Sparse optimization for nonconvex group penalized estimation

Sangin Lee[1], Miae Oh[2] and Yongdai Kim[3]

[1] *University of Iowa,* [2] *Korea Institute for Health and Social Affairs and* [3] *Seoul National University*

Abstract

We consider a linear regression model where there are group structures in covariates. The group LASSO has been proposed for group variable selections. Many nonconvex penalties such as SCAD and MCP were extended to group variable selection problems. The group coordinate descent (GCD) algorithm is used popularly for fitting these models. However, the GCD algorithms are hard to be applied to nonconvex group penalties due to computational complexity unless the design matrix is orthogonal. In this paper, we propose an efficient optimization algorithm for nonconvex group penalties by combining the concave convex procedure and the group LASSO algorithm. We also extend the proposed algorithm for generalized linear models. We evaluate numerical efficiency of the proposed algorithm compared to existing GCD algorithms. In addition, we perform simulation studies and real data analysis to compare the nonconvex group penalties and the group LASSO.

Key words: Concave convex procedure, group LASSO, nonconvex penalty, variable selection.

1 Introduction

We consider the linear regression model with K groups of covariates

$$\mathbf{y} = \sum_{k=1}^{K} \mathbf{X}_k \boldsymbol{\beta}_k + \boldsymbol{\varepsilon}, \tag{1}$$

where $\mathbf{y} = (y_1, \ldots, y_n)'$ is the vector of response variables, $\mathbf{X}_k$ is the $n \times p_k$ design matrix corresponding to the kth group with $\mathbf{x}_{ik}, i = 1, \ldots, n$ being $p_k \times 1$ vector, $\boldsymbol{\beta}_k = (\beta_{k1}, \ldots, \beta_{kp_k})' \in \mathbb{R}^{p_k}$ is the vector of the regression coefficients in the kth group and $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)'$ is the random error vector with $E(\boldsymbol{\varepsilon}) = 0$ and $Var(\boldsymbol{\varepsilon}) = \sigma^2 \mathbf{I}_n$ for some $0 < \sigma^2 < \infty$. Here, the total number of covariates is $p = \sum_{k=1}^{K} p_k$.

In many regression problems, explanatory variables can often be naturally grouped. For example, categorical covariates are represented by a group of dummy variables. In nonparametric additive models, continuous covariates can be represented by a linear combination of a set of basis functions. In these cases, we are naturally interested in selecting important groups of covariates rather than individual ones.

There has been much work on the penalized method for group variable selection. Yuan and Lin (2006) proposed the group LASSO as a natural extension of the LASSO (Tibshirani, 1996). Kim *et al.* (2006) developed the blockwise sparse regression, which is an extension of the group LASSO for generalized linear models. The group LASSO penalty uses the ℓ_2-norm of the coefficients within each group. Several authors have studied theoretical properties of the group LASSO, but found that a certain irrepresentable condition is required for group selection consistency. (Bach, 2008; Huang and Zhang, 2010; Wei and

Huang, 2010). To overcome this drawback, nonconvex penalized methods have been proposed for group variable selection. Wang *et al.* (2007) and Huang *et al.* (2012a) proposed the group smoothly clipped absolute deviation (SCAD) penalty and group minimax concave penalty (MCP), respectively. They showed that these nonconvex methods satisfy the oracle property in group selection (Huang *et al.*, 2012b).

In this paper, we consider an optimization algorithm for nonconvex group penalized methods. The group coordinate descent (GCD) algorithm has been used in Yuan and Lin (2006) for the group LASSO and Wei and Zhu (2012) for the group MCP, assuming that the design matrix of each group is orthogonal. However, Friedman *et al.* (2010) pointed out that the solution with the orthogonality assumption will not be a solution of the original problem. The GCD algorithm is not easy to use without the orthogonality assumption since the closed form solution in each iteration does not exist. For the group LASSO, Foygel and Drton (2010) and Qin *et al.* (2010) proposed the GCD algorithm without the orthogonal assumption. For the group SCAD, Wang *et al.* (2007) used the local quadratic approximation (LQA) algorithm which does not require the orthogonality assumption. However, the LQA algorithm is incapable of producing an exact sparse solution since it relies on the quadratic approximation of the penalty function, and is computationally inefficient since it requires the repeated factorization of large matrices. With authors's knowledge, there is no optimization algorithm which gives an exact solution for the group SCAD and group MCP without the orthogonality assumption.

