

# Adaptive Regularization through Entire Solution Surface

BY SEONGHO WU, XIAOTONG SHEN AND CHARLES J. GEYER

*School of Statistics, University of Minnesota, 313 Ford Hall, 224 Church Street S. E.,  
Minneapolis, Minnesota 55455, U. S. A.*

swu@stat.umn.edu xshen@stat.umn.edu charlie@stat.umn.edu

## SUMMARY

Several sparseness penalties have been suggested for delivery of good predictive performance in automatic variable selection within the framework of regularization. All assume that the true model is sparse. We propose a penalty, a convex combination of the  $L_1$ - and  $L_\infty$ -norms, that adapts to a variety of situations including sparseness and nonsparseness, grouping and nongrouping. The proposed penalty performs grouping and adaptive regularization. In addition, we introduce a novel homotopy algorithm utilizing subgradients for developing regularization solution surfaces involving multiple regularizers. This permits efficient computation and adaptive tuning. Numerical experiments are conducted via simulation. In simulated and real examples, the proposed penalty compares well against popular alternatives.

*Some key words:* Homotopy; Least squares;  $L_1$ -norm;  $L_\infty$ -norm; Subdifferential; Support vector machine; Variable grouping and selection.

## 1. INTRODUCTION

There has been great interest in various sparseness penalties for high-dimensional analysis, particularly the  $L_1$ -penalty. One distinct feature of the  $L_1$ -penalty is that it permits

automatic variable selection, especially when the number of candidate variables greatly exceeds the sample size. This unique feature, however, requires the true model be sparse, which is difficult if not impossible to verify in many situations. To seek high-dimensional structures leading to high predictive performance, we introduce a new penalty that adapts to a variety of situations including sparseness and nonsparseness, grouping and nongrouping. This new penalty, together with our new solution surface algorithm, yields adaptive regularization.

The  $L_2$ -penalty has been used in regression (Hoerl and Kennard, 1988) and support vector machines (SVMs, Vapnik, 1995). The  $L_1$ -penalty has been used in least squares regression (Tibshirani, 1996), SVMs (Bradley and Mangasarian, 1998), and sparse overcomplete representations (Donoho et al. 2006). The elastic net penalty, a convex combination of the  $L_1$ - and  $L_2$ -penalties, encourages grouping of highly correlated predictors (Zou and Hastie, 2005; Wang et al., 2006). Other relevant penalties include simultaneous variable selection and clustering (Bondell and Reich, 2008; Liu and Wu, 2008; Wang and Zhu, 2008), the Dantzig selector (Candes and Tao, 2007; Bickel et. al, 2008), and Sure Independence Screening (Fan and Lv, 2006).

To deliver high predictive performance, especially in a high-dimensional situation, we propose a new penalty, a convex combination of the  $L_1$ - and the  $L_\infty$ -penalties. This penalty not only enables automatic variable selection but also seeks grouping among predictors to enhance predictive performance. Furthermore, it permits efficient computation.

For high-dimensional data analysis, efficient computation requires realizing high predictive performance through tuning. Toward this end, we develop a subdifferential based homotopy method for computing entire solution surfaces, in addition to subgradient surfaces. This approach dramatically differs from the existing Kuhn-Tucker method; see Section 3 for details.

For model selection, we propose a model selection criterion for prediction error in least squares regression, where the designs can be random or fixed. The criterion can be expressed in terms of a covariance penalty plus a correction term taking into account extrapolation from random designs.

## 2. METHODOLOGY

Consider a problem of estimating  $f$  based on a random sample  $(x_i, y_i)_{i=1}^n$ , where  $x_i = (x_{i1}, \dots, x_{ip})$  is a  $p$ -dimensional vector and  $y_i$  is a scalar. We estimate  $f$  by minimizing the empirical loss  $\sum_{i=1}^n l(y_i, f(x_i))$  over a candidate function class  $f \in \mathcal{F}$ .

### 2.1. Regularization

The method of regularization is often employed to prevent overfitting in estimating  $f$ . To regularize parameters, penalties are added to the loss  $l(\cdot, \cdot)$  for various purposes. Thus the regularized loss is

$$\sum_{i=1}^n l(y_i, f(x_i)) + \lambda^T J(f) \quad (1)$$

where  $\lambda$  is a vector of nonnegative tuning parameters,  $J(f)$  is a vector of penalties regularizing  $f$ , and the superscript  $T$  denotes the transpose. The framework (1) covers least squares regression with  $l(y_i, f(x_i)) = (y_i - f(x_i))^2$  and SVM classification with  $l(y_i, f(x_i)) = [1 - y_i f(x_i)]_+$ , where  $x_+$  denotes the nonnegative part of  $x$ .

The penalty function  $J(f)$  is designed to achieve specific objectives. In variable selection, when  $f(x) = \beta^T h(x)$ , where  $h(x)$  is a vector valued basis function,  $J(f)$  can be chosen to be the  $L_1$ -norm  $\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$ . Then (1) leads to Lasso (Tibshirani, 1996) when  $l(\cdot, \cdot)$  is the least squares loss. The advantage of using the  $L_1$ -penalty in (1) is that it performs automatic variable selection by yielding a sparse solution of (1). In other words, the  $L_1$ -penalty can control a model's complexity effectively even when the dimension of  $\beta$  greatly

exceeds the sample size, c.f., Wang and Shen (2007) for classification. When  $J(f)$  is chosen to be the  $L_2$ -penalty, (1) shrinks  $\beta$  by grouping  $\beta_j$ 's corresponding to highly correlated variables; unfortunately, it does not produce a sparse model, c.f., Zou and Hastie (2005) for regression, and Wang et al. (2006) for classification.

To seek a sparse model, Zou and Hastie (2005) propose the elastic net penalty  $J(f)$ , a convex combination of  $L_1$ - and  $L_2$ - penalties. They show that it combines the advantages of the  $L_1$ - and  $L_2$ - penalties. Despite its performance, the elastic net penalty has two aspects requiring further attention. First, grouping predictors alone in regression or classification may not deliver good predictive performance in variable selection when the response is ignored, because correlations among predictors can be irrelevant to outcomes of the response. See the example in Section 6. Second, this penalty does not permit an efficient algorithm through a piecewise linear regularization solution surface.

In a recent paper, Bondell and Reich (2008) proposed the OSCAR penalty using the  $L_1$  and pairwise  $L_\infty$ -penalties for variable selection and grouping. However, efficient solution algorithms do not exist due to difficulty in treating overcomplete representation of the penalty.

## 2.2. Adaptive penalty

To enhance predictive performance of the method of regularization, we propose a new penalty to achieve three goals. First, the penalty is adaptive in that it adapts to a variety of situations including both sparse or nonsparse situations. Second, it seeks grouping among predictors in variable selection, for better predictive accuracy. Third, it permits efficient computation through a homotopy approach (Allgower and Georg, 1990).

The proposed penalty, which we call the  $L_1L_\infty$ -penalty has the form in the case  $f(x) = \beta^T h(x)$

$$J(f) \equiv J(\beta) = (1 - \alpha)\|\beta\|_1 + \alpha\|\beta\|_\infty, \quad (2)$$

where the  $L_\infty$ -norm is  $\|\beta\|_\infty = \max_{1 \leq j \leq p} |\beta_j|$ . This penalty retains the advantages of the two extreme penalties within the class  $L_p$ ,  $1 \leq p \leq \infty$ . The  $L_1$ -penalty is sparse and thresholds some coefficients at zero, whereas the  $L_\infty$ -penalty is nonsparse and groups highly correlated predictors that are relevant to the response. To adapt to the degree of sparseness,  $\alpha$  can be tuned.  $L_1L_\infty$  appears to be more adaptive than other penalties mentioned. Most importantly, (2) is piecewise linear in  $\beta$ . This is critical for efficient computation through the method of homotopy, see Section 3 for details. In contrast, the elastic net penalty is quadratic in  $\beta$ , which explains its computational limitation discussed above.

Placing (2) into (1), we obtain our regularized loss  $\sum_{i=1}^n l(y_i, \beta^T h(x_i)) + \tau((1 - \alpha)\|\beta\|_1 + \alpha\|\beta\|_\infty)$ , equivalently

$$\sum_{i=1}^n l(y_i, \beta^T h(x_i)) + \lambda_1 \|\beta\|_1 + \lambda_\infty \|\beta\|_\infty, \quad (3)$$

where  $\lambda_1 = \tau(1 - \alpha)$  and  $\lambda_\infty = \tau\alpha$  are nonnegative tuning parameters.

### 3. REGULARIZATION SOLUTION SURFACES

The minimizer of (3)  $\hat{\beta} \equiv \hat{\beta}_\lambda$ , if exists, is a function of  $\lambda = (\lambda_1, \lambda_\infty)^T$ . Consequently, computing an entire solution surface  $\lambda \mapsto \hat{\beta}_\lambda$  is critical for adaptive tuning.

In the literature, there exist several solution path algorithms computing an entire solution path for one-dimensional  $\lambda$ , c.f., Efron et al. (2004), Rosset and Zhu (2007), and Park and Hastie (2007) for regression, and Zhu et al. (2003), Hastie et al. (2004), and Wang et al. (2006) for classification. These algorithms use the Kuhn-Tucker conditions for tracking the piecewise linear solution path in the one-dimensional case. Unfortunately, there does not seem to exist such an algorithm in the higher dimensional case. This is partly because of difficulty of applying the Kuhn-Tucker conditions when many slack variables are involved.

To compute a high-dimensional solution surface, we use the subdifferential approach (Rockafellar and Wets, 2003), a systematic method of handling nonsmooth functions, which

has several advantages. First, the properties of nonsmooth functions at nondifferentiable points are completely characterized by subgradients. Second, subgradients can be tracked as well as  $\hat{\beta}_\lambda$ , facilitating computation in multi-dimensional situations.

### 3.1. General algorithm

First we describe a general algorithm for a convex objective function  $G_\lambda(\beta)$ , and then apply it to two specific situations.

