COSSO-type Penalized Likelihood Method for Simultaneous Nonparametric Regression and Model Selection in Exponential Families

Hao Helen Zhang and Yi Lin

North Carolina State University and University of Wisconsin at Madison

Abstract: This paper extends the component selection and smoothing operator (COSSO), a nonparametric variable selection approach recently developed in Lin and Zhang (2002), to exponential families. We propose the COSSO-type penalized likelihood method for nonparametric regression models in the framework of tensor product splines. Instead of using the roughness penalty, the COSSO imposes the reproducing kernel Hilbert space norm penalty on functional components and yields a soft-thresholding estimate. Thus the COSSO-type penalized likelihood method conducts model fitting and variable selection simultaneously. We shall prove the representer theorem, which ensures the maximum penalized likelihood estimator lies in a finite dimensional function space. The connection between the COSSO-type penalized likelihood method and standard smoothing spline ANOVA models is discussed. For computation, we develop an efficient algorithm that iterates between a reweighed smoothing spline and the constrained quadratic programming. Automatic selection of smoothing parameters via cross validation criterion is used. Bernoulli examples, with both independent and correlated covariate structures, are presented to illustrate the performances of the proposed method.

Key words and phrases: penalized likelihood, variable selection, COSSO, nonparametric regression, smoothing spline ANOVA, generalized linear models.

1. Introduction

We consider the nonparametric regression and model selection problem in the exponential family framework. Suppose we are interested in predicting a response variable Y given input vector $\mathbf{X} = \left(X^{(1)}, ..., X^{(d)}\right) \in \mathcal{X} \subseteq \mathcal{R}^d$, and assume that conditioning on $\mathbf{X} = \mathbf{x}$, Y follows an exponential family distribution with the canonical density form

$$\exp[\{yf(\mathbf{x}) - B(f(\mathbf{x}))\}/A(\phi) + C(y, \phi)], \tag{1.1}$$

where $A > 0, B$, and C are known functions, and ϕ are dispersion parameters. The goal of the regression problem is to estimate $f(\mathbf{x})$ based on an independently and identically distributed sample $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$. In many practical situations, the number of input variables is large and some of the input variables are superfluous. In such situations, effective variable selection can improve both the accuracy and the interpretability of the estimated model.

Many variable selection methods have been proposed for the commonly studied linear regression model. Traditional methods include forward selection, backward elimination, and best subset selection. Recent developments include the nonnegative garotte (Breiman (1995)), the LASSO (Tibshirani (1996)), the SCAD (Fan and Li (2001)), and the least angle regression (Efron et al. (2004)). These methods implement variable selection and coefficient shrinkage at the same time, and improve on the traditional methods in terms of estimation accuracy and stability of the solution.

The nonparametric regression model allows more flexibility than the linear model and is the topic of this paper. Many popular proposals for variable selection and estimation in nonparametric regression, such as CART (Breiman, Friedman, Olshen and Stone (1984)) , TURBO (Friedman and Silverman (1989)), BRUTO (Hastie (1989)), and MARS (Friedman (1991)), use greedy search type of algorithm for variable selection. This is similar to the forward selection and backward elimination in the linear regression. The greedy search type of algorithm can often be made very fast, but may suffer from being myopic at each step of the search since it looks only one step ahead, thus may not be taking the globally optimal step. Given the successes of the global penalized likelihood methods such as LASSO and the SCAD in the linear model, it is desirable to develop global algorithms based on penalized likelihood for nonparametric regression models.

In the Gaussian nonparametric regression setting, Lin and Zhang (2002) proposed the component selection and smoothing and operator (COSSO) for variable selection and estimation in the smoothing spline ANOVA model. The smoothing spline ANOVA model provides a general framework for high dimensional function estimation, and has been successfully applied to many practical problems. See Wahba (1990), Wahba, Wang, Gu, Klein and Klein (1995) , and Gu (2002). The COSSO is a penalized likelihood method. It differs from the common smoothing spline method in that the penalty functional in the COSSO is the sum of component norms, whereas the penalty functional in the common smoothing spline is the sum of squared component norms. This difference between the COSSO and the smoothing spline is similar to that between the LASSO and the ridge regression. In fact, as noted in Lin and Zhang (2002), the LASSO can be seen as the COSSO applied to the linear model. Lin and Zhang (2002) showed that this new penalty tends to shrink insignificant functional components to exact zeros, and enjoys excellent model selection and estimation properties. In this paper we shall generalize the COSSO to the more general setting of nonparametric regression in exponential families, and develop a general algorithm using the iteratively-reweighted least squares (IRLS) estimation procedure. The general framework developed in this paper allows the treatment of many types of responses, such as non-normal responses, binary and polychotomous responses, and the event counts data, with the COSSO-type penalized likelihood methods.

This paper is organized as follows. In Section 2 we introduce the formulation of the COSSO-type penalized likelihood method, and give some theoretical properties of the estimator. In Section 3, we give the computation algorithm and discuss the issue of choosing the tuning parameters. Sections 4 and 5 show some numerical results for both simulation and real examples. A summary and discussion are given in Section 6.

2. COSSO-type Penalized Likelihood Method

2.1 Smoothing Spline ANOVA Model

The functional ANOVA decomposition of a multivariate function f is

$$f(\mathbf{x}) = b + \sum_{j=1}^{d} f_j(x^{(j)}) + \sum_{j=1}^{d} \sum_{k=j+1}^{d} f_{jk}(x^{(j)}, x^{(k)}) + \cdots, \tag{2.1}$$

where b is a constant, f_j's are the main effects, f_{jk}'s are the two-way interactions, and so on. The identifiability of the terms in (2.1) is assured by side conditions through averaging operators. In the smoothing spline ANOVA model, we assume $f_j \in H^{(j)}$, where $H^{(j)}$ is a space of functions of $x^{(j)}$ over $[0,1]$, and the full function space for f is the tensor product space $\otimes_{j=1}^{d} H^{(j)}$.

When $x^{(j)}$ is a continuous covariate, typically $H^{(j)}$ is chosen to be the mth order Sobolev Hilbert space $\{h : h, h', ..., h^{(m-1)}$ are absolutely continuous, $h^{(m)} \in \mathcal{L}_2[0,1]\}$. We will use the second order Sobolev Hilbert space with $m = 2$ in our computation. When endowed with the norm

$$\|h\|^2 = \{\int_0^1 h(t)dt\}^2 + \{\int_0^1 h'(t)dt\}^2 + \int_0^1 \{h''(t)\}^2 dt,$$

the second order Sobolev space is an RKHS with the reproducing kernel (RK) $K(s,t) = 1 + \bar{K}(s,t)$, where $\bar{K}(s,t) = k_1(s)k_1(t) + k_2(s)k_2(t) - k_4(|s-t|)$, $k_1(t) = t - \frac{1}{2}$, $k_2(t) = \frac{1}{2}\{k_1^2(t) - \frac{1}{12}\}$, and $k_4(t) = \frac{1}{24}\{k_1^4(t) - \frac{1}{2}k_1^2(t) + \frac{7}{240}\}$. See Wahba (1990) and Gu (2002). When $x^{(j)}$ is a categorical covariate taking values on the discrete domain $\mathcal{X} = \{1, ..., L\}$, a function on $\mathcal{X}$ is simply an L-vector and the evaluation functional is the coordinator extraction. We define the squared norm of such an L-vector to be the $1/L$ of the common Euclidean space squared norm to ensure that functions on categorical variables with different number of categories have comparable norms. Under this norm we have $H^{(j)} = \{1\} \oplus \bar{H}^{(j)}$, where $\bar{H}^{(j)}$ is an RKHS with reproducing kernel $\bar{K}(s,t) = L\delta(s,t) - 1$. Here $\delta(\cdot,\cdot)$ is the Kronecker notation. That is, $\delta(s,t) = 1$ if $s = t$; $= 0$ otherwise.

In the smoothing spline ANOVA model, usually only lower order interactions are retained in decomposition for easy computation and interpretability. Correspondingly, the function space assumed for the smoothing spline ANOVA model is a subspace $\mathcal{F} \in \otimes_{j=1}^d H^{(j)}$ which can be written as $\mathcal{F} = \{1\} \oplus_{\alpha=1}^p \mathcal{F}^\alpha$, where $\mathcal{F}^1, ..., \mathcal{F}^p$ are p orthogonal subspaces of $\mathcal{F}$. For the additive model, $p = d$ and $\mathcal{F}^\alpha$'s are the main effect subspaces. For the two-way interaction model, $p = d(d+1)/2$, $\mathcal{F}^\alpha$'s are the main effect and two-way interaction subspaces. For the saturated SS-ANOVA model, $p = 2^d - 1$. We denote the norm in $\mathcal{F}$ by $\|\cdot\|$. This is the norm induced by the norm in $\otimes_{j=1}^d H^{(j)}$.

2.2 COSSO-type Penalized Likelihood Method in Exponential Families

The COSSO-type penalized likelihood method solves the following problem:

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \big[-l\{y_i, f(\mathbf{x}_i)\}\big] + \tau^2 \sum_{\alpha=1}^p J(f), \quad \text{with} \quad J(f) = \sum_{\alpha=1}^p \|P^\alpha f\|, \qquad (2.2)$$

where $P^\alpha f$ is the orthogonal projection of f onto $\mathcal{F}^\alpha$, τ is a smoothing parameter, and

$$l\{y, f(\mathbf{x})\} \equiv yf(\mathbf{x}) - B\{f(\mathbf{x})\} \qquad (2.3)$$

is the log likelihood for the exponential family distribution (1.1). In the important special case of the additive model, this becomes

$$\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\big[-l\{y_i, f(\mathbf{x}_i)\}\big] + \tau^2\sum_{j=1}^{d}\|f_j\|, \quad \text{with} \quad f(\mathbf{x}) = b + \sum_{j=1}^{d} f_j(x^{(j)}). \tag{2.4}$$

3.1 Properties of Solutions

Since the functional $\mathcal{L}(f) = \frac{1}{n}\sum_{i=1}^{n}\big[-l\{y_i, f(\mathbf{x}_i)\}\big]$ is continuous and convex in f, so is the objective functional in (2.2). Theorem 2.1 shows that the solution to (2.2) exists in $\mathcal{F}$. Technical proof is given in Appendix 1.