In this paper, we consider the class of the nonconvex penalties including the group SCAD and group MCP, and propose an optimization algorithm which can be applied to all of the penalties in this class. The main idea of the proposed algorithm is that we convert the nonconvex group penalty to the group LASSO penalty via the concave convex procedure (CCCP) in Yuille and Rangarajan (2003), and then we apply the GCD algorithm of Foygel and Drton (2010). The proposed method is easy to implement and always converges to a local minimum. In addition, we extend the proposed algorithm for generalized linear models.

This paper is organized as follows. In Section 2, we describe the existing penalties and corresponding GCD algorithms. In Section 3, we introduce the class of nonconvex penalties and propose an optimization algorithm. Section 4 presents numerical efficiencies of the existing GCD algorithm and proposed algorithm through analysing simulated as well as real data sets.

Concluding remarks are presented in Section 5.

2 Group coordinate descent algorithm

2.1 Group LASSO

For a given $\lambda > 0$, the group LASSO estimator proposed by Yuan and Lin (2006) is defined as the minimizer of

$$Q_\lambda(\boldsymbol{\beta}) = \frac{1}{2n}\Big\|\mathbf{y} - \sum_{k=1}^{K}\mathbf{X}_k\boldsymbol{\beta}_k\Big\|_2^2 + \lambda\sum_{k=1}^{K}\sqrt{p_k}\|\boldsymbol{\beta}_k\|_2, \tag{2}$$

where $\lambda > 0$ is a regularization parameter and $\|\cdot\|_2$ stands for the ℓ_2-norm. For a computation of the group LASSO, Yuan and Lin (2006) used the GCD algorithm which is a natural extension of the coordinate descent algorithm (Fu, 1998; Friedman *et al.*, 2007). The GCD algorithm optimizes the objective function with respect to each group iteratively until the solution converges. To explain the GCD algorithm for the group LASSO, we consider the group coordinate step. That is, for fixed coefficients $(\tilde{\boldsymbol{\beta}}_l,\ l \neq k)$, we are to minimize (2) with respect to the kth group coefficients $\boldsymbol{\beta}_k$. Using some algebra, it can be shown that this problem is equivalent to minimizing $\tilde{Q}_\lambda(\boldsymbol{\beta}_k)$ defined as

$$\tilde{Q}_\lambda(\boldsymbol{\beta}_k) = \frac{1}{2n}\Big\|\mathbf{r}_k - \mathbf{X}_k\tilde{\boldsymbol{\beta}}_k\Big\|_2^2 + \lambda\sqrt{p_k}\|\boldsymbol{\beta}_k\|_2 \tag{3}$$

where $\mathbf{r}_k = \mathbf{y} - \sum_{l\neq k}\mathbf{X}_l\tilde{\boldsymbol{\beta}}_l$ is the partial residual vector. With the orthogonal assumption for each group, i.e., $\mathbf{X}_k'\mathbf{X}_k/n = \mathbf{I}_{p_k}$, Yuan and Lin (2006) showed that the minimizer of $\tilde{Q}_\lambda(\boldsymbol{\beta}_k)$ in (3) has the explicit form as

$$\hat{\boldsymbol{\beta}}_k = \left(1 - \frac{\lambda\sqrt{p_k}}{\|\mathbf{s}_k\|_2}\right)_+ \mathbf{s}_k, \tag{4}$$

where $\mathbf{s}_k = \mathbf{X}_k'\mathbf{r}_k/n$ and the subscript '+' indicates the positive part. The group LASSO solution can be obtained by iteratively applying (4) to each group until it converges. If the covariates in each group are not orthogonal, this approach requires to orthogonalize them before applying the GCD algorithm. However, as noted by Friedman *et al.* (2010), it will not provide the solution of the original problem.

