A Framework for Estimation and Inference in Generalized Additive Models with Shape and Order Restrictions

Mary C. Meyer
Colorado State University

Abstract. Methodology for the partial linear generalized additive model is presented, where components for continuous predictors may be modeled with shape-constrained regression splines, and components for ordinal predictors may have partial orderings. The estimated mean function is obtained through a projection (or iteratively re-weighted projections) onto a polyhedral convex cone; this is key for formally derived inference procedures. Point-wise confidence bands and hypothesis tests for the individual components, as well as a model selection method, are proposed. These methods are available in the R package cgam.

Key words and phrases: monotone, convex, partial linear, confidence interval.

1. BACKGROUND

Estimation and inference under shape and order constraints dates back to Hildreth (1954), who formulated maximum likelihood estimators for convex functions. Contributions to estimation and inference involving complete and partial orderings were made by Brunk (1955), Ayer et al. (1955), VanEeden (1956), and others. These early estimators were not smooth; the isotonic regression estimator is piece-wise constant, and the convex regression estimator is piece-wise linear. Isotonic smoothing spline estimators were proposed by Tantiyaswasdikul and Woodroofe (1994), and various ideas for shape-constrained kernel regression estimators were given by Mammen (1991), Hall and Huang (2001), and Du et al. (2013).

Regression splines are attractive from an inferential perspective because the shape-constrained estimator can be expressed as a mixture of linear estimators. Monotone regression spline estimators were introduced by Ramsay (1988), who defined the $I$-splines, shown in Figure 1 in the left-hand plot. These quadratic spline basis functions have the property that for each knot (marked by the dotted vertical lines), exactly one basis function has non-zero slope. Hence, non-negative coefficients comprise necessary and sufficient conditions for linear combinations of these basis functions to be non-decreasing. Any non-decreasing quadratic spline
function can be expressed as a linear combination of the \textit{I}-spline basis functions with non-negative coefficients, plus an unconstrained intercept. Meyer (2008) defined the \textit{C}-splines, shown in the middle panel of Figure 1. These cubic spline basis functions are constructed so that at each knot, exactly one has non-zero second derivative. Any convex cubic spline function can be expressed as a linear combination of the \textit{C}-spline basis functions with non-negative coefficients, plus an unconstrained linear function. Functions with combinations of monotonicity and convexity assumptions can be modeled with the \textit{C}-splines as well. Meyer et al. (2018) derived convergence rates for the constrained splines, under mild conditions, if the number of knots increases on the order of \(n^{1/7}\) for the quadratic splines and on the order of \(n^{1/9}\) for the cubic splines.

The third plot in Figure 1 shows increasing and concave least-squares fits to a scatterplot of height versus diameter of white spruce trees, available as the \texttt{whitespruce} data set in the \texttt{R} package \texttt{ConSpline}. Constrained scatterplot smoothers are more robust to tuning parameters, compared to unconstrained smoothers, because of the shape restriction, increasing the number of knots does not necessarily lead to severe over-fitting. The constraints prevent the wiggling typically associated with over-fitting; hence, the increasing and concave fits to the white spruce data using 4-8 knots are almost identical.

With constrained regression splines, the least-squares estimator is found by projection onto a polyhedral convex cone, and for the generalized regression, through iteratively re-weighted cone projections. A set \(C\) in \(\mathbb{R}^n\) is a cone if for any \(\eta \in C\), positive multiples of \(\eta\) are also in \(C\). The cone is polyhedral if it is finitely generated, i.e., there is a finite set of vectors in the cone so that any vector in the cone can be expressed as a linear combination of these vectors, with linearly constrained coefficients.

The projection onto a polyhedral cone lands on a face of the cone, and coincides with the projection onto a linear space defined by this face. The projection of another realization of the response might land on a different face, so coincides with a projection onto another linear space. Thus the constrained estimator is a mixture of ordinary linear estimators, and inference methods use this important property. Meyer (2008) provided a test of constant versus increasing regression function, where the alternative is estimated with \textit{I}-splines, and a test for linear versus convex regression function, where the alternative is estimated with \textit{C}-splines. These have, under the normal errors assumption, a likelihood ratio test.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{Basis functions for constrained splines, and example fits to a data set of 36 observations of height and diameter of white spruce trees to demonstrate robustness to knot choices.}
\end{figure}
where the null distribution is that of a mixture of beta random variables. The partial linear constrained spline least-squares model was developed by Meyer (2017), who derived inference methods for the linear term that also take advantage of the polyhedral cone formulation.


In this paper we extend these ideas and give a framework for estimation and inference for the generalized additive partial linear model, where the components for continuous predictors are modeled with splines, and the components for ordinal predictors are modeled with partial orderings. The focus is on inference methods for the components. We consider generalized regression models for independent observations $Y_1, \ldots, Y_n$, where the probability distribution can be expressed as

$$p(y_i; \eta_i, \tau) = \exp \left\{ \{y_i \eta_i - b(\eta_i)\} \tau - c(y_i, \tau) \right\}, \quad i = 1, \ldots, n.$$  

The family of distributions determine the functions $b$ and $c$, and the parameter $\tau$ is related to the dispersion. For example, if $Y_i$ is a binary response, we can define $b(\eta_i) = \log(1 + e^{\eta_i})$ to get the logistic model with $\mu_i = \mathbb{E}(Y_i) = e^{\eta_i} / (1 + e^{\eta_i})$. For the normal-errors model, $\eta_i$ is the expected value of $Y_i$, $b(\eta_i) = \eta_i^2 / 2$, $\tau = 1/\sigma^2$, where $\sigma^2$ is the model variance. In any case, $\mu_i \equiv \mathbb{E}(Y_i) = b'(\eta_i)$.

The vector $\eta = (\eta_1, \ldots, \eta_n)^\top$ is determined by the predictor values. We consider predictor functions of the form

$$\eta_i = f_1(t_{1i}) + \cdots + f_L(t_{Li}) + g_1(z_{1i}) + \cdots + g_R(z_{Ri}) + x_i^\top \beta$$  

where the predictors $t_\ell$ are treated as continuous and $f_\ell$ are functions to be estimated with shape-constrained splines, for $\ell = 1, \ldots, L$. The $z_r$ predictors are ordinal and a partial ordering will be imposed on the components $g_r$, $r = 1, \ldots, R$. The parametrically modeled predictors are included in the vector $x$.

In the next section, the $f_\ell$ and $g_r$ components of the model will be formulated in terms of convex cones. The component cones and linear spaces are then combined to form a single convex cone. For the least-squares model, the estimate $\hat{\eta}$ is simply the projection onto the single cone, and the component estimates are readily determined. For the generalized regression models, $\hat{\eta}$ is estimated through iteratively re-weighted cone projections. Hypothesis tests concerning the individual components are given in Section 3. Point-wise confidence intervals for $\eta$ and the model components are given in Section 4 and shown through simulations to have good coverage and small lengths compared to other methods. A model selection method, where the shapes do not have to be specified a priori, is given in Section 5, and some discussion is provided in Section 6.
2. BUILDING THE MODEL CONE

2.1 Methodology

We consider three types of model components: constrained regression splines to model the effect of a continuous predictor, components for ordinal predictors with partial or complete orderings, and components for parametrically modeled covariates such as nominal or linear predictors. The estimate of each model component is found in a polyhedral convex cone or a linear space. We combine the individual cones and linear spaces into a large cone so that the maximum likelihood estimator $\hat{\eta}$ is found by projections onto a single cone, after which we sort out the individual component estimates.

