j}\}. \tag{39}$$

In this case x_v can only enter the approximating model $\hat{f}(\mathbf{x})$ interacting with x_j. Purely additive (main) effects involving only x_v are not meaningful. Also, interactions between x_v and other variables have no meaning unless they also involve x_j in each such interaction. Due to the tensor product form of MARS approximations (31), this type of variable nesting is straightforward to implement.

In order for nested variables to be treated properly one must ensure that each one only contributes to the model when its value has meaning, as defined by the corresponding value of its nestor. In the context of MARS modeling this constraint can be met by requiring that any basis function (31) involving a nested variable x_v in one of its factors also involves a factor of the form $I(x_j \in A)$, where x_j is the variable to which x_v is nested, and $A \subseteq A_{vj}$ (39). This ensures that any basis function involving x_v will have the value zero when values of x_v have no meaning. This in turn can be accomplished through a minor modification to the forward stepwise part of the MARS algorithm (Sections 2.2 and 3.0); when a factor involving a nested variable x_v is being considered in the optimization loop, only previous basis functions $B_\ell(\mathbf{x})$ that include a factor $I(x_j \in A)$, $A \subseteq A_{vj}$ (39), are made eligible to multiply it, and its complement (10c) (25). Although this modification alone properly constrains the MARS model with respect to nested variables, it places them at a competitive disadvantage to other (nonnested) variables in entering the model due to the forward stepwise (greedy) nature of the MARS algorithm (Section 2.2).

Unlike other variables, nested variables are not permitted to enter additively (main effect), but must wait for their nestors to enter, before they can enter the model at all. It is not unusual for a nested variable to have considerably more predictive power than its nestor variable, especially when the sole purpose of the nesting is to define the existence of values for the nestie (missing values – Section 3.4). In order to place nested variables on an equal footing with other variables in the context of the MARS algorithm, it is necessary to regard interactions with their nestors on the same level as main effects for other variables. This motivates a slightly more involved modification of the basic MARS algorithm for handling nested variables.

At each $[(M+1)\text{st}]$ iteration of the basic MARS algorithm (Sections 2.2 and 3.0) one considers multiplying a basis function $B_\ell(\mathbf{x})$, entered at a previous iteration, by a factor $b(x_v \mid p)$ (and its complement $\bar{b}(x_v \mid p)$) involving one of the predictor variables x_v. If x_v is ordinal $b(x_v \mid p)$ is a truncated power spline function (3) (10c) and the parameter p is an (optimized) knot location. If it is categorical $b(x_v \mid p)$ is an indicator function (25) and p is an (optimized) value subset. All possible pairings of variables and previous basis functions

$$\{B_\ell(\mathbf{x}), b(x_v \mid p)\}_{\ell=0}^{2M}\;{}_{v=1}^{\;n} \tag{40}$$

are considered. The best pairing gives rise to two new basis functions (10c) (25) to be entered into the current model. The modification to this strategy for nested variables is as follows. Whenever a factor $b(x_v \mid p)$ involves a nested variable x_v, each previous basis function $B_\ell(\mathbf{x})$ (40) is examined. If it contains a factor $I(x_j \in A)$ involving x_v's nestor variable x_j, and the corresponding value subset has the property $A \subseteq A_{vj}$ (39), then the pairing $[B_\ell(\mathbf{x}), b(x_v \mid p)]$ is treated in the usual manner. If $B_\ell(\mathbf{x})$ contains such a factor, but with $A \not\subseteq A_{vj}$, then the pairing is made ineligible to be a solution. If $B_\ell(\mathbf{x})$ does not contain a factor involving x_v's nestor, x_j, then it is provisionally modified to include it

$$B_\ell(\mathbf{x}) \leftarrow B_\ell(\mathbf{x}) I(x_j \in A_{vj}) \tag{41}$$

before being paired with $b(x_v \mid p)$ (40). If this particular pairing turns out to be the best one (solution) then (up to) four (rather than two) new basis functions are entered into the model at this iteration in analogy with (10c) (25). They are:

$$\begin{aligned}
&B_\ell(\mathbf{x}) I(x_j \in A_{vj}) \\
&B_\ell(\mathbf{x}) I(x_j \notin A_{vj}) \\
&B_\ell(\mathbf{x}) I(x_j \in A_{vj}) b(x_v \mid p) \\
&B_\ell(\mathbf{x}) I(x_j \in A_{vj}) \bar{b}(x_v \mid p).
\end{aligned} \tag{42}$$

The second basis function in (42) need not be entered at this step if it already exists in the model.

This modified strategy (41) (42) ensures that all basis functions involving a nested variable x_v will be zero when the value for x_v is not defined. This in turn ensures that the final approximation $\hat{f}(\mathbf{x})$ (31) will exhibit a dependence on x_v only when its value is defined. Introduction of the partial "look ahead" feature (41) for nested variables into the MARS algorithm gives nested predictors the same opportunity to enter the model as other (nonnested) ones. Putting the (first) two additional basis functions (42), not involving x_v, into the model is necessary to preserve the full generality of the forward stepwise procedure. They are available to serve as ingredients for the construction of future basis functions in later iterations. If they turn out not to be needed they will likely be removed during the backward stepwise basis function deletion part of the procedure (Section 2.3). This strategy also ensures that a MARS model involving nested variables has the same form (31) as any ordinary MARS model. Thus all interpretational tools such as the ANOVA decomposition and slicing (Section 2.4) and the categorical-ordinal decomposition (Section 3.2) can be directly used in the same manner as described above with no special consideration being needed for the nested nature of some of the variables.

3.4. Missing Values. One of the most useful applications of variable nesting in MARS is in dealing with missing values among the predictor variables. In many problems one is forced to do prediction and/or training in the presence of incomplete data; values for some (or many) of the predictor variables are missing. This often has serious consequences for many learning (regression) procedures, either severely degrading their performance or rendering their application impossible.

One method that is often used is to replace each missing value with the mean of the corresponding variable over the training data. In the case of linear modeling this removes the influence of that variable for the observations missing its value without distorting the function estimate itself (slope values). For nonlinear modeling however this is not the case and such an approach can lead to severe distortion of the model estimate. Another standard method for treating missing values is to delete all training observations that are incomplete. In a problem with ten predictor variables, each one of which (independently) stands a 20% chance of having a missing value, roughly 10% of the observations will be complete. This strategy would then delete 90% of the information, even though only 20% of it was missing. (Also such a strategy provides no mechanism for doing prediction with missing values.) It is important that the degree of reduced performance in the presence of missing values bear some reasonable relation to the amount of information that is actually missing. In particular if there are strong associations among certain sets of predictors the information loss, if one (or more) of them is missing, is small since the same information is present in the remaining others that are highly correlated with it. A missing value strategy should be able to take advantage of such redundancy to mitigate the damage associated with missing values.

Missing values among the predictor variables can be handled through variable nesting (Section 3.3). One introduces an additional indicator (dummy) variable $x_{v'}$ for each (original) variable x_v with missing values. These new variables indicate the presence of a (nonmissing) value for each corresponding original variable

$$x_{v'} = \begin{cases} 0 & \text{if } x_v \text{ is missing,} \\ 1 & \text{otherwise .} \end{cases} \tag{43}$$

Each original variable x_v (with missing values) is nested within its corresponding indicator variable $x_{v'}$ to the value $x_{v'} = 1$ $[A_{vv'} = \{1\}, (39)]$. The strategy for variable nesting described above (42) ensures that the approximation $\hat{f}(\mathbf{x})$ (31) will exhibit a dependence on each variable x_v with missing values only when a value for that variable is present ($x_{v'} = 1$). The partial "look ahead" for nested variables (41) ensures that variables with missing values compete for entry into the model on the same basis as those with no missing values to the extent that their values are present.

This strategy also allows variables that are highly associated with one another to act as "surrogates" for each other (Breiman, et al., 1984) when their values are missing. This opportunity is provided by the introduction of the second basis function in (42). This ensures that every time a basis function involving a variable x_v with missing values is entered $[B_\ell(\mathbf{x})I(x_{v'} = 1)b(x_v \mid p)]$, a corresponding basis function $B_\ell(\mathbf{x})I(x_{v'} = 0)$ is also entered. This latter basis function can then serve (in future iterations of the forward stepwise algorithm) as a multiplier to create a basis function of the form $B_\ell(\mathbf{x})I(x_{v'} = 0)b(x_u \mid p)$. If x_u is highly associated with x_v then this new basis function will serve as a surrogate for the original one when x_v is missing. If x_u also has missing values, this latter basis function will have the form

$$B_\ell(\mathbf{x})I(x_{v'} = 0)I(x_{u'} = 1)b(x_u \mid p)$$

(42) and x_v and x_u will serve as surrogates for each other. In this case an additional basis function

$$B_\ell(\mathbf{x})I(x_{v'} = 0)I(x_{u'} = 0)$$

is also produced (42) that enables the creation of future basis functions to serve as surrogates when both x_v and x_u are missing. In all cases if these extra basis functions turn out not to be useful they will likely be deleted in the backward stepwise part of the procedure (Section 2.3).

This missing value strategy allows (in principle) the MARS algorithm to produce a different model for each possible combination of missing values among the predictor variables. If there are m variables with missing values that enter the model, there are potentially 2^m such models. Each of these models will be highly correlated, however, sharing many factors and basis functions in common. When the model is to be evaluated (response prediction) with some combination of the predictor values missing, the appropriate model for that missing combination is automatically selected and evaluated provided that the corresponding (missing) dummy variables $\{x_{v'}\}_{v'=n+1}^{n+m}$ are included with the predictor variable vector.

Each of the separate models produced for different combinations of missing values can be identified through categorical-ordinal decomposition (37). They can be separately produced for interpretation through the slicing technique (Section 2.4). Each corresponding slicing vector $\mathbf{z}$ (17) would simultaneously condition on all of the missing dummy variables $\{x_{v'}\}_{n+1}^{n+m}$ thereby producing a function involving only the original variables. The particular model associated with a given missing/nonmissing combination is specified by setting the corresponding (missing-dummy) components of the slicing vector to 0/1. For interpretational purposes the most useful model is likely to be the one corresponding to no missing values since it exhibits the dependence on all of the (original) predictor variables that enter the model. This model is obtained by setting all of the (missing-dummy) components of the slicing vector to one

$$\{x_{v'} = 1\}_{v'=n+1}^{n+m}. \tag{44}$$

This model can then be interpreted in the identical manner as any other that does not involve missing values through the ANOVA and categorical-ordinal decompositions, and further slicing (if necessary) on selected (original) variables.