Foygel and Drton (2010) and Qin *et al.* (2010) proposed the GCD algorithm which does not require the orthogonalization. They showed that the exact solution for any single group problem in (3) without the orthogonality assumption can be obtained by an efficient application of Newton's method as in the Algorithm 1.

2.2 Group SCAD and group MCP

The group LASSO is constructed by replacing the LASSO penalty with the ℓ_2-norms of coefficients within the groups. The group LASSO has many attractive properties, but it does not possess the group level selection consistency. In fact, it tends to select more groups (variables) than necessary. Nonconvex penalized methods satisfying the oracle property have been proposed for group variable selection. A

Algorithm 1 Group coordinate descent algorithm for group LASSO

Choose any initial vector $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^p$ and compute the eigen-decomposition of $\mathbf{X}_k'\mathbf{X}_k/n = \mathbf{U}_k'\mathbf{D}_k\mathbf{U}_k$ with $\mathbf{D}_k = \text{diag}\{d_1^k, \ldots, d_{p_k}^k\}$, for all k.

repeat

  for $k = 1, \ldots, K$ **do**

    Calculate $\mathbf{r}_k = \mathbf{y} - \sum_{l \neq k} \mathbf{X}_l \tilde{\boldsymbol{\beta}}_l$ and $\mathbf{v}_k = \mathbf{U}_k \mathbf{X}_k' \mathbf{r}_k$.

    if $\|\mathbf{v}_k\|_2 \leq \lambda\sqrt{p_k}$ **then**

      $\hat{\boldsymbol{\beta}}_k = 0$.

    else

      Find the unique $\delta > 0$ satisfying $f(\delta) = \sum_{j=1}^{p_k} (\mathbf{v}_k)_j^2/(d_j^k\delta + \lambda\sqrt{p_k})^2 = 1$.

      Update $\hat{\boldsymbol{\beta}}_k = \mathbf{U}_k'(\mathbf{D}_k + \delta^{-1}\lambda\sqrt{p_k} I_{p_k})^{-1}\mathbf{v}_k$.

    end if

  end for

  Update $\tilde{\boldsymbol{\beta}}$ by $\hat{\boldsymbol{\beta}}$.

until convergence.

nonconvex group penalized estimator is defined as the minimizer of

$$Q_\lambda(\boldsymbol{\beta}) = \frac{1}{2n}\Big\|\mathbf{y} - \sum_{k=1}^{K} \mathbf{X}_k\boldsymbol{\beta}_k\Big\|_2^2 + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\beta}_k\|_2), \tag{5}$$

where $J_{\lambda_k}(\cdot)$ are the nonconvex penalty functions and λ_k are the regularization parameters. We set $\lambda_k = \lambda\sqrt{p_k}$ for some $\lambda > 0$. If $J_\lambda(\cdot)$ is the SCAD penalty in Fan and Li (2001), the estimator becomes the group SCAD (Wang *et al.*, 2007). On the other hand, the group MCP estimator of Huang *et al.* (2012b) is obtained by using the MCP of Zhang (2010) for $J_\lambda(\cdot)$.

Breheny and Huang (2012) and Wei and Zhu (2012) extended the GCD algorithm of Yuan and Lin (2006) to find the group SCAD and group MCP estimators. They obtained the explicit form of the solution for each iteration under the orthogonality assumption within each group. The closed form solutions of the group SCAD and group MCP in the kth group are given as follows:

$$\hat{\boldsymbol{\beta}}_k^{gMCP} = \begin{cases} \frac{a}{a-1} S(\mathbf{s}_k, \lambda_k) & \text{if } \|\mathbf{s}_k\|_2 \leq a\lambda_k, \\ \mathbf{s}_k & \text{if } \|\mathbf{s}_k\|_2 > a\lambda_k, \end{cases} \tag{6}$$

$$\hat{\boldsymbol{\beta}}_k^{gSCAD} = \begin{cases} S(\mathbf{s}_k, \lambda_k) & \text{if } \|\mathbf{s}_k\|_2 \leq 2\lambda_k, \\ \frac{a-1}{a-2} S\big(\mathbf{s}_k, \frac{a\lambda_k}{a-1}\big) & \text{if } 2\lambda_k < \|\mathbf{s}_k\|_2 \leq a\lambda_k, \\ \mathbf{s}_k & \text{if } \|\mathbf{s}_k\|_2 > a\lambda_k, \end{cases} \tag{7}$$