Theorem 2.1 *Let $\mathcal{F}$ be a reproducing kernel Hilbert space of functions over an input space $\mathcal{X}$. Assume that $\mathcal{F}$ can be decomposed as in the orthogonal sum of some functional subspaces. Then there exists a minimizer of (2.2) in $\mathcal{F}$.*

Kimeldorf and Wahba (1971) showed the representer theorem for the ordinary SS-ANOVA models, which implies that the optimal solution lies in a finite dimensional subspace even though $\mathcal{F}$ is infinite dimensional. Now we show that the representer theorem also holds for the COSSO-type penalized likelihood estimation, which substantially reduces the difficulty in computation.

Theorem 2.2 (Representer Theorem) *Let the minimizer of (2.2) be $\hat{f} = \hat{b} + \sum_{\alpha=1}^{p}\hat{f}_\alpha$, with $\hat{f}_\alpha \in \mathcal{F}^\alpha$. Then $\hat{f}_\alpha \in span\{R_\alpha(\mathbf{x}_i,\cdot), i = 1,...,n\}$, where $R_\alpha(\cdot,\cdot)$ is the reproducing kernel of the space $\mathcal{F}^\alpha$.*

Proof. For any $f \in \mathcal{F}$, we can write $f = b + \sum_{\alpha=1}^{p} f_\alpha$ with $f_\alpha \in \mathcal{F}^\alpha$. Let the projection of f_α onto span$\{R_\alpha(\mathbf{x}_i,\cdot), i = 1,...,n\} \subset \mathcal{F}^\alpha$ be denoted by g_α, and its orthogonal complement by h_α. Then $f_\alpha = g_\alpha + h_\alpha$, and $\|f_\alpha\|^2 = \|g_\alpha\|^2 + \|h_\alpha\|^2$, $\alpha = 1,...,p$. Since the reproducing kernel of $\mathcal{F}$ is $R = 1 + \sum_{\alpha=1}^{p} R_\alpha$ is, we have

$$f(\mathbf{x}_i) = \langle 1 + \sum_{\alpha=1}^{p} R_\alpha(\mathbf{x}_i,\cdot), b + \sum_{\alpha=1}^{p}(g_\alpha + h_\alpha)\rangle = b + \sum_{\alpha=1}^{p}\langle R_\alpha(\mathbf{x}_i,\cdot), g_\alpha\rangle,$$

where $\langle\cdot,\cdot\rangle$ is the inner product in $\mathcal{F}$. Therefore (2.2) can be written as

$$\frac{1}{n}\sum_{i=1}^{n}\big[-l\{y_i, b + \sum_{\alpha=1}^{p}\langle R_\alpha(\mathbf{x}_i,\cdot), g_\alpha\rangle\}\big] + \tau^2\sum_{\alpha=1}^{p}(\|g_\alpha\|^2 + \|h_\alpha\|^2)^{1/2}.$$

Therefore any minimizer f satisfies $h_\alpha = 0$, $\alpha = 1,...,p$.

3.2. Equivalent Formulation

We propose an equivalent but easy-to-solve formulation for the optimization problem in (2.2). Let $\theta = (\theta_1, ..., \theta_p)^{\mathrm{T}}$. For any positive integer q, consider

$$\min_{f \in \mathcal{F}, \theta \geq \mathbf{0}} \frac{1}{n} \sum_{i=1}^{n} \big[- l\{y_i, f(\mathbf{x}_i)\}\big] + \lambda_0 \sum_{\alpha=1}^{p} \theta_\alpha^{-q} \|P^\alpha f\|^2 + \lambda \sum_{\alpha=1}^{p} \theta_\alpha^q, \tag{2.5}$$

where $\mathbf{0}$ is the vector of zeros, $\lambda_0 > 0$ is a constant, and λ is the smoothing parameter. If $\theta_\alpha = 0$, then the minimizer is taken to satisfy $\|P^\alpha f\|^2 = 0$. We take the convention $0/0 = 0$ throughout this paper. A series of optimization problems are defined in (2.5). Lemma 2.1 shows that solving (2.2) is equivalent to solving (2.5) with an arbitrary choice of q.

Lemma 2.1 *Fix $q > 0$. Set $\lambda = \tau^4/(4\lambda_0)$. (i) If $\hat{f}$ minimizes (2.2), set $\hat{\theta}_\alpha = \lambda_0^{1/2}\lambda^{-1/2}\|P^\alpha \hat{f}\|$, then the pair $(\hat{\theta}, \hat{f})$ minimizes (2.5). (ii) On the other hand, if a pair $(\hat{\theta}, \hat{f})$ minimizes (2.5), then $\hat{f}$ minimizes (2.2).*

Proof. Denote the functional in (2.2) by $A(f)$, and the functional in (2.5) by $B_q(\theta, f)$ for any fixed q. For any $\theta_\alpha \geq 0, f \in \mathcal{F}$, we have $\lambda_0 \theta_\alpha^{-q}\|P^\alpha f\|^2 + \lambda\theta_\alpha^q \geq 2\lambda_0^{1/2}\lambda^{1/2}\|P^\alpha f\|^2 = \tau^2\|P^\alpha f\|$. If $\theta_\alpha = 0$, let the minimizer f satisfy $\|P^\alpha f\|^2 = 0$. Therefore $B_q(\theta, f) \geq A(f)$ for any $\theta_\alpha \geq 0, f \in \mathcal{F}$, and "=" holds if and only if $\theta_\alpha = \lambda_0^{1/2}\lambda^{-1/2}\|P^\alpha f\|$, $\alpha = 1, ..., p$.

In particular, the choice of $q = 1$ implies the connection between the COSSO method and the standard smoothing spline. When $q = 1$, (2.5) becomes

$$\min_{f \in \mathcal{F}, \theta \geq \mathbf{0}} \frac{1}{n} \sum_{i=1}^{n} \big[- l\{y_i, f(\mathbf{x}_i)\}\big] + \lambda_0 \sum_{\alpha=1}^{p} \theta_\alpha^{-1} \|P^\alpha f\|^2 + \lambda \sum_{\alpha=1}^{p} \theta_\alpha, \tag{2.6}$$

which is very similar to the common smoothing spline with multiple smoothing parameters, except the additional penalty on θ's. Note λ is the only smoothing parameter in (2.6). We generally fix λ_0 at some value for computational consideration. From the smoothing spline literature, with fixed θ's the minimizer of (2.6) has the form $f(\mathbf{x}) = b + \sum_{i=1}^{n} c_i R_\theta(\mathbf{x}_i, \mathbf{x})$. Since $R_\theta = 1 + \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha$, the COSSO-type penalized likelihood estimator has the form

$$f(\mathbf{x}) = b + \sum_{i=1}^{n} c_i \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha(\mathbf{x}, \mathbf{x}_i).$$

4. Algorithms

For fixed λ_0 and λ, we will solve (2.6) using the Newton-Raphson iteration. Given a current solution f^0, the conditional mean of Y at $\mathbf{x}_i$ is $\mu_i^0 = \dot{B}(f^0(\mathbf{x}_i))$ and the conditional variance is $V_i^0 = \ddot{B}(f^0(\mathbf{x}_i))$. Define $\nu_i = -y_i + \mu_i^0$ and $w_i = V_i^0$. The second-order Taylor expansion of $-y_i f(\mathbf{x}_i) + B(f(\mathbf{x}_i))$ at $f^0(\mathbf{x}_i)$ is

$$\begin{aligned} & -y_i f^0(x_i) + B(f^0(\mathbf{x}_i)) + \nu_i\big[f(\mathbf{x}_i) - f^0(\mathbf{x}_i)\big] + \frac{1}{2}w_i\big[f(\mathbf{x}_i) - f^0(\mathbf{x}_i)\big]^2 \\ & = \frac{1}{2}w_i\big[f(\mathbf{x}_i) - f^0(\mathbf{x}_i) + \frac{\nu_i}{w_i}\big]^2 + \beta_i, \end{aligned}$$

where β_i is independent of $f(\mathbf{x}_i)$. Define the adjusted dependent variable

$$z_i = f^0(\mathbf{x}_i) + (y_i - \mu_i^0)/w_i,$$

then the Newton iteration update of (2.6) becomes

$$\min_{f\in\mathcal{F},\theta\geq\mathbf{0}} \frac{1}{n}\sum_{i=1}^{n} w_i\big[z_i - f(\mathbf{x}_i)\big]^2 + \lambda_0\sum_{\alpha=1}^{p}\theta_\alpha^{-1}\|P^\alpha f\|^2 + \lambda\sum_{\alpha=1}^{p}\theta_\alpha. \tag{4.1}$$

For Bernoulli distribution with the logit link, $w_i = \mu_i^0(1-\mu_i^0)$, $z_i = f^0(\mathbf{x}_i) + (y_i - \mu_i)/w_i$. For Poisson distribution with the log link, $w_i = \mu_i^0$, and $z_i = f^0(\mathbf{x}_i) + (y_i - \mu_i^0)/w_i$. The first part of (4.1) is actually a weighted least squares with the weights changing in each iteration. This iteratively-reweighted least-squares (IRLS) is commonly used for computing the maximal likelihood estimates in generalized linear and additive models. See Hastie and Tibshirani (1990).