To estimate $f_\ell$ with constrained regression splines, we define knots $\xi_{\ell,1} < \cdots < \xi_{\ell,K_\ell}$ where the $t_{\ell,i}$ values fall in $[\xi_{\ell,1}, \xi_{\ell,K_\ell}]$, $i = 1, \ldots, n$. If $f_\ell$ is assumed to be “increasing” or “decreasing,” we define a set of $K_\ell$ quadratic I-spline basis functions with the property that at each knot, exactly one basis function has non-zero slope. We estimate $f_\ell$ as a linear combination of the basis functions plus an intercept; necessary and sufficient conditions for monotonicity are that the coefficients are non-negative. If the constraint is “convex” or “concave” we define a set of $K_\ell$ cubic C-spline basis functions with the property that at each knot, exactly one basis function has non-zero second derivative. We model $f_\ell$ as a linear combination of these basis functions with positive coefficients, plus an unconstrained multiple of $t_\ell$ as well as an intercept; these are necessary and sufficient conditions for convexity. If the constraints involve both monotonicity and convexity, such as “increasing and convex,” we use the $K_\ell$ cubic C-spline basis functions and include $t_\ell$ as a basis function with a constrained coefficient. For a more detailed treatment of constrained regression splines, see Meyer (2008).

For each $\ell = 1, \ldots, L$, let the spline basis functions be $\delta_{\ell,1}(t), \ldots, \delta_{\ell,m_\ell}(t)$, where $m_\ell = K_\ell$ for monotone or convex splines and $m_\ell = K_\ell + 1$ for combinations of monotonicity and convexity. Let $\Delta_\ell$ be an $n \times m_\ell$ matrix with $\Delta_{\ell,ji} = \delta_{\ell,j}(t_i)$. The cone over which the sum of squared residuals is to be minimized, or the likelihood to be maximized, is

$$C^{(1)}_\ell = \{ \eta \in \mathbb{R}^n : \eta = \Delta_\ell \alpha + X_0, \alpha_0, \text{ for } \alpha \geq 0 \},$$

where for constraints involving monotonicity, $X_{0,\ell} = 1 = (1, \ldots, 1)^T$, and for convex or concave constraints, $X_{0,\ell} = [1 \ t_\ell]$. (The intercepts when $L > 1$ will be combined later.)

For ordinal predictors, we assume that a partial ordering is known. For each $r = 1, \ldots, R$, define $\eta_r \in \mathbb{R}^n$ such that $\eta_{r,i} = g_r(z_{r,i})$; Meyer (2013) showed that the set of possible components for the ordinal predictor can be described as

$$C^{(2)}_r = \{ \eta \in \mathbb{R}^n : A_r \eta \geq 0 \text{ and } B_r \eta = 0 \},$$

where the inequality constraints using $A_r$ impose the partial ordering, and the equality constraints using $B_r$ ensure that $\eta_{r,i} = \eta_{r,j}$ whenever $z_{r,i} = z_{r,j}$. Let $V_r$ be the largest linear space contained in $C^{(2)}_r$.

For a “toy” example, suppose a treatment variable has three levels, one of which is a placebo, and there are three subjects assigned to each group, so that $n = 9$. Suppose subjects 1, 2, 3 are in the placebo group, subjects 4, 5, 6 are given
Treatment 1, and subjects 7, 8, 9 are given Treatment 2. The researchers want to assume that the two treatments have at least as great an effect as the placebo, without imposing an order on the two treatments (this is called a tree ordering). Here the constraint matrix $A_r$ has two rows:

$$A_r = \begin{pmatrix} -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix},$$

and $B_r$ ensures that the effect of this predictor is the same within each of the three groups:

$$B_r = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix}.$$

The linear space $V_r$ contained in the cone is the one-dimensional space of the constant vectors.

Define $D_r$ so that its columns are a basis for the linear space that is orthogonal both to $V_r$, and to the space spanned by the rows of $B_r$. Proposition 2.2 of Meyer (2013) determined that if $A_r$ is full row rank, then defining $\Gamma_r = D_r(A_rA_r^T)^{-1}$ we have

$$C_r^{(2)} = \{ \eta \in \mathbb{R}^n : \eta = v + \Gamma_r \gamma \text{ for } \gamma \geq 0, \ v \in V_r \}.$$

For the above tree ordering example, $\Gamma_r$ is a multiple of the matrix

$$\begin{pmatrix} -1 & -1 & -1 & -1 & -1 & 2 & 2 & 2 \\ -1 & -1 & -1 & 2 & 2 & 2 & -1 & -1 \end{pmatrix}^T,$$

and the effect of the categorical variable is a constant plus positive multiples of the columns of $\Gamma_r$.

Next we combine the component cones and linear spaces into a single cone. For the parametrically modeled covariates, let the $n \times p$ matrix $X$ have rows $x_i$, $i = 1 \ldots, n$. The component cones may contain same linear spaces, so let the columns matrix $X_c$ represent a basis for the Minkowski sum of all these $L + R$ linear spaces. For example the column space of $X_c$ contains the constant vectors and any $t_\ell$ where $f_\ell$ is modeled as convex or concave. We estimate $\eta$ to be in the large cone

$$\begin{align*}
C &= \{ \eta \in \mathbb{R}^n : \eta = \Delta_1 \alpha_1 + \cdots + \Delta_L \alpha_L + \Gamma_1 \gamma_1 + \cdots + \Gamma_R \gamma_R \\
&\quad + X_c \beta_c + X \beta, \text{ where } \alpha_\ell \geq 0, \ \ell = 1, \ldots, L, \\
&\quad \text{and } \gamma_r \geq 0, \ r = 1, \ldots, R \}. 
\end{align*}

Create an overall $n \times m$ “design” matrix for the constrained components

$$\Delta = [\Delta_1 | \cdots | \Delta_L | \Gamma_1 | \cdots | \Gamma_R]$$

as well as an $m \times 1$ constrained coefficient vector $\zeta^T = [\alpha_1^T, \ldots, \alpha_L^T, \gamma_1^T, \ldots, \gamma_R^T]$. We assume that the columns of $\Delta$, $X$, and $X_c$ form a linearly independent set;
otherwise, the component effects are not identifiable. Linear dependence in the columns results from highly or completely correlated predictors and/or poorly spaced knots. For example, if the knot spacing for \( t_\ell \) is such that there are no observed \( t_{i,j} \) between two knots, this may lead to linear dependence. This may be corrected by changing the knot spacing.

In Step 3, we compare the distance between \( \eta \) and \( \eta' \). If the distance is “small,” then the iteration ends; otherwise increment \( k \) and go to (2).

In Step 3, we compare the distance between \( \eta(k) \) and \( \eta'(k) \), rather than the distance between and \( \mu(k) \) and \( \mu'(k) \), because for models such as binomial and Poisson, the latter can be unbounded. For example, if the binomial responses corresponding to the last knot interval are all successes, then the maximum likelihood spline estimate for the log-odds might be infinite at the end-point. However, the
estimate for the mean tends to one. For the iteratively re-weighted algorithm, the cone projection routine is modified so that the first guess for $J$ at iteration $k$ is the $J$ from iteration $k-1$; this makes the routine almost as fast as for the unconstrained fit.

To demonstrate the method, we consider the data set pima in the R package MASS with the following description: “A population of women who were at least 21 years old, of Pima Indian heritage and living near Phoenix, Arizona, was tested for diabetes according to World Health Organization criteria. The data were collected by the US National Institute of Diabetes and Digestive and Kidney Diseases.” The response variable is whether or not the subject has diabetes, and three predictors are treated as continuous: skin-fold measurement, body mass index, and age. We also have the number of pregnancies, which is treated as ordinal. We assume that we know a priori that the probability of diabetes is increasing in each of the predictors.