A subgradient of  $G_\lambda$  at  $\beta$  is any vector  $b \in \mathbb{R}^p$  satisfying  $G_\lambda(\gamma) \geq G_\lambda(\beta) + b^T(\gamma - \beta)$ , for all  $\gamma \in \mathbb{R}^p$ , and the subdifferential of  $G_\lambda$  at  $\beta$ , denoted  $\partial G_\lambda(\beta)$ , is the set of all such  $b$ . The subdifferential of a convex function is a nonempty, convex, compact set, and  $0 \in \partial G_\lambda(\hat{\beta}_\lambda)$  is a necessary and sufficient condition that  $\hat{\beta}_\lambda$  is a global minimizer of  $G_\lambda$  (Rockafellar and Wets, 2003, pp. 308–311).

We give two examples, which will be used in our algorithms. The subgradient of  $\beta \mapsto \|\beta\|_1$  at  $\hat{\beta}_\lambda$  is a vector  $b_\lambda^1$  whose components satisfy  $b_{\lambda,j}^1 = \text{sign}(\hat{\beta}_{\lambda,j})$  if  $\hat{\beta}_{\lambda,j} \neq 0$  and  $-1 \leq b_{\lambda,j}^1 \leq 1$  otherwise, where  $\text{sign}(x) = 1$  if  $x > 0$ ,  $= 0$  if  $x = 0$ , and  $= -1$  otherwise. The subgradient of  $\beta \mapsto \|\beta\|_\infty$  (Rockafellar, 1970, p. 215) at  $\hat{\beta}_\lambda \neq 0$  is a vector  $b_\lambda^\infty$  whose components satisfy (a)

$$\sum_{j=1}^p |b_{\lambda,j}^\infty| = 1, \quad (4)$$

(b)  $b_{\lambda,j}^\infty = 0$  when  $|\hat{\beta}_{\lambda,j}| < \|\hat{\beta}_\lambda\|_\infty$ , (c)  $|b_{\lambda,j}^\infty| \leq 1$ , and (d)  $\text{sign}(b_{\lambda,j}^\infty) \text{sign}(\hat{\beta}_{\lambda,j}) \geq 0$ . The subgradient of  $\beta \mapsto \|\beta\|_\infty$  at  $\hat{\beta}_\lambda = 0$  is a vector  $b_\lambda^\infty$  whose components satisfy  $\sum_{j=1}^p |b_{\lambda,j}^\infty| \leq 1$ .

We encapsulate the case splitting in these characterizations by partitioning the indices for each into strongly active, weakly active, and inactive sets. An index  $j$  is strongly (resp. weakly) active for the  $L_1$  term at  $\lambda$  if  $|\hat{\beta}_{\lambda,j}| = 0$  and  $|b_{\lambda,j}^1| < 1$  (resp.  $= 1$ ). An index  $j$  is strongly (resp. weakly) active for the  $L_\infty$  term at  $\lambda$  if  $|\hat{\beta}_{\lambda,j}| = \|\hat{\beta}_\lambda\|_\infty$  and  $b_{\lambda,j}^\infty \neq 0$  (resp.  $= 0$ ). An index  $j$  is inactive if it is not one of these active cases.

For the least squares loss, we can write the optimality condition  $0 \in \partial G_\lambda(\hat{\beta}_\lambda)$  as

$$-2 \sum_{i=1}^n x_{ij} (y_i - x_i^T \hat{\beta}_\lambda) + \lambda_1 b_{\lambda,j}^1 + \lambda_\infty b_{\lambda,j}^\infty = 0, \quad j = 1, \dots, p. \quad (5)$$

If we know the index sets, that determines certain components of  $\hat{\beta}_\lambda$ ,  $b_\lambda^1$ , and  $b_\lambda^\infty$  and allows (5) to determine the rest. If the result satisfies the characterization of the subgradients and index sets, then this is the solution. Otherwise we must check with other index sets again until we identify the correct index sets at  $\lambda$ .

Since this check process is slow, we use more properties of  $\hat{\beta}_\lambda$ ,  $b_\lambda^1$ , and  $b_\lambda^\infty$ . They are each piecewise linear in a known smooth function of  $\lambda$  (Theorem 1 below). Moreover the change in the index sets at a transition point where the slope of some piecewise linear function changes is usually regular: one index goes from strongly active to weakly active to inactive or vice versa as  $\lambda$  goes through the transition point. In this regular case, no check process is necessary. If multiple indices are weakly active simultaneously, then the check process is necessary. More specifically we let one weakly active index be either strongly active or inactive, and repeat this process with other weakly active indices until we obtain the new index sets satisfying (5). Thus it is easy to compute the solution surface proceeding along straight lines in  $\lambda$  space.

The optimality condition becomes more complicated for SVM because the SVM hinge loss,  $l(y_i, f(x_i)) = [1 - y_i(\sum_{j=1}^p x_{ij}\beta_j + \beta_0)]_+$ , is nonsmooth and also needs subgradients and index sets. Then (5) is replaced by (6) below, but the general principles are the same. The solutions and subgradients are piecewise constant in  $\lambda$  or piecewise linear in a known smooth function  $\lambda$  (Theorem 2 below). When the index sets are fixed (6) determines solutions and subgradients. The behavior of subgradients at transition points is usually regular, so checking index sets is unnecessary, when the solution surface is followed along straight lines in  $\lambda$  space.

Following the above discussion, to compute  $\lambda \mapsto \hat{\beta}_\lambda$ , we specify a set of evaluation points

at which  $\hat{\beta}_\lambda$  will be computed. Starting from any evaluation point  $\lambda$  and evaluating  $\hat{\beta}_\lambda$  at all evaluation points by moving along straight lines from one evaluation point to another yields the entire solution surface.

This leads to a new homotopy algorithm.

**Algorithm 1:**

**Step 1:** (Initialization) At an initial  $\lambda = \lambda^0$ , compute  $\hat{\beta}_\lambda$ , which initializes the strongly active, weakly active, and inactive sets.

**Step 2:** (Directional derivatives) At a current point, either evaluation or transition, compute the directional derivative of  $\hat{\beta}_\lambda$ ,  $b_\lambda^1$ ,  $b_\lambda^\infty$ , and (in SVM) the subgradient of the loss function, along a line toward the next evaluation point. If the current point is a transition point, then the check process is applied. Determine the next transition point along the direction.

**Step 3:** (Updating or Extrapolation) At the current point, if no transition occurs before reaching the evaluation point, then extrapolate linearly to compute the value of  $\hat{\beta}_\lambda$  at the evaluation point from the current point. Update the current point by the evaluation point when no transition occurs, otherwise, by the next transition point and update the corresponding index sets. Terminate if  $\hat{\beta}_\lambda$  has been computed at all evaluation points, otherwise, go to Step 2.

Different  $\lambda$  can produce the same solution; in particular, for sufficiently large  $\lambda_1$  or  $\lambda_\infty$ , the solution  $\hat{\beta}_\lambda = 0$ . By construction, **Algorithm 1** identifies the unique global  $\hat{\beta}_\lambda$ , see Section 3.2 for the specification of evaluation points.

### 3.2. Least squares regression

In least squares regression, as  $\lambda$  varies,  $\hat{\beta}_\lambda$ ,  $b_\lambda^1$ , and  $b_\lambda^\infty$  change to satisfy (5), resulting in piecewise linearity of  $\hat{\beta}_\lambda$ ,  $b_\lambda^1$ , and  $b_\lambda^\infty$ .

**Theorem 1** (Piecewise linearity). *The solution  $\hat{\beta}_\lambda$  is piecewise linear in  $\lambda$  while  $b_\lambda^1$  and  $b_\lambda^\infty$  are piecewise linear in  $(\lambda_1/\lambda_\infty, 1/\lambda_\infty)^T$  and  $(\lambda_\infty/\lambda_1, 1/\lambda_1)^T$ , respectively.*

Let  $\mathcal{S}_\lambda^1$  and  $\mathcal{W}_\lambda^1$  denote the strongly and weakly active sets for  $\|\hat{\beta}_\lambda\|_1$ , similarly  $\mathcal{S}_\lambda^\infty$  and  $\mathcal{W}_\lambda^\infty$  for  $\|\hat{\beta}_\lambda\|_\infty$ , and  $\mathcal{I}_\lambda$  the inactive set. Also let  $\mathcal{A}_\lambda^\infty = \mathcal{S}_\lambda^\infty \cup \mathcal{W}_\lambda^\infty$  and  $\mathcal{A}_\lambda^1 = \mathcal{S}_\lambda^1 \cup \mathcal{W}_\lambda^1$ .

As an initial evaluation point, we take  $\lambda^0 = (\lambda_1^0, 0)^T$  where  $\lambda_1^0 = 2 \max_j |\sum_{i=1}^n x_{ij} y_i|$ . In (5), if  $\lambda_1 > \lambda_1^0$  and  $\lambda_\infty = 0$ , then  $|b_{\lambda,j}^1| < 1$  for all  $j$ , hence  $\hat{\beta}_\lambda = 0$ . Thus if  $\lambda_1 = \lambda_1^0$  and  $\lambda_\infty = 0$ , then a coefficient  $\hat{\beta}_{\lambda,j}$  becomes nonzero as  $|b_{\lambda,j}^1| = 1$ . Indeed  $|\hat{\beta}_{\lambda^0,j}| = \|\hat{\beta}_{\lambda^0}\|_\infty$  and  $|b_{\lambda^0,j}^\infty| = 1$  because  $\hat{\beta}_{\lambda^0,j}$  is the only nonzero variable. Consequently, initial index sets become  $\mathcal{A}_{\lambda^0}^\infty = \{j\}$ ,  $\mathcal{I}_{\lambda^0} = \emptyset$ , and  $\mathcal{A}_{\lambda^0}^1 = (\mathcal{A}_{\lambda^0}^\infty)^c$ , where  $A^c$  denotes the complement of  $A$ .