The missing value strategy outlined above does not assume that the probability of missing values for a predictor variable is independent of the response value, values of it or other predictor variables, or the fact that other predictor variables have missing values. If the missing probabilities change with different response values then there will be predictive information in the missing-dummy variables $\{x_{v'}\}_{n+1}^{n+m}$. Since these are treated on an equal basis with the other variables, they are eligible to enter by themselves (main effect) or in interactions with other variables that are not (necessarily) nested to them. The MARS procedure in this way automatically attempts to use any information present in the missing frequencies (even when nonmissing values of their corresponding (original) variable may have no association with the response).

This approach to missing values does assume that the relative missing frequencies in the training data are representative of those in future data to be predicted. If not, some loss in efficiency (predictive power) will result. In this sense the procedure treats missing values like any other predictor variable values. If the design produced by the training data is not representative of the joint distribution of future data, reduced predictive power will likely result. In particular the procedure does not produce a model corresponding to missing values in a predictor variable if that variable has no missing values in the training data. If a predictor has too few missing values in the training data, compared to future data (to be predicted), then the algorithm will devote too little effort in producing a model for those missing values. On the other hand, if it has correspondingly too many missing values, too much effort will be devoted to this situation at the expense of other aspects of the model.

If the frequency of missing values for a predictor variable is too high in the training data there is nothing that can be done. There is an inherent loss of information in this case. On the other hand, if it is known (or suspected) that missing values in future data will be more likely than that represented in the training data, remedial action is possible by increasing the sample size through resampling. One can produce an additional sample of training data by randomly drawing (with replacement) observations from the original training sample and flagging predictor variables as being missing. The training procedure (MARS algorithm) is then applied to the combined data set consisting of the original data and the additional (randomly) selected data.

This resampling procedure consists of first specifying the fraction of missing values desired for each predictor variable in the final training sample (original plus resampled data). The starting sample is the original training data. Each predictor variable for which the fraction of missing data in the current (original plus so far resampled) data set is less than that specified is considered. An additional observation is drawn at random from the current sample and then added to the sample with that variable flagged as missing. This is repeated until the fraction of missing values for that variable is increased to the prespecified level. This resampling is then applied to the next variable with too few missing values in the current sample, and so on. Repeated passes are made over the variables in this manner until the procedure converges with the fraction of missing values on each variable in the resulting total sample being (nearly) equal to their prespecified values.

Unlike missing values that appear naturally in the original data, this resampling scheme produces (by construction) missing values independently at random. If this is not the case in future data, some potential prediction accuracy may be lost. If one has knowledge of the dependencies of missing value probabilities on the other aspects of the problem, this could be incorporated into a (modified) resampling scheme.

3.5. Other Strategies. In this section an attempt is made to provide some insight into the MARS approach by comparing and contrasting it to other commonly used methods for regression modeling with categorical variables. These are the "dummy variable" technique, projection, and recursive partitioning.

The "dummy variable" technique treats all of the variables as ordinal. Each categorical variable is transformed to a set of ordinal variables by introducing an indicator function for each of its potential values. Thus, categorical variable x_i with K_i values is transformed to K_i 0/1-valued ordinal variables. If there are n_c categorical variables, then this produces

$$n_D = \sum_{i=1}^{n_c} K_i$$

"dummy variables" that are combined with the intrinsically ordinal variables, thereby producing a set of variables that are entirely ordinal. The primary limitation of this approach is that it provides no "smoothing" on the categorical variables (see Section 3.0). If there are only a few categorical variables (n_c small) each of which takes on only a few values (all K_i small) then this lack of the ability to smooth may not be a serious limitation. On the other hand, for high dimensional (multi-way) tables (n_c not small) and/or some of the categorical variables taking on more than a few values (K_i not small) smoothing becomes essential to moderate the high variance associated with the inherent sparseness of the contingency table associated with the categorical variables.

With the projection technique regularization (smoothing) is introduced (only) through projection. The regression model for the corresponding multi-way table is taken to be the sum of lower dimensional tables. No smoothing is done within these low dimensional components. As a result the model usually is limited to (one dimensional) main effects, and sometimes (if there is sufficient data) selected two-variable interactions. Projection methods have been highly successful in modeling low dimensional tables ($n \lesssim 2$) especially when the corresponding number of cells is not too large. Success has been more limited when it has been applied to high dimensional sparse tables with many cells.

The limitations of the projection approach when applied to large sparse contingency tables center on its singular ability to smooth only by projection. Even if a high dimensional table can be well approximated by a sum of lower dimensional ones (weak higher order interactions) additional smoothing within each low dimensional table can often be beneficial, especially if they contain many cells. If an adequate model requires higher order interaction effects then smoothing only by low dimensional projection is at a strong disadvantage. In problems involving mixed categorical and ordinal variables, the ordinal variables are discretized into a relatively small number of values and treated as categorical, unless interactions between ordinal and categorical variables are not allowed. The limitations discussed above thereby apply to this situation as well. In addition, this approach does not produce approximations with a continuous dependence on the ordinal variables, nor does it attempt to take advantage of any continuity properties of the target function with respect to them.

Recursive partitioning methods such as CART (Breiman, Friedman, Olshen, and Stone, 1984), AID (Morgan and Sonquist, 1963), and CHAID (Kass, 1980), are specifically intended for application in situations involving a large number of variables, some, many, or all of which are categorical – possibly taking on many values. All variables are basically treated as categorical. Ordinal variables are discretized into intervals. (With some recursive partitioning methods (CART, AID) this discretization is done adaptively as part of the procedure to improve goodness-of-fit to the training data.) With recursive partitioning smoothing is performed by clustering; there is no smoothing by projection. The corresponding n-way table is partitioned into smaller subtables by recursively splitting it into blocks (clusters). The function estimate is taken to be a constant within each such block. Each subtable is divided into two (or more) smaller tables by a split(s) on one of the predictor variables. (The procedure begins with the original full contingency table.) For ordinal variables, potential splits divide the (current) range into two intervals. For categorical variables all possible splits of the current set of values into two (or more) subsets are considered. Each split optimizes the goodness-of-fit of the current model (conditioned on previous splits) jointly with respect to the splitting variable and corresponding parameter (split point or categorical value subset). This partitioning continues until further splitting fails to improve the model. The result is a piecewise constant approximation of the target function $f(\mathbf{x})$ (1).

Recursive partitioning methods have met with considerable success, especially in situations where the projection approach tends to fail. These are large sparse multi-way tables for which the target function involves high order interaction effects. This is due to the fact that the approximations it produces must intrinsically involve higher order interactions in order to capture multivariable dependencies. Each split of a subtable on a new variable, not yet used to delineate that subtable, introduces a higher order interaction effect involving an additional predictor variable. Thus, as the recursive partitioning proceeds, higher and higher order interactions are thereby introduced. This strongly limits the ability of recursive partitioning to provide good approximations in (often occurring) situations where the target function is dominated by main effects and/or interactions of low order compared to the number of predictor variables n.

Unlike both projection (with categorical-ordinal interactions) and recursive partitioning, the MARS approach produces strictly continuous approximations with respect to the ordinal variables. It thereby attempts to use to advantage any such continuity present in the target function. When applied with categorical variables it can be viewed as a hybrid between the (complementary) approaches represented by projection and recursive partitioning. It can simultaneously smooth both by projection and clustering. It directly (and adaptively) estimates how much of each to do in any particular situation based on the training data. In situations characterized by high order interactions, it will tend mainly to use clustering, whereas in cases where the target function is dominated by low order interactions, especially involving highly structured dependencies, the smoothing will tend to be mostly by projection. MARS modeling contains pure projection and recursive partitioning approximations as (extreme) special cases; it has the potential ability to introduce smoothing entirely by clustering or entirely by projection if dictated by the data. The hope is that it will be competitive in these extreme cases, and in addition provide superior performance in those situations that form the large spectrum of problems in between them.

4.0. Illustrative Examples. In this section, applications of the MARS approach to several data sets involving both ordinal and categorical variables are presented. The first two examples are artificially generated so that the results can be compared with the known truth. Data for the third example is taken from a sample survey, whereas the fourth and fifth use data presented in Andrews and Herzberg (1985). The goal is to illustrate the type of information that can be obtained from this kind of analysis.

4.1. Artificial Data. The purpose of this example is to explore the ability of MARS to simultaneously smooth both by clustering and projecting. There are two categorical and two ordinal variables. Each of the categorical variables, x_1 and x_2, (randomly) assume ten values (0–9) independently from a uniform distribution. The ordinal variables (x_3 and x_4) are randomly generated from a joint uniform distribution with values in the range zero to one. The target function is taken to be

$$f(x_3, x_4) = \begin{cases} 0 & \text{if } x_1 = \text{even and } x_2 = \text{even} \\ 2\sin(\pi x_3 x_4) & \text{if } x_1 = \text{odd and } x_2 = \text{even} \\ \cos(2\pi x_3) + 0.5\log(10 x_4) & \text{if } x_1 = \text{even and } x_2 = \text{odd} \\ 2\sin(\pi x_3 x_4) + \cos(2\pi x_3) + 0.5\log(10 x_4) & \text{if } x_1 = \text{odd and } x_2 = \text{odd.} \end{cases} \tag{45}$$

The values of the categorical variables (x_1 and x_2) can be viewed as forming a 10×10 contingency table. The target function (45) is a particular function of the ordinal variables (x_3 and x_4) in each of the 100 cells of this table. Values of the response y were generated by adding (independent homoscedastic) normal noise to the target function

$$y_i = f(\mathbf{x}_i) + \sigma\epsilon_i, \quad i = 1, N, \tag{46}$$

with $\epsilon_i \sim N(0,1)$ and the value of σ $(= .36)$ chosen so that the signal to noise ratio $\mathrm{var}(f)/\,\mathrm{var}(\epsilon)$ is 3/1.

Obtaining even a reasonable estimate of $f(x_3, x_4)$ (45) separately in each cell of the 10×10 (x_1, x_2)-contingency table would require a moderate number ($\sim$ 100) of observations in each one. The overall sample size would thereby have to be quite large ($\sim$ 10000). The hope is that the MARS procedure can take advantage of the fact that the target function (45) involves only three variable interactions, and more importantly, that its dependence on the categorical variables is very smooth; it depends only on the (even/odd) parity of their (categorical) values.