for some $a > 2$ in the group SCAD and $a > 1$ in the group MCP, where $S(\mathbf{z}, \lambda) = (1 - \lambda/\|\mathbf{z}\|_2)_+\mathbf{z}$ is the multivariate soft-thresholding operator. The GCD algorithm can be easily applied to the group SCAD and group MCP by the update rules in (6) and (7). Since these update rules are simple, the

corresponding algorithms are efficient and stable. Also, Tseng (2001) established that the GCD algorithm has the descent property, which means that the objective function decreases in each iteration, hence the algorithms converge to a local minimum (Breheny and Huang, 2012; Wei and Zhu, 2012).

2.3 Discussion for the orthogonality

The algorithm of Section 2.2 can not be directly applied to a model without the orthogonality of each group. For general cases, the algorithm can be applied to the transformed design matrix after orthogonalizing the design matrix for each group as follows. Each group of covariates $\mathbf{X}_k$ can be orthogonalized by using a Cholesky decomposition, i.e., $\mathbf{X}_k'\mathbf{X}_k/n = \mathbf{U}_k'\mathbf{U}_k, \forall k$, where $\mathbf{U}_k$ is an upper triangular matrix. Let $\mathbf{Z}_k = \mathbf{X}_k\mathbf{U}_k^{-1}$ and $\boldsymbol{\theta}_k = \mathbf{U}_k\boldsymbol{\beta}_k$ so that $\mathbf{Z}_k'\mathbf{Z}_k/n = \mathbf{I}_{p_k}$ and $\mathbf{Z}_k\boldsymbol{\theta}_k = \mathbf{X}_k\boldsymbol{\beta}_k$. Then, the algorithm can be applied to the transformed minimization problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \left\{ \frac{1}{2n} \left\| \mathbf{y} - \sum_{k=1}^{K} \mathbf{Z}_k\boldsymbol{\theta}_k \right\|_2^2 + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\theta}_k\|_2) \right\}, \tag{8}$$

where $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_K)$ is the solution with the transformed design matrix. Finally, the solution $\hat{\boldsymbol{\theta}}$ is transformed back to the original problem with $\hat{\boldsymbol{\beta}}_k = \mathbf{U}_k^{-1}\boldsymbol{\theta}_k, \forall k$. As mentioned earlier, the resulting solution is not the minimizer of $Q_\lambda(\boldsymbol{\beta})$ in (5), which is the objective function with original covariates. In fact, the objective function in (8) is exactly same as the objective function

$$\frac{1}{2n} \left\| \mathbf{y} - \sum_{k=1}^{K} \mathbf{X}_k\boldsymbol{\beta}_k \right\|_2^2 + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\beta}_k\|_{\Sigma_k}), \tag{9}$$

where $\|\boldsymbol{\beta}_k\|_{\Sigma_k} = (\boldsymbol{\beta}_k'\Sigma_k\boldsymbol{\beta}_k)^{1/2}$ and $\Sigma_k = \mathbf{X}_k'\mathbf{X}_k/n$. Without the orthogonality assumption, the existing GCD algorithm gives a minimization of (9), not that of the original problem $Q_\lambda(\boldsymbol{\beta})$ in (5).

3 The proposed algorithm

3.1 Class of nonconvex penalties

We first introduce the class of nonconvex penalties considered in Kim and Kwon (2012). Let $\nabla J_\lambda(t)$ be the first derivative of $J_\lambda(t)$ with respect to t. Consider a class of nonconvex penalties $J_\lambda(\cdot)$ that satisfy the following three conditions

(P1) $\nabla J_\lambda(\cdot)$ is nonnegative, nonincreasing and continuous over $(0, \infty)$,

(P2) $\lim_{t\to 0+} \nabla J_\lambda(t) = \lambda$ and $\nabla J_\lambda(t) = 0$ for $t \geq a\lambda$,