Three of the estimated log odds components are shown in Figure 2. For the spline fits, the default knot choice in cgam is used, which for $n = 530$ is nine equally spaced knots. The estimated components deviate substantially from linearity; in particular, for age and BMI, the log-odds increase rapidly at the beginning of the range before leveling off. The skin-fold variable does not seem to have much effect on the response, given the effects of the other predictors. Of course, we would like to determine the statistical significance of each of the components, and determine confidence bounds for the probability estimates. These inference methods are addressed in the next two sections.

3. HYPOTHESIS TESTING FOR MODEL COMPONENTS

3.1 Tests concerning the parametrically modeled components

For inference about the parameters $\beta$, we start with the least-squares model. Suppose $y = \eta + \varepsilon$, where the components of $\eta$ are as in (1.1) and $\varepsilon$ is a mean-zero random vector with covariance matrix $\sigma^2 I$. Suppose that the projection of $y$ onto the cone $C$ falls on the face $F_J$. Meyer (2018) showed that, if $P_J$ represents the projection matrix for the space $L_J$ spanned by the columns of $\Delta_J$ and $X_c$, then

$$\hat{\beta} = (X^\top(I - P_J)X)^{-1}X^\top(I - P_J)y.$$
The same paper also gave conditions for root-$n$ convergence of $\hat{\beta}$, where Theorem 4 states that under mild assumptions,

$$\sqrt{n}(\hat{\beta} - \beta) = \sqrt{n}[X^\top(I - P_f)X]^{-1}X^\top(I - P_f)e + o_p(1).$$

Approximate $t$ and $F$ statistics are constructed from the estimated covariance matrix

$$\tilde{\text{cov}}(\hat{\beta}) = \hat{\sigma}^2[X^\top(I - P_f)X]^{-1},$$

where $\hat{\sigma}^2$ is the model variance estimate given by Meyer and Woodroofe (2000):

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(y_i - \hat{\mu}_i)^2}{n - d}$$

where $d = \min(1.5D, m)$ and $D$ is the dimension of $L_J$ for the observed $J$.


For weighted least-squares regression with positive-definite weight matrix $W$, we transform into the unweighted case by substituting $W^{1/2}y$ for $y$, $W^{1/2}\Delta$ for $\Delta$, and $W^{1/2}X$ for $X$. For the generalized regression model, the converged value of $\xi$ is used in place of the response $y$ in the weighted least-squares regression with the converged value of the weight vector.

Next we use simulations to compare these hypothesis tests for the linear terms in the binomial model, coded in the R package cgam (Liao and Meyer (2018)), with those of scam (Pya and Wood (2015)) and the routine gam in the mcv package. We use a continuous predictor $t$ with values equally spaced in $[0, 1]$, and a nominal covariate $x$ with three levels. The model is $η_i = f(t_i) + \beta_0 + \beta_1I\{x_i = 1\} + \beta_2I\{x_i = 2\}$, for $i = 1, \ldots, n$. We imagine that we know a priori that $f$ is smooth and increasing, so we model $f$ with constrained $I$-splines. The null hypothesis is that there is no effect of the covariate $x$ on the response (i.e., $H_0: \beta_1 = \beta_2 = 0$ versus $H_1$: at least one of $\beta_1$ or $\beta_2$ is not zero), and an approximate $F$ test for sub-models is used. To construct this test statistic, we use the final values $\xi$ and $W$ from the iteratively re-weighted least-squares procedure, and use the weighted sums of squared residuals from the the null ($SSE_0$) and alternative fits ($SSE_1$). Because we do not assume the model variance $\hat{\sigma}^2$ is known, the test statistic is

$$F = \frac{(SSE_0 - SSE_1)/2}{SSE_1/(n - df_1)},$$

where $df_1$ is the dimension of $L_J$ for the alternative hypothesis cone projection. Under $H_0$, this has approximately an $F(2, n - df_1)$ distribution.

For the simulated data sets, the values of $x$ are related to the values of $t$, so that $x = 1$ is more likely for larger values of $t$, and $x = 2$ is more likely for smaller values of $t$. Specifically, we generate independent uniform random numbers $u_1, \ldots, u_n$, and if $u_i \in (0, .05 + .85t_i^4)$ we set $x_i = 1$, if $u_i \in [.4 + .55t_i^{1/4}, 1]$, we set $x_i = 2$; otherwise $x_i = 3$. For the simulations, the true log odds is either linear: $f(x) = 4x - 2$, or $f(t) = -1 + 60(t - 1/2)^{1/4} + \beta_1I\{x = 1\}$, where $(\cdot)_+ = \max(0, \cdot)$.

An example data set with the non-linear log-odds, $\beta_1 = -1$, and $n = 200$ is shown in Figure 3, along with fits from the three methods. The constrained fits are similar; the difference in the hypothesis test results is due to the testing method rather than the fit. We generated 10,000 data sets for each of the two
functions, four sample sizes, and two values of $\beta_1$, to compare test size and power, which are shown in Table 1. For the scam test, we use both the default tuning parameters, and the tuning parameters that “match” those of cgam. That is, we set the penalty parameter to zero, and use the same number of knots as the cgam default. We find that for the linear log-odds, scam with the default choices outperforms the other methods. However, for the non-linear log-odds, the test size for scam is inflated if the default tuning parameters are used, especially for the smaller sample sizes, while the power for our method is consistently greater than that for gam. For the matching tuning parameters, the cgam and scam results are similar.

3.2 Testing $H_0 : \eta \in \mathcal{V}$ versus $H_1 : \eta \in \mathcal{C} \setminus \mathcal{V}$

Next, we turn to hypothesis tests concerning the constrained components. Let $\mathcal{V}$ be the linear space spanned by the columns of $\mathbf{X}_c$ and $\mathbf{X}$; this is the largest linear space contained in the cone $\mathcal{C}$. The traditional test of $H_0 : \eta \in \mathcal{V}$ versus $H_1 : \eta \in \mathcal{C} \setminus \mathcal{V}$, for the normal-errors model, was first presented by Bartholomew (1959), in terms of the one-way ANOVA model, where the null hypothesis corresponds to constant means, versus the completely ordered alternative. A more general treatment of these one-sided tests was given by Raubertas et al. (1986). Here, the null hypothesis is that none of constrained predictors is related to the response; i.e., $\alpha_1 = \cdots = \alpha_L = \gamma_1 = \cdots = \gamma_R = 0$, and the alternative is that at least one of the constrained predictors has a non-zero coefficient. Sen and Meyer (2017) formulated a “double-cone” test against a linear model, where the alternative
shape does not need to be specified.

We use the following notation. A random variable $T$ has a mixture of chi-squared distributions with mixing parameters $p \geq 0$, $\sum_{j=d_1}^{d_2} p_j = 1$, written $T \sim \bar{\chi}_{d_1 : d_2}^2(p)$, if for $c > 0$,

$$P(T \leq c) = \sum_{j=d_1}^{d_2} P(\chi_j^2 \leq c)p_j,$$

where $\chi_j^2$ is a chi-squared random variable with $j$ degrees of freedom. Proofs of the following result are available in Raubertas et al. (1986), Robertson et al. (1988), and Meyer (2003).