Let $\boldsymbol{y} = (y_1, ..., y_n)^{\mathrm{T}}, \boldsymbol{z} = (z_1, ..., z_n)^{\mathrm{T}}, \boldsymbol{f} = (f(\mathbf{x}_1), ..., f(\mathbf{x}_n))^{\mathrm{T}}, \boldsymbol{c} = (c_1, ..., c_n)^{\mathrm{T}}$, $W = \mathrm{diag}[w_1, ..., w_n]$. Define $\mathbf{1}_n$ to be the column vector of n ones and I_n to be the identity matrix of dimension n. With some abuse of notations, we also use R_α for the matrix $\{R_\alpha(\mathbf{x}_i, \mathbf{x}_j)\}_{i,j=1}^n$, and R_θ for the matrix $\sum_{\alpha=1}^p \theta_\alpha R_\alpha$. Then $\boldsymbol{f} = b\mathbf{1}_n + R_\theta\boldsymbol{c}$, $\sum_{\alpha=1}^p \|P^\alpha f\|^2 = \sum_{\alpha=1}^p \boldsymbol{c}^{\mathrm{T}} R_\alpha \boldsymbol{c} = \boldsymbol{c}^{\mathrm{T}} R_\theta \boldsymbol{c}$, and (4.1) becomes

$$\min_{f\in\mathcal{F},\theta\geq\mathbf{0}} \frac{1}{n}\sum_{i=1}^{n} w_i\big[z_i - f(\mathbf{x}_i)\big]^2 + \lambda_0\boldsymbol{c}^{\mathrm{T}}R_\theta\boldsymbol{c} + \lambda\sum_{\alpha=1}^{p}\theta_\alpha. \tag{4.2}$$

Two algorithms will be developed to solve θ and $(b, \boldsymbol{c})$ iteratively.

4.1. Full Basis Algorithm

1. With θ's fixed, solving (4.2) is equivalent to

$$\min_{b,\boldsymbol{c}}(\boldsymbol{z} - b\mathbf{1}_n - R_\theta\boldsymbol{c})^{\mathrm{T}}W(\boldsymbol{z} - b\mathbf{1}_n - R_\theta) + n\lambda_0\boldsymbol{c}^{\mathrm{T}}R_\theta\boldsymbol{c}.$$

   Define $\boldsymbol{z}_w = W^{1/2}\boldsymbol{z}$,$R_{w\theta} = W^{1/2}R_\theta W^{1/2}$,$\boldsymbol{c}_w = W^{-1/2}\boldsymbol{c}$, and $\boldsymbol{s}_w = W^{1/2}\mathbf{1}_n$. It is easy to derive (in Appendix 2) the following smoothing spline problem

$$\min_{b,\boldsymbol{c}_w}||\boldsymbol{z}_w - b\boldsymbol{s}_w - R_{w\theta}\boldsymbol{c}_w||^2 + n\lambda_0\boldsymbol{c}_w^{\mathrm{T}}R_{w\theta}\boldsymbol{c}_w. \tag{4.3}$$

2. With $(b, \boldsymbol{c})$ fixed, solving (4.2) becomes

$$\min_{\theta\geq\mathbf{0}}(\boldsymbol{z}-b\mathbf{1}_n-\sum_{\alpha=1}^{p}\theta_\alpha R_\alpha\boldsymbol{c})^{\mathrm{T}}W(\boldsymbol{z}-b\mathbf{1}_n-\sum_{\alpha=1}^{p}\theta_\alpha R_\alpha\boldsymbol{c})+n\lambda_0\sum_{\alpha=1}^{p}\theta_\alpha(\boldsymbol{c}^{\mathrm{T}}R_\alpha\boldsymbol{c})+n\lambda\sum_{\alpha=1}^{p}\theta_\alpha. \tag{4.4}$$

   Let G be the matrix with its αth column being $\boldsymbol{g}_\alpha = R_\alpha\boldsymbol{c}$, $\alpha = 1, ..., p$, $G_w = W^{1/2}G$, and $\boldsymbol{u}_w = \boldsymbol{z}_w - b\mathbf{1}_n - \frac{n}{2}\lambda_0\boldsymbol{c}_w$. It is easy to show that, for some $M \geq 0$, (4.4) is equivalent to

$$\min_{\theta}||\boldsymbol{u}_w - G_w\theta||^2 \quad \text{subject to} \qquad \mathbf{1}^{\mathrm{T}}\theta \leq M, \quad \theta \geq \mathbf{0} \tag{4.5}$$

   The tuning parameter M in (4.5) is equivalent to the λ in (4.4). Note (4.5) has the same formulation as the non-negative garrote approach, and we use the MATLAB optimization function "quadprog" to solve it.

We now give the algorithm for solving the COSSO with fixed λ_0 and M. The issue of tuning parameter is very important and will be discussed in a later section.

Algorithm 1

step 1: Initialize $\theta_\alpha = 1$ for $\alpha = 1, ..., p$. Initialize $f_i = \bar{Y}, \mu_i = \dot{B}(f_i), w_i = \ddot{B}(f_i), z_i = f_i + (y_i - \mu_i)/w_i$ for $i = 1, ..., n$. Let W=diag$[w_1, ..., w_n]$.

step 2: Calculate $\boldsymbol{z}_w = W^{1/2}\boldsymbol{z}, \boldsymbol{s}_w = W^{1/2}\mathbf{1}$ and $R_{w\theta}$. Solve (4.3) for $(b, \boldsymbol{c}_w)$.

step 3: Calculate $\boldsymbol{u}_w = \boldsymbol{z}_w - b\mathbf{1}_n - \frac{n}{2}\lambda_0\boldsymbol{c}_w$ and G_w. Solve (4.5) for θ's.

step 4: With current $(b, \boldsymbol{c})$ and θ's, calculate $\boldsymbol{f} = b\mathbf{1}_n + R_\theta\boldsymbol{c}$. Update the weights $w_i = \ddot{B}(f_i), i = 1, ..., n$, and the weight matrix W.

step 5: Go to step 2, until the convergence criterion meets.

4.2. Subset Basis Algorithm

When n is large, the computation can be very intensive. An efficient subset basis algorithm is developed using the parsimonious basis approach. The idea is to subsample N points from the data and use them to generate basis functions and span the function space. Gu and Kim (2001) shows that N can be much smaller than n without degrading the performance of the estimation. This approach is also used by Xiang and Wahba (1998), Ruppert and Carroll (2000), Lin, Wahba, Xiang, Gao, Klein and Klein (2000), and Yau, Kohn and Wood (2002). We use the simple random scheme to subsample the points. Alternatively, a cluster algorithm in Xiang and Wahba (1998) can be used. Denote the sampled points by $\{\mathbf{x}_{1*}, ..., \mathbf{x}_{N*}\}$. The minimizer f has the expression

$$f(\mathbf{x}) = b + \sum_{i=1}^{N} c_i R_\theta(\mathbf{x}_{i*}, \mathbf{x}) = b + \sum_{i=1}^{N} c_i \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha(\mathbf{x}_{i*}, \mathbf{x}). \tag{4.6}$$

Define $\boldsymbol{c} = (c_1, ..., c_N)^{\mathrm{T}} \in \mathcal{R}^N$. Let R_α^{**} be the matrix $\{R_\alpha(\mathbf{x}_{i*}, \mathbf{x}_{k*})\}_{i,k=1}^{N}$, and R_α^* the $n \times N$ matrix $\{R_\alpha(\mathbf{x}_i, \mathbf{x}_{k*})\}$, $i = 1, ..., n$ and $k = 1, ..., N$. Define $R_\theta^{**} = \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha^{**}$ and $R_\theta^* = \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha^*$. Then $\boldsymbol{f} = b\mathbf{1}_n + R_\theta^* \boldsymbol{c}$, and (4.2) becomes

$$\min_{\theta \geq \mathbf{0}, b, \boldsymbol{c}} \frac{1}{n} \sum_{i=1}^{n} w_i \big[z_i - f(\mathbf{x}_i)\big]^2 + \lambda_0 \boldsymbol{c}^{\mathrm{T}} R_\theta^{**} \boldsymbol{c} + \lambda \sum_{\alpha=1}^{p} \theta_\alpha. \tag{4.7}$$

1. With θ's fixed, (4.7) is equivalent to

$$\min_{b, \boldsymbol{c}} ||\boldsymbol{z}_w - b\boldsymbol{s}_w - R_{w\theta}^* \boldsymbol{c}||^2 + n\lambda_0 \boldsymbol{c}^{\mathrm{T}} R_\theta^{**} \boldsymbol{c}, \tag{4.8}$$

where $\boldsymbol{z}_w = W^{1/2}\boldsymbol{z}, \boldsymbol{s}_w = W^{1/2}\mathbf{1}_n$, and $R_{w\theta}^* = W^{1/2} R_\theta^*$. Letting $Q = (R_\theta^{**})^{-1}(R^*)_{w\theta}^{\mathrm{T}}$, the solution to (4.8) is

$$\begin{bmatrix} \boldsymbol{c} \\ b \end{bmatrix} = \begin{bmatrix} QR_{w\theta}^* + \lambda_0 I_N & Q\boldsymbol{s}_w \\ \boldsymbol{s}_w^{\mathrm{T}} R_{w\theta}^* & \boldsymbol{s}_w^{\mathrm{T}} \boldsymbol{s}_w \end{bmatrix}^{-1} \begin{bmatrix} Q\boldsymbol{z}_w \\ \boldsymbol{s}_w^{\mathrm{T}} \boldsymbol{z}_w \end{bmatrix}.$$

2. With $(b, \boldsymbol{c})$ fixed, (4.7) is equivalent to

$$\min_{b, \boldsymbol{c}} ||\boldsymbol{z}_w - b\boldsymbol{s}_w - \sum_{\alpha=1}^{p} \theta_\alpha R_{w\alpha}^* \boldsymbol{c})||^2 + n\lambda_0 \sum_{\alpha=1}^{p} \theta_\alpha (\boldsymbol{c}^{\mathrm{T}} R_\alpha^{**} \boldsymbol{c}) + n\lambda \sum_{\alpha=1}^{p} \theta_\alpha. \tag{4.9}$$

Let G_w be the matrix with the αth column $\boldsymbol{g}_{w\alpha} \equiv R_{w\alpha}^* \boldsymbol{c} = W^{1/2} R_\alpha^* \boldsymbol{c}$, and $\boldsymbol{J}_{\lambda_0}$ be the vector with the αth element being $n\lambda_0 \boldsymbol{c}^{\mathrm{T}} R_\alpha^{**} \boldsymbol{c}$, $\alpha = 1, ..., p$. Define $\boldsymbol{z}_{bw} = \boldsymbol{z}_w - b\boldsymbol{s}_w$, then (4.9) becomes

$$\min_{\theta} ||\boldsymbol{z}_{bw} - G_w \theta||^2 + \boldsymbol{J}_{\lambda_0}^{\mathrm{T}} \theta \quad \text{subject to} \quad \mathbf{1}^{\mathrm{T}} \theta \leq M, \quad \theta \geq \mathbf{0}. \tag{4.10}$$

Algorithm 2: In Algorithm 1, replace (4.3) by (4.8), and replace (4.5) by (4.10).