(P3) $\nabla J_\lambda(t) \geq (\lambda - t/a)_+ I(0 < t < a\lambda)$ for $t > 0$,

for some $a > 0$. This class includes the SCAD penalty (Fan and Li, 2001)

$$\nabla J_\lambda(t) = \lambda I(0 < t < \lambda) + (a\lambda - t)_+/(a-1) I(t \geq \lambda),$$

and MCP (Zhang, 2010)

$$\nabla J_\lambda(t) = (\lambda - t/a)_+ I(0 < t < a\lambda).$$

Note that the penalties in this class satisfy the oracle property. Also, the CCCP algorithm can be applied to the penalties in this class since $\tilde{J}_\lambda(t) = J_\lambda(t) - \lambda|t|$ is always a differentiable concave function (Kim *et al.*, 2008; Lee *et al.*, 2012).

3.2 Proposed optimization algorithm

The GCD algorithm described in Section 2.2 can be applied only to the models when the design matrix of each group is orthogonal. If the design matrices are not orthogonal, there is no closed form solution in each iteration, and hence the GCD algorithm can not be directly applied to the nonconvex group penalized methods.

In this section, we propose an optimization algorithm which can be applied to nonconvex group penalties without the orthogonality assumption. The main idea of the proposed algorithm is that we convert the objective function to a convex problem via the CCCP, and then apply the GCD algorithm of Foygel and Drton (2010).

The CCCP algorithm of Yuille and Rangarajan (2003) is one of the powerful optimization algorithm for nonconvex problems and has been used popularly in many areas including nonconvex penalized estimators (Kim *et al.*, 2008) and semi-supervised learning problems (Collobert *et al.*, 2006; Shen *et al.*, 2003). The key idea of the CCCP algorithm is to update the solution by the minimizer of the tight convex upper bound of the objective function at the current solution. To explain more details, let $Q(\boldsymbol{\beta})$ be the objective function to be minimized. Suppose that $Q(\boldsymbol{\beta})$ consist of a sum of convex and concave functions such that $Q(\boldsymbol{\beta}) = Q_{vex}(\boldsymbol{\beta}) + Q_{cav}(\boldsymbol{\beta})$, where Q_{vex} is convex and Q_{cav} is concave. For a given current solution $\tilde{\boldsymbol{\beta}}$, the tight convex upper bound is defined by

$$U(\boldsymbol{\beta}) = Q_{vex}(\boldsymbol{\beta}) + \{\partial Q_{cav}(\tilde{\boldsymbol{\beta}})/\partial\boldsymbol{\beta}\}'\boldsymbol{\beta}.$$

Then we update the current solution by the minimizer of $U(\boldsymbol{\beta})$ and iterate this procedure until the solution converges. Since $U(\boldsymbol{\beta})$ is the convex function, we can easily find the minimizer using various convex optimization algorithms. One important property of the CCCP algorithm is that after each iteration, the objective function always decreases and the solution converges to a local minimum (Yuille and Rangarajan, 2003).

Recall that the nonconvex group penalized estimator is defined as a minimizer of

$$Q_\lambda(\boldsymbol{\beta}) = \frac{1}{2n}\Big\|\mathbf{y} - \sum_{k=1}^{K}\mathbf{X}_k\boldsymbol{\beta}_k\Big\|_2^2 + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\beta}_k\|_2), \tag{10}$$

where $J_\lambda(\cdot)$ is a nonconvex penalty function. Define $\tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta}) = \sum_{k=1}^{K}\{J_{\lambda_k}(||\boldsymbol{\beta}_k||_2) - \lambda\sqrt{p_k}||\boldsymbol{\beta}_k||_2\}$. Then

we rewrite the objective function (10) as

$$Q_\lambda(\boldsymbol{\beta}) = \frac{1}{2n}\Big\|\mathbf{y} - \sum_{k=1}^{K}\mathbf{X}_k\boldsymbol{\beta}_k\Big\|_2^2 + \tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta}) + \lambda\sum_{k=1}^{K}\sqrt{p_k}\|\boldsymbol{\beta}_k\|_2. \tag{11}$$