Table 1 displays the ANOVA decomposition (Section 2.4) of the model obtained by applying MARS to (45) (46) with a sample of two hundred ($N = 200$) observations. The first line gives the GCV estimate (12) of the squared multiple correlation coefficient R^2 for the fit using the optimal complexity parameter $d = 8.59$ (13) estimated by 20-fold cross-validation. The cross-validated estimate of the corresponding goodness-of-fit was $CV = 0.84$ so that the GCV estimate appears here to be somewhat pessimistic. The MARS model has seven ANOVA functions (15) that involve two-variable interactions in all four variables and a three-variable interaction between variables 1, 3, and 4. The second column of Table 1 gives the standard deviation of each ANOVA function; this is a measure of the relative importance of each one. The third column provides another such measure by giving the R^2_{GCV} of the MARS model with the corresponding ANOVA function removed. This can be compared to that for the full model (first line) to gauge the contribution of each respective ANOVA function. The fourth column gives the number of basis functions comprising each ANOVA function and the last column gives the variables associated with each one.

Table 1
ANOVA Decomposition of the MARS Model on
the Artificial Data of Section 4.1

R^2_{GCV}(full model) = .72

ANOVA function	standard deviation	$\backslash R^2_{GCV}$	#basis functions	variables
1	.78	.29	1	1
2	.29	.69	1	2
3	.62	.54	2	1 4
4	.64	.51	1	1 3
5	.92	.22	2	2 3
6	.27	.66	1	2 4
7	.40	.69	2	1 3 4

One sees from Table 1 that the resulting MARS model on these data involves interactions involving at most three variables. Furthermore the three-variable ANOVA function (last line) seems to be making a fairly small contribution. Reapplying MARS to these data restricting the model to involve at most two-variable interactions gives a (20-fold) cross-validated $R^2_{CV} = 0.77$ (compared to 0.84 for full MARS modeling); restricting the model to be additive (no interactions) yields $R^2_{CV} = 0.44$. Thus, the three-variable interaction component of the model appears to be making a small but nonnegligible contribution, whereas the two-variable interactions appear to be crucial.

Figure 1 shows (graphically) the categorical-ordinal decomposition (37) of the full MARS model. It consists of three functions of x_3 and/or x_4. The first $f_3(x_3)$ (upper left) is a function only of x_3 and the second $f_4(x_4)$ (upper right) is a function only of x_4. The bottom two frames show a function of x_3 and x_4, $f_{34}(x_3, x_4)$, from opposite perspective views. Each of these functions is conditionally added to the model depending on the values of x_1 or x_2. The corresponding logical condition is indicated below the frames of the respective functions. As can be seen, the MARS estimate for these data is

$$\hat{f}(\mathbf{x}) = I(x_2 \in \text{ odd})[f_3(x_3) + f_4(x_4)] + I(x_1 \in \text{odd}) f_{34}(x_3, x_4)$$

with f_3, f_4, and f_{34} shown (respectively) in Figure 1. The (scaled) predictive squared error

$$pse(\hat{f}) = \int [f(\mathbf{x}) - \hat{f}(\mathbf{x})]^2 d^4x \Big/ \int [f(\mathbf{x}) - \bar{f}]^2 d^4x, \tag{47}$$

with

$$\bar{f} = \int f(\mathbf{x}) d^4x,$$

of this approximation is $pse(\hat{f}) = 0.048$ as estimated from 5000 additional data points generated according to (45). Thus, the approximation accounts for about 95% of the variance of the target function (45) over the range of the predictor variables. The individual function estimates (f_3, f_4, f_{34}) shown in Figure 1 bear a reasonable resemblance to the corresponding ones in the target function (45). This can be judged by reference to Figure 2 which shows the corresponding estimate $\hat{f}(\mathbf{x})$ for $N = 400$ observations. The predictive squared error for this (latter) estimate is $pse(\hat{f}) = 0.024$.

This example indicates that the MARS approach is able to exploit to advantage highly smooth dependencies on both ordinal and categorical variables with smoothness on categorical variables defined as in Section 3.0. Of course, this example is purposely contrived to be favorable for the MARS strategy in order to illustrate this point. Target functions with a lower degree of smoothness will give rise to estimates that are either less accurate or require larger training samples for comparable accuracy. Similarly, simpler and/or smoother functions will be easier to estimate. For this example the categorical/ordinal decomposition (37) provided a fairly interpretable representation of the approximation. This is because it involves a relatively small number of (ordinal) functions $\{f_\ell(\mathbf{x})\}$ (37) which is a consequence of the form of the target function (45). Not all (successful) applications of the MARS procedure result in such parsimonious categorical-ordinal decompositions. For these (more complicated) situations the categorical-ordinal decomposition is less useful as an interpretational tool and the "slicing" approach (Section 2.4) becomes (relatively) more valuable for interpreting the multivariate nature of the approximation.

4.2. Missing Values. The purpose of this example is to test the missing value strategy based on variable nesting described in Section 3.4. The data are artificially generated so that the relative information loss caused by presence of missing values can be gauged. There are ten predictor variables (all ordinal) uniformly generated in $[0,1]$. The target function was taken to be (Friedman and Silverman, 1989)

$$f(\mathbf{x}) = 0.1e^{4x_1} + 4/(1+e^{-20(x_2-0.5)}) + 3x_3 + 2x_4 + x_5 \tag{48}$$

and the response variable obtained by adding standard normal noise

$$y_i = f(\mathbf{x}_i) + \epsilon_i, \quad \epsilon_i \sim N(0,1), \tag{49}$$

to produce a sample of $N = 200$ $(1 \leq i \leq 200)$ training observations. The target function is seen to be additive in the first five predictor variables with the second five (noise variables) being unrelated to the response. The (true) target function accounts for $R^2(f) = 0.86$ of the variance of the response.

Table 2 shows summary results from a Monte Carlo study consisting of 100 independent replications of (48) (49). For each replication the predictor variables $\{\mathbf{x}_i\}_1^{200}$, as well as the noise $\{\epsilon_i\}_1^{200}$, were independently regenerated. The rows of Table 2 give results for each of four situations. The first is for the target function (48). The second is for the MARS estimate with no missing values. The third row represents the case where each of the ten predictor variables is replaced by a missing value independently with 20% probability. Thus on average each training observation has two missing predictor values, and only 21 (of the 200) observations in the training sample are complete (no missing values). The situation represented by the fourth row of Table 2 is the same as for the third except that a simple correlational structure is introduced among the predictor variables. After they have been independently generated, the last five ($x_6 - x_{10}$) are recalculated as

$$x_{i+5} \leftarrow 0.9x_i + 0.1x_{i+5}, \quad i = 1,5 \tag{50}$$

and then missing values are assigned as described above. This introduces a strong association between the pairs x_i and x_{i+5} $(1 \leq i \leq 5)$ but no other associations among the predictor variables.

The columns of Table 2 give three summary statistics of the MARS solutions computed over the 100 replications for each of the situations represented by the rows. (The quantities in parentheses are the respective standard deviations of the corresponding quantities over the 100 replications. The corresponding standard error on each statistic in Table 2 is one tenth this standard deviation.) These statistics were obtained by generating an independent set of 5000 ("test") observations according to (48) (49) for each respective situation (missing values), and computing from them the corresponding quantities from the MARS solutions based on the 100 independent training samples, each of size $N = 200$. The entries in Table 2 thereby reflect the expected performance accuracy over future data not part of the training data. The first column of Table 2 is the squared multiple correlation coefficient on future data for which predictor values are missing by the same mechanism as that for the corresponding training data. The second column gives the corresponding (scaled) predictive squared error (47) which reflects target function (48) prediction error (squared) in the presence of missing values in the (test) observations to be predicted. The third column gives this same quantity but for future (test) observations that are complete (no missing values). This last quantity reflects the accuracy of the actual function (48) estimate and thereby provides information as to the accuracy loss solely due to missing values in the training data, as opposed to those in future test data as well. It is also important because this is the expected accuracy of the estimated MARS model for no missing values, produced by slicing the missing ("dummy") variables $\{x_{v'} = 1\}_{11}^{20}$ (44). As discussed in Section 3.4, it is this model that is most useful for interpretation.

Table 2
Performance of Mars with Missing Values in Both
Training Data and Future (Test) Data

situation	R^2 (all)	pse (all)	pse (no missing)
true model	.86	0	0
no missing	.84 (.01)	.025 (.01)	.025 (.01)
20% missing (no correlation)	.45 (.10)	.48 (.12)	.078 (.04)
20% missing (correlation)	.67 (.04)	.22 (.04)	.064 (.02)

Examination of Table 2 indicates considerable loss in prediction accuracy when predictor variable values are missing at this very high (20%) rate (row 3). On average each observation to be predicted has one important predictor variable ($x_1, \cdots, x_5$) missing; many have more than one. Results shown in the third column however indicate that most of this information loss is due to missing values in (test) observations to be predicted and not estimation error due to missing values in the training data. While the former error increases by a factor of 19 (column 2), the latter increases only by a factor of three (column 3) and is still fairly small. Comparing the third and fourth rows of Table 2 one sees that the missing value strategy is able to use "surrogate" information to advantage. The (function) prediction error (column 2) is reduced by over a half by using information in correlated variables when predictors are missing (row 4). This illustrates that the MARS missing value strategy has been at least partially successful in encoding this surrogate information. Note that on average each future test observation has a 20% chance that both an important predictor variable $\{x_i\}_1^5$ and its surrogate (x_{i+5}) will *both* be missing. In this case there is no additional surrogate information due to the way these particular data (50) were generated. In many settings with observational data however there are often strong associations among variable subsets of higher cardinality. At least in principle, the MARS missing value strategy should be able to capture this to further reduce prediction error.

In the preceding example, the same (probabilistic) mechanism produced missing values in both the training data and future (test) data to be predicted. As noted in Section 3.4, this may not always be the case. In particular, a resampling strategy (on the training data) was proposed to improve accuracy if a larger fraction of missing values was expected in future test data than was present in the training sample. This resampling approach is now examined in the same context as the preceding example (48) (49). In this case each training sample ($N = 200$) had no missing values. The resampling scheme described in Section 3.4 was then applied to derive (much larger) training samples with each (of the ten) predictor variables having a prespecified fraction p of their values missing. (Examples with $p = 10\%$, 15%, and 20% were run.) MARS was then applied to these larger training sets with (induced) missing values. The resulting MARS models were then used to predict 5000 newly generated test observations (48) (49) with each predictor variable missing independently at random with 15% probability.