**Lemma 3.1.** Let $C$ be a polyhedral cone where the largest linear space contained in the cone has dimension $d_1$, and the smallest linear space containing the cone has dimension $d_2$. If $\varepsilon \sim N(0, \sigma^2 I)$, then

$$\frac{1}{\sigma^2} \|\Pi(\varepsilon|C)\|^2 \sim \bar{\chi}_{d_1 : d_2}^2(p)$$

where $p = (p_{d_1}, \ldots, p_{d_2})$ and $p_d$ is the probability that the projection of $\varepsilon$ onto $C$ falls on a face where the dimension of the corresponding linear space is $d$.

The intuition behind this lemma is as follows: The projection of $\varepsilon$ onto the cone lands on a face $F_J$, and coincides with the projection of $\varepsilon$ onto $L_J$. We know that $\|\Pi(\varepsilon|L_J)\|^2/\sigma^2 \sim \chi^2(\dim(L_J))$; because another realization of the response might have a projection that lands on a different face, we have a mixture of chi-squared random variables.

First we consider a linear space $V_0 \subseteq V$, and the test of $H_0 : \eta \in V_0$ versus $H_a : \eta \in C \setminus V_0$. We start by defining a matrix $X_0$ whose columns form a basis for $V_0$, a projection matrix $P_0$ for the linear space $V_0$, and a matrix $X_1$ whose columns form a basis for $V \setminus V_0^\perp$, where $V_0^\perp$ denotes the linear space orthogonal to $V_0$. Then define $E = (I - P_0)\Delta$ and the cone can be written as

$$C = \{\eta \in \mathbb{R}^n : \eta = E\alpha + X_0\beta_0 + X_1\beta_1, \text{ for } \alpha \geq 0\}.$$

The projection of $y$ onto $C$ (the alternative hypothesis fit) can be written as

$$\hat{\eta}_1 = E_J\hat{\alpha} + X_0\hat{\beta}_0 + X_1\hat{\beta}_1,$$

while the null hypothesis fit is $\hat{\eta}_0 = X_0\hat{\beta}_0$ (the $\hat{\beta}_0$ is the same for the null and alternative fits due to orthogonality).

Let $SSR_0 = \|y - \hat{\eta}_0\|^2$ be the sum of squared residuals under the null hypothesis fit, and let $SSR_1 = \|y - \hat{\eta}_1\|^2$ be the sum of squared residuals under the alternative fit. By orthogonality,

$$SSR_0 - SSR_1 = \|E_J\hat{\alpha} + X_1\hat{\beta}_1\|^2,$$

and $E_J\hat{\alpha} + X_1\hat{\beta}_1$ is the projection of $y$ onto the cone

$$C_1 = \{\eta \in \mathbb{R}^n : \eta = E\alpha + X_1\beta_1, \text{ for } \alpha \geq 0\}.$$

imsart-sts ver. 2014/10/16 file: sts-framework-rev2.tex date: July 16, 2018
A FRAMEWORK FOR CONSTRAINED REGRESSION

Such an orthogonality argument is well understood in the linear model setting; it applies to cones as well. For, if a cone is contained in a linear space orthogonal to a second linear space, then each face of the cone is orthogonal to the second linear space. If two cones are contained in orthogonal subspaces, then each face of one cone is orthogonal to every face of the other cone.

Under $H_0: \eta \in V_0$, the expected value of $y$ is in a space orthogonal to $C_1$, so we can apply Lemma 3.1 if the normal errors assumption holds. The likelihood ratio test statistic distribution is that of a mixture of chi-squared random variables:

$$T = \frac{1}{\sigma^2}(SSR_0 - SSR_1) \sim \chi^2_{d_1:d_2}(p),$$

where $d_1$ is the dimension of the largest linear space contained in $C_1$, and $d_2$ is the dimension of the smallest linear space containing $C_1$. The mixing distribution is determined through simulations where the response vector is generated as a normal random vector with mean zero and identity covariance matrix.

If $\sigma^2$ is unknown, we use a test statistic with a mixture-of-betas null distribution. Define

$$B = \frac{T}{T + SSR_1/\sigma^2} = \frac{SSR_0 - SSR_1}{SSR_0},$$

then for $c \in (0, 1)$,

$$P(B \leq c) = \sum_{d=d_1}^{d_2} B(d/2, (n - d_1 - d)/2)p_d,$$

where $B(a, b)$ is a beta random variable with parameters $a$ and $b$. The mixing parameter $p$ is the same as for the known-variance case.

For the normal errors model, this test is exact, at least to the precision of the estimated mixing parameters. For the generalized model, the null hypothesis fit is obtained, and as in the test for linear effects, the final values of $\xi$ and $W$ in the iterative re-weighting algorithm are treated as response vector and weights, respectively. The mixing parameters are obtained using the alternative hypothesis cone that is transformed by the weights. The test is not exact for the generalized model, but the following simulations demonstrate its good qualities.

To compare this test with the methods in \texttt{gam} and \texttt{scam}, we use the same simulation set-up as in the previous example, but we test $H_0: f \equiv 0$ versus $H_1: f$ is increasing. Specifically, the log-odds in the binomial model is assumed to be of the form $\eta(t_i, x_i) = f(t_i) + \beta_1 I\{x_i = 1\} + \beta_2 I\{x_i = 2\}$, but this time we fix $\beta_1 = 2$ and $\beta_2 = -1$. We use $f(t) \equiv 0$ to find the test size, and $f(t) = 30(t - 1/2)^4$ is used to compare the powers of the tests. To find the mixing parameters in \texttt{cgam}, 1000 replications are used. These results are presented in Table 3.2. The default tuning parameters in \texttt{scam} give conservative test sizes and lower power than the other methods, but the test sizes for the “matching” tuning parameters are inflated. The power for the \texttt{cgam} method, compared to the test in \texttt{gam}, is substantially higher, mostly due to comparing a one-sided test with a two-sided test.

3.3 Tests concerning individual constrained components

Next we consider the case where there are several constrained components and we want to test the significance of a single component of the model. In this case our
null hypothesis is not described by a linear space; instead we have a test against a sub-cone. Suppose the null hypothesis is $H_0 : \alpha_1 = 0$ versus $H_1 : \alpha_1 \geq 0$. Define

$$
\Delta_0 = [\Delta_2 | \cdots | \Delta_L | \Gamma_1 | \cdots | \Gamma_R], \quad \text{and} \quad \zeta_0 = \begin{bmatrix} \alpha_2 \\ \vdots \\ \alpha_L \\ \gamma_1 \\ \vdots \\ \gamma_R \end{bmatrix},
$$

so that we can write $\eta = \Delta_1 \alpha_1 + \Delta_0 \zeta_0 + X_c \beta_e + X \beta$. Let $P_v$ be the projection matrix for $V$, the space spanned by the columns of $X_c$ and $X$. Let $\hat{\Delta}_1 = (I - P_v)\Delta_1$ and $\hat{\Delta}_0 = (I - P_v)\Delta_0$, so $\eta$ can be written in the form $\hat{\Delta}_1 \alpha_1 + \Delta_0 \zeta_0 + X_c \beta_e + X \beta$, with $\alpha_1 \geq 0$ and $\alpha_0 \geq 0$.