It can be easily shown that $\tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta})$ is a differentiable concave function with respect to $\boldsymbol{\beta}$. That is, the objective function $Q_\lambda(\boldsymbol{\beta})$ in (11) consists of the sum of concave and convex functions. Thus, we can apply the CCCP algorithm. Let $\partial\tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta})$ be the gradient of $\tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta})$. For a given current solution $\tilde{\boldsymbol{\beta}}$, the tight convex upper bound of $Q_\lambda(\boldsymbol{\beta})$ is defined as

$$U_\lambda(\boldsymbol{\beta}) = \frac{1}{2n}\Big\|\mathbf{y} - \sum_{k=1}^{K}\mathbf{X}_k\boldsymbol{\beta}_k\Big\|_2^2 + \partial\tilde{\mathbf{J}}_\lambda(\tilde{\boldsymbol{\beta}})'\boldsymbol{\beta} + \lambda\sum_{k=1}^{K}\sqrt{p_k}\|\boldsymbol{\beta}_k\|_2.$$

We then update the current solution by the minimizer of $U_\lambda(\boldsymbol{\beta})$, which can be obtained easily by group LASSO algorithm of Foygel and Drton (2010). The proposed algorithm is summarized in Algorithm 2.

Algorithm 2 The proposed optimization algorithm for nonconvex group penalties

Find an initial estimator $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^p$.
Compute the spectral-decomposition of $\mathbf{X}_k'\mathbf{X}_k/n = \mathbf{U}_k'\mathbf{D}_k\mathbf{U}_k$ with $\mathbf{D}_k = \mathrm{diag}\{d_1^k, \ldots, d_{p_k}^k\}$, for all k.
repeat
  Calculate $\partial\tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta})$ at $\tilde{\boldsymbol{\beta}}$, and denote by $\partial\tilde{\mathbf{J}}_\lambda(\tilde{\boldsymbol{\beta}})$.
  repeat
    for $k = 1, \ldots, K$ **do**
      Calculate $\mathbf{v}_k = -\mathbf{U}_k(-\mathbf{X}_k'\mathbf{y}/n + \partial\tilde{\mathbf{J}}_\lambda(\tilde{\boldsymbol{\beta}})_k + 2\mathbf{X}_k'\mathbf{X}_{-k}\tilde{\boldsymbol{\beta}}_{-k}/n)$.
      if $\|\mathbf{v}_k\|_2 \le \lambda\sqrt{p_k}$ **then**
        $\hat{\boldsymbol{\beta}}_k = 0$.
      else
        Find the unique $\delta > 0$ satisfying $f(\delta) = \sum_{j=1}^{p_k}(\mathbf{v}_k)_j^2/(d_j^k\delta + \lambda\sqrt{p_k})^2 = 1$.
        Update $\hat{\boldsymbol{\beta}}_k = \mathbf{U}_k'(\mathbf{D}_k + \delta^{-1}\lambda\sqrt{p_k}I_{p_k})^{-1}\mathbf{v}_k$.
      end if
    end for
  until convergence.
  Update $\tilde{\boldsymbol{\beta}}$ by $\hat{\boldsymbol{\beta}}$.
until convergence.

Furthermore, we check whether the least square estimator of $\boldsymbol{\beta}_k$ is a solution when $\|\tilde{\boldsymbol{\beta}}_k\|_2 > a\lambda\sqrt{p_k}$ in each iteration of the algorithm. As a result, we only apply the group LASSO algorithm to a subset of groups. ($\mathcal{N}$ in Algorithm 3). These modification makes the algorithm slightly faster when the number of nonzero groups with the strong signal are large. The modified proposed algorithm summarized in Algorithm 3.

Algorithm 3 The modified proposed optimization algorithm for nonconvex group penalties

Find an initial estimator $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^p$.

Compute the spectral-decomposition of $\mathbf{X}_k'\mathbf{X}_k/n = \mathbf{U}_k'\mathbf{D}_k\mathbf{U}_k$ with $\mathbf{D}_k = \text{diag}\{d_1^k, \ldots, d_{p_k}^k\}$, for all k.

repeat

Define $\mathcal{A} = \{k; \|\tilde{\boldsymbol{\beta}}_k\|_2 \geq a\lambda\sqrt{p_k}\}$ and $\mathcal{N} = \mathcal{A}^c$.