To specify other evaluation points, we compute  $\lambda_1^0$  and  $\lambda_\infty^0 = 2 \sum_{j=1}^p |\sum_{i=1}^n y_i x_{ij}|$  where  $\hat{\beta}_{(\lambda_1,0)^T} = 0$  for  $\lambda_1 > \lambda_1^0$  and  $\hat{\beta}_{(0,\lambda_\infty)^T} = 0$  for  $\lambda_\infty > \lambda_\infty^0$ , respectively. In the  $\lambda$ -plane, then we locate a set of evaluation points that are uniformly distributed in the rectangle with four corners  $(0,0)$ ,  $(\lambda_1^0, 0)$ ,  $(\lambda_1^0, \lambda_\infty^0)$ , and  $(0, \lambda_\infty^0)$ . Starting from  $\lambda = (\lambda_1^0, 0)^T$ ,  $\hat{\beta}_\lambda$  can be evaluated through the moving process: (a) Move  $\hat{\beta}_\lambda$  along the  $\lambda_1$  axis by decreasing  $\lambda_1$  until reaching the  $\lambda_\infty$  axis, (b) move  $\hat{\beta}_\lambda$  along the  $\lambda_\infty$  axis by increasing  $\lambda_\infty$  to the next evaluation point, (c) move  $\hat{\beta}_\lambda$  parallel to the  $\lambda_1$  axis by increasing  $\lambda_1$  until reaching  $\tilde{\lambda}$  where  $\hat{\beta}_{\tilde{\lambda}} = 0$ , (d) move  $\hat{\beta}_\lambda$  to the nearest evaluation point with  $\lambda_1 < \tilde{\lambda}_1$  and  $\lambda_\infty > \tilde{\lambda}_\infty$ , and (e) iterate (a)–(d) to reach  $(0, \lambda_\infty^0)$ .

Given index sets, directional derivatives of  $\hat{\beta}_\lambda$ ,  $b_\lambda^1$ , and  $b_\lambda^\infty$  are obtained by solving the derivatives with respect to  $\lambda$  of (4) and (9)–(11). In moving along a direction, a transition occurs when one of the following events occurs: (a) An index  $j$  in  $\mathcal{I}_\lambda$  moves to  $\mathcal{W}_\lambda^\infty$  when  $|\hat{\beta}_{\lambda,j}|$  becomes  $\|\hat{\beta}_\lambda\|_\infty$  retaining  $|b_{\lambda,j}^\infty| = 0$ , (b) an index  $j$  in  $\mathcal{S}_\lambda^\infty$  moves to  $\mathcal{W}_\lambda^\infty$  when  $|b_{\lambda,j}^\infty|$  becomes 0 retaining  $|\hat{\beta}_{\lambda,j}| = \|\hat{\beta}_\lambda\|_\infty$ , (c) an index  $j$  in  $\mathcal{I}_\lambda$  moves to  $\mathcal{W}_\lambda^1$  when  $|\hat{\beta}_{\lambda,j}|$  becomes 0 retaining  $|b_{\lambda,j}^1| = 1$ , or (d) an index  $j$  in  $\mathcal{S}_\lambda^1$  moves to  $\mathcal{W}_\lambda^1$  when  $|b_{\lambda,j}^1|$  becomes 1 retaining  $|\hat{\beta}_{\lambda,j}| = 0$ .

Then Step 3 of **Algorithm 1** is applied.

### 3.3. Classification

We now apply (2) to the hinge loss  $l(y_i, f(x_i)) = [1 - y_i(\sum_{j=1}^p x_{ij}\beta_j + \beta_0)]_+$  with  $y_i \in \{1, -1\}$ .

In addition to the index sets for  $\|\beta\|_1$  and  $\|\beta\|_\infty$ , the index sets for the hinge loss are specified. Let  $z_i(\beta, \beta_0) = 1 - y_i(\sum_{j=1}^p x_{ij}\beta_j + \beta_0)$  and  $\alpha_{\lambda,i}$  denote subgradients at  $z_{\lambda,i} \equiv z_i(\hat{\beta}_\lambda, \hat{\beta}_{\lambda,0})$  of  $z_{\lambda,i} \mapsto [z_{\lambda,i}]_+$ . Then  $\alpha_{\lambda,i} = 0$ , if  $z_{\lambda,i} < 0$ ,  $= 1$ , if  $z_{\lambda,i} > 0$ , and  $= [0, 1]$ , otherwise,  $i = 1, \dots, n$ . From this characterization, the strongly active hinge set is  $\mathcal{S}_\lambda^H = \{i : z_{\lambda,i} = 0 \text{ and } 0 < \alpha_{\lambda,i} < 1\}$ . The left weakly active hinge and right weakly active hinge sets are  $\mathcal{W}_\lambda^{LH} = \{i : z_{\lambda,i} = 0 \text{ and } \alpha_{\lambda,i} = 1\}$  and  $\mathcal{W}_\lambda^{RH} = \{i : z_{\lambda,i} = 0 \text{ and } \alpha_{\lambda,i} = 0\}$ . Let  $\mathcal{H}_\lambda = \mathcal{S}_\lambda^H \cup \mathcal{W}_\lambda^{LH} \cup \mathcal{W}_\lambda^{RH}$ . The left and right inactive sets are defined as  $\mathcal{L}_\lambda = \{i : z_{\lambda,i} > 0 \text{ and } \alpha_{\lambda,i} = 1\}$  and  $\mathcal{R}_\lambda = \{i : z_{\lambda,i} < 0 \text{ and } \alpha_{\lambda,i} = 0\}$ .

The optimality condition  $0 \in \partial G_\lambda(\hat{\beta}_\lambda)$  can be written as

$$-\sum_{i=1}^n \alpha_{\lambda,i} y_i x_{ij} + \lambda_1 b_{\lambda,j}^1 + \lambda_\infty b_{\lambda,j}^\infty = 0, \quad j = 1, \dots, p, \quad \text{and} \quad \sum_{i=1}^n \alpha_{\lambda,i} y_i = 0. \quad (6)$$

As  $\lambda$  varies,  $\hat{\beta}_\lambda, b_\lambda^1, b_\lambda^\infty$ , and  $\alpha_\lambda = (\alpha_{\lambda,1}, \dots, \alpha_{\lambda,n})^T$  change to satisfy (6), leading to the following theorem.

**Theorem 2** (Piecewise constancy). *The solutions  $\hat{\beta}_\lambda$  and  $\hat{\beta}_{\lambda,0}$  are piecewise constant in  $\lambda$ . Furthermore,  $\alpha_\lambda$  is piecewise linear in  $\lambda$  while  $b_\lambda^\infty$  and  $b_\lambda^1$  are piecewise linear in  $(\lambda_1/\lambda_\infty, 1/\lambda_\infty)^T$  and  $(\lambda_\infty/\lambda_1, 1/\lambda_1)^T$ , respectively.*

In this case, minimizing (3) for the hinge loss numerically at  $\lambda^0$  yields an initial solution and its corresponding index sets.

Directional derivatives of  $b_\lambda^1, b_\lambda^\infty$ , and  $\alpha_\lambda$  are obtained by solving the derivatives with respect to  $\lambda$  of (4) and (12)–(15). However, (6) does not contain  $\hat{\beta}_\lambda$  and  $\hat{\beta}_{\lambda,0}$ , so to track

them, we use the hinge relationship  $\hat{\beta}_{\lambda,0} + \sum_{j \in \mathcal{I}_\lambda} x_{ij} \hat{\beta}_{\lambda,j} + \sum_{j \in \mathcal{A}_\lambda^\infty} x_{ij} \text{sign}(\hat{\beta}_{\lambda,j}) \|\hat{\beta}_\lambda\|_\infty = y_i$ ,  $i \in \mathcal{H}_\lambda$ . These are  $\text{card}(\mathcal{H}_\lambda)$  equations to be solved for  $\text{card}(\mathcal{I}_\lambda) + 2$  unknowns,  $\beta_{\lambda,j}$ ,  $j \in \mathcal{I}_\lambda$ ,  $\hat{\beta}_{\lambda,0}$ , and  $\|\hat{\beta}_\lambda\|_\infty$ . To obtain  $\hat{\beta}_\lambda$  and  $\hat{\beta}_{\lambda,0}$ , we apply the cardinality relationship  $\text{card}(\mathcal{H}_\lambda) = \text{card}(\mathcal{I}_\lambda) + 2$  as explained in the proof of Theorem 2.

In moving along a homotopy direction, a transition occurs when one of the events (a), (b), (c), or (d) in Section 3.2 occurs or one of the following events occurs: (a) An index  $i$  in  $\mathcal{L}_\lambda$  moves to  $\mathcal{W}_\lambda^{LH}$  when  $z_{\lambda,i}$  becomes 0 retaining  $\alpha_{\lambda,i} = 1$ , (b) an index  $i$  in  $\mathcal{S}_\lambda^H$  moves to  $\mathcal{W}_\lambda^{LH}$  when  $\alpha_{\lambda,i}$  becomes 1 retaining  $z_{\lambda,i} = 0$ , (c) an index  $i$  in  $\mathcal{R}_\lambda$  moves to  $\mathcal{W}_\lambda^{RH}$  when  $z_{\lambda,i}$  becomes 0 retaining  $\alpha_{\lambda,i} = 0$ , or (d) an index  $i$  in  $\mathcal{S}_\lambda^H$  moves to  $\mathcal{W}_\lambda^{RH}$  when  $\alpha_{\lambda,i}$  becomes 0 retaining  $z_{\lambda,i} = 0$ .

Evaluation points are located as in the least squares problem and then Step 3 of **Algorithm 1** is applied.

#### 4. CHOICE OF TUNING PARAMETER

This section is devoted to model selection, particularly for selection of the optimal tuning parameter  $\lambda$ . Specifically, we focus our attention on least squares regression and binary classification, where the design points can be fixed or random. Our focus is on estimation of the prediction error and generalization error.