4.3. Smoothing Parameter Selection

Smoothing parameters balance the tradeoff between the likelihood fit and the penalty on function components. For an exponential family, the Kullback-Leibler (KL) distance between the distributions parameterized by (f, ϕ) and $(\hat{f}, \phi)$ is $\mathrm{KL}(f, \hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\big[\mu(\mathbf{x}_i)\{f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i)\} - \{B(f(\mathbf{x}_i)) - B(\hat{f}(\mathbf{x}_i))\}\big]/A(\phi)$. Comparative KL distance is obtained by dropping terms that do not involve $\hat{f}$,

$$\mathrm{CKL}(f, \hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\big[-\mu(\mathbf{x}_i)\hat{f}(\mathbf{x}_i) + B\{\hat{f}(\mathbf{x}_i)\}\big].$$

Since μ is generally unknown, we use k-fold cross validation (CV) to tune the parameters adaptively. We divide the data into k subsets of (approximately) equal size. We fit the penalized likelihood regression model k times, each time leaving out one of the subsets from fitting, but using only the omitted subset to compute the testing error. In our simulation, we use five-fold cross validation to choose the λ_0 and M. The grid search is applied. In the following is given the complete algorithm for the COSSO penalized likelihood estimation cooperated with parameter tuning.

1. Fix $\theta_\alpha = 1, \alpha = 1, ..., p$. Initialize f_i, μ_i, w_i, z_i and W in the same way as Algorithm 1 step 1. For each fixed λ_0, we repeat the following two steps.
   (a) Calculate $\boldsymbol{z}_w, \boldsymbol{s}_w, R_{w\theta}$ and solve (4.3) or (4.8) $(b, \boldsymbol{c}_w)$.
   (b) Compute $\boldsymbol{f} = b\mathbf{1}_n + R_\theta \boldsymbol{c}$ and the new weights w_i's. Go to (a) until the convergence criterion meets.
2. Choose the best λ_0 according to the cross validation (CV) score. Fix λ_0 at the chosen value in all later steps.
3. For each fixed M in a reasonable range, apply Algorithm 1 (or 2 for large datasets) with M. Choose the best M according to the CV score. The solution corresponding to this chosen M is the final solution.

6. Simulations

We study the empirical performances of the COSSO-type penalized likelihood method in Bernoulli examples. Given $\mathbf{x}$, the response Y takes value 1 with probability $p(\mathbf{x})$. To measure the model accuracy of estimating the conditional probabilities, we calculate the CKL distance between $\hat{p}(\mathbf{x})$ and $p(\mathbf{x})$. To measure the classification accuracy in terms of label prediction, we calculate the expected risk $\mathrm{ER} = \mathrm{E}_{\mathbf{X}}[(2Y-1) \neq \mathrm{sign}\{\hat{p}(\mathbf{X}) - \frac{1}{2}\}]$. The following principles are followed in all the examples.

- Simple random scheme is used to subsample the points for basis functions.
- The smoothing parameter is tuned using 5-fold cross validation.
- Empirical misclassification rate (EMR) is evaluated on 10000 testing points.
- Each example is repeated 100 times, The average CKL, EMR, model size, and time (spent in both parameter tuning and model fitting) are reported.

Example 1: Independent continuous covariates Consider a simple additive model with ten continuous covariates, independently generated from Uniform$[0, 1]$. The sample size $n = 250$. The true conditional logit function

$$f(\mathbf{x}) = 3x^{(1)} + \pi sin(\pi x^{(2)}) + 8(x^{(3)})^5 + \frac{2}{(e-1)} e^{x^{(4)}} - 6,$$

thus $X^{(5)}, ..., X^{(10)}$ are uninformative. We fit the additive COSSO model first with the full basis algorithm. Figure 6.1 plots the true and estimated function components. Notice the components are centered according to the ANOVA decomposition. Four important variables are selected correctly, and the penalized likelihood method provides very good estimates for the true probability function components.

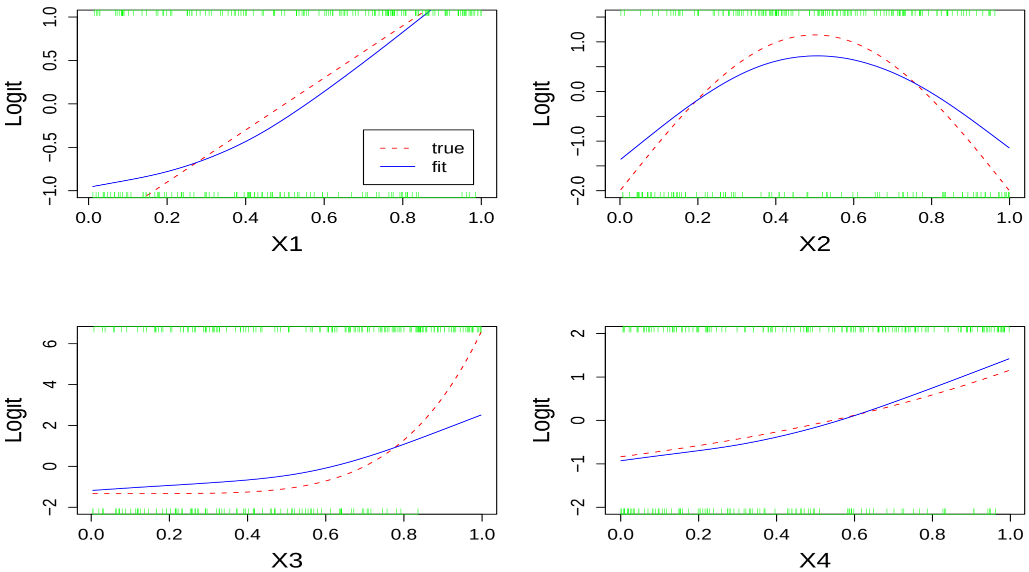

Figure 6.1: The estimated component functions (dashed lines) and the true component functions (solid lines) in one run of Example 1. Full basis is used. Shown are the components for variables $X^{(1)}, X^{(2)}, X^{(3)}$, and $X^{(4)}$. For the other variables, both the true and estimated component functions are zero.

We then fit the additive model using the subset basis algorithm with different numbers of basis: $N = 25, 50, 100$. Each experiment is repeated 100 times, and the results are summarized in Tables 1 and 2. The models obtained from the subset and full basis algorithms have the same accuracy as shown by their average CKL and EMR values. Table 2 presents the number of times each variable appears in the 100 final models, and the average model size with the standard error of the model size in the parentheses. The subset and full basis algorithms have similar results, and particularly, the algorithm with $N = 100$ gives identical frequencies for all the ten variables as the full basis algorithm. The COSSO penalized likelihood method almost always selects $X^{(1)}, X^{(2)}, X^{(3)}$ (in more than 99% runs), and selects $X^{(4)}$ in more than 92% runs. Figure 6.2 plots the 5th, 50th, 95th best functional estimates chosen by the EMR with $N = 100$. We conclude that the subset basis algorithm performs equally well as the full basis algorithm in both variable selection and functional estimation.

Table 1: The average CKL and EMR of 100 final models (Example 1).