Table 3 shows summary statistics over 100 replications of this procedure. The first and third columns give the squared multiple correlation coefficient between the model predictions and the test data; the second and fourth show the corresponding (scaled) predictive squared error (47) of the function (48) estimate. The first two columns are for the case of no (population) associations among the predictor variables, while the last two are for the correlational structure given by (50). (As in Table 2, the quantities in parentheses are the respective standard deviations of each corresponding quantity over the 100 replications.)

The results shown in Table 3 indicate that prediction accuracy with (15%) missing data increases as one increases the number of missing values in the training data through resampling, although there seems to a somewhat diminishing return for the largest number. More striking is the decrease in variability of this quantity with increasing (resampled) training data. The only price paid for this is increased computation. In this particular example where there are no missing values in the original training data (of size N) and the fraction of missing values for all ($n = 10$) predictor variables is p, the size of the final resampled training data set is approximately $N/(1-p)^n$. The results of Table 3 also indicate the "surrogate" effect (last two columns); associations among the predictor variables are used to advantage by this strategy to reduce information loss in the presence of missing predictor values.

Table 3
Performance of MARS on Predicting with 15%
Missing Values Using Training Data Resampling

training	no correlation		correlation	
% missing	R^2	pse	R^2	pse
10	.47 (.26)	.46 (.30)	.68 (.11)	.21 (.13)
15	.56 (.10)	.35 (.12)	.76 (.04)	.12 (.04)
20	.61 (.03)	.29 (.03)	.78 (.01)	.10 (.02)

4.3. Sample Survey Data. The data for this example came from questionnaires filled out by shopping mall patrons throughout the San Francisco Bay Area (Impact Resources, Columbus, Ohio). Among the questions asked were 14 that involved demographic information. Table 4 lists these variables and their possible values. Variables marked with a "*" are considered categorical, whereas the others are treated as ordinal variables. In this exercise the response is taken to be household income, x_6, with the predictors being the other 13 variables listed in Table 4. Data from $N = 1371$ questionnaires were used for this analysis.

Table 4
Demographic Variables used in Sample Survey Example
with Their Possible Values (Section 4.3)

*1. sex (male / female)
*2. marital status (married / living together–not married / divorced or separated / widowed / single–never married)
3. age (14–17 / 18–24 / 25–34 / 35–44 / 45–54 / 55–64 / over 65)
4. education (grade 8 or less / grades 9–11 / graduated high school / 1–3 years of college / college graduate / graduate study)
*5. occupation (professional–managerial / sales worker / factory worker, laborer, driver / clerical, service worker / homemaker / student / military / retired / unemployed)
6. annual household income (< $10K / $10K–$15K / $15K–$20K / $20K–$25K / $25K–$30K / $30K–$40K / $40K–$50K / $50K–$75K / > $75K)
7. how long in Bay Area (< 1 year / 1–3 years / 4–6 years / 7–10 years / > 10 years)
*8. dual incomes (if married) (not married / yes / no)
9. persons in household
10. fraction in household under 18
*11. household status (own / rent / live with family)
*12. type of home (house / condo / apartment / mobile home / other)
*13. ethnic classification (american indian / asian / black / east indian / hispanic / pacific islander / white / other)
*14. language spoken in home (english / spanish / other)

* = categorical

Applying MARS, restricting the maximum number of interacting variables to one, two, and three, respectively yielded cross-validated R^2_{CV} of .44, .45, and .43. Since these estimated future prediction errors are all quite similar, the additive (no interaction) model is chosen for interpretation. Table 5 gives the ANOVA decomposition for this model. It is seen to involve six of the 13 predictor variables; two of them are ordinal (age and education) and the other four are categorical. Since there are no interactions the ANOVA and categorical-ordinal decompositions are equivalent. Figure 3 gives the respective contributions of age and education to the model for household income. Not surprisingly, income grows monotonically with both (accounting for the contributions of the other variables), linearly with education and with a decreasing slope as age increases. Table 6 gives the functions of the categorical variables entering into the additive MARS model. (Note that all functions are translated to have zero minimum value.) Even though the sample size is fairly large, a substantial amount of smoothing has been applied to these estimates, as well as to those for the ordinal variables (Figure 3). This is likely due to the high noise level.

Table 5

ANOVA Decomposition of the MARS Model
for the Sample Survey Data (Section 4.3)

R^2_{GCV}(full model) = 0.414

ANOVA function	standard deviation	$\backslash R^2_{GCV}$	# basis functions	variable
1	.36	.409	1	age
2	.48	.402	1	householder status
3	.68	.376	2	occupation
4	.61	.381	1	marital status
5	.42	.403	1	education
6	.30	.409	1	type of home

Table 6

Additive Contributions of the Categorical Variables Selected by the MARS Model on Househould Income (Section 4.3)

Householder status:	$f_{11}(x_{11})$
own	0.99
rent	0.0
live with family	0.0
Occupation:	$f_5(x_5)$
professional – managerial	1.73
sales worker	1.05
military	1.05
factory worker, laborer, driver	0.68
clerical, service worker	0.68
homemaker	0.68
student	0.0
retired	0.0
unemployed	0.0
Marital status:	$f_2(x_2)$
married	1.22
living together – not married	1.22
divorced or separated	0.0
widowed	0.0
single – never married	0.0
Type of home:	$f_{12}(x_{12})$
house	0.68
condominium	0.68
other	0.68
apartment	0.0
mobile home	0.0

The MARS model for household income based on these data contains few surprises. The dependence on marital status probably reflects the fact that the response variable is household rather than individual income. The association with type of home and householder status is likely a reversal of cause and effect in that income probably dictates these variables rather than vice versa. One possible surprise is that military income appears relatively high after accounting for other variables that entered the model. Another possible surprise is that certain variables (sex, ethnic classification) do not contribute to household income with sufficient strength to gain entry into the model (again after accounting for those that do enter).

4.4. Social Grooming Habits of River Otters. These data are taken from Table 54.1 of Andrews and Herzberg (1985). They consist of $N = 394$ observations on the grooming habits of North American river otters. Each observation involves watching a specific pair of animals for a period of time. There are 14 different otters arranged in five groups. All animals within a group were observed simultaneously. The (response) variable of interest is the frequency of grooming. Table 7 lists the five predictor variables for this study. Four are categorical and one (time observed) is ordinal.

Applying MARS to these data, restricting the maximum number of interacting variables to be 1, 2, 3, 4, and 5 (no restriction), respectively produced (20-fold) cross-validated R^2_{CV} of 0.42, 0.40, 0.51, 0.48, 0.48. Thus, MARS modeling favors an approximation involving three-variable interactions. Table 8 shows the ANOVA decomposition of this model. The relative importance of a variable (Table 8) is defined as the square root of the GCV (12) of the model with all basis functions involving that variable removed, minus the square root of the GCV score of the corresponding full model, scaled so that the relative importance of the most important variable (using this definition) has a value of 100. Table 8 tends to confirm that three-variable interactions are important to the model, and indicates that the sex of the recipient is substantially less influential than the other variables in predicting grooming frequency.

Table 7
Predictor Variables for North American Otter Data
(Section 4.4) with Their Possible Values

variable	values
1. group	A/B/C/D/H
2. season	breeding/nonbreeding
3. time observed	minutes
4. sex of groomer	female/male
5. sex of recipient	female/male

Table 8

ANOVA Decomposition of the MARS Model
for River Otter Data (Section 4.4)

R^2_{GCV}(full model) = 0.52

ANOVA function	standard deviation	$\backslash R^2_{GCV}$	# of basis functions	variables
1	1.58	0.48	2	group
2	0.74	0.50	1	time
3	1.95	0.43	1	time, groomer
4	1.59	0.48	1	group, recipient
5	2.32	0.42	2	group, season, time
6	1.24	0.48	1	season, time, groomer

Relative variable importance:

group	season	time	groomer	recipient
100	67	88	58	26

Table 9 and Figure 4 give the categorical-ordinal decomposition (37) of this model. Table 9 shows the pure categorical contribution $f_c(\mathbf{x}_c)$ which is an interaction between the group and sex of the groomer. Since there is only one ordinal variable (time observed), all of the other contributions to the categorical-ordinal decompostion $f_o(\mathbf{x}_o)$, $\{f_\ell(\mathbf{x}_o)\}$ (37) are functions of that single variable. Figure 4 shows these contributions. There is a nearly negligible pure ordinal contribution $f_o(\mathbf{x}_o)$ (upper left frame) and there are four categorical-ordinal interactions $\{f_\ell(\mathbf{x}_o)\}_1^4$ of various strengths. The nearly linear dependence of all of these could be more readily understood if the response variable were in fact the total number of grooming incidents observed, rather than grooming frequency (as reported). One would expect this number to increase linearly with observation time if the length of each grooming incident tended not to depend on observation time. The slope of the linear dependence would then be an estimate of the grooming rate. As indicated in Figure 4, this rate has a fairly strong dependence on the other (categorical) variables. The most marked increase in grooming rate is for otters in group D during the breeding season (middle left frame), irrespective of the sex of the groomer or recipient. The otters in group D are adult siblings; whereas none of the other groups contain similarly related animals.

Table 9

Pure Categorical Contribution to the Categorical-Ordinal Decomposition of the MARS Model for the River Otter Data (Section 4.4)

	groomer	
group	female	male
A	0.0	0.0
B	4.5	4.5
C	0.0	0.0
D	15.0	5.9
H	0.0	0.0

4.5. Auto Insurance in Sweden. The data for this example are from Andrews and Herzberg (1985), Table 68.1. It consists of data concerning Swedish third party motor insurance for 1977 presented as a large four-way contingency table. The predictor variables are given in Table 10. Two of the variables (zone an make) are categorical, and the other two (travel and bonus) are taken to be ordinal. The response variable for this study is the number of claims per number insured ($\times 10^5$). Each cell of the contingency table was weighed in proportion to the number of insured, thereby taking each policy as an observational unit; the response can be interpreted as the probability of a claim per policy ($\times 10^5$).