Let $P_0$ be the projection matrix for the space spanned by the columns of $\hat{\Delta}_0$. Then

$$
\hat{\Delta}_1 \alpha_1 + \hat{\Delta}_0 \zeta_0 = (I - P_0)\hat{\Delta}_1 \alpha_1 + P_0 \Delta_1 \alpha_1 + \hat{\Delta}_0 \zeta_0
= (I - P_0)\hat{\Delta}_1 \alpha_1 + \hat{\Delta}_0 \left[ (\hat{\Delta}_0 \hat{\Delta}_0)^{-1} \hat{\Delta}_0 \hat{\Delta}_1 \alpha_1 + \zeta_0 \right]
= E_1 \alpha_1 + \hat{\Delta}_0 a,
$$

and $\eta = E_1 \alpha_1 + \hat{\Delta}_0 a + X_c \beta_e + X \beta$. To estimate the coefficients $\beta_e$ and $\beta$, we project $y$ onto the linear space $V$. To estimate $a$ and $\alpha_1$, we minimize $\|y - (E_1 \alpha_1 + \hat{\Delta}_0 a)\|^2$ subject to

$$
\begin{bmatrix} I & M \\ 0 & I \end{bmatrix} \begin{bmatrix} a \\ \alpha_1 \end{bmatrix} \geq 0
$$

where $M = -(\hat{\Delta}_0 \hat{\Delta}_0)^{-1} \hat{\Delta}_0 \hat{\Delta}_1$. By orthogonality, the estimate of $\eta$ is the sum of the two projections.

Because the space spanned by the columns of $E_1$ is orthogonal to the space spanned by the columns of $\hat{\Delta}_0$ and also orthogonal to $V$, we have

$$
\|y - (E_1 \hat{\alpha}_1 + \hat{\Delta}_0 \hat{\alpha}) + X_c \hat{\beta}_e + X \hat{\beta})\|^2 = \|y - (\hat{\Delta}_0 \hat{\alpha} + X_c \hat{\beta}_e + X \hat{\beta})\|^2 + \|E_1 \hat{\alpha}_1\|^2.
$$

Again by orthogonality, the last term in the above equation is the squared length of the projection of $y$ onto the cone defined by the columns of $E_1$. Therefore by
Lemma 3.1, if $H_0$ is true so that $E(y)$ is orthogonal to the column space of $E_1$, we have

$$T_1 = \frac{1}{\sigma^2} \|E_1 \hat{\alpha}_1\|^2 \sim \chi^2_{0:m_1}(p),$$

where $m_1$ is the number of columns of $E_1$. The mixing coefficients in $p$ can be approximated through simulations to the desired precision, similarly to the method for previous test.

If $\sigma^2$ is known, the test with statistic $T_1$ is exact under the normal errors assumption, with larger weights supporting the alternative hypothesis. In the more usual case of unknown model variance, we can construct a mixture of betas test statistic. Let $\hat{\eta}_1 = E_1 \hat{\alpha}_1 + \Delta_0 \hat{\alpha} + X_v \hat{\beta} + X \hat{\beta}$, and consider

$$T_2 = \frac{1}{\sigma^2} \|y - \hat{\eta}_1\|^2 = \frac{1}{\sigma^2} \|(I - P_J)y\|^2 = \frac{1}{\sigma^2} \|(I - P_J)(\eta + \varepsilon)\|^2$$

where $P_J$ is the projection matrix for the face of $C$ on which the projection of $y$ lands. Then if $\hat{\eta} = P_J \eta$,

$$T_2 = \frac{1}{\sigma^2} \|(I - P_J)\varepsilon\|^2 + \frac{1}{\sigma^2} \|\|\eta - \hat{\eta}\|^2 + 2\varepsilon^T (\eta - \hat{\eta})\|

= \frac{1}{\sigma^2} \|(I - P_J)\varepsilon\|^2 + o_p(n),$$

by the approximation error rate for the spline functions.

The term $\|(I - P_J)\varepsilon\|^2/\sigma^2$ has a $\chi^2_{(n-m_1-m-d_V);(n-d_V)}$ distribution, where $d_V$ is the dimension of $V$. However, the exact mixing distribution cannot be found through simulations. Instead, we approximate the distribution of this term with a chi-squared random variable with degrees of freedom $n - d_{obs}$, where $d_{obs}$ is the observed dimension of the face of $C$ on which the projection of $y$ lands. If $n$ is large compared to the dimension of the cone, this approximation is reasonable. The statistics $T_1$ and $T_2$ are independent by orthogonality of the spaces into which they project (the two cones are contained in orthogonal subspaces), so $T = T_1/(T_1 + T_2)$ has approximately a mixture-of-betas distribution. Specifically, for $c \in (0, 1)$,

$$P(T \leq c) \approx \sum_{d=0}^{m_1} P(B(d/2, (n-d_{obs})/2) \leq c) p_d.$$

To compare this test with the tests in $\text{gam}$ and $\text{scam}$, we generated 10,000 data sets from the model $y_i = f_1(t_{1i}) + f_2(t_{2i}) + \varepsilon_i$, for $i = 1, \ldots, n$, where $f_1(t) = 40(t_1 - 1/2)^+ t_2$, and $(\cdot)^+ = \max(\cdot, 0)$. The $\varepsilon_i$ are independent standard normal, but we do not assume the model variance is known. We considered the

<table>
<thead>
<tr>
<th>$f_2(t_2) = 0$</th>
<th>$f_2(t_2) = \sqrt{T_2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{gam}$ (def.)</td>
<td>$\text{scam}$ (def.)</td>
</tr>
<tr>
<td>n</td>
<td>$\delta$</td>
</tr>
<tr>
<td>100</td>
<td>$0.065$</td>
</tr>
<tr>
<td>200</td>
<td>$0.058$</td>
</tr>
<tr>
<td>400</td>
<td>$0.056$</td>
</tr>
</tbody>
</table>

Table 3
Proportions of 10,000 data sets for which $H_0 : f_2 = 0$ is rejected, with test size $\alpha = 0.05$. The predictors $t_1$ and $t_2$ are positively correlated, and $f_1(t_1) = 40(t_1 - 1/2)^+$. A FRAMEWORK FOR CONSTRAINED REGRESSION 13
test of $H_0 : f_2$ is constant versus $H_1 : f_2$ is increasing, using increasing regression splines to estimate both terms. The $t_{1i}$ values were generated as uniform on the unit interval, and the $t_{2i}$ values were the $t_{1i}$ values plus a normal random variable with standard deviation $1/4$, then scaled to be in the unit interval. The average correlation between the predictors is about .76. Simulation results are in Table 3.3, for three sample sizes and two underlying functions. The test sizes for \texttt{gam} are slightly over the target .05, but getting closer to the target as $n$ increases. The \texttt{scam} test sizes with the default tuning parameters are getting substantially smaller than the target, and with the “matching” tuning parameters the test size is inflated. The \texttt{cgam} test has appropriate size and the best power; its main advantage over \texttt{gam} being that it is a one-sided test.

For the sub-cone test in the generalized linear model, we first fit the null hypothesis model, and retain the final $\xi$ as a response vector and $W$ as the weights, from the iteratively re-weighted cone projection algorithm. We perform the test with the transformed response and the transformed cones. For the diabetes data used in the Section 2 example, we can test for the statistical significance of the skin-fold predictor. As expected, we get a large $p$-value (.71) and conclude that this predictor is not important for predicting diabetes, when BMI, age, and number of pregnancies are controlled for. We proceed to test each of the three remaining predictors, while controlling for the effects of the other two. We get small $p$-values for BMI and age ($3 \times 10^{-7}$ and $2 \times 10^{-6}$, respectively) but for the number of pregnancies, $p = .10$.