N (number of basis)	CKL	EMR
25	0.476 (0.018)	0.235 (0.018)
50	0.477 (0.017)	0.236 (0.016)
100	0.478 (0.017)	0.235 (0.016)
200 (full)	0.477 (0.017)	0.236 (0.016)

Table 2: The frequency of appearance of the variables and the average model size for 100 final models, with different basis functions. (Example 1)

N	1	2	3	4	5	6	7	8	9	10	model size
25	99	99	100	91	10	8	11	9	13	14	4.54 (1.36)
50	100	100	100	92	14	10	16	12	13	16	4.73 (1.40)
100	100	100	100	93	12	10	17	12	14	17	4.75 (1.42)
full	100	100	100	93	12	10	17	12	14	17	4.54 (1.42)

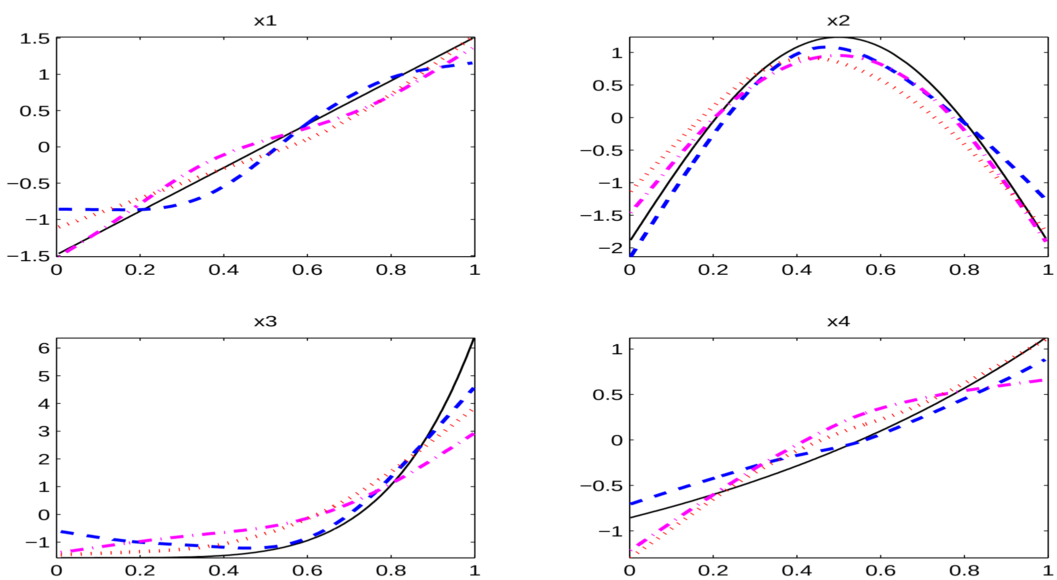

Figure 6.2: The estimated component functions when $n = 250$ and $N = 100$. The black solid lines are the true components. The blue dashed lines are the 5th best; the magenta dash-dot lines are the 50th best; the red dotted lines indicate the 95th best.

Example 2: Correlated continuous covariates Consider the situation when some degree of correlation exists among the covariates. The following four functions on $[0,1]$ are used as building blocks of regression functions:

$$g_1(t) = t; \quad g_2(t) = (2t-1)^2; \quad g_3(t) = \frac{\sin(2\pi t)}{2 - \sin(2\pi t)};$$

$$g_4(t) = 0.1\sin(2\pi t) + 0.2\cos(2\pi t) + 0.3\sin^2(2\pi t) + 0.4\cos^3(2\pi t) + 0.5\sin^3(2\pi t).$$

The true model is $f(\mathbf{x}) = 5g_1(x^{(1)}) + 3g_2(x^{(2)}) + 4g_3(x^{(3)}) + 6g_4(x^{(4)})$, thus $X^{(5)}, ..., X^{(10)}$ are uninformative. Consider two types of covariance structure:

(trimmed) AR(1): Generate $W_1, ..., W_{10}$ independently from $N(0,1)$. Let $X^{(1)} = W_1, X^{(j)} = \rho X^{(j-1)} + (1-\rho^2)^{1/2} W_j, j \geq 2$. Trim X's into [-2.5, 2.5].

Compound Symmetry (CS): Generate $W_1, ..., W_{10}$ and U independently from Uniform$[0,1]$. For any $t \geq 0$, define $X^{(j)} = (W_j + tU)/(1+t)$. Then $Corr(X^{(j)}, X^{(k)}) = t^2/(1+t^2)$ for $j \neq k$.

Various experiments are conducted in the combination setting of three sample sizes ($n = 100, 200, 500$) and three degrees of correlation (low,moderate,high).

1. zero correlation: (trimmed) AR(1) with $\rho = 0$

Table 3 shows that as n increases from $100, 200$ to 500, the CKL decreases from $0.46, 0.37$ to 0.33, and the EMR decreases from $0.22, 0.17$ to 0.15. In the last column is the average computation time, which is the total spent on parameter tuning time and model fitting time. In all the settings, the subset basis algorithm performs as well as the full algorithm, but can be much faster.

Table 3: The average CKL, EMR, and computation time for 100 final models, with AR(1) covariance structure $\rho = 0$ (Example 2)

n	N	CKL	EMR	Time
100	50	0.462 (0.049)	0.221 (0.033)	24.8
	100	0.463 (0.049)	0.221 (0.032)	64.5
200	50	0.371 (0.029)	0.172 (0.020)	57.3
	100	0.371 (0.026)	0.172 (0.019)	185.9
	200	0.371 (0.027)	0.172 (0.020)	589.0
500	50	0.325 (0.017)	0.148 (0.014)	274.3
	100	0.326 (0.016)	0.148 (0.014)	697.6
	200	0.327 (0.016)	0.148 (0.014)	929.2
	500	0.328 (0.013)	0.151 (0.007)	1179.1

Table 4 shows as n increases, the selection of important variables becomes more accurate. When $n = 100$ or 200, $X^{(2)}$ is selected in fewer than 40% of runs; however when $n = 500$, it is selected in average 80% of runs. Furthermore, when n is small, the model tends to retain some unimportant variables. As n increases, the appearance frequency of unimportant variables decreases quickly, and the average model size is closer to the true model size 4.

Table 4: The frequency of appearance of the variables and the average model size for the 100 final models, with AR(1) covariance structure $\rho = 0$ (Example 2)

n	N	1	2	3	4	5	6	7	8	9	10	model size
100	50	86	32	99	95	32	30	29	30	31	29	4.93 (2.07)
	100	87	33	99	95	35	33	30	31	32	30	5.05 (2.13)
200	50	93	40	100	100	10	14	13	7	12	12	4.01 (1.36)
	100	95	39	100	100	10	14	12	10	12	14	4.06 (1.43)
	200	95	39	100	100	10	14	12	10	12	16	4.08 (1.45)
500	50	100	85	100	100	6	8	6	8	7	3	4.23 (1.06)
	100	100	84	100	100	6	7	7	7	7	3	4.21 (1.05)
	200	100	83	100	100	6	7	7	7	7	3	4.20 (1.05)
	500	100	77	100	100	7	7	3	3	9	6	4.12 (0.99)

2. moderate correlation: (trimmed) AR(1) with $\rho = 0.5$, and CS with $t = 1$

The COSSO penalized likelihood method performs slightly worse than in the independent situation, it still does a very good job in functional estimation and variable selection. For example, when $n = 500$, the important variables $X^{(1)}, X^{(3)}$ and $X^{(4)}$ are never missed, and $X^{(2)}$ is chosen in average 80% runs. The subset basis algorithm performs as well as the full algorithm.

Table 5: The average CKL, EMR, and computation time of the 100 final models, with AR(1) covariance structure $\rho = 0.5$ (Example 2)

n	N	CKL	EMR	Time
100	50	0.477 (0.054)	0.239 (0.035)	25.4
	100	0.466 (0.053)	0.231 (0.033)	95.9
200	50	0.477 (0.054)	0.239 (0.035)	79.1
	100	0.392 (0.027)	0.186 (0.018)	189.1
	200	0.392 (0.027)	0.186 (0.018)	744.8
500	50	0.351 (0.014)	0.165 (0.014)	331.3
	100	0.352 (0.013)	0.165 (0.013)	786.2
	200	0.352 (0.013)	0.165 (0.013)	2027.5
	500	0.351 (0.011)	0.165 (0.006)	> 5000

Table 6: The frequency of appearance of the variables and the average model size for 100 final models, with AR(1) covariance structure $\rho = 0.5$ (Example 2)

n	N	1	2	3	4	5	6	7	8	9	10	model size
100	50	82	45	100	90	38	27	31	33	22	27	4.95 (2.28)
	100	88	41	100	90	30	31	21	32	29	35	4.97 (2.10)
200	50	97	49	100	100	15	16	15	16	18	11	4.37 (1.69)
	100	99	48	100	100	14	18	15	18	17	12	4.40 (1.73)
	200	98	48	100	100	14	18	15	18	17	12	4.40 (1.73)
500	50	100	79	100	100	7	7	6	10	8	10	4.27 (1.25)
	100	100	81	100	100	4	10	7	11	6	10	4.29 (1.13)
	200	100	79	100	100	4	10	7	11	6	10	4.27 (1.14)
	500	100	78	100	100	12	4	11	5	9	7	4.26 (1.25)

Table 7: The average CKL, EMR, and computation time of 100 final models, with CS covariance structure t=1 (Example 2)

n	N	CKL	EMR	Time
100	50	0.468 (0.050)	0.222 (0.023)	35.0
	100	0.457 (0.049)	0.216 (0.026)	101.6
200	50	0.379 (0.022)	0.180 (0.017)	77.1
	100	0.380 (0.022)	0.180 (0.017)	197.4
	200	0.381 (0.023)	0.199 (0.016)	119.1
500	50	0.339 (0.012)	0.158 (0.007)	335.3
	100	0.337 (0.010)	0.158 (0.007)	739.1
	200	0.339 (0.012)	0.158 (0.007)	2533.8
	500	0.339 (0.012)	0.158 (0.007)	> 5000

Table 8: The frequency of appearance of the variables and the average model size in the 100 finals models, with CS covariance structure t=1 (Example 2)

n	N	1	2	3	4	5	6	7	8	9	10	model size
100	50	74	38	100	81	38	33	30	33	34	35	4.96 (2.51)
	100	78	45	96	92	34	36	30	33	40	32	5.16 (2.49)
200	50	84	35	100	100	8	17	12	13	15	13	3.97 (1.49)
	100	83	37	100	100	11	18	14	12	17	14	4.06 (1.57)
	200	83	37	100	100	11	19	14	12	17	14	4.07 (1.57)
500	50	100	77	100	100	14	9	12	7	11	14	4.44 (1.43)
	100	100	71	100	100	6	7	6	7	5	6	4.08 (0.94)
	200	100	75	100	100	14	9	10	8	11	16	4.43 (1.34)
	500	100	75	100	100	14	9	10	8	11	15	4.42 (1.34)

3. high correlation: CS with $t = 3$

The correlation among the covariates is very high, with $Corr(X^{(j)}, X^{(k)}) = 0.9$ for any $j \neq k$. The model accuracies of the COSSO estimates are comparable to those obtained in low or moderate correlation situations. The COSSO model tends to choose more variables due to the existence of high correlation. However, when n gets larger, it selects important variables more correctly. The subset basis algorithm works almost same as the full algorithm.