##### 4.1. Least squares regression

Consider a regression model  $Y_i = \mu(X_i) + e_i$ ,  $i = 1, \dots, n$ , with  $\mu(x) = x^T \beta$ , where  $X_i$  follows an unknown distribution  $P$ , and  $e_i$  is random error with  $E(e_i) = 0$  and  $\text{var}(e_i) = \sigma^2$ , and  $e_i$  is independent of  $X_i$ , for all  $i$ . Let  $\hat{\mu}_\lambda(x)$  be an estimate of  $\mu_\lambda(x)$  obtained from (5), based on the sample  $(X^n, Y^n) = (X_i, Y_i)_{i=1}^n$ .

The performance of  $\hat{\mu}_\lambda$  is evaluated by the prediction error,  $PE(\hat{\mu}_\lambda) = E(Y - \hat{\mu}_\lambda(X))^2$ ,

where the expectation  $E$  is taken over  $(X, Y)$ . This prediction error measures predictive performance with respect to not only  $Y$  but also  $X$ , which differs from the conventional conditional prediction error given  $X$ . See Breiman and Spector (1992) for a detailed discussion of the difference between this  $PE(\hat{\mu}_\lambda)$  and the conditional prediction error.

To derive a model selection criterion, we apply an argument similar to that in Theorem 1 of Wang and Shen (2006), to yield an (approximately) optimal unbiased estimator of  $PE(\hat{\mu}_\lambda)$  in the form of  $OPE(\hat{\mu}_\lambda) = n^{-1} \sum_{i=1}^n (Y_i - \hat{\mu}_\lambda(X_i))^2 + 2n^{-1} \sum_{i=1}^n \text{cov}(Y_i, \hat{\mu}_\lambda(X_i)|X^n) + D_{1n}(X^n, \hat{\mu}_\lambda) + D_{2n}(X^n)$  where  $D_{1n} = E[E(E(Y|X) - \hat{\mu}_\lambda(X))^2] - n^{-1} \sum_{i=1}^n (E(Y_i|X_i) - \hat{\mu}_\lambda(X_i))^2|X^n)$  and  $D_{2n} = E[\text{var}(Y|X)] - n^{-1} \sum_{i=1}^n \text{var}(Y_i|X_i)$ . For comparison, it suffices to use  $OPE(\hat{\mu}_\lambda) - D_{2n}$  because the term  $D_{2n}$  is independent of  $\hat{\mu}_\lambda$ . This leads to our proposed model selection criterion, denoted by the generalized degrees of freedom (GDF):

$$\widehat{\text{GDF}}(\hat{\mu}_\lambda) = n^{-1} \sum_{i=1}^n (Y_i - \hat{\mu}_\lambda(X_i))^2 + 2n^{-1} \sum_{i=1}^n \widehat{\text{cov}}(Y_i, \hat{\mu}_\lambda(X_i)|X^n) + \widehat{D}_{1n}(X^n, \hat{\mu}_\lambda). \quad (7)$$

In the case of fixed design,  $\widehat{D}_{1n} \equiv 0$  and hence  $\widehat{\text{GDF}}(\hat{\mu}_\lambda)$  reduces to the covariance penalty  $C_p(\hat{\mu}_\lambda) = n^{-1} \sum_{i=1}^n (Y_i - \hat{\mu}_\lambda(X_i))^2 + 2n^{-1} \sum_{i=1}^n \widehat{\text{cov}}(Y_i, \hat{\mu}_\lambda(X_i))$ .

To estimate  $\sum_{i=1}^n \text{cov}(Y_i, \hat{\mu}_\lambda(X_i)|X^n)$ , we define the degrees of freedom  $\text{df}(\hat{\mu}_\lambda)$  for  $\hat{\mu}_\lambda$  to be  $\sum_{i=1}^n \text{cov}(Y_i, \hat{\mu}_\lambda(X_i)|X^n)/\sigma^2$ .

**Theorem 3.** *For the  $L_1L_\infty$  estimate,  $\text{df}(\hat{\mu}_\lambda) = E[\text{card}(\mathcal{I}_\lambda)] + 1$ , and  $\widehat{\text{df}}(\hat{\mu}_\lambda) = \text{card}(\mathcal{I}_\lambda) + 1$  is an unbiased estimate of  $\text{df}(\hat{\mu}_\lambda)$ .*

Therefore  $\sum_{i=1}^n \widehat{\text{cov}}(Y_i, \hat{\mu}_\lambda(X_i)|X^n)$  can be estimated through  $\widehat{\text{df}}(\hat{\mu}_\lambda)\sigma^2$ . When  $\sigma^2$  is unknown, an (approximately) unbiased estimate  $\hat{\sigma}^2$  is used, see Section 4 in Efron et al. (2004).

For  $\widehat{D}_{1n}$ , we apply the data perturbation technique in Shen and Huang (2006). First, we perturb  $X_i$  to generate pseudo data  $X_i^* = X_i + \tau(\tilde{X}_i - X_i)$ ,  $i = 1, \dots, n$ , where  $\tilde{X}_i$  is sampled from its empirical distribution and  $0 \leq \tau \leq 1$  is the size of perturbation. Second,

we perturb  $Y_i$  to yield  $Y_i^* = Y_i + \tau(\tilde{Y}_i - Y_i)$ ,  $i = 1, \dots, n$  with  $\tilde{Y}_i \sim N(Y_i, \sigma_i^2)$ . The first term  $E(E(Y|X) - \hat{\mu}_\lambda(X))^2$  in  $D_{1n}$  is estimated by  $n^{-1} \sum_{i=1}^n (\hat{\mu}_\lambda(X_i) - \hat{\mu}_\lambda^*(X_i))^2$ , where  $\hat{\mu}_\lambda^*$  is estimated through  $(X_i^*, Y_i^*)_{i=1}^n$ , while the second term  $n^{-1} \sum_{i=1}^n (E(Y_i|X_i) - \hat{\mu}_\lambda(X_i))^2 |X^n$  in  $D_{1n}$  is estimated by  $n^{-1} \sum_{i=1}^n (\hat{\mu}_\lambda(X_i^*) - \hat{\mu}_\lambda^*(X_i^*))^2$ . Consequently,  $\widehat{\text{GDF}}(\hat{\mu}_\lambda)$  is obtained by the perturbed data, and can be computed via Monte Carlo approximation as described in Wang and Shen (2006). In what follows, we fix  $\tau = 0.5$  throughout, see Shen and Huang (2006) for a sensitivity study with regard to the choice of  $\tau$ .

#### 4.2. Classification

In classification, a model selection criterion  $\widehat{\text{GDF}}$  that is similar to (7) has been obtained in Wang and Shen (2006) through a different data perturbation scheme. The reader may consult their paper for more details.

### 5. NUMERICAL EXAMPLES

We now demonstrate effectiveness of the proposed penalty and compare it against the elastic net, Lasso, and the  $L_\infty$ -penalty through simulated and benchmark examples. In least squares regression and binary classification, we examine the case of small  $p$  and large  $n$  and additionally consider the case of large  $p$  and small  $n$  where the number of candidate variables  $p$  can greatly exceed the sample size  $n$ , which is of great current interest.

In each simulated example, a training sample is generated together with an independent test sample. In each benchmark example, a training and a test sample are created by randomly dividing the original data into equal halves. In each example,  $\hat{\beta}_\lambda$  is computed on a training sample, and its predictive performance is evaluated on a test sample.

To adaptively tune, we compute  $\hat{\beta}_\lambda$  through a regularization solution surface by applying Algorithm 1. For the  $L_1L_\infty$ -penalty, we locate 200 evaluation points on the  $\lambda$ -plane as

described in Section 3.2.

For the  $L_2$ -component of the elastic net and the  $L_2$ -penalty, we choose a set of uniform grid points between  $10^{-3}$  and  $10^3$ . For a fair comparison, the number of grid points for the  $L_2$ -component of the elastic net and  $L_2$ -penalty is fixed to be that of evaluation points on the  $\lambda_\infty$  axis in the  $L_1L_\infty$ -penalty.

In each example, we compute the test error with  $\hat{\beta}_\lambda$  for the squared loss in regression and the 0–1 loss in classification by obtaining the minimizer  $\hat{\lambda}$  of  $\widehat{\text{GDF}}$ .

### 5.1. Simulated examples in least squares regression

Five simulated examples are examined. The first three examples are modified from those in Tibshirani (1996) and Zou and Hastie (2005), the fourth one is taken from Yuan and Lin (2006), and the last one considers the case of large  $p$  and small  $n$ . In each example, a linear model is used, where the response  $Y_i$  is generated from

$$Y_i = X_i^T \beta + e_i, \quad e_i \sim N(0, \sigma^2), \quad i = 1, \dots, n \quad (8)$$

where  $X_i = (X_{i1}, \dots, X_{ip})^T$  is a vector of predictors, and is independent of  $e_i$ . For each example, a training sample of size 50 and a test sample of size  $10^3$  are generated. Details of the five examples are as follows.

**Example 1 (Sparse predictors).** In (8),  $X_i$  is sampled from  $N(0, \Sigma)$  with  $p = 10$ , where the  $jk$ -th element of  $\Sigma$  is  $0.5^{|j-k|}$ . Here  $\beta = (3, -1.5, 0, 0, 1, 0, 0, 0, 2, 0)^T$  and  $\sigma = 3$ .

**Example 2 (Nonsparse grouped predictors).** This example is the same as Example 1 except that  $\beta = (0.85, \dots, 0.85)^T$ .

**Example 3 (Sparse grouped predictors).** In (8),  $X_i$  is sampled from  $N(0, \Sigma)$  with  $p = 20$ , where the diagonal and off-diagonal elements of  $\Sigma$  are 1 and 0.5, respectively. Here  $\beta = (0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2)^T$  and  $\sigma = 9$ .