4. CONFIDENCE AND PREDICTION INTERVALS

Buja et al. (1989) provide an overview of linear scatterplot smoothers, where the least-squares estimator can be expressed as $\hat{\eta} = Sy$ for an $n \times n$ matrix $S$ that does not depend on $y$. For regression splines, $S$ is a projection matrix for a linear subspace spanned by the spline basis functions, but for kernel, smoothing spline, and penalized spline methods, $S$ is not idempotent. The true $\eta$ is not assumed to be in the space of possible estimators, so there is a bias component of the fit as well as a variance component. Reducing the amount of smoothing by increasing the number of knots or decreasing the bandwidth, smoothing parameter, or penalty parameter, will decrease the bias because the estimator can get closer to the true function, but this is at the expense of increasing the variance. The optimal convergence rate Stone (1980) is attained by balancing the bias and variance. Similarly, Huang (2001) characterized the bias as approximation error and the variance as estimation error, and noted that increasing the dimension of the estimation space decreases approximation error but increases estimation error, and showed that setting the two error rates equal gave optimal rates of convergence.

The estimated covariance matrix for the fit, $\text{var}(\hat{\eta}) = \hat{\sigma}^2 SS^\top$ can be used (with a suitable estimator $\hat{\sigma}^2$ for the model variance $\sigma^2$) to construct the confidence intervals as

$$\hat{\eta}_i \pm z_{\alpha/2} \left[ \text{var}(\hat{\eta}) \right]_{ii}^{1/2},$$

but because this does not take into account the bias, the coverage probability will be low unless the estimator is undersmoothed. Zhou et al. (1998) showed that for regression splines, the knots must increase at a faster-than-optimal rate to achieve asymptotically the desired confidence level.
The necessity for undersmoothing the fit to get good coverage is seen in other methods of non-parametric function estimation. The scatterplot in Figure 4 was simulated from the dashed curve with independent standard normal errors. The upper left plot shows the fit and point-wise confidence intervals using the popular \texttt{gam} function in the R package \texttt{mgcv} \cite{wood2018} with the default parameters, and the coverage probabilities estimated from 10,000 such fits are the top right. The default parameters do not provide good coverage where the curve is steep. In the second row a fit to the same scatterplot is shown with options \texttt{k=40, sp=.01} which provide more flexibility so that the approximation error or bias is minimal. The coverage probabilities are good, but there is spurious wiggling in the fit and the confidence bands are wider. In the final two rows are examples of constrained fits and confidence interval simulations results for the method proposed here. In the case where there are \textit{a priori} shape assumptions about the regression function \(f\), constrained regression splines obviate the problem of bias-variance trade-off. Many knots can be used with monotone or convex regression splines, because the constraints do not allow the spurious wiggling. Both good coverage and smaller-length intervals can be attained, because with more knots the bias is dominated by the variance. The default number of knots in \texttt{cgam} for \(n = 100\) is \(K = 7\); for \(n = 200\) we use \(K = 8\), and for \(n = 400\), \(K = 9\).

The projection \(\hat{\eta}\) lands on a face \(\mathcal{F}_J\) of the cone \(\mathcal{C}\). If \(P_J\) denotes the projection matrix for the linear space \(\mathcal{L}_J\), then

\[
\hat{\eta} = \sum_J P_J y I\{y \in \mathcal{C}_J\},
\]

where \(\mathcal{C}_J \subset \mathbb{R}^n\) is the set of \(y\) whose projection onto \(\mathcal{C}\) lands on \(\mathcal{F}_J\). To construct the point-wise confidence intervals at the design points, we propose

\[
\hat{\text{var}}(\hat{\eta}) = \hat{\sigma}^2 \sum_J P_J \hat{p}_J,
\]

where \(\hat{p}_J\) is the estimated probability that \(y\) is in \(\mathcal{C}_J\), and \(\hat{\sigma}^2\) is defined in \eqref{eq:variance}. The \(\hat{p}_J\) values are obtained by simulating many normal random vectors with mean \(\hat{\eta}\) and covariance matrix \(\hat{\sigma}^2 I\), and recording the resulting sets \(J\).

For estimating the variance of \(\hat{f}_1(t_1) + \cdots + \hat{f}_L(t_L) + \hat{g}_1(z_1) + \cdots + \hat{g}_R(z_R) + x^\top \hat{\beta}\) at arbitrary values of the predictors, we construct \(u\) to be a predictor vector as follows. Let \(w_c\) be a vector of length \(d_c\), where \(d_c\) is the number of columns of \(X_c\), and the first element is one, and the other elements are \(t_\ell\), for \(\ell\) such that \(f_\ell\) is concave or convex. Let \(b_\ell = (b_{\ell,1}(x_1), \ldots, b_{\ell,m_c}(x_1))\top\), where \(b_{\ell,j}\) is the \(j\)th spline basis function defined for the predictor \(x_\ell\). Let \(d_c\) contain the row of \(\Gamma_r\) appropriate for the value \(z_r\), and let \(b = [b_1\top, \ldots, b_L\top, d_1\top, \ldots, d_R\top]\top\). Finally, \(w = (b\top, w_c\top, x\top)\top\). Then \(\hat{f}_1(t_1) + \cdots + \hat{f}_L(t_L) + \hat{g}_1(z_1) + \cdots + \hat{g}_R(z_R) + x^\top \hat{\beta} = w^\top [\hat{\alpha}_c, [\hat{\beta}_c|\beta]\]. \(\hat{\alpha}_c, \hat{\beta}_c,\) and \(\beta\) are the coefficients for the projection of \(y\) onto the cone defined in \eqref{eq:cone_def}.

Then \(\hat{\text{var}}(\hat{f}(t_1, \ldots, t_L, z_1, \ldots, z_R, x)) = \hat{\sigma}^2 w^\top \hat{C} w\), where \(\hat{C} = \sum_J C^{(J)} \hat{p}_J\) and \(C^{(J)}\) is an \(M \times M\) matrix, where \(M = m + d_c + p\) and \(C_J\) is constructed as follows. For \(J \in \{1, \ldots, m\}\), define \(\Delta_J\) to have the columns of \(\Delta\) (defined in \eqref{eq:delta}) that are indexed by the elements of \(J\). Define \(X_J = [\Delta_J|X_c|X]\), and let \(m_J\) be the number of elements in \(J\). For \(k, \ell \in \{1, \ldots, m\}\),

\[
C^{(J)}_{k,\ell} = \begin{cases} (X_J^\top X_J)^{-1} & \text{if } j_k\text{th element of } J = k \text{ and } j_\ell\text{th element of } J = \ell \\ 0 & \text{if } k \notin J \text{ or } \ell \notin J. \end{cases}
\]
Fig 4. Example scatterplot smoothers with pointwise confidence bands; the first two rows use the \texttt{gam} function in the \texttt{R} package \texttt{mgcv}. The bottom two rows display the proposed constrained spline method. The data shown in the plots on the left was generated from the true regression function shown as the dashed curve with iid normal errors. The plots on the right show coverage probabilities (upper dots, left scale) and interval lengths (lower dots, right scale).
For $k \in \{1, \ldots, m\}$ and $\ell \in \{d_c + p + 1, \ldots, M\}$,  
\[
C_{k,\ell}^{(J)} = \begin{cases} 
(X_j^\top X_J)^{-1}_{j_k,\ell} & \text{if } j_k \text{th element of } J \text{ is } k \\
0 & \text{if } k \notin J.
\end{cases}
\]

For $k, \ell \in \{d_c + p + 1, \ldots, M\}$,  
\[
C_{k,\ell}^{(J)} = (X_j^\top X_J)^{-1}_{k+m_j-\ell+m_j-m}.
\]

The proposed confidence interval for $\eta(t_1, \ldots, t_L, z_1, \ldots, z_R, x)$ is  
\[
\hat{f}_1(t_1) + \cdots + \hat{f}_L(t_L) + \hat{g}_1(z_1) + \cdots + \hat{g}_L(z_R) + z^\top \beta \pm \frac{\alpha/2}{\hat{\sigma}^2 w C w^\top}^{1/2}. 
\]
This expression is consistent with (4.1); if $D = [\Delta | X_0 | Z]$, then $\text{var}(\hat{\eta}) = \hat{\sigma}^2 D C D^\top$.