Table 9: The average CKL, EMR, and computation time of the 100 final models, with CS covariance structure $t = 3$ (Example 2).

n	N	CKL	EMR	Time
100	50	0.397 (0.034)	0.196 (0.014)	41.4
	100	0.392 (0.036)	0.192 (0.016)	94.5
200	50	0.348 (0.018)	0.176 (0.013)	92.6
	100	0.348 (0.015)	0.176 (0.013)	222.8
	200	0.348 (0.020)	0.174 (0.016)	690.6
500	50	0.318 (0.008)	0.157 (0.007)	255.0
	100	0.317 (0.014)	0.157 (0.014)	710.5
	200	0.318 (0.008)	0.158 (0.008)	1056.8
	500	0.318 (0.008)	0.158 (0.008)	1468.6

Table 10: The frequency of appearance of the variables and the average model size for the 100 final models, with CS covariance structure t=3 (Example 2)

n	N	1	2	3	4	5	6	7	8	9	10	model size
100	50	72	79	94	96	61	65	62	56	57	55	6.97 (2.28)
	100	75	75	96	96	61	68	63	59	57	61	7.11 (2.35)
200	50	57	60	96	100	35	37	33	28	30	32	5.08 (2.20)
	100	53	58	96	100	32	40	34	33	29	33	5.08 (2.18)
	200	70	51	98	100	32	37	40	34	37	40	5.39 (2.32)
500	50	69	58	100	100	14	14	20	11	13	17	4.16 (1.70)
	100	78	60	100	100	11	19	16	19	15	13	4.31 (1.70)
	200	68	61	100	100	19	17	17	15	15	20	4.32 (1.72)
	500	68	61	100	100	19	17	17	15	15	20	4.32 (1.72)

Example 3: Categorical Covariates Included Generate eleven variables from Uniform$[0, 1]$, and then make the last four variables be categorical by setting

$$X^{(8)} = I(X^{(8)} < 0.5) + 2I(X^{(8)} \geq 0.5)$$
$$X^{(9)} = I(X^{(9)} < \tfrac{1}{3}) + 2I(\tfrac{1}{3} \leq X^{(9)} < \tfrac{2}{3}) + 3I(X^{(9)} > \tfrac{2}{3})$$
$$X^{(10)} = I(X^{(10)} < \tfrac{1}{4}) + 2I(\tfrac{1}{4} \leq X^{(9)} < \tfrac{2}{4}) + 3I(\tfrac{2}{4} \leq X^{(9)} < \tfrac{3}{4}) + 4I(X^{(9)} \geq \tfrac{3}{4})$$
$$X^{(11)} = I(X^{(11)} < \tfrac{1}{3}) + 2I(\tfrac{1}{3} \leq X^{(11)} < \tfrac{2}{3}) + 3I(X^{(11)} \geq \tfrac{2}{3})$$

We use the same building blocks of regression functions as in Example 2. The underlying logit function is

$$f(\mathbf{x}) = 5g_1(x^{(1)}) + 3g_2(x^{(2)}) + 4g_3(x^{(3)}) + 6g_4(x^{(4)}) - 4.5x^{(8)} + 2.5\sqrt{x^{(10)}} - 2.4.$$

Among the important variables, $X^{(8)}$ and $X^{(10)}$ are categorical. Assume the CS covariance structure: $\mathrm{corr}(X^{(j)}, X^{(k)}) = t^2/(1+t^2)$ for $j \neq k$, with $t = 0, 0.5, 1$. The additive COSSO model is fitted.

1. zero correlation CS structure with $t = 0$

The correct model size is 6. Figure 6.3 plots the estimated function components with the true function components when $n = 200, N = 100$.

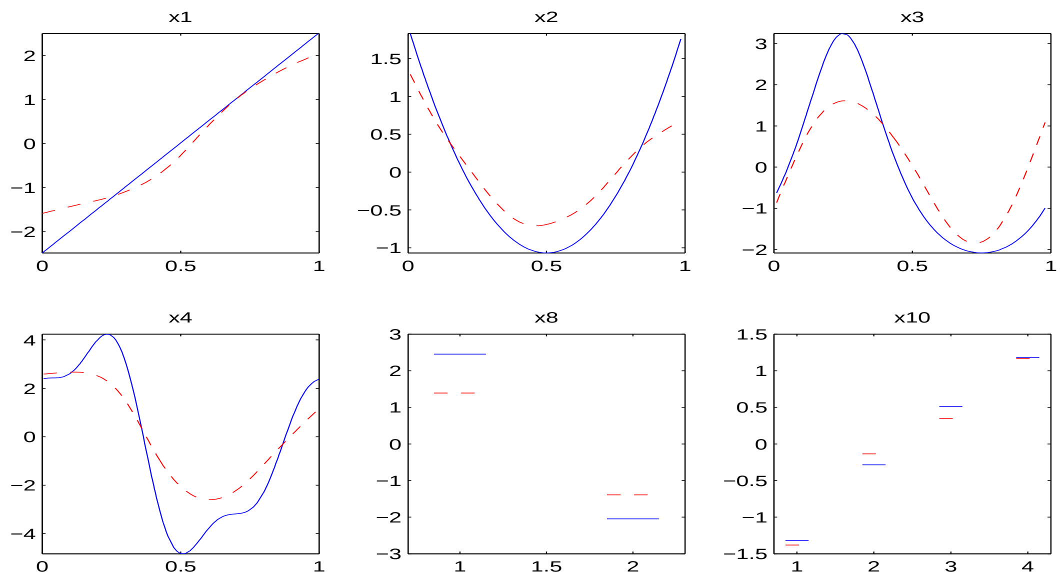

Figure 6.3: The estimated component functions (dashed line) and the true component functions (solid line) in one run of Example 3. $n = 200, N = 100$. Shown are the components for variable $X^{(1)}, ..., X^{(4)}, X^{(8)}$, and $X^{(10)}$. For the other variables, both the true and estimated component functions are zero.

Figure 6.4 plots the 5th, 50th, 95th best estimates for 100 runs with $N = 100$. Table 12 shows that $X^{(8)}$ is never missed in any setting. $X^{(10)}$ is selected in about 80% runs when $n = 200$, and selected in at least 98% runs when $n = 500$.

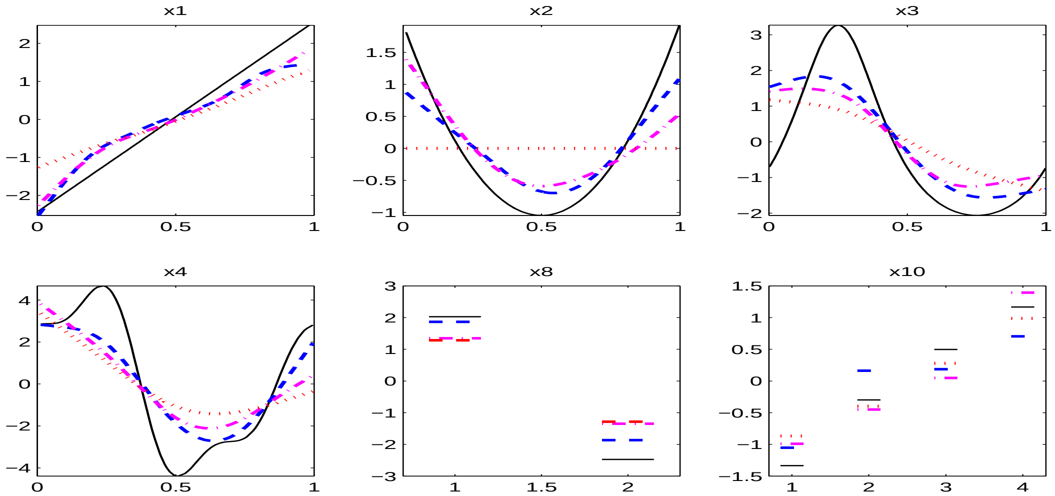

Figure 6.4: The COSSO estimates for $n = 200, N = 100$. The black solid lines are the true components. The blue dashed lines are 5th best; the magenta dash-dot lines 50th best; the red dotted lines are 95th best. Note the 95th best run misses variable $X^{(2)}$.

Table 11: The average CKL, EMR, and computation times of 100 final models, with CS covariance structure $t = 0$ (Example 3).

n	N	CKL	EMR	Time
100	100	0.536 (0.080)	0.265 (0.047)	75.5
200	100	0.351 (0.032)	0.160 (0.016)	207.7
	200	0.351 (0.033)	0.159 (0.017)	825.8
500	100	0.284 (0.009)	0.125 (0.006)	871.6
	200	0.284 (0.009)	0.125 (0.006)	2549.9
	500	0.284 (0.009)	0.125 (0.006)	> 5000

Table 12: The frequency of appearance of the variables and the average model size for the 100 final models, with CS covariance structure $t = 0$ (Example 3)

n	N	1	2	3	4	5	6	7	8	9	10	11	model size
100	100	83	34	78	93	15	18	18	100	41	73	37	5.90 (2.37)
200	100	100	64	100	100	11	18	10	100	8	82	9	6.02 (1.41)
	200	100	73	99	100	7	16	16	100	8	79	5	6.03 (1.34)
500	100	100	100	100	100	19	10	13	100	2	98	1	6.43 (0.74)
	200	100	100	100	100	13	10	9	100	1	100	1	6.35 (0.64)
	500	100	100	100	100	19	11	13	100	2	98	1	6.44 (0.77)

2. moderate correlation: CS structure with $t = 1$

$Corr(X^{(j)}, X^{(k)}) = 0.5$ for $j \neq k$. Tables 13 and 14 show that the additive COSSO performs slightly worse than the independent case in terms of model accuracy and variable selection. However when n is large enough, the CPL selects the correct model structure more correctly. The subset basis algorithm works as effectively as the full basis algorithm.