**Example 4 (Grouping among predictors).** In (8), let  $X_i$  be  $(W_{i1}, W_{i1}^2, W_{i1}^3, \dots, W_{i5}, W_{i5}^2, W_{i5}^3)^T$ , where  $W_{ik} = (U_{ik} + V)/\sqrt{2}$ ,  $k = 1, \dots, 5$ , and  $U_{i1}, \dots, U_{i5}$  and  $V$  are generated from  $N(0, 1)$  independently. Here  $\beta = (0, 0, 0, 1, 1, 1, 0, 0, 0, 2/3, -1, 1/3, 0, 0, 0)^T$  and  $\sigma = 3$ .

**Example 5 (Large  $p$  but small  $n$ ).** In (8),  $X_i$  is sampled from  $N(0, \sigma)$  with  $p = 200$ , where the  $jk$ -th element of  $\Sigma$  is  $0.5^{|j-k|}$ . Here elements 1–5 of  $\beta$  are 3, elements 6–10 of  $\beta$  are  $-1.5$ , elements 11–15 are 1, elements 16–20 are 2, and the rest are zeros.

Table 1 indicates that the  $L_1L_\infty$ -penalty performs equally as well as the Lasso and elastic net penalties in both of the large  $n$  small  $p$  and large  $p$  small  $n$  sparse cases (Examples 1, 4, and 5) and as well as the  $L_\infty$ -penalty in the nonsparse case (Example 2), adapting to a variety of situations by changing the value of  $\lambda$ . Interestingly, it outperforms the others in the sparse grouped predictors case (Example 3), which says that its grouping property provides a way of dimensionality reduction. As a result, it tends to identify a simpler model. In fact, the number of distinct nonzero coefficients identified by the  $L_1L_\infty$ -penalty is close to those of the elastic net and Lasso in the sparse situation (Examples 1, 4 and 5), and becomes much smaller in the nonsparse situation (Examples 2 and 3).

With regard to the quality of estimation of  $PE(\hat{\mu}_\lambda) = E(Y - \hat{\mu}_\lambda(X))^2$ , GDF performs well as compared to the Oracle test error that is the minimum value of the empirical  $PE(\hat{\mu}_\lambda)$  evaluated through the test samples on the pre-specified  $\lambda$  values.

[Table 1 about here.]

## 5.2. Simulated examples in classification

Three examples are examined, which are slightly modified from those used in Wang et al. (2006). In the case of small  $p$  and large  $n$ , we generate a training sample of size  $n = 50$  and  $p = 10$ , with 50% of them having the positive class. In the case of large  $p$  and small

$n$ , the size of a training sample remains to be the same with  $p = 300$ . In each example, a test sample of size  $10^4$  is used to evaluate the performance of each method after adaptive tuning through GDF.

**Example 1 (Independent predictors).** First,  $X_i$  is generated from  $N(\mu, I_{p \times p})$  with  $\mu = (0.5, 0.5, 0.5, 0.5, 0.5, 0, \dots, 0)^T \in \mathbb{R}^p$  and assign  $Y_i = 1, i = 1, \dots, [n/2]$ . Second,  $X_i$  is generated from  $N(-\mu, I_{p \times p})$  and assign  $Y_i = -1, i = [n/2] + 1, \dots, n$ .

**Example 2 (High correlations among predictors).** First,  $X_i$  is generated from  $N(\mu, \Sigma)$  with  $\mu = (1, 1, 1, 1, 1, 0, \dots, 0)^T \in \mathbb{R}^p$ , and assign  $Y_i = 1, i = 1, \dots, [n/2]$ . Second,  $X_i$  is generated from  $N(-\mu, \Sigma)$  and assign  $Y_i = -1, i = [n/2] + 1, \dots, n$ . Here

$$\Sigma = \begin{pmatrix} \Sigma^* & 0_{5 \times (p-5)} \\ 0_{(p-5) \times 5} & I_{(p-5) \times (p-5)} \end{pmatrix}$$

where the diagonal and the off-diagonal elements of  $\Sigma^*$  equal to 1 and 0.8, respectively.

**Example 3 (Power decay correlations among predictors).** This example remains the same as Example 2 except that the  $jk$ -th component of  $\Sigma^*$  is  $0.8^{|j-k|}$ .

[Table 2 about here.]

As indicated in Table 2, the  $L_1L_\infty$ -SVM outperforms its three competitors in every single case. This suggests that the  $L_1L_\infty$ -penalty goes beyond what each of its  $L_1$  and  $L_\infty$  counterparts can offer in terms of adaptation. However, the amount of improvement varies over the competitors, with the largest amount occurring in the case of large  $p$  and small  $n$ .

### 5.3. Breast cancer classification

The Wisconsin Breast Cancer Data (WBCD), collected at University of Wisconsin Hospitals, develop a prediction model for discriminating benign from malignant breast tissue samples through nine clinical diagnostic characteristics. These characteristics are assigned

integer values in  $[1, 10]$ , with lower and high values indicating the most normal and abnormal states. Detailed descriptions of WBCD can be found in Wolberg and Mangasarian (1990).

For WBCD, we apply the  $L_1$ -,  $L_2$ -,  $L_\infty$ -, and  $L_1L_\infty$ -SVMs. To cross validate our analysis, we randomly divide the 682 issue samples there into equal halves for training and testing. Test errors over 100 random partitions are reported in Table 3 for optimal models after tuning.

[Table 3 about here.]

As suggested by Table 3, the  $L_1L_\infty$ -SVM outperforms its competitors in terms of predictive accuracy. It appears that WBCD is a nonsparse case as the  $L_1$ -SVM performs worst.

#### 5.4. *Microarray*

The leukemia DNA microarray data studied in Golub et al. (1999) concerns prediction of two types of acute leukemia, lymphoblastic and myeloid, through gene expression profiles. Of particular interest is selecting a subset of genes, among 7129 candidate genes, as a prediction marker of acute leukemia. For the 7129 genes, 1059 genes remain after a pre-screening process consisting of thresholding, filtering, and standardization, c.f., Dudoit et al. (2002). The data contain 72 tissue samples of the two types of acute leukemia, among which 57 samples are lymphoblastic, together with expression profiles of 1059 candidate genes. Details can be found at <http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi>.

For the data, we apply the  $L_1$ -,  $L_2$ -,  $L_\infty$ -, and  $L_1L_\infty$ -SVMs, to 38 training samples, and use additional 34 for testing as in Golub et al. (1999). The classification results are reported in Table 4.

[Table 4 about here.]

As suggested in Table 4, the  $L_1L_\infty$ -SVM performs best in terms of predictive accuracy, while identifying 65 important genes. In contrast, the maximum number of important genes that can be selected by  $L_1$ -SVM is no greater than the training sample size 38, which may be too small to be realistic, whereas the  $L_2$ - and  $L_\infty$ -SVMs select all 1,059 genes. In a similar study, the elastic net selects 78 out of 2,308, see Wang et al. (2006). This result is comparable to what we obtain in here.

## 6. THEORY

This section investigates statistical aspects of the grouping, adaptation, and shrinkage properties of the  $L_1L_\infty$ -penalty.

**Theorem 4** (Grouping). *In least squares regression, let  $c(\hat{\beta}_{\lambda,j}) = -\sum_{i=1}^n x_{ij}(y_i - x_i^T \hat{\beta}_\lambda)$  denote the correlation between predictors and residuals. For any  $\lambda$  and  $j = 1, \dots, p$ , if  $|c(\hat{\beta}_{\lambda,j})| > \lambda_1/2$  and  $\hat{\beta}_\lambda \neq 0$ , then  $\hat{\beta}_{\lambda,j} = \text{sign}(c(\hat{\beta}_{\lambda,j}))\|\hat{\beta}_\lambda\|_\infty$ . Furthermore, in case of orthonormal predictors,*

$$\hat{\beta}_{\lambda,j} = \begin{cases} \left[ \left| \sum_{i=1}^n x_{ij}y_i \right| - \frac{\lambda_1}{2} \right]_+ \text{sign}(c(\hat{\beta}_{\lambda,j})), & |c(\hat{\beta}_{\lambda,j})| \leq \frac{\lambda_1}{2} \\ \frac{1}{\text{card}(\mathcal{A}_\lambda^\infty)} \left( \sum_{j \in \mathcal{A}_\lambda^\infty} \left( \left| \sum_{i=1}^n x_{ij}y_i \right| - \frac{\lambda_1}{2} \right) - \frac{\lambda_\infty}{2} \right) \text{sign}(c(\hat{\beta}_{\lambda,j})), & |c(\hat{\beta}_{\lambda,j})| > \frac{\lambda_1}{2}. \end{cases}$$

Theorem 4 says that  $\hat{\beta}_{\lambda,j}$  is grouped at  $\text{sign}(c(\hat{\beta}_{\lambda,j}))\|\hat{\beta}_\lambda\|_\infty$  if the sample correlation between the predictor  $x_j$  and the residuals exceeds  $\lambda_1/2$ . When  $|c(\hat{\beta}_{\lambda,j})| > \lambda_1/2$ , as suggested in the orthonormal case,  $\sum_{i=1}^n x_{ij}y_i$ , the  $j$ -th component of the ordinary least squares estimate, gets shrunk by both the  $L_1$ - and  $L_\infty$ -components of the  $L_1L_\infty$ -penalty, and  $\hat{\beta}_{\lambda,j}$  is pulled down to  $\text{sign}(c(\hat{\beta}_{\lambda,j}))\|\hat{\beta}_\lambda\|_\infty$ . When  $|c(\hat{\beta}_{\lambda,j})| \leq \lambda_1/2$ , the  $L_1L_\infty$ -penalty performs a Lasso-type thresholding (Tibshirani, 1996). Consequently, the grouping property is incorporated into shrinkage, which enables the  $L_1L_\infty$ -penalty to yield a simple model regardless of sparseness. In the sparse situation,  $\lambda_1$  can be greater than  $\lambda_\infty$  to yield some coefficients

to be shrunken to zero. In the nonsparse case, it is just opposite, and the  $L_1L_\infty$ -penalty forces some predictors to be grouped. In any case, the  $L_1L_\infty$ -penalty produces a simple model.