Calculate $\mathbf{r}_k = \mathbf{y} - \sum_{l \neq k} \mathbf{X}_l\tilde{\boldsymbol{\beta}}_l$ and $\mathbf{s}_k = (\mathbf{X}_k'\mathbf{X}_k)^{-1}\mathbf{X}_k'\mathbf{r}_k$.

for $k \in \mathcal{A}$ **do**

if $\|\mathbf{s}_k\|_2 \geq \lambda\sqrt{p_k}$ **then**

$\hat{\boldsymbol{\beta}}_k = \mathbf{s}_k$.

else

Update the set $\mathcal{N}$ by $\mathcal{N} \cup \{k\}$.

end if

end for

Calculate $\partial\tilde{\mathbf{J}}_\lambda(\boldsymbol{\beta})$ at $\tilde{\boldsymbol{\beta}}$, and denote by $\partial\tilde{\mathbf{J}}_\lambda(\tilde{\boldsymbol{\beta}})$.

repeat

for $k \in \mathcal{N}$ **do**

Calculate $\mathbf{v}_k = -\mathbf{U}_k(-\mathbf{X}_k'\mathbf{y}/n + \partial\tilde{\mathbf{J}}_\lambda(\tilde{\boldsymbol{\beta}})_k + 2\mathbf{X}_k'\mathbf{X}_{-k}\tilde{\boldsymbol{\beta}}_{-k}/n)$.

if $\|\mathbf{v}_k\|_2 \leq \lambda\sqrt{p_k}$ **then**

$\hat{\boldsymbol{\beta}}_k = 0$.

else

Find the unique $\delta > 0$ satisfying $f(\delta) = \sum_{j=1}^{p_k}(\mathbf{v}_k)_j^2/(d_j^k\delta + \lambda\sqrt{p_k})^2 = 1$.

Update $\hat{\boldsymbol{\beta}}_k = \mathbf{U}_k'(\mathbf{D}_k + \delta^{-1}\lambda\sqrt{p_k}I_{p_k})^{-1}\mathbf{v}_k$.

end if

end for

until convergence.

Update $\tilde{\boldsymbol{\beta}}$ by $\hat{\boldsymbol{\beta}}$.

until convergence.

In the following proposition, we state formally a convergence result for the proposed algorithm, which follows directly from Theorem 2 of Yuille and Rangarajan (2003).

Proposition 1 *Let $\boldsymbol{\beta}^{(s)}$ denote the coefficient vector after s iterations for s=0,1,2,.... Then for a fixed $\lambda > 0$, $Q_\lambda\big(\boldsymbol{\beta}^{(s+1)}\big) \leq Q_\lambda\big(\boldsymbol{\beta}^{(s)}\big)$, for all s. Furthermore, every limit point of the sequence $\{\boldsymbol{\beta}^{(s)} : s \geq 0\}$ is a stationary point of $Q_\lambda(\boldsymbol{\beta})$.*

3.3 Regularization parameter selection

Given a current solution $\tilde{\boldsymbol{\beta}}$, the proposed algorithm finds the minimizer of

$$\frac{1}{2n}\Big\|\mathbf{y} - \sum_{k=1}^{K} \mathbf{X}_k\boldsymbol{\beta}_k\Big\|_2^2 + \partial\tilde{\mathbf{J}}_\lambda(\tilde{\boldsymbol{\beta}})'\boldsymbol{\beta} + \lambda\sum_{k=1}^{K}\sqrt{p_k}\|\boldsymbol{\beta}_k\|_2,$$

until $\tilde{\boldsymbol{\beta}}$ converges to $\hat{\boldsymbol{\beta}}$. The nonzero elements of the final solution $\hat{\boldsymbol{\beta}}$ satisfy the Karush-Kuhn-Tucker (KKT) condition