Table 13: The average CKL, EMR, and computation times of 100 final models, with CS covariance structure $t = 1$ (Example 3)

n	N	CKL	EMR	Time
100	100	0.486 (0.081)	0.231 (0.045)	109.6
200	100	0.367 (0.025)	0.165 (0.014)	253.5
	200	0.367 (0.026)	0.165 (0.014)	601.0
500	100	0.306 (0.013)	0.134 (0.007)	755.6
	200	0.306 (0.013)	0.134 (0.008)	2521.3
	500	0.306 (0.014)	0.134 (0.008)	> 5000

Table 14: The frequency of appearance of the variables and the average model size for the 100 final models, with CS covariance structure $t = 1$ (Example 3)

n	N	1	2	3	4	5	6	7	8	9	10	11	model size
100	100	43	31	85	97	19	22	19	100	32	72	30	5.50 (2.40)
200	100	67	51	99	100	18	14	9	100	8	80	15	5.61 (1.74)
	200	67	51	99	100	18	13	10	100	8	80	14	5.60 (1.75)
500	100	97	83	100	100	15	12	11	100	4	98	5	6.25 (1.09)
	200	98	81	100	100	12	13	10	100	2	100	3	6.19 (1.00)
	500	98	82	100	100	12	13	10	100	2	100	3	6.20 (0.99)

Example 4: Two-way interaction model There are $d = 4$ continuous covariates, independently distributed from Uniform$[0,1]$. The true logit function f is

$$f(\mathbf{x}) = 4x^{(1)} + \pi sin\big(\pi x^{(1)}\big) + 6x^{(2)} - 8\big(x^{(2)}\big)^3 + 3\cos\big(2\pi(x^{(1)} - x^{(2)})\big) - 5.$$

This is a two-factor interaction model, with the important components including $X^{(1)}, X^{(2)}$ and their interaction effect. Consider two settings: $n = 200$ and $n = 500$. We apply the two-way interaction CPL model with the full basis algorithm, and the subset basis algorithm with $N = 50, 100$. Table 15 shows the model accuracy of the COSSO estimate is getting better when n becomes larger. Table 16 shows the important interaction term f_{12} is selected in all the experiments. The subset basis algorithm performs almost the same as the full basis algorithm, but with much less computation time.

Table 15: The average CKL, EMR, and computation times of 100 final models (Example 4)

n	N	CKL	EMR	Time
200	50	0.515 (0.005)	0.247 (0.004	30.59
	100	0.515 (0.005)	0.247 (0.004)	57.59
	200	0.514 (0.004)	0.248 (0.003)	99.71
500	50	0.448 (0.003)	0.192 (0.002)	187.30
500	100	0.444 (0.008)	0.191 (0.007)	384.72

Table 16: The frequency of appearance of the variables and the average model size in the 100 final models (Example 4)

n	N	1	2	3	4	5	6	7	8	9	10	model size
200	50	100	99	19	18	100	14	13	10	18	7	3.98 (1.56)
	100	100	99	19	18	100	14	13	10	18	7	3.98 (1.56)
	200	100	100	19	17	100	15	13	13	20	9	4.06 (1.64)
500	50	100	100	11	13	100	12	6	9	8	1	3.60 (0.74)
500	100	100	100	13	15	100	15	11	10	12	5	3.81 (0.92)

7. Real Examples

We apply the COSSO-type penalized likelihood method to four benchmark datasets, which are all available at the UCI machine learning repository. Both the additive and two-way interaction COSSO are applied to each dataset. If $n \leq 200$, we use the full basis algorithm; otherwise we use the subset basis algorithm with $N = 200$. The tuning parameter is chosen by five-fold cross validation.

Cleveland Heart Disease. This dataset was contributed by R. Detrano from Cleveland Clinic Foundation. The problem concerns the prediction of the presence or absence of heart disease given various medical tests. There are 296 instances with seven categorical covariates and six continuous covariates.

BUPA liver Disorder. This dataset was donated by R. S. Forsyth from BUPA Medical Research Ltd. The problem is to predict whether or not a male patient has a liver disorder based on blood tests and alcohol consumption. There are 345 observations with six continuous variables.

PIMA Indian Diabetes. This dataset was provided by V. Sigillito. The patients are females older than twenty-one years from Pima Indian heritage, Arizona. The problem is to predict the positive test for diabetes given a number of physiological measurements and medical test results. There are eight variables. Since the original data contains some impossible values (like zero body mass index), we remove those instances and consider the remaining 532 observations.

Wisconsin Breast Cancer. This breast cancer database was donated by Dr. William H. Wolberg at the University of Wisconsin Hospitals, Madison. The problem is to predict whether a tissue sample taken from a patient is malignant or benign. There are 699 observations with some missing values, thus our results are based on 683 complete records. Nine features are clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, and so on. Mangasarian and Wolberg (1990) applied the linear programming on this dataset.

We shall compare the CPL with other classification methods in terms of the EMR, which is calculated using the ten-fold cross validation. Lim and Loh (2000) conducted a thorough comparison among twenty-two decision tree, nine statistical, and two neural network algorithms on these datasets. Table 17 presents the EMRs of the additive COSSO, two-way interaction COSSO, and the best rule of the thirty-three algorithms reported in Lim and Loh (2000),which is referred as “best(LL2000)” from now on. For the BUPA dataset, the additive COSSO is best, the next is the best(LL2000) approach, and the two-way interaction COSSO is slightly worse than the best(LL2000). For the PIMA and WBC datasets, the two-way CPL works best; the additive COSSO and the best(LL2000) work equally well. For the Cleveland dataset, both COSSO estimators are worse than best(LL2000). For model selection, we summarize the average model size of the additive and two-way interaction COSSO models.

Table 17: EMRs of the additive COSSO, two-way interaction COSSO, and the best(LL2000).

	additive COSSO	two-way COSSO	best(LL2000)
BUPA	0.257 (0.022)	0.286 (0.023)	0.28
Cleveland	0.169 (0.022)	0.165 (0.018)	0.14
PIMA	0.220 (0.014)	0.214 (0.014)	0.22
WBC	0.031 (0.008)	0.026 (0.005)	0.03

Table 18: Average model size of the additive and two-way interaction COSSO models

	BUPA	Cleveland	PIMA	WBC
additive COSSO	6.0 (0.03)	9.4 (2.95)	6.0 (0.94)	8.8 (0.63)
two-way COSSO	12.2 (4.9)	10.0 (3.43)	6.6 (2.16)	14.6 (1.27)

Acknowledgment

The authors thank Grace Wahba for helpful discussions. Hao Helen Zhang's research was partially supported by Faculty Research and Professional Development Fund, North Carolina State University. Yi Lin's work was supported by National Science Foundation grant DMS-0134987.

Appendix 1
Proofs

Proof of Theorem 1. Denote the functional to be minimized in (2.2) by $A(f)$, then $A(f)$ is convex and continuous. Without loss of generality, we assume $\tau = 1$.

Because $\sum_{\alpha=1}^{p} \|P^\alpha f\|^2 \le J^2(f) \le p \sum_{\alpha=1}^{p} \|P^\alpha f\|^2$, we have that $J(f) \ge \|f\|$, for any $f \in \mathcal{F}_1$. Let $R_{\mathcal{F}_1}$ be the reproducing kernel of $\mathcal{F}_1$ and $\langle \cdot, \cdot \rangle_{\mathcal{F}_1}$ be the inner product in $\mathcal{F}_1$. Denote $a = \max_{i=1}^{n} R_{\mathcal{F}_1}^{1/2}(\mathbf{x}_i, \mathbf{x}_i)$. By the definition of reproducing kernel, we have for any $f \in \mathcal{F}_1$ and $i = 1, ..., n$,

$$\begin{aligned} |f(\mathbf{x}_i)| &= |\langle f(\cdot), R_{\mathcal{F}_1}(\mathbf{x}_i, \cdot)\rangle_{\mathcal{F}_1}| \le \|f\| \langle R_{\mathcal{F}_1}(\mathbf{x}_i, \cdot), R_{\mathcal{F}_1}(\mathbf{x}_i, \cdot)\rangle_{\mathcal{F}_1}^{1/2} \\ &= \|f\| R_{\mathcal{F}_1}^{1/2}(\mathbf{x}_i, \mathbf{x}_i) \le a\|f\| \le aJ(f). \end{aligned} \tag{A1}$$

Denote $\rho = \max_{i=1}^n (y_i^2 + |y_i| + 1)$. Consider the set

$$D = \{f \in \mathcal{F} : f = b + f_1, \text{with } b \in \{1\}, f_1 \in \mathcal{F}_1, J(f) \le \rho, |b| \le \rho^{1/2} + (a+1)\rho\}.$$

Then D is a closed, convex, and bounded set. Therefore by Theorem 4 of Tapia and Thompson (1978, page 162), there exists a minimizer of (2.2) in D. Denote the minimizer by $\bar{f}$. Then $A(\bar{f}) \le A(0) < \rho$.

On the other hand, for any $f \in \mathcal{F}$ with $J(f) > \rho$, clearly $A(f) \ge J(f) > \rho$; for any $f \in \mathcal{F}$ with $J(f) \le \rho$, $f = b + f_1$, $b \in \{1\}$, $f_1 \in \mathcal{F}$, and $|b| > \rho^{1/2} + (a+1)\rho$, we use (A1) to get that, for any $i = 1, ..., n$,

$$|b + f_1(\mathbf{x}_i) - y_i| > [\rho^{1/2} + (a+1)\rho] - a\rho - \rho = \rho^{1/2}.$$

Therefore we have $A(f) > \rho$. Hence for any $f \notin D$, we have $A(f) > A(\bar{f})$. Therefore $\bar{f}$ is a minimizer of (2.2) in $\mathcal{F}$.