For the generalized regression model, we use the terminal $\xi$ and $W$ in the iteratively re-weighted projection, and transform both $\xi$ and the cone $C$ using the weights. We determine the confidence bands as above, within the transformed model, and transform back to $\eta$ and $\mu$.

To compare our method to that of scam, we ran simulations with the binomial model with a single continuous predictor. We use two sample sizes and the underlying true mean function
\[
\mu(t) = .3 + \frac{.5 \exp(10t - 5)}{1 + \exp(10t - 5)},
\]
and again we use two sets of tuning parameters for the scam method. In Figure 5 the coverage probabilities and average interval lengths of the methods are compared. Although scam with the default tuning parameters produces the smallest intervals, the coverage is poor and does not improve with the larger sample size. The method with sp=0 and the same number of knots as the default in cgam produces similar results to cgam.

Finally, we apply the method to the diabetes data set used for examples in the last two sections. Using BMI and age, the two predictors found to be significantly related to the probability of diabetes, we obtain 95% point-wise confidence bands for the probabilities for all values of BMI, and three values of age; these are shown in Figure 6.
Fig 6. Estimated probability of diabetes in Pima Indian women, at three ages and all levels of BMI (solid curves). The 95% point-wise confidence bands for the probabilities are shown as dashed curves.

5. SHAPE AND MODEL SELECTION

For a model with many predictors, some of them correlated, the shapes might not be known a priori. In fact, the scientific question may concern the shapes. Is the probability of a cure decreasing in age, while controlling for the effects of other predictors? Does the strength of a filament increase with conductivity, given the thickness? A model selection method may be considered where predictors are chosen along with the shapes of the relationships. The cone information criterion (CIC) given by Meyer (2013) was derived as an estimate of the predicted squared error, and is given by

$$CIC = \psi(\eta) + \log \left( \frac{2(E_0(D))}{n - 1.5E_0(D) + 1} \right),$$

where $\psi$ is the negative log-likelihood function and $E_0(D)$ is the expected null degrees of freedom of the model. The second term of the CIC penalizes the complexity of the model, and hence it would be inappropriate to use observed degrees of freedom. The dimension of the observed face tends to be larger when the predictors are more strongly related to the response, and in the case where none of the constrained predictors are important, the observed face could simply be the linear space contained in the cone. Using observed face would result in larger penalties for better models. To determine the expected null degrees of freedom, simulations are necessary for each model in the list of possible models, where the response is generated from the appropriate model with no predictor effects. The observed degrees of freedom for each simulated response will be between $d_1$, the largest linear space contained in the cone, and $d_2$, the smallest linear space that contains the cone. The average of these observed degrees of freedom is averaged over many responses to get $E_0(D)$.

Several model or variable selection methods have been proposed involving isotonic or ordering restrictions, where the relationships are assumed to be monotone and variables can be chosen or not. Anraku (1999) proposed an information criterion specific to the order-restricted models, and Zhao and Peng (2002) used this for an isotonic dose-response problem, to determine at which levels the probability of a success increases. Peddada et al. (2003) proposed a method for selecting and clustering genes in gene expression data, based on traditional order-restricted methods.
models. Rueda (2013) proposed a variable selection method with unsmoothed isoton-
tic regression, based on the observed degrees of freedom of the fit. A method
similar to that proposed here, where shape is chosen from several different op-
tions, is found in Moisen et al. (2016) and is currently used for monitoring forest
disturbances via Landsat signals. In that application, different forest disturbances
produce different shapes in the Landsat time series.

To demonstrate the proposed CIC method, we look at the data set \texttt{trade.union}
in the R package \texttt{SemiPar}. The response variable is the logarithm of the wages,
for \( n = 532 \) workers in 1985. Three predictors are modeled with splines: age,
years of experience, and years of education. Each of the three spline functions
can take any of the nine following shapes: flat, increasing, decreasing, convex,
concave, or any of the four combinations of monotonicity and convexity. There
are seven nominal predictors: sex, occupation, whether or not the worker is in a
union, whether the worker is married, whether the worker is in the south, race
of worker, and sector (manufacturing, construction, other). With the constrained
splines, we can choose the shape: if each of the three “continuous” variables can
be either flat, increasing, decreasing, convex, concave, or one of the four combi-
nations of monotonicity and convexity, we have \( 9^3 \times 2^7 = 93,312 \) models. For
each model the calculation of \( E_0(D) \) involves simulations from the null model, so
the fitting of all models takes about two days. For this and larger data sets, we
can use a genetic algorithm to search for the best model.

We define \textit{phenotypes} as strings having a one-to-one map with the possible
models. Each element of the string describes one of the variables, where the
shapes are coded with numbers such as “1=increasing”, “2=decreasing,” etc.,
coding each of the eight shapes, and reserving the code “0” to mean “not in the
model.” For the nominal predictors, 1 means “in the model” and 0 means “not
in the model.” For example, the string \((5, 1, 4, 1, 1, 0, 0, 1, 0, 0)\) represents a model
where \( \eta \) is increasing and convex in \( x_1 \), increasing in \( x_2 \), and concave in \( x_3 \). The
remaining digits represent the nominal covariates.

To begin the genetic algorithm, we make an initial population of a few hundred
phenotypes. For each phenotype, we can compute its \textit{fitness}, i.e., the negative
CIC value for this model with the observed data. Then we let the population
evolve. In each generation, there are the following steps. For the mutation step,
we randomly choose a small number of phenotypes in the population, and for each,
we randomly choose a gene and randomly change it to one of the allowed values.
For the reproduction step, we randomly choose two members of the population.
We randomly choose half the genes of one and combine these with the remaining
genes of the other, to get an offspring. One of the old members of the population
is replaced with the offspring. For the reproduction step, the more fit members
of the population must be more likely to reproduce. To do this, tournament
methods are common: choose two pairs and have each pair “fight” where the
more fit are more likely to win; then the winners get to reproduce. Another way
is to randomly assign genders to the phenotypes: all females but only the most
fit males will reproduce. By mimicking “real” evolution, the genetic algorithm
converges to the best model.

There is no proof that a genetic algorithm always produces the optimal pheno-
type. The standard method for checking validity of the answer is to use another
starting population and run the algorithm again. If a different answer is obtained,
a larger starting population is needed. For the trade.union data, the genetic algorithm reliably gives the correct answer (minimizing the CIC over all models) with a population of 400 phenotypes and 10% mutation. The ShapeSelect function in the R package cgam allows the user to choose population size and mutation rate, as well as the set of shapes for each variable.

The genetic algorithm finds the best model for the trade union data in about half an hour. It chooses two of the continuous predictors, so that the fit to wages is increasing in years of education, increasing and concave in years of experience, and flat in age. Three nominal variables chosen are sex, occupation, and union membership. The fits for this model are shown in Figure 7, where the surface is shown with the effects of each of the three nominal predictors. It would be difficult to determine a priori some parametric regression shapes that would be appropriate, so non-parametric methods are valuable in this situation. However fits that are only “smooth” are difficult to interpret. The set of possible shapes provides a rich field and minimal assumptions; it is appropriate if there is confidence that the true components of the relationship do not “wiggle.” The results are interpretable in the context of the problem and provide information about the nature of the relationships.