As mentioned in Section 2.1, grouping by the elastic net uses the correlations among predictors while grouping by the  $L_1L_\infty$ -penalty deals with  $c(\hat{\beta}_\lambda)$ . It is clear that the correlations among the predictors do not determine the correlations between the predictors and the response. Thus grouping by the former may not reduce estimation variance and may result in degradation of variable selection. To confirm this intuition, we examine a simple example. We sample  $X_1$  from  $\text{Unif}[-10, 10]$ , and let  $X_k$  be  $e \times X_1$ ,  $k = 2, \dots, 10$ , where  $e$  follows  $N(3, 1)$  and is independent of  $X_1$ . Then the response  $Y$  is  $3X_1 + e$ , where  $e \sim N(0, 3)$ . This makes  $X_1$  and  $X_k$ ,  $k = 2, \dots, 10$  highly correlated but  $Y$  is conditionally independent of  $(X_2, \dots, X_{10})$  given  $X_1$ . Consequently, the selected model should contain  $X_1$  only. We generate 100 datasets, each with a training and test sample of 50 and 500 observations. Tuning is performed as in Section 5.

[Table 5 about here.]

Table 5 shows that predictive performance of the elastic net penalty is worse than the  $L_1L_\infty$ -penalty. In fact, the elastic net penalty selects more than 8 variables while the  $L_1L_\infty$ -penalty selects 4.52. This example demonstrates that prediction accuracy and variable selection are not directly related with the correlations among predictors.

The next theorem explains how the grouping property of the  $L_1L_\infty$ -penalty leads to adaptive regularization.

**Theorem 5** (Adaptation). *In least squares regression if  $\hat{\beta}_\lambda \neq 0$ , then the regularized loss (3) reduces to*

$$\sum_{i=1}^n (y_i - x_i^{cT} \hat{\beta}_\lambda^c)^2 + \sum_{j=1}^{\text{card}(\mathcal{I}_\lambda)+1} \lambda_j^c |\hat{\beta}_{\lambda,j}^c|$$

where  $x_i^c = (x_{ik_1}, \dots, x_{ik_{\text{card}(\mathcal{I}_\lambda)}}, \sum_{j \in \mathcal{A}_\lambda^\infty} x_{ij} \text{sign}(\hat{\beta}_{\lambda,j}))^T$ , where  $\hat{\beta}_\lambda^c = (\hat{\beta}_{\lambda,k_1}, \dots, \hat{\beta}_{\lambda,k_{\text{card}(\mathcal{I}_\lambda)}}, \|\hat{\beta}_\lambda\|_\infty)^T$  with  $\{k_1, \dots, k_{\text{card}(\mathcal{I}_\lambda)}\} \in \mathcal{I}_\lambda$ , and  $\lambda^c$  is a vector whose first  $\text{card}(\mathcal{I}_\lambda)$  elements are  $\lambda_1$  and the last element is  $\text{card}(\mathcal{A}_\lambda^\infty)\lambda_1 + \lambda_\infty$ . And  $\mathcal{I}_\lambda$  and  $\mathcal{A}_\lambda^\infty$  are defined in Section 3.2.

Theorem 5 says that  $\|\hat{\beta}_\lambda\|_\infty$  is regularized by  $\text{card}(\mathcal{A}_\lambda^\infty)\lambda_1 + \lambda_\infty$  while  $\hat{\beta}_{\lambda,j}$ ,  $j \in \mathcal{I}_\lambda$  is controlled by  $\lambda_1$ . Because  $\text{card}(\mathcal{A}_\lambda^\infty)\lambda_1 + \lambda_\infty > \lambda_1$ , it indicates the  $L_1L_\infty$ -penalty regularizes  $\|\hat{\beta}_\lambda\|_\infty$  more than the components of  $\hat{\beta}_\lambda$  in  $\mathcal{I}_\lambda$ . In other words, the  $L_1L_\infty$ -penalty achieves adaptive regularization based on  $c(\hat{\beta}_{\lambda,j})$  because any  $\hat{\beta}_{\lambda,j}$  with  $c(\hat{\beta}_{\lambda,j}) > \lambda_1/2$  is grouped at  $\text{sign}(c(\hat{\beta}_{\lambda,j}))\|\hat{\beta}_\lambda\|_\infty$ .

## 7. DISCUSSION

This paper introduces an adaptive regularization method based on the  $L_1L_\infty$ -penalty, for improving predictive accuracy in both sparse and nonsparse situations. The proposed method is implemented through solution surfaces based on a subdifferential approach. In contrast to the existing penalties such as Lasso and the elastic net penalties, the new penalty reduces the estimation error by collapsing predictors that are highly correlated with the residuals into one group. As a result, it leads to a model with a simple representation regardless of the degree of sparseness.

The proposed method may be generalized to allow for a general polyhedral penalty, defined as  $J(\beta) = \{\beta : v_i^T \beta \leq 1, i = 1, \dots, m\}$ , where  $v_i$  is a vector in  $\mathbb{R}^p$  and  $m$  is the number of constraints. This covers the case of the fused Lasso (Tibshirani et al., 2005).

## ACKNOWLEDGEMENT

This research was supported in part by grants from the U. S. A. National Science Foundation and National Institute of Health.

APPENDIX: PROOFS

**Proof of Theorem 1.** If the index sets remain unchanged in an interval, then it suffices to prove the piecewise linearity of  $\hat{\beta}_\lambda$ ,  $b_\lambda^\infty$ , and  $b_\lambda^1$  in this interval. We can write (5) at  $\lambda$  as

$$-2 \sum_{i=1}^n x_{ij} \left( y_i - \sum_{k \in \mathcal{I}_\lambda} x_{ik} \hat{\beta}_{\lambda,k} - \sum_{k \in \mathcal{A}_\lambda^\infty} x_{ik} \text{sign}(\hat{\beta}_{\lambda,k}) \|\hat{\beta}_\lambda\|_\infty \right) + \lambda_1 \text{sign}(\hat{\beta}_{\lambda,j}) + \lambda_\infty \text{sign}(\hat{\beta}_{\lambda,j}) |b_{\lambda,j}^\infty| = 0, \quad j \in \mathcal{A}_\lambda^\infty \quad (9)$$

$$-2 \sum_{i=1}^n x_{ij} \left( y_i - \sum_{k \in \mathcal{I}_\lambda} x_{ik} \hat{\beta}_{\lambda,k} - \sum_{k \in \mathcal{A}_\lambda^\infty} x_{ik} \text{sign}(\hat{\beta}_{\lambda,k}) \|\hat{\beta}_\lambda\|_\infty \right) + \lambda_1 \text{sign}(\hat{\beta}_{\lambda,j}) = 0, \quad j \in \mathcal{I}_\lambda \quad (10)$$

$$-2 \sum_{i=1}^n x_{ij} \left( y_i - \sum_{k \in \mathcal{I}_\lambda} x_{ik} \hat{\beta}_{\lambda,k} - \sum_{k \in \mathcal{A}_\lambda^\infty} x_{ik} \text{sign}(\hat{\beta}_{\lambda,k}) \|\hat{\beta}_\lambda\|_\infty \right) + \lambda_1 b_{\lambda,j}^1 = 0, \quad j \in \mathcal{A}_\lambda^1. \quad (11)$$

We first prove for  $\hat{\beta}_\lambda$ . Eliminating  $b_{\lambda,j}^\infty$  from (9) through (4), we obtain

$$-2 \sum_{j \in \mathcal{A}_\lambda^\infty} \text{sign}(\hat{\beta}_{\lambda,j}) \sum_{i=1}^n x_{ij} \left( y_i - \sum_{k \in \mathcal{I}_\lambda} x_{ik} \hat{\beta}_{\lambda,k} - \sum_{k \in \mathcal{A}_\lambda^\infty} x_{ik} \text{sign}(\hat{\beta}_{\lambda,k}) \|\hat{\beta}_\lambda\|_\infty \right) + \text{card}(\mathcal{A}_\lambda^\infty) \lambda_1 + \lambda_\infty = 0.$$

Solving this equation and (10) for  $\hat{\beta}_{\lambda,j}$ ,  $j \in \mathcal{I}_\lambda$  and  $\|\hat{\beta}_\lambda\|_\infty$  shows that they are linear in  $\lambda$ . This proves that  $\hat{\beta}_\lambda$  is piecewise linear in  $\lambda$ . Using this solution for  $\hat{\beta}_\lambda$ , we can now solve (9) for  $b_{\lambda,j}^\infty$ ,  $j \in \mathcal{A}_\lambda^\infty$  and (11) for  $b_{\lambda,j}^1$ ,  $j \in \mathcal{A}_\lambda^1$ , and this implies that  $b_\lambda^\infty$  is piecewise linear in  $(\lambda_1/\lambda_\infty, 1/\lambda_\infty)^T$  and  $b_\lambda^1$  is piecewise linear in  $(\lambda_\infty/\lambda_1, 1/\lambda_1)^T$ .  $\square$

**Proof of Theorem 2.** In an interval on which the index sets remain unchanged, (6) can

be written as

$$-\sum_{i \in \mathcal{H}_\lambda} \alpha_{\lambda,i} y_i x_{ij} - \sum_{i \in \mathcal{L}_\lambda} y_i x_{ij} + \lambda_1 \text{sign}(\hat{\beta}_{\lambda,j}) + \lambda_\infty \text{sign}(\hat{\beta}_{\lambda,j}) |b_{\lambda,j}^\infty| = 0, \quad j \in \mathcal{A}_\lambda^\infty \quad (12)$$

$$-\sum_{i \in \mathcal{H}_\lambda} \alpha_{\lambda,i} y_i x_{ij} - \sum_{i \in \mathcal{L}_\lambda} y_i x_{ij} + \lambda_1 \text{sign}(\hat{\beta}_{\lambda,j}) = 0, \quad j \in \mathcal{I}_\lambda \quad (13)$$

$$-\sum_{i \in \mathcal{H}_\lambda} \alpha_{\lambda,i} y_i x_{ij} - \sum_{i \in \mathcal{L}_\lambda} y_i x_{ij} + \lambda_1 b_{\lambda,j}^1 = 0, \quad j \in \mathcal{A}_\lambda^1 \quad (14)$$

$$\sum_{i \in \mathcal{H}_\lambda} \alpha_{\lambda,i} y_i + \sum_{i \in \mathcal{L}_\lambda} y_i = 0. \quad (15)$$