Table 10

Predictor Variables for the Swedish Auto Insurance Data (Section 4.5) with Their Possible Values

1. Kilometers traveled per year (1: < 1000 / 2: 1000–15000 / 3: 15000–20000 / 4: 20000–25000 / 5: > 25000)
2. Geographical zone (1: Stockholm, Göteborg, Malmo / 2: other bigger cities / 3: smaller cities in south / 4: rural areas in south / 5: smaller cities in north / 6: rural areas in north / 7: Gotland)
3. Claims bonus (1–7: number of years since last claim or start of policy, plus one)
4. Make of auto (1–8: different specified car models)

Since the claims bonus (x_3) reflects frequency of previous claims, it should indirectly capture all risk factors (at least in principal). It is of interest to see to what extent (if any) the other three observed variables can be used to increase prediction accuracy. Figure 5 shows a MARS regression of the response on claims bonus. The cross-validated $R^2_{CV} = 0.41$ for this model indicates moderate predictability based on this variable alone. The estimated response dependence is seen to monotonically decrease with increasing claims bonus, with the highest slopes at the extremes. Applying MARS to these data incorporating all four predictor variables (Table 10) yielded a model with $R^2_{CV} = 0.77$ that involved three-variable interactions. Restricting the model to two-variable interactions also gave an $R^2_{CV} = 0.77$, whereas further restricting it to be additive (no interactions) resulted in a model with $R^2_{CV} = 0.65$. Thus, including the other variables (with claims bonus) seems to substantially improve predictive performance.

Table 11 presents the ANOVA decomposition of the two-variable interaction model. The last ANOVA function (interaction between zone and make) is seen to make at best a very minor contribution to the model. Its standard deviation is much smaller than the others, and its removal imperceptibly (to two significant digits) decreases R^2_{GCV}. Rerunning MARS prohibiting interactions between zone and make gives a cross-validated $R^2_{CV} = 0.77$ indicating further the lack of importance of this ANOVA function. Claims bonus is seen to be the most important predictive variable in the model with zone and make making substantial contributions. Distance traveled seems to have somewhat less importance.

Table 11

ANOVA Decomposition of the MARS Model for the Swedish Car Insurance Data (Section 4.5)

R^2_{GCV}(full model) = 0.78

ANOVA function	std. dev. ($\times 10^{-2}$)	$\backslash R^2_{GCV}$	# of basis functions	variables
1	3.5	0.57	3	bonus
2	1.9	0.72	2	make
3	1.6	0.74	2	zone
4	0.54	0.75	1	travel
5	1.6	0.68	5	bonus, make
6	0.90	0.75	3	zone, bonus
7	0.28	0.78	1	zone, make

Relative variable importance:

travel	zone	bonus	make
21	40	100	57

After removing the last ANOVA function, the resulting model's interaction effects all involve claims bonus. Furthermore, for fixed values of this variable the resulting model on the other variables is seen from the ANOVA decomposition (Table 11), to be additive. This suggests that choosing this variable for slicing (Section 2.4) would give rise to interpretable additive (sliced) models.

The contribution of distance traveled to the MARS model is seen to be additive (Table 11) so that it does not interact with claims bonus. This makes its contribution the same irrespective of the value of the claims bonus variable. Figure 6 shows this contribution. The response estimate is seen to have a (weak) linear dependence on travel. Note however that this variable is a (discretized) nonlinear function of the actual kilometers traveled (Table 10).

Figure 7 shows a graphical representation of the (additive) sliced model on (categorical) make and zone for three different values (slices) of claims bonus. These represent the smallest, middle, and largest values of this variable. One sees a marked dependence on the two categorical variables especially for the smallest claims bonus value. Some automobile makes are associated with a much higher claims risk than others, and some geographical zones seem more dangerous than others. As the claims bonus increases the overall predictive importance of make decreases but the relative contributions of each of its values stays roughly the same. Its magnitude is changing, but not its "shape." The contribution of zone to the model as claims bonus increases exhibits a somewhat different behavior. The claims risk associated with most geographical zones stays the same (in absolute value) as the claims bonus changes. A dramatic exception is zone 1 and to a lesser extent zones 4 and 6.

The diminishing relative predictive value of both the make and zone variables as claims bonus increases might have to do with the inherent nature of the latter variable. A large claims bonus indicates a good driving record for a long time, at least in terms of insurance claims, thereby providing a reliable forecast of (good) future behavior. On the other hand, a small claims bonus indicates either a recent claim (possibly indicating bad future behavior) or a new policy (no information at all). It may be that the relative lack of information provided by this variable when its value is small leaves more variance for the other variables to explain there, giving them the potential to be relatively more helpful. As claims bonus increases in value, it indirectly captures the contributions of the other variables to policy risk, causing them to be less needed.

Further support for this interpretation is presented in Figure 8. The upper frame shows the average squared residual (ASR) from the regression of policy risk on claims bonus alone (Figure 5), as a function of claims bonus. The lack-of-fit is seen to decrease monotonically for increasing values of claims bonus. The ASR for the lowest bonus value is five times larger than at the highest bonus value. Thus, claims bonus alone is an increasingly good predictor of (lower) policy risk as the bonus value increases. The lower frame of Figure 8 shows the average squared residual from the full MARS model (Table 11) on all of the variables (Table 10), again as a function of claims bonus. Although the ASR is monotonically decreasing here as well, the effect is far less dramatic. The gain in prediction accuracy achieved by the full model, over that of one based on claims bonus alone, is clearly (much) larger for smaller values of claims bonus.

5.0. Discussion. The examples in the previous sections illustrate the need for being able to do regression modeling in situations involving both ordinal and categorical predictor variables and the ability of MARS to accomplish it. The second example (Section 4.2) indicates that MARS can be successfully applied in situations involving (possibly many) missing predictor values. The analyses indicate that the MARS approach may have the potential to serve as a useful adjunct to other commonly used methods. It may also prove to be competitive for the analysis of large sparse contingency tables where all of the variables are categorical. The principal features of this approach are strictly continuous approximations with respect to the ordinal variables, and the ability to smooth simultaneously both by clustering and projection on the categorical variables. This should help improve accuracy over existing methods in some situations. In addition, interpretational tools like the ANOVA and categorical-ordinal decompositions, as well as slicing, provide some ability to probe the (multivariate) nature of the derived approximation (function estimate), thereby providing some insight into the predictive relationship.

The amount of (correct) insight that is gained depends on the power of the interpretational tools to probe the approximation and the degree to which the approximation reflects the properties under study of the target function. This latter concern is one of statistical inference. The MARS procedure described here is a fairly complex highly nonlinear method. As such, it is unlikely that inferential tools based on linear fitting can serve even as useful approximations. Sample reuse methods (so far) provide the best hope for gauging the reliability of inferences concerning the target function based on the derived approximation. Cross-validation (Stone, 1974) can provide an unbiased estimate of prediction accuracy and bootstrapping (Efron and Tibshirani, 1986) can be used to judge the stability (under sampling fluctuations) of any aspect of the model. The computational properties of the MARS procedure permit it to be used in conjunction with these sample reuse methods, except perhaps for very large problems on small computers.

A Fortran program implementing the MARS procedure is available from the author.

Acknowledgement. The sample survey data used in Section 4.3 is part of a data base provided by Impact Resources, Inc., Columbus, OH. The author is grateful for its use.

References

ANDREWS, D. F. and HERZBERG, A. M. (1985). *Data. A Collection of Problems from Many Fields for the Student and Research Worker.* Springer-Verlag, New York.

BELLMAN, R. E. (1961). *Adaptive Control Processes.* Princeton University Press, Princeton, NJ.

BREIMAN, L. (1989). Submodel selection and evaluation in regression I. The x-fixed case and little bootstrap. Department of Statistics, University of California, Berkeley, Technical Report No. 169.

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

BREIMAN, L. AND PETERS, S. (1988). Comparing automatic bivariate smoothers (A public service enterprise). Department of Statistics, University of California, Berkeley, Technical Report No. 161.

CRAVEN, P. and WAHBA, G. (1979). Smoothing noisy data with spline functions. Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numer. Math.,* **31**, 317–403.

DE BOOR, C. (1978). *A Practical Guide to Splines.* Springer-Verlag, New York, NY.

EFRON, B. (1983). Estimating the error rate of a prediction rule: improvement on cross-validation. *J. Amer. Statist. Assn.* **78**, 316–331.

EFRON, B. and TIBSHIRANI, R. (1986). Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science,* **1**, 54–77.

EUBANK, R. L. (1988). *Spline Smoothing and Nonparametric Regression.* Marcel Dekker, New York.

FRIEDMAN, J. H. (1991). Multivariate adaptive regression splines (with discussion). *Ann. Statist.,* **19**, 1–141.

FRIEDMAN, J. H. and SILVERMAN, B. W. (1989). Flexible parsimonious smoothing and additive modeling (with discussion). *Technometrics* **31**, 3–39.

Gu, C. and Wahba, G. (1991). Discussion of Friedman, *Ann. Statist.,* **19**, 115–123.

James. W. and Stein, C. (1961). Estimation with quadratic loss. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1. Berkeley: University of California Press, 361–379.

Kass, G. V. (1980). An exploratory technique for investigating large quantities of categorical data. *Appl. Statist.* **29**, 119–127.

Hocking, R. R. (1977). Selection of the best subset of regression variables. In *Statistical Methods for Digital Computers*, K. Enslein, A. Ralston and H. Wilf, eds. Wiley-Interscience, New York, 39–57.

Morgan, J. N. and Sonquist, J. A. (1963). Problems in the analysis of survey data, and a proposal. *J. Amer. Statist. Assn.* **58**, 415–434.

Parzen, E. (1962). On estimation of a probability density function and mode. *Ann. Math. Statist.,* **33**, 1065–1076.

Shumaker, L. L. (1976). Fitting surfaces to scattered data. In *Approximation Theory III*, G. G. Lorentz, C. K. Chui, and L. L. Shumaker, eds. Academic Press, New York, 203–268.

Shumaker, L. L. (1984). On spaces of piecewise polynomials in two variables. In *Approximation Theory and Spline Functions*, S. P. Singh et al. (eds.). D. Reidel Publishing Co., 151–197.