6. DISCUSSION

Interest is growing for estimation and inference procedures where a minimum of assumptions is required. Linear models are convenient and inference methods are well known, but serious problems with model misspecification arise from making strong, invalid assumptions. Consider the data in Figure 8, where $y_i = f(x_i) + \varepsilon_i$ and $f(x) = 5e^{10x-5}/(1 + e^{10x-5})$ with i.i.d. standard normal $\varepsilon_i$. A categorical covariate was also simulated, strongly related to $x$ as shown in the plot. The scatterplot might suggest a linear relationship with different intercepts for the levels of the covariate. The results of the linear model, however, erroneously suggest that the covariate is a significant predictor, and ordinary residual analysis might not detect departures from the model. If we assume only that the relationship between $y$ and $x$ is smooth and increasing, the fit on the right in Figure 8 is obtained, with a large $p$-value for the covariate effect. In addition, the fitted function is quite close to the true mean. Especially when there are many predictors that are related to each other, parametric relationships are difficult to determine. Strong parametric assumptions that may be unjustified will lead to
making the wrong conclusions, and the probability of making the wrong conclusion actually increases with the sample size. Non-parametric function estimation methods make only vague assumptions about the components, such as “smooth” and “increasing.” Minimizing the assumptions will minimize the model misspecification errors. Using shape constraints as well as smoothing, either as a priori assumptions or through shape and variable selection, leads to flexible fits that are interpretable in the context of the problem.

If unsmoothed shape-constrained estimators are used for continuous predictors, the dimension of the smallest linear space containing the cone can be as large as $n$, the sample size. Dimension reduction through regression splines is an important advantage as well as having a more satisfactory estimator when the smoothness assumption is valid. Modeling continuous predictors with regression splines allows a much smaller model dimension, which in turn allows for the inference methods such as the sub-cone test. The choice of knots, crucial for unconstrained splines, is rendered less important by the robustness of the constrained fits to knot choices. We can choose a “generous” number of knots for each component, because constraints disallow the “wiggling” usually associated with over-fitting in scatterplot smoothers.

For the scam methods, Pya and Wood (2015) implemented a Bayesian approach to obtain approximately normal distributions for the spline basis coefficients, attributed to Wahba (1983) and Silverman (1985). The methods coded in cgam use the cone formulation with formally derived inference methods, specifically for estimation with constraints.

ACKNOWLEDGEMENTS

This work was partially funded by NSF-MMS-1533804. The author would like to express great appreciation for Dr. Xiyue Liao, who maintains the cgam package on CRAN.

REFERENCES


**APPENDIX A: R CODE FOR THE SIMULATIONS**

Table 1, cgam

```r
nloop=10000
n=800
x=0:(n-1)/(n-1)
eta0=-1+60*(x-1/2)^4;eta0[x<1/2]=-1
pr1=.05+.85*x^4
pr2=.4+.55*x^(1/4)
pval=1:nloop
for(iloop in 1:nloop){
  z=1:n*0+3
  u=runif(n)
  z[u<pr1]=1;z[u>pr2]=2
  eta=eta0
  # eta[z==1]=eta[z==1]-1 (uncomment for power)
}```
mu=exp(eta)/(1+exp(eta))
y=1:n*0
y[runif(n)<mu]=1
ans=cgam(y~s.incr(x)+factor(z),family="binomial")
pval[iloop]=anova(ans)$coefficients1[3]
}
sum(pval<=.05)/iloop

**Table 1, scam**
nloop=10000
n=100
x=0:(n-1)/(n-1)
et0=-1+60*(x-1/2)^4;eta0[x<1/2]=-1
pr1=.05+.85*x^-4
pr2=.4+.55*x^(1/4)
pval=1:nloop
for(iloop in 1:nloop){
z=1:n*0+3
u=runif(n)
z[u<pr1]=1;z[u>pr2]=2
eta=eta0
# eta[z==1]=eta[z==1]-1
mu=exp(eta)/(1+exp(eta))
y=1:n*0
y[runif(n)<mu]=1
# ans=scam(y~s(x,bs='mpi',k=10)+factor(z),family="binomial",sp=0) #(match)
ans=scam(y~s(x,bs='mpi')+factor(z),family="binomial") ## (default)
pval[iloop]=anova(ans)$pTerms.pv
}
sum(pval<=.05)/iloop

**Table 2, cgam**
nloop=10000
n=800
x=0:(n-1)/(n-1)
et0=30*(x-1/2)^4;eta0[x<1/2]=0
# eta0=1:n*0
pr1=.05+.85*x^-4
pr2=.4+.55*x^(1/4)
pval=1:nloop
for(iloop in 1:nloop){
z=1:n*0+3
u=runif(n)
z[u<pr1]=1
z[u>pr2]=2
eta=eta0
eta[z==1]=eta[z==1]+2
eta[z==2]=eta[z==2]-1
mu=exp(eta)/(1+exp(eta))
A FRAMEWORK FOR CONSTRAINED REGRESSION

\begin{verbatim}
y=1:n*0
y[runif(n)<mu]=1
ans=cgam(y~s.incr(x)+factor(z),family="binomial",nsim=0)
pval[iloop]=anova(ans)$coefficients2[3]
}
hist(as.numeric(pval))
sum(pval<.05)/nloop

Table 2, scam
nloop=10000
n=800
x=0:(n-1)/(n-1)
#eta0=30*(x-1/2)^4;eta0[x<1/2]=0
eta0=1:n*0
pr1=.05+.85*x^4
pr2=.4+.55*x^(1/4)
pval=1:nloop
for(iloop in 1:nloop){
    z=1:n*0+3
    u=runif(n)
    z[u<pr1]=1
    z[u>pr2]=2
    eta=eta0
    eta[z==1]=eta[z==1]+2
    eta[z==2]=eta[z==2]-1
    mu=exp(eta)/(1+exp(eta))
    y=1:n*0
    y[runif(n)<mu]=1
    ans=scam(y~s(x,bs='mpi',k=10)+factor(z),family="binomial",sp=0)  ## match
    # ans=scam(y~s(x,bs='mpi')+factor(z),family="binomial")  ## default
    pval[iloop]=summary(ans)$s.pv
}
sum(pval[1:nloop]<.05,na.rm=TRUE)/(nloop-sum(is.na(pval)))

Table 3, cgam
nloop=10000
n=100
x1=0:(n-1)/(n-1)
x2=x1+rnorm(n)/2
x2=(x2-min(x2))/(max(x2)-min(x2))
pval=1:nloop
eta1=40*(x1-1/2)^4;eta1[x1<1/2]=0
#eta2=1:n*0  ## null
eta2=sqrt(x2)  ## alternative
for(iloop in 1:nloop){
    y=eta1+eta2+rnorm(n)
    ans=cgam(y~s.incr(x1)+s.incr(x2),nsim=0)
    pval[iloop]=anova(ans)$coefficients2[2,3]
}
\end{verbatim}
Table 3, scam

\begin{verbatim}
Table 3, scam
nloop=10000
n=100
x1=0:(n-1)/(n-1)
x2=x1+rnorm(n)/2
x2=(x2-min(x2))/(max(x2)-min(x2))
pval=1:nloop
eta1=40*(x1-1/2)^4;eta1[x1<1/2]=0
eta2=1:n*0  ## null
#eta2=sqrt(x2)  ## alternative
for(iloop in 1:nloop){
  y=eta1+eta2+rnorm(n)
  # ans=cgam(y~s.incr(x1)+s.incr(x2),nsim=0)
  ans=scam(y~s(x1,bs='mpi',k=8)+s(x2,bs='mpi',k=8),sp=c(0,0))
  pval[iloop]=summary(ans)$s.pv[2]
}
sum(pval<.05)/nloop
\end{verbatim}