For these  $\text{card}(\mathcal{H}_\lambda)$  equations to solve for  $\text{card}(\mathcal{I}_\lambda) + 2$  number of unknowns,  $\hat{\beta}_{\lambda,j}$ ,  $j \in \mathcal{I}_\lambda$ ,  $\hat{\beta}_{\lambda,0}$ , and  $\|\hat{\beta}_\lambda\|_\infty$ , we impose  $\text{card}(\mathcal{H}_\lambda) = \text{card}(\mathcal{I}_\lambda) + 2$ . From the fact that this system of equations is independent of  $\hat{\beta}_{\lambda,j}$ ,  $j \in \mathcal{I}_\lambda$ ,  $\hat{\beta}_{\lambda,0}$ , and  $\|\hat{\beta}_\lambda\|_\infty$ , it follows that  $\hat{\beta}_\lambda$  and  $\hat{\beta}_{\lambda,0}$  are piecewise constant. Eliminating  $b_{\lambda,j}^\infty$  from (12) through (4), we obtain

$$\sum_{j \in \mathcal{A}_\lambda^\infty} \text{sign}(\hat{\beta}_{\lambda,j}) \left( \sum_{i \in \mathcal{H}_\lambda} \alpha_{\lambda,i} y_i x_{ij} - \sum_{i \in \mathcal{L}_\lambda} y_i x_{ij} \right) + \text{card}(\mathcal{A}_\lambda^\infty) \lambda_1 + \lambda_\infty = 0.$$

Solving this equation, (13), and (15) for  $\alpha_{\lambda,i}$ ,  $i \in \mathcal{H}_\lambda$  yields that they are linear in  $\lambda$ , which proves that  $\alpha_\lambda$  is piecewise linear in  $\lambda$ . Using these solutions for  $\hat{\beta}_\lambda$  and  $\alpha_\lambda$  and solving (12) for  $b_{\lambda,j}^\infty$ ,  $j \in \mathcal{A}_\lambda^\infty$  and (14) for  $b_{\lambda,j}^1$ ,  $j \in \mathcal{A}_\lambda^1$  implies that  $b_\lambda^\infty$  is piecewise linear in  $(\lambda_1/\lambda_\infty, 1/\lambda_\infty)^T$  and  $b_\lambda^1$  is piecewise linear in  $(\lambda_\infty/\lambda_1, 1/\lambda_1)^T$ .  $\square$

**Proof of Theorem 3.** This proof employs Theorems 1 and 2 in Zou et al. (2007), and Theorem 5. Following the notation in Theorem 5, let  $X^c$  be the matrix whose rows are  $x_i^{cT}$ ,  $i = 1, \dots, n$ . If  $\lambda$  is not a transition point, then  $d(\sum_{i=1}^n (y_i - x_i^{cT} \hat{\beta}_\lambda^c)^2 + \sum_{j=1}^{\text{card}(\mathcal{I}_\lambda)+1} \lambda_j^c |\hat{\beta}_{\lambda,j}^c|) / d\beta_\lambda^c = 0$  yields  $\hat{\beta}_\lambda^c(y) = (X^{cT} X^c)^{-1} (X^{cT} y - w/2)$ , where the vector  $w = (\text{sign}(\hat{\beta}_{\lambda,1}) \lambda_1, \dots, \text{sign}(\hat{\beta}_{\lambda, \text{card}(\mathcal{I}_\lambda)}) \lambda_1, \text{card}(\mathcal{A}_\lambda^\infty) \lambda_1 + \lambda_\infty)^T$ . Observe that  $\hat{\mu}_\lambda(y) = X^c \hat{\beta}_\lambda^c(y) = P_\lambda(y) y - W_\lambda(y)$ , where  $P_\lambda(y) = X^c (X^{cT} X^c)^{-1} X^{cT}$  and  $W_\lambda(y) = X^c (X^{cT} X^c)^{-1} w/2$ .

Now we compute an infinitesimal change of  $\hat{\mu}$ , denoted  $\Delta \hat{\mu}$ , when  $y$  changes infinitesimally, which is essential to apply Stein's lemma. By Theorem 1 in Zou et al. (2007), there

exists a sufficiently small  $\varepsilon$  such that  $\|\Delta y\|_2 < \varepsilon$  keeping the index sets unchanged. Accordingly, for such a sufficiently small change of  $y$ , we have  $P_\lambda(y + \Delta y) = P_\lambda(y)$  and  $W_\lambda(y + \Delta y) = W_\lambda(y)$ , and hence  $\partial \hat{\mu}_\lambda(y) / \partial y = P_\lambda(y)$ . By Theorem 2 in Zou et al. (2007),  $\hat{\mu}_\lambda(y)$  is almost differentiable with respect to  $y$ . Then by Stein's lemma (Stein, 1981), we obtain  $\widehat{\text{df}}(\hat{\mu}_\lambda) = \text{tr}(\partial \hat{\mu}_\lambda(y) / \partial y) = \text{tr}(P_\lambda(y)) = \text{card}(\mathcal{I}_\lambda) + 1$  and  $\text{df}(\hat{\mu}_\lambda) = \text{E}[\text{card}(\mathcal{I}_\lambda) + 1]$ .  $\square$

**Proof of Theorem 4.** Suppose  $|c(\hat{\beta}_{\lambda,j})| > \lambda_1/2$ . Then from (5),  $|b_{\lambda,j}^1| < 1$  implies  $|b_{\lambda,j}^\infty| > 0$ . On the other hand,  $|b_{\lambda,j}^1| < 1$  implies  $|\hat{\beta}_{\lambda,j}| = 0$  and hence  $|\hat{\beta}_{\lambda,j}| < \|\hat{\beta}_\lambda\|_\infty$ , which means  $b_{\lambda,j}^\infty = 0$  because  $\hat{\beta}_\lambda \neq 0$  by the assumption. Thus  $|b_{\lambda,j}^1| < 1$  does not satisfy  $|c(\hat{\beta}_{\lambda,j})| > \lambda_1/2$ , and we must have  $|b_{\lambda,j}^1| = 1$ . Then  $|b_{\lambda,j}^\infty| > 0$  and hence  $|\hat{\beta}_{\lambda,j}| = \|\hat{\beta}_\lambda\|_\infty$ . Now the characteristics of  $b_{\lambda,j}^1$  and  $b_{\lambda,j}^\infty$  imply  $\text{sign}(\hat{\beta}_{\lambda,j}) = \text{sign}(b_{\lambda,j}^1) = \text{sign}(b_{\lambda,j}^\infty)$ ,  $j \in \mathcal{A}_\lambda^\infty$ . Then from (5), we obtain  $\text{sign}(\hat{\beta}_{\lambda,j}) = \text{sign}(c(\hat{\beta}_{\lambda,j}))$ ,  $j \in \mathcal{A}_\lambda^\infty$ . Since  $|c(\hat{\beta}_{\lambda,j})| > \lambda_1/2$  is equivalent to  $j \in \mathcal{A}_\lambda^\infty$ ,  $\hat{\beta}_{\lambda,j} = \text{sign}(c(\hat{\beta}_{\lambda,j}))\|\hat{\beta}_\lambda\|_\infty$  if  $|c(\hat{\beta}_{\lambda,j})| > \lambda_1/2$ .

In the orthonormal case, (9)–(11) become

$$-2 \sum_{i=1}^n x_{ij}y_i + 2\hat{\beta}_{\lambda,j} + \lambda_1 \text{sign}(\hat{\beta}_{\lambda,j}) + \lambda_\infty \text{sign}(\hat{\beta}_{\lambda,j})|b_{\lambda,j}^\infty| = 0, \quad j \in \mathcal{A}_\lambda^\infty \quad (16)$$

$$-2 \sum_{i=1}^n x_{ij}y_i + 2\hat{\beta}_{\lambda,j} + \lambda_1 \text{sign}(\hat{\beta}_{\lambda,j}) = 0, \quad j \in \mathcal{I}_\lambda \quad (17)$$

$$-2 \sum_{i=1}^n x_{ij}y_i + \lambda_1 b_{\lambda,j}^1 = 0, \quad j \in \mathcal{A}_\lambda^1. \quad (18)$$

Applying  $\text{sign}(\hat{\beta}_{\lambda,j}) = \text{sign}(\sum_{i=1}^n x_{ij}y_i)$ ,  $j \in \mathcal{I}_\lambda$ , from (17) we get  $\hat{\beta}_{\lambda,j} = \sum_{i=1}^n x_{ij}y_i - (\lambda_1/2) \text{sign}(\sum_{i=1}^n x_{ij}y_i)$ ,  $j \in \mathcal{I}_\lambda$ . The requirement  $|b_{\lambda,j}^1| \leq 1$  in (18) yields  $|\sum_{i=1}^n x_{ij}y_i| \leq \lambda_1/2$  if and only if  $j \in \mathcal{A}_\lambda^1$  because  $|\sum_{i=1}^n x_{ij}y_i| > \lambda_1/2$  implies  $\hat{\beta}_\lambda \neq 0$ . Since  $j \in \mathcal{I}_\lambda \cup \mathcal{A}_\lambda^1$  is equivalent to  $|c(\hat{\beta}_{\lambda,j})| \leq \lambda_1/2$ , we obtain

$$\hat{\beta}_{\lambda,j} = \left[ \sum_{i=1}^n x_{ij}y_i - \frac{\lambda_1}{2} \right]_+ \text{sign}(c(\hat{\beta}_{\lambda,j})), \quad |c(\hat{\beta}_{\lambda,j})| \leq \frac{\lambda_1}{2}.$$