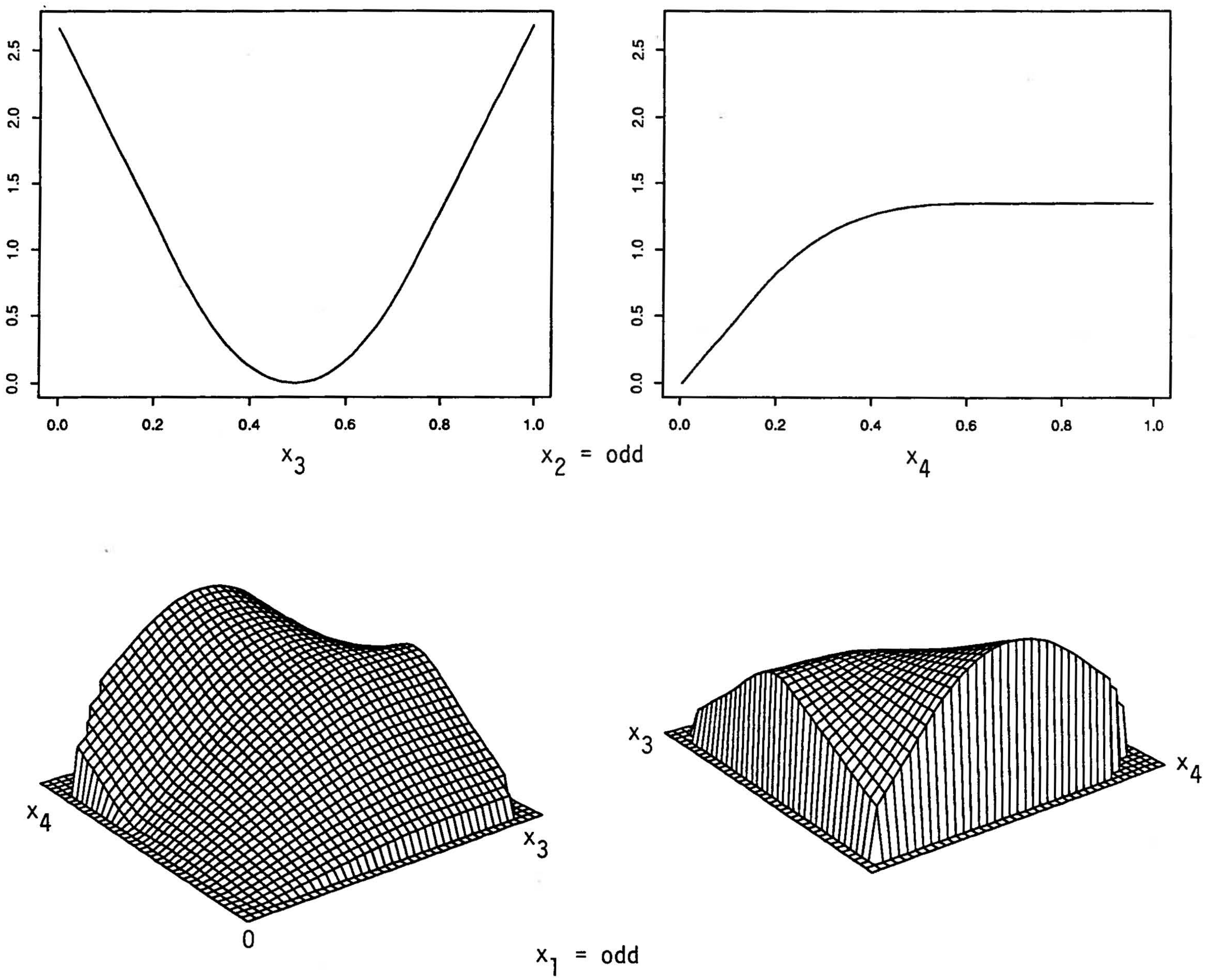
Smith, P. L. (1982). Curve fitting and modeling with splines using statistical variable selection techniques. NASA, Langley Research Center, Hampton, VA, Report NASA 166034.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictors (with discussion). *J. R. Statist. Soc.,* **B36**, 111–147.

Wahba, G. (1990). *Spline Models for Observational Data.* Monograph: SIAM, CBMS–NSF Regional Conference Series in Applied Mathematics, Vol. 59.

Figure Captions

Figure 1: Categorical-ordinal decomposition of the MARS model for the artificial data example of Section 4.1 using $N = 200$ observations.
Figure 2: Categorical-ordinal decomposition of the MARS model for the artificial data example of Section 4.1 using $N = 400$ observations.
Figure 3: Additive contributions of age and education to the MARS model for household income, using the sample survey data of Section 4.3.
Figure 4: Categorical-ordinal interactions of the MARS model for the grooming frequency of North American river otters, Section 4.4.
Figure 5: MARS estimate of the dependence of policy risk on claims bonus alone, for the Swedish auto insurance data, Section 4.5.
Figure 6: Additive contribution of distance traveled to the MARS model for the Swedish auto insurance data of Section 4.5.
Figure 7: MARS model for Swedish auto insurance data, Section 4.5, along three slices on claims bonus. Left/right frames are the respective contributions of make and zone. The top/middle/bottom frames are for slices on claims bonus = 1, 4, and 7 respectively.
Figure 8: Average squared residual as a function of claims bonus for two MARS models on the Swedish auto insurance data, Section 4.5. Upper frame is for a model based on claims bonus alone, whereas the bottom frame is for the MARS regression on all of the variables.



x_2 = odd

x_1 = odd

FIGURE 1

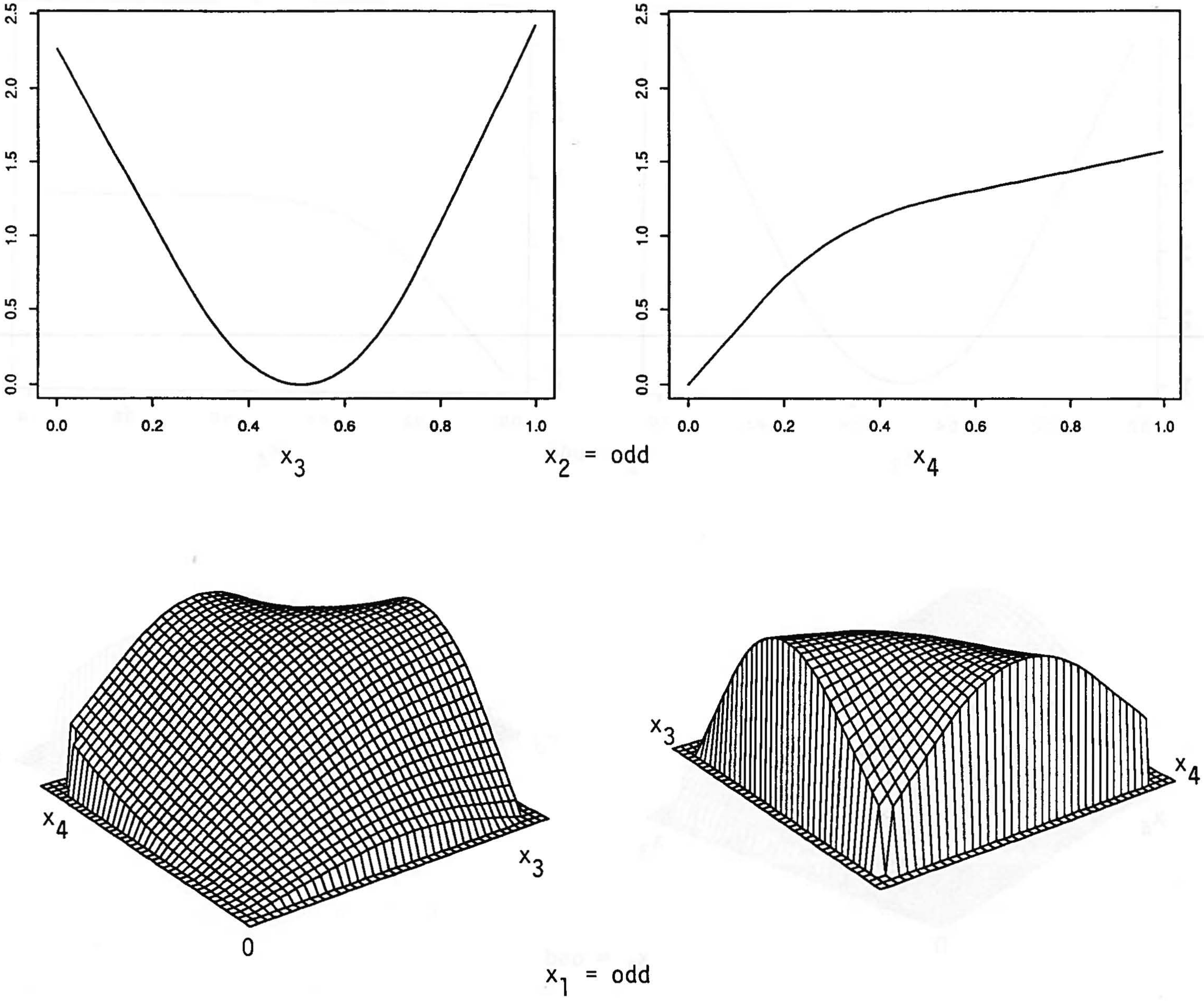
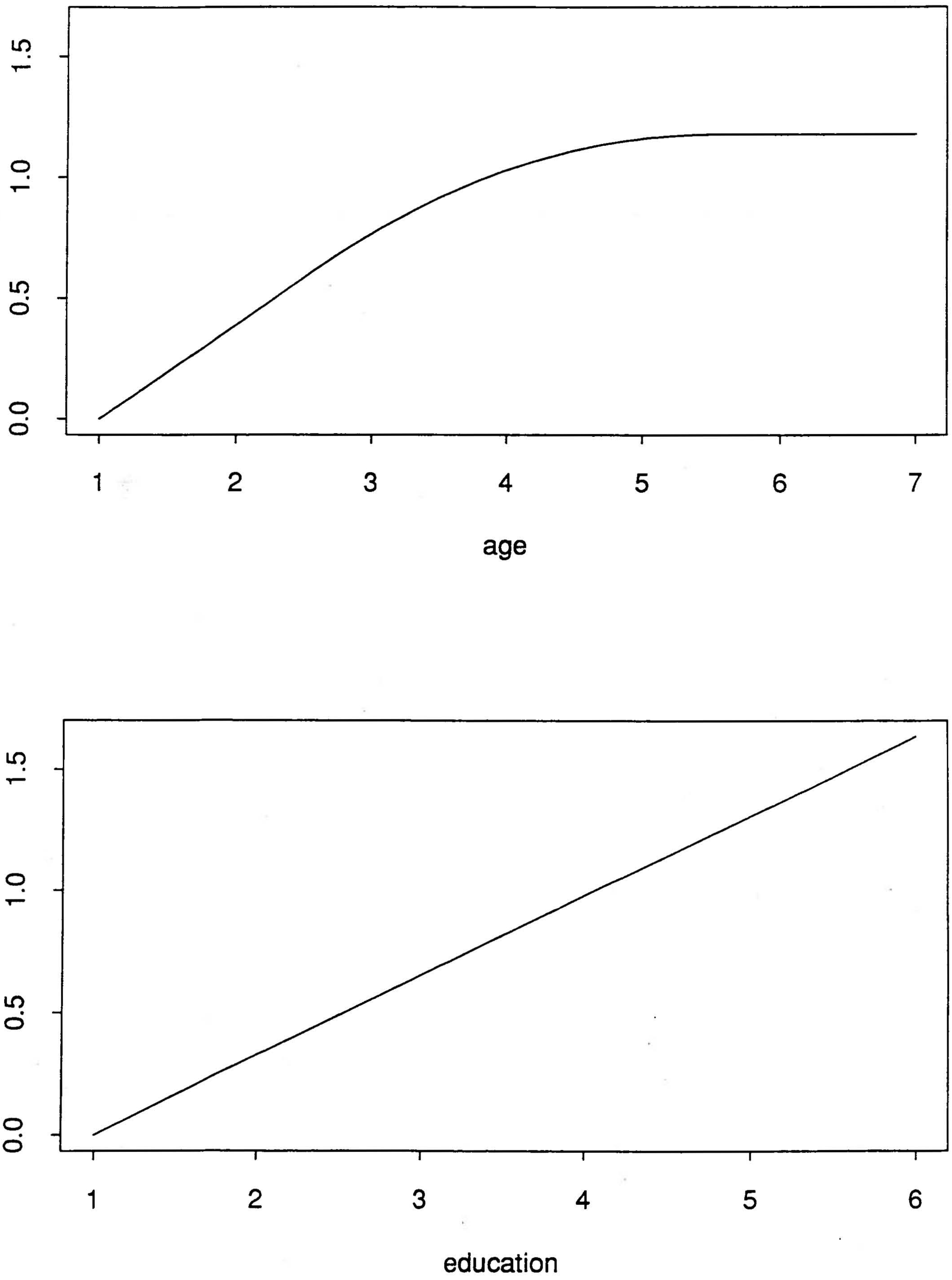