$$-\mathbf{X}_{\mathcal{A}}'(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\hat{\boldsymbol{\beta}}_{\mathcal{A}})/n + \partial\tilde{\mathbf{J}}_\lambda(\hat{\boldsymbol{\beta}}_{\mathcal{A}}) + \lambda\sum_{k\in\mathcal{A}}\mathbf{w}_k'\hat{\boldsymbol{\beta}}_k = \mathbf{0},$$

where $\mathcal{A} = \{k : \|\hat{\boldsymbol{\beta}}_k\|_2 \neq 0\}$, $\mathbf{X}_{\mathcal{A}}$ is the submatrix of $\mathbf{X}$ whose columns are in $\mathcal{A}$, $\hat{\boldsymbol{\beta}}_{\mathcal{A}} = (\hat{\boldsymbol{\beta}}_k, k \in \mathcal{A})$, $\tilde{\mathbf{J}}_\lambda(\hat{\boldsymbol{\beta}}_{\mathcal{A}}) = \sum_{k\in\mathcal{A}}\tilde{J}_{\lambda_k}(\|\hat{\boldsymbol{\beta}}_k\|_2)$ and $\mathbf{w} = (\mathbf{w}_1, \ldots, \mathbf{w}_K)$ with $\mathbf{w}_k = (\sqrt{p_k}/\|\hat{\boldsymbol{\beta}}_k\|_2, \ldots, \sqrt{p_k}/\|\hat{\boldsymbol{\beta}}_k\|_2) \in \mathbb{R}^{p_k}$. Thus, we can write the predicted vector of the response as

$$\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}_{\mathcal{A}}\{\mathbf{X}_{\mathcal{A}}'\mathbf{X}_{\mathcal{A}} + n\mathbf{H}(\hat{\boldsymbol{\beta}}_{\mathcal{A}})\}^{-1}\mathbf{X}_{\mathcal{A}}'\mathbf{y},$$

where $\mathbf{H}(\hat{\boldsymbol{\beta}}_{\mathcal{A}}) = \mathrm{diag}(\partial\tilde{\mathbf{J}}_\lambda(\hat{\boldsymbol{\beta}}_{\mathcal{A}})/\partial\hat{\boldsymbol{\beta}}_{\mathcal{A}} + \lambda\mathbf{w}_{\mathcal{A}})$. Hence, when $p < n$, to select the regularization parameter λ we can use the BIC with the generalized degrees of freedom given as

$$\mathrm{BIC}(\lambda) = \log\big(\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)\|_2^2/n\big) + \mathrm{df}(\lambda)\log(n)/n,$$

where $\hat{\boldsymbol{\beta}}(\lambda)$ is the solution for a given λ and the generalized degrees of freedom $\mathrm{df}(\lambda)$ is defined as (Tibshirani, 1996; Fan and Li, 2001)

$$\mathrm{df}(\lambda) = \mathrm{trace}\Big[\{\mathbf{X}_{\mathcal{A}}'\mathbf{X}_{\mathcal{A}} + n\mathbf{H}(\hat{\boldsymbol{\beta}}_{\mathcal{A}})\}^{-1}\mathbf{X}_{\mathcal{A}}'\mathbf{X}_{\mathcal{A}}\Big].$$

When $p > n$, we may use the cross validation method.

3.4 Extension to generalized linear models

We can extend the algorithm described in Section 3.2 to generalized linear models with grouped covariates. We consider the penalized log-likelihood estimator using a nonconvex group penalty. Suppose that the likelihood belongs to the exponential family, where the generic density form can be written as (McCullagh and Nelder, 1989)

$$f(y|\mathbf{x}, \boldsymbol{\beta}) = c(y)\exp(y\mathbf{x}'\boldsymbol{\beta} - b(\boldsymbol{\beta})),$$

where $b(\cdot)$ and $c(\cdot)$ are known functions. Let $\{(\mathbf{x}_i, y_i) : \mathbf{x}_i \in \mathbb{R}^p, y_i \in \mathbb{R},\ i = 1, \ldots, n\}$ be n pairs of p-dimensional covariates and a response, where covariates are divided into K groups as in the linear model (1). The nonconvex group penalized estimator for the generalized linear model is defined as a minimizer of

$$Q_\lambda(\boldsymbol{\beta}) = L(\