Appendix 2
Derivation for Algorithms

1. Full basis algorithm

   For fixed θ's

$$\begin{aligned} & (\boldsymbol{z} - R_\theta \boldsymbol{c} - b\mathbf{1}_n)' W (\boldsymbol{z} - R_\theta \boldsymbol{c} - b\mathbf{1}_n) + n\lambda_0 \boldsymbol{c}' R_\theta \boldsymbol{c} \\ = \; & (W^{1/2}\boldsymbol{z} - W^{1/2} R_\theta \boldsymbol{c} - W^{1/2} b\mathbf{1}_n)'(W^{1/2}\boldsymbol{z} - W^{1/2} R_\theta \boldsymbol{c} - W^{1/2} b\mathbf{1}_n) + n\lambda_0 \boldsymbol{c}' R_\theta \boldsymbol{c} \\ = \; & (\boldsymbol{z}_w - R_{w\theta} W^{-1/2}\boldsymbol{c} - b\mathbf{1}_n)'(\boldsymbol{z}_w - R_{w\theta} W^{-1/2}\boldsymbol{c} - b\mathbf{1}_n) \\ & + \lambda_0 \boldsymbol{c}' W^{-1/2} W^{1/2} R_\theta W^{1/2} W^{-1/2} \boldsymbol{c} \\ = \; & (\boldsymbol{z}_w - R_{w\theta}\boldsymbol{c}_w - b\mathbf{1}_n)(\boldsymbol{z}_w - R_{w\theta}\boldsymbol{c}_w - b\mathbf{1}_n) + n\lambda_0 \boldsymbol{c}_w' R_{w\theta} \boldsymbol{c}_w, \end{aligned}$$

   The solutions $\boldsymbol{c}_w$ and b are given by

$$\begin{bmatrix} \boldsymbol{c}_w \\ b \end{bmatrix} = \begin{bmatrix} R_{w\theta} + n\lambda_0 I_n & \boldsymbol{s}_w \\ \boldsymbol{s}_w' & 0 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{z}_w \\ 0 \end{bmatrix},$$

and $\boldsymbol{c} = W^{1/2}\boldsymbol{c}_w$.

For fixed $\boldsymbol{c}, b$

$$
\begin{aligned}
& (\boldsymbol{z} - \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha \boldsymbol{c} - b\mathbf{1}_n)' W (\boldsymbol{z} - \sum_{\alpha=1}^{p} \theta_\alpha R_\alpha \boldsymbol{c} - b\mathbf{1}_n) + n\lambda_0 \sum_{\alpha=1}^{p} \theta_\alpha (\boldsymbol{c}' R_\alpha \boldsymbol{c}) \\
= \ & (\boldsymbol{z} - G\theta - b\mathbf{1}_n)' W (\boldsymbol{z} - G\theta - b\mathbf{1}_n) + n\lambda_0 c' G\theta \\
= \ & ||\boldsymbol{z}_w - G_w\theta - b\boldsymbol{s}_w||^2 + n\lambda_0 \boldsymbol{c}_w' G_w \theta,
\end{aligned}
$$

where we define $G_w = W^{1/2}G$ in the last equation. The following nonnegative garrote problem need to be solved:

$$
\begin{aligned}
\min_{\theta} \quad & ||\boldsymbol{u}_w - G_w\theta||^2 \\
\text{subject to} \quad & \textstyle\sum_{\alpha=1}^{p} \theta_\alpha \le M \quad \text{and} \quad \theta_\alpha \ge 0, \alpha = 1, ..., p
\end{aligned}
$$

with $\boldsymbol{u}_w = \boldsymbol{z}_w - \frac{n}{2}\lambda_0 \boldsymbol{c}_w - b\boldsymbol{s}_w$

2. Subset basis algorithm

For fixed θ's,

$$
\begin{aligned}
& (\boldsymbol{z} - R_\theta^* \boldsymbol{c} - b\mathbf{1}_n)' (W^{1/2})' W^{1/2} (\boldsymbol{z} - R_\theta^* \boldsymbol{c} - b\mathbf{1}_n) + \lambda_0 \boldsymbol{c}' R_\theta^{**} \boldsymbol{c} \\
= \ & (W^{1/2}\boldsymbol{z} - W^{1/2} R_\theta^* \boldsymbol{c} - W^{1/2} b\mathbf{1}_n)' (W^{1/2}\boldsymbol{z} - W^{1/2} R_\theta^* \boldsymbol{c} - W^{1/2} b\mathbf{1}_n) + \lambda_0 \boldsymbol{c}' R_\theta^{**} \boldsymbol{c} \\
= \ & ||\boldsymbol{z}_w - R_{w\theta}^* \boldsymbol{c} - b\boldsymbol{s}_w||^2 + \lambda_0 \boldsymbol{c}' R_\theta^{**} \boldsymbol{c}
\end{aligned}
$$

The solutions $\boldsymbol{c}$ and b are given by

$$
\begin{bmatrix} \boldsymbol{c} \\ b \end{bmatrix} = \begin{bmatrix} R_{w\theta}^{*'} R_{w\theta}^* + \lambda_0 R_\theta^{**} & R_{w\theta}^{*'} \boldsymbol{s}_w \\ \boldsymbol{s}_w' R_{w\theta}^* & \boldsymbol{s}_w^{\mathrm{T}} \boldsymbol{s}_w \end{bmatrix}^{-1} \begin{bmatrix} R_{w\theta}^{*'} \boldsymbol{z}_w \\ s_w' \boldsymbol{z}_w \end{bmatrix}. \qquad \text{(A2)}
$$

Alternatively, if we define $Q = (R_\theta^{**})^{-1} R_{w\theta}^{*'}$, then

$$
\begin{bmatrix} \boldsymbol{c} \\ b \end{bmatrix} = \begin{bmatrix} Q R_{w\theta}^* + \lambda_0 I_N & Q\boldsymbol{s}_w \\ \boldsymbol{s}_w' R_{w\theta}^* & \boldsymbol{s}_w' \boldsymbol{s}_w \end{bmatrix}^{-1} \begin{bmatrix} Q\boldsymbol{z}_w \\ \boldsymbol{s}_w' \boldsymbol{z}_w \end{bmatrix}. \qquad \text{(A3)}
$$

When λ_0 is very small, (A3) has a better condition than (A2), However when θ's are very small, $(R_\theta^{**})^{-1}$ tends to be ill-posed, and (A2) has a better condition than (A3).

Bibliography

BREIMAN, L. (1995). Better subset selection using the nonnegative garrotte. *Technometrics* **37** 373–384.

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

EFRON, B., HASIE, T., JOHNSTONE, I. and TIBSHIRANI, R. (2004). Least angle regression. *Journal of American Statistical Association* **32** 1–24.

FAN, J. and LI, R. Z. (2001). Variable selection via penalized likelihood. *J. Am. Statist. Assoc.* **96** 1348–1360.

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

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

GU, C. (2002). *Smoothing Spline ANOVA Models*. Springer-Verlag.

GU, C. and KIM, Y. J. (2001). Penalized likelihood regression: General formulation and efficient approximation To appear in *Canadian Journal of Statistics*.

HASTIE, T. (1989). Discussion of "flexible parsimonious smoothing and additive modelling" by j. friedman and b. silverman. *Techometrics* **31** 3–39.

HASTIE, T. and TIBSHIRANI, R. (1990). *Generalized Additive Models*. Chapman and Hall.

KIMELDORF, G. and WAHBA, G. (1971). Some results on Tchebycheffian spline functions. *J. Math. Anal. Applic.* **33** 82–85.

Lim, T. and Loh, W. Y. (2000). A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning* **40** 203–229.

Lin, X., Wahba, G., Xiang, D., Gao, F., Klein, R. and Klein, B. (2000). Smoothing spline ANOVA models for large data sets with Bernoulli observations and the randomized GACV. *Ann. Statist.* **28** 1570–1600.

Lin, Y. and Zhang, H. H. (2002). Component selection and smoothing in smoothing spline analysis of variance models. Tech. Rep. 1072, University of Wisconsin – Madison.

Mangasarian, O. L. and Wolberg, W. H. (1990). Cancer diagnosis via linear programming. *SIAM News* **23-5** 1–18.

Ruppert, D. and Carroll, R. (2000). Spatially-adaptive penalties for spline fitting. *Australian and New Zealand Journal of Statistics* **45** 205–223.

Tibshirani, R. J. (1996). Regression shrinkage and selection via the lasso. *Journal of Royal Statistical Society, B* **58** 267–288.

Wahba, G. (1990). *Spline Models for Observational Data*, vol. 59. SIAM. CBMS-NSF Regional Conference Series in Applied Mathematics.

Wahba, G., Wang, Y., Gu, C., Klein, R. and Klein, B. (1995). Smoothing spline ANOVA for exponential families, with application to the WESDR. *Ann. Statist.* **23** 1865–1895.

Xiang, D. and Wahba, G. (1998). Approximate smoothing spline methods for large data sets in the binary case. *Proceedings of ASA Joint Statistical Meetings, Biometrics Section* 94–98.

Yau, P., Kohn, R. and Wood, S. (2002). Bayesian variable selection and model averaging in high dimensional multinomial nonparametric regression. *J. Comp. Graph. Statist.* To appear.

North Carolina State University

E-mail: (hzhang2@stat.ncsu.edu)

University of Wisconsin at Madison
E-mail: (yilin@cs.wisc.edu)