FIGURE 2

FIGURE 3



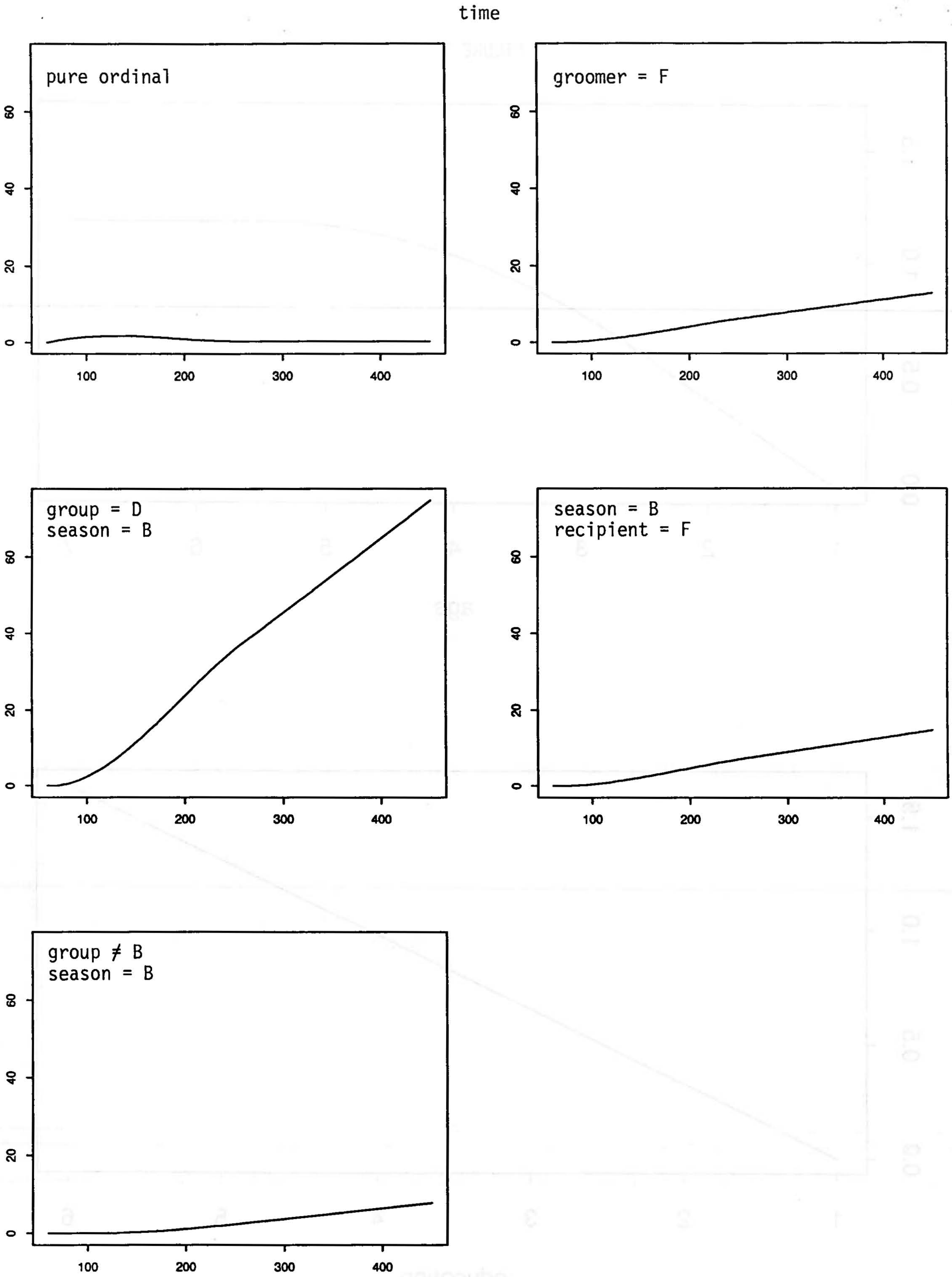

time
pure ordinal
groomer = F
group = D
season = B
season = B
recipient = F
group ≠ B
season = B
0
20
40
60
100
200
300
400

FIGURE 4

FIGURE 5

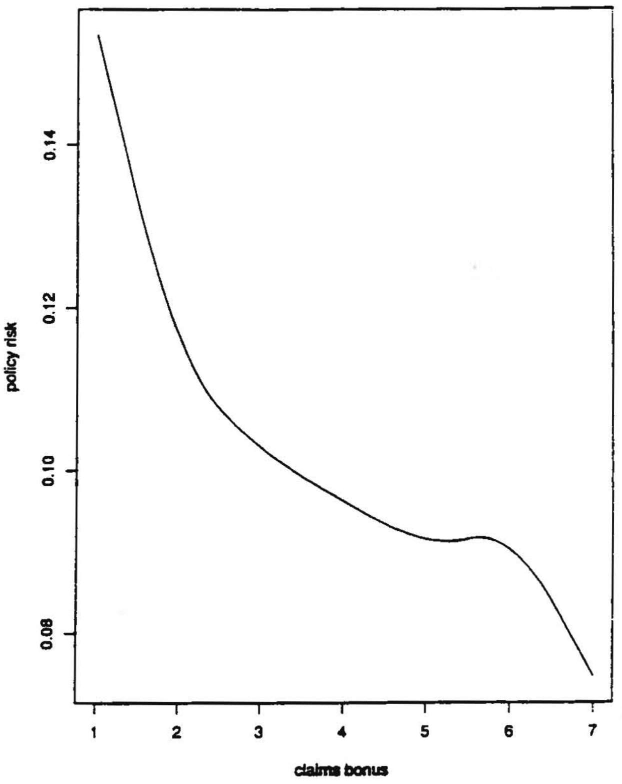


FIGURE 6

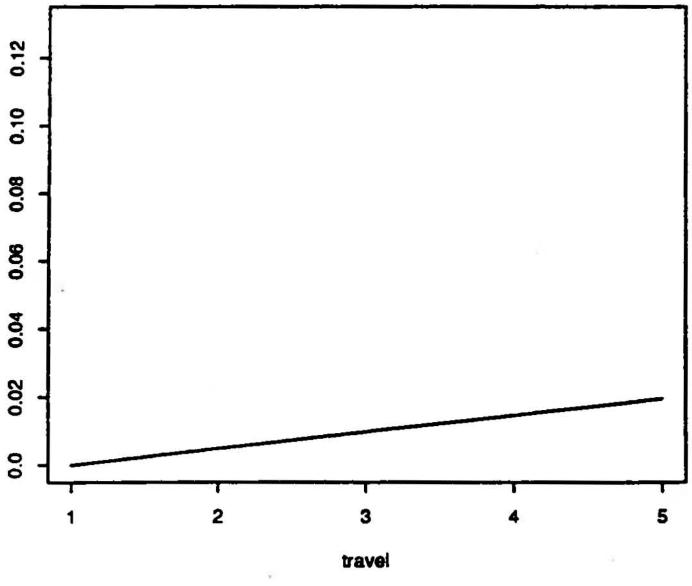
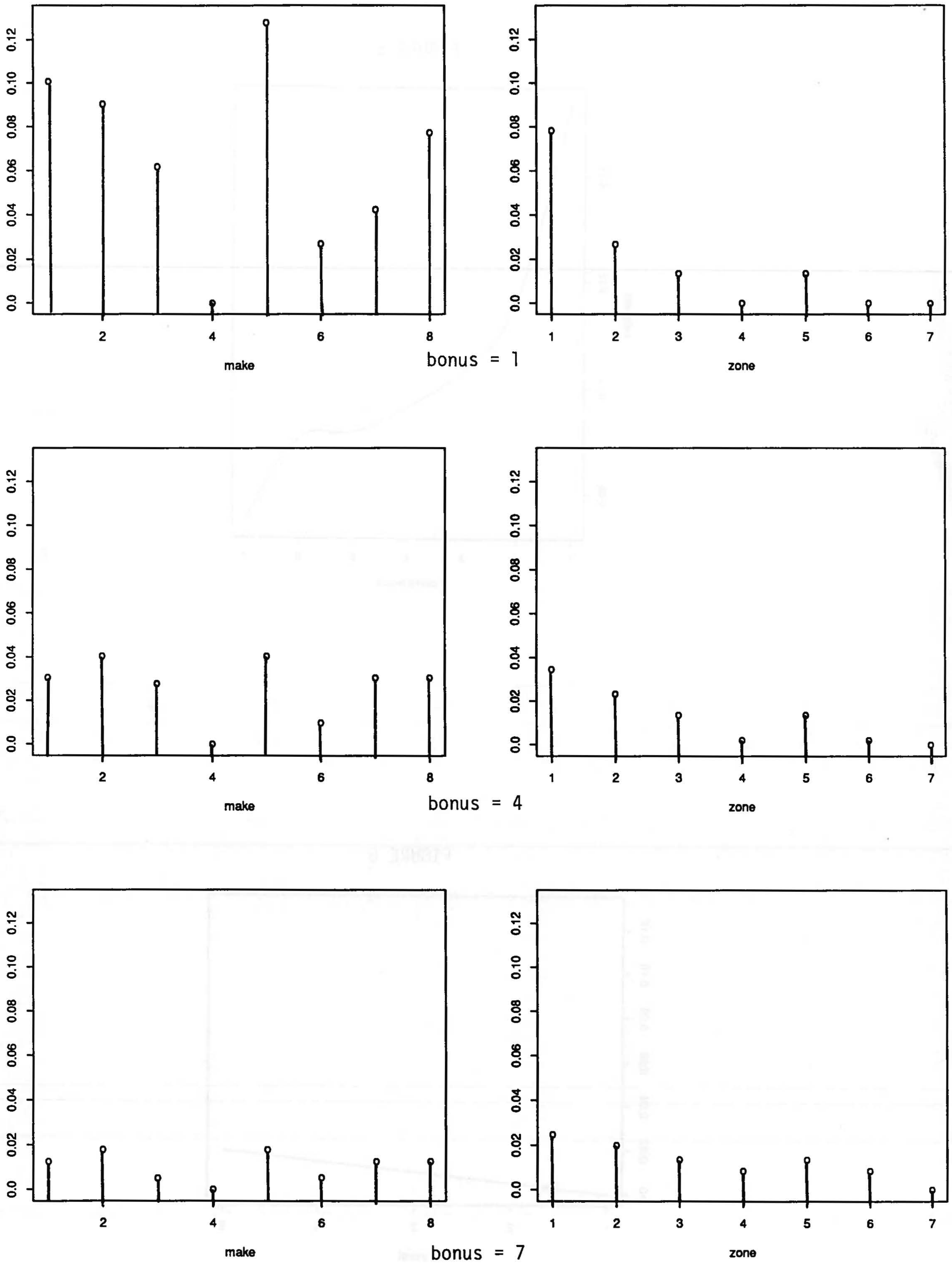


FIGURE 7

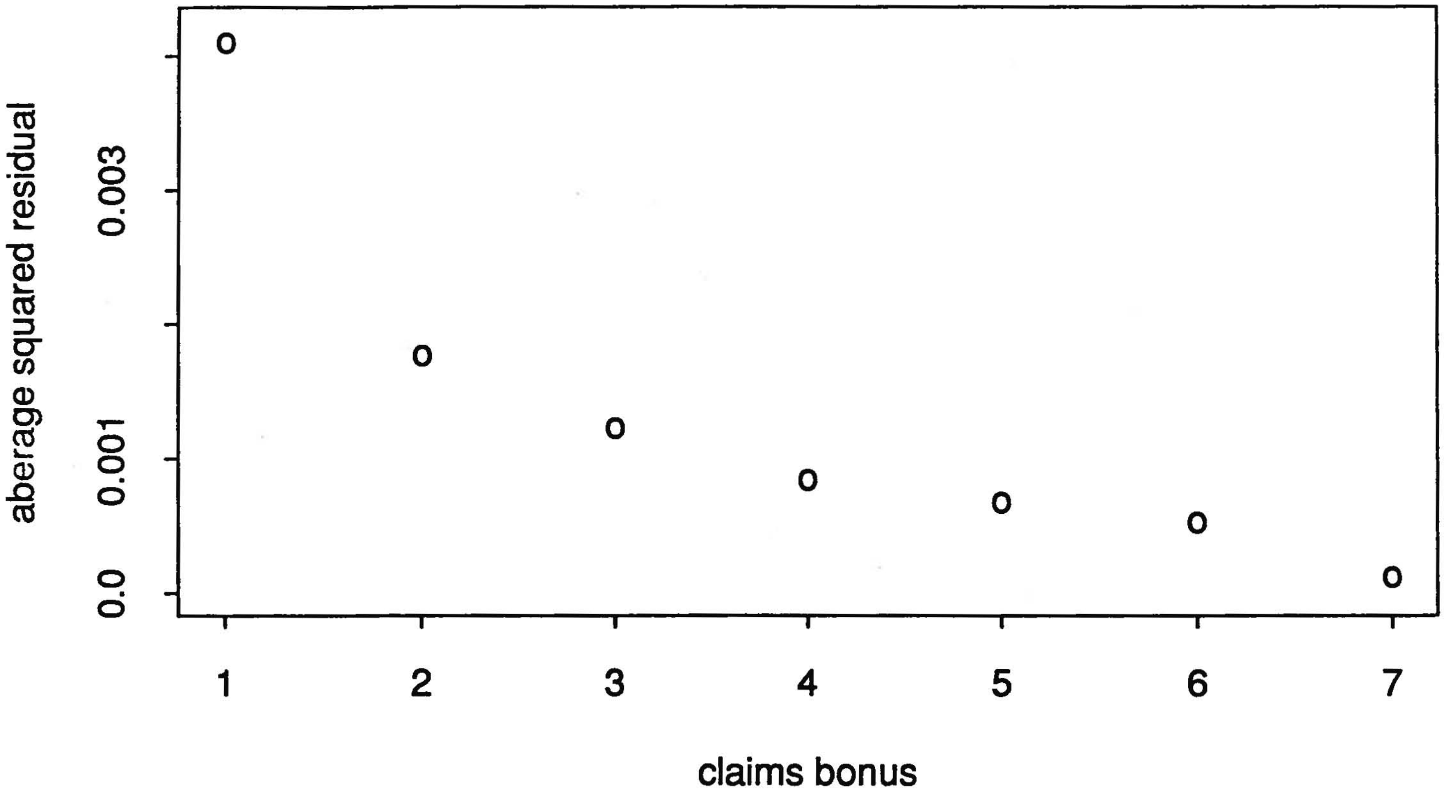


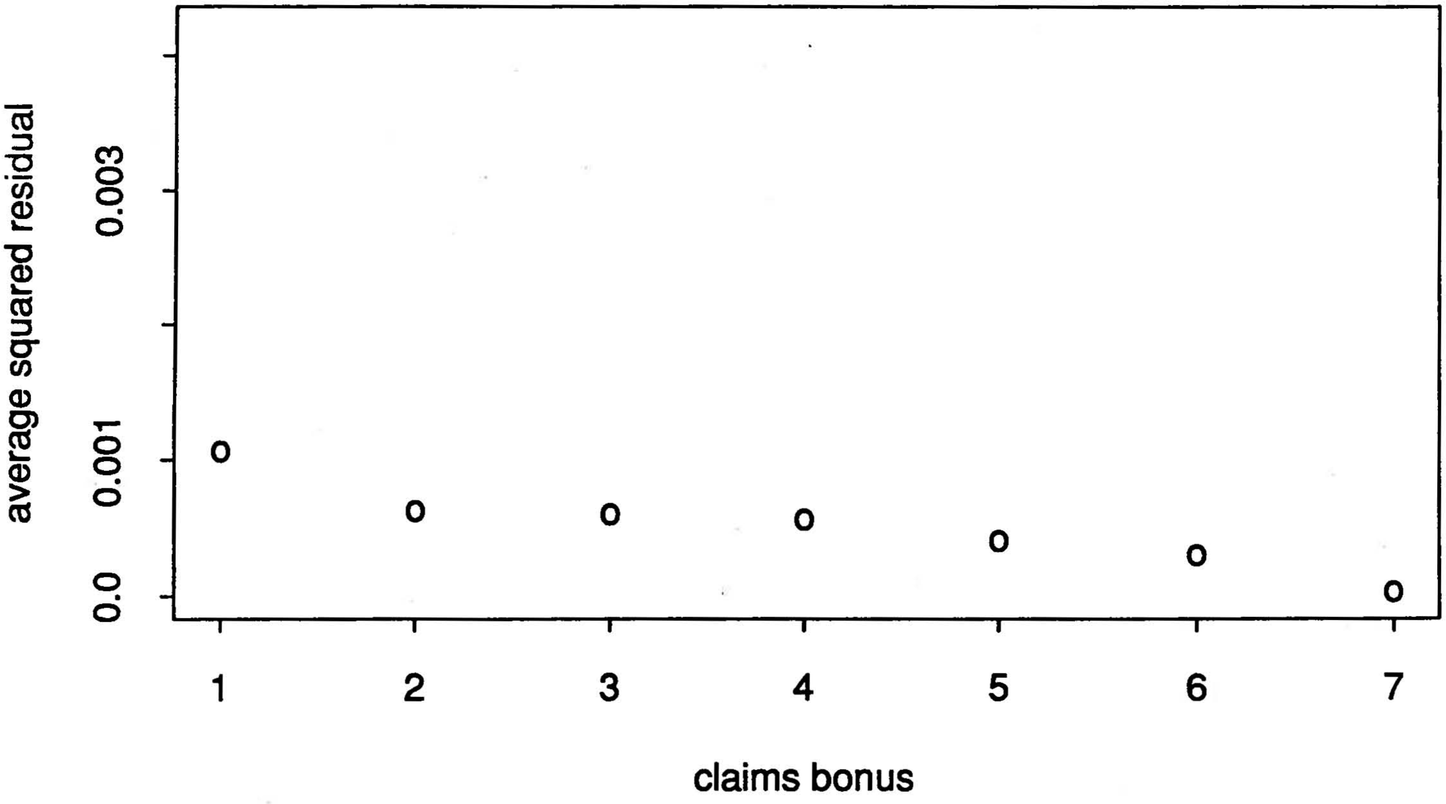
bonus = 1

bonus = 4

bonus = 7

FIGURE 8


aberage squared residual
0.0
0.001
0.003
1
2
3
4
5
6
7
claims bonus


average squared residual
0.0
0.001
0.003
1
2
3
4
5
6
7
claims bonus