Observe that  $\hat{\beta}_{\lambda,j} = \|\hat{\beta}_\lambda\|_\infty \text{sign}(\sum_{i=1}^n x_{ij}y_i)$ ,  $j \in \mathcal{A}_\lambda^\infty$ . This allows us to write (16) as

$-2|\sum_{i=1}^n x_{ij}y_i| + 2\|\hat{\beta}_\lambda\|_\infty + \lambda_1 + \lambda_\infty|b_{\lambda,j}^\infty| = 0$ ,  $j \in \mathcal{A}_\lambda^\infty$ . Now through (4), we obtain

$$\hat{\beta}_{\lambda,j} = \frac{1}{\text{card}(\mathcal{A}_\lambda^\infty)} \left( \sum_{j \in \mathcal{A}_\lambda^\infty} \left( \sum_{i=1}^n x_{ij}y_i - \frac{\lambda_1}{2} \right) - \frac{\lambda_\infty}{2} \right) \text{sign}(c(\hat{\beta}_{\lambda,j})), \quad |c(\hat{\beta}_{\lambda,j})| > \frac{\lambda_1}{2}.$$

□

**Proof of Theorem 5.** The proof is straightforward hence omitted. □

## REFERENCES

- [1] ALLGOWER E. L AND GEORGE, K. (2003). *Introduction to Numerical Continuation Methods*. The Society for Industrial and Applied Mathematics.
- [2] BICKEL, P. J., RITOV, Y. AND TSYBAKOV A. B. (2008). Simultaneous analysis of lasso and dantzig selector. *Unpublished manuscript. Available at arxiv.org/abs/0801.1095v1*.
- [3] BONDELL, H. D. AND REICH, B. J. (2008). Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with OSCAR. *Biometrics*, **64**, 115–123.
- [4] BRADLEY, P. S. AND MANGASARIAN, O. L. (1998). Feature selection via concave minimization and support vector machines. *Machine Learning Proceedings of the Fifteenth International Conference(ICML 1998)*, 82–90.
- [5] BREIMAN, L. AND SPECTOR, P. (1992). Submodel selection and evaluation in regression - the X random case. *Technometrics* **60**, 291–319.
- [6] CANDÈS, E. AND TAO, T. (2007). The Dantzig selector: Statistical estimation when  $p$  is much larger than  $n$ . *Annals of Statistics* **35**, 2313-2351.

- [7] DONOHO, D. L., ELAD, M. AND TEMLYAKOV V. N. (2006). Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory* **52**, 6–18.
- [8] DUDOIT, S., FRIDLAND, J. AND SPEED, T. P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association* **97**, 77–87.
- [9] EFRON, B., JOHNSTONE, I., HASTIE, T. AND TIBSHIRANI, R. (2004). Least angle regression. *Annals of Statistics* **32**, 407–499.
- [10] FAN, J. AND LV, J. (2006). Sure independence screening for ultra-high dimensional feature space. *Unpublished manuscript. Available at [arxiv.org/abs/math/0612857v1](http://arxiv.org/abs/math/0612857v1)*.
- [11] GOLUB, T., SLONIM, D., TAMAYO, P., HUARD, C., GAASENBEEK, M., MESIROV, J., COLLER, H., LOH, M., DOWNING, J. AND CALIGIURI, M. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* **286**, 513–536.
- [12] HASTIE, T., ROSSET, S., TIBSHIRANI, R. AND ZHU, J. (2004). The entire regularization path for the support vector machine. *Journal of Machine Learning Research* **5**, 1391–1415.
- [13] HOERL, A. AND KENNARD, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* **12**, 55–67.
- [14] LIU, Y. AND WU, Y. (2007). Variable selection via a combination of the  $L_0$  and  $L_1$  penalties. *Journal of Computational and Graphical Statistics* **16**, 782–798.
- [15] PARK, M. AND HASTIE, T..  $L_1$  regularization-path algorithm for generalized linear models. *Journal of the Royal Statistical Society, Series B* **69**, 659–677.

- [16] ROCKAFELLAR, R. T. (1970). *Convex Analysis*. Princeton University Press.
- [17] ROCKAFELLAR, R. T. AND WETS, R. J. (2003). *Variational Analysis*. Springer-Verlag.
- [18] ROSSET, S. AND ZHU J. (2007). Piecewise linear regularized solution paths. *Annals of Statistics*. To appear.
- [19] SHEN, X. AND HUANG, H. -C. (2006). Optimal model assessment, selection and combination. *Journal of the American Statistical Association* **101**, 554–568.
- [20] STEIN, C. (1981). Estimation of the mean of a multivariate normal distribution. *Annals of Statistics* **9**, 1135–1151.
- [21] TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B* **58**, 267–288.
- [22] TIBSHIRANI, R., SAUNDERS, M., ROSSET, S., ZHU, J. AND KNIGHT, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society, Series B* **67**, 91–108.
- [23] VAPNIK, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag.
- [24] WANG, J. AND SHEN, X. (2006). Estimation of generalization error: random and fixed inputs. *Statistica Sinica* **16**, 569–588.
- [25] WANG, L. AND SHEN, X. (2007). On L1-norm multi-class support vector machines: methodology and theory. *Journal of the American Statistical Association*. To appear.
- [26] WANG, L., ZHU, J. AND ZOU, H. (2006). The doubly regularized support vector machine. *Statistica Sinica* **16**, 589–616.
- [27] WANG, S. AND ZHU, J. (2008). Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, **64**, 440–448.

- [28] WOLBERG, W. H. AND MANGASARIAN, O. L. (1990). Multisurface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology. *Proceedings of the National Academy of Sciences* **87**, 9193–9196.
- [29] YUAN, M. AND LIN, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B* **68**, 49–67.
- [30] ZHU, J., ROSSET S., HASTIE T. AND TIBSHIRANI R. (2003). 1-norm support vector machines. *Neural Information Processing Systems 2003*.
- [31] ZOU, H. AND HASTIE, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B* **67**, 301–320.
- [32] ZOU, H., HASTIE, T. AND TIBSHIRANI, R. (2007). On the “degrees of freedom” of the Lasso. *Annals of Statistics*. To appear.

Table 1: Regression. Averaged test errors and standard errors (in parenthesis), and number of distinct nonzero coefficients (in square brackets) of the four methods with optimal tuning by Oracle and GDF, based on 100 simulation replications. The smallest error over the four methods is underlined.

Ex	Model selection	Method			
		$L_1L_\infty$	Elastic	Lasso	$L_\infty$
1	Oracle	10.35(0.11)[7.73]	<u>10.32</u> (0.09)[7.16]	10.42(0.11)[7.58]	10.97(0.12)[9.72]
	GDF	11.14(0.16)[6.30]	11.12(0.15)[5.69]	<u>10.96</u> (0.13)[6.82]	11.55(0.16)[9.42]
2	Oracle	<u>9.65</u> (0.08)[4.52]	10.85(0.11)[9.22]	10.91(0.11)[9.15]	9.67(0.08)[4.15]
	GDF	10.35(0.11)[3.31]	11.83(0.09)[7.60]	11.31(0.11)[8.46]	<u>10.13</u> (0.12)[3.53]
3	Oracle	<u>86.91</u> (0.41)[9.61]	91.145(0.57)[13.17]	91.53(0.59)[13.39]	88.78(0.41)[9.41]
	GDF	<u>90.78</u> (0.55)[5.57]	97.30(0.83)[10.15]	93.60(0.71)[12.95]	92.40(0.57)[7.53]
4	Oracle	<u>10.36</u> (0.10)[9.78]	10.67(0.10)[9.58]	10.76(0.10)[9.60]	11.47(0.17)[12.53]
	GDF	<u>11.38</u> (0.14)[8.34]	11.57(0.14)[8.36]	11.43(0.15)[8.58]	12.17(0.20)[12.74]
5	Oracle	35.16(1.21)[33.02]	<u>34.33</u> (1.17)[40.78]	39.76(1.07)[38.70]	72.31(0.71)[31.85]
	GDF	<u>39.38</u> (1.79)[28.92]	39.76(1.98)[30.30]	42.49(1.14)[37.57]	78.66(1.13)[44.45]

Table 2: Binary SVM classification. Averaged test errors and standard errors (in parenthesis) of the four SVMs with optimal tuning by GDF over 100 simulation replications. The smallest error over the four methods is underlined.

Ex	$n$ & $p$	SVM			
		$L_1L_\infty$	$L_1$	$L_2$	$L_\infty$
1	$n \gg p$	<u>0.133</u> (0.001)	0.143(0.001)	0.145(0.001)	0.145(0.002)
	$n \ll p$	<u>0.167</u> (0.003)	0.197(0.005)	0.321(0.003)	0.366(0.005)
2	$n \gg p$	<u>0.138</u> (0.001)	0.142(0.001)	0.139(0.001)	0.143(0.001)
	$n \ll p$	<u>0.140</u> (0.001)	0.147(0.001)	0.175(0.001)	0.298(0.005)
3	$n \gg p$	<u>0.118</u> (0.001)	0.120(0.001)	0.121(0.001)	0.123(0.001)
	$n \ll p$	<u>0.121</u> (0.001)	0.131(0.002)	0.162(0.001)	0.283(0.005)

Table 3: Averaged test errors and standard errors (in parenthesis) of the four SVMs for WBCD over 100 random partitions.

SVM	TE
$L_1L_\infty$	0.025(0.001)
$L_1$	0.028(0.001)
$L_2$	0.026(0.001)
$L_\infty$	0.027(0.001)

Table 4: Test errors and numbers of selected genes for four SVMs for the leukemia data.

SVM	TE	# Genes
$L_1L_\infty$	0/34	65
$L_1$	1/34	16
$L_2$	2/34	1059
$L_\infty$	2/34	1059

Table 5: Comparison of averaged test errors and numbers of distinct nonzero coefficients (DNC) of  $L_1L_\infty$ -penalty and the elastic net with optimal tuning by GDF over 100 simulation replications.

Method	TE	# DNC
$L_1L_\infty$	3.450(0.151)	4.52
Elastic net	5.373(0.272)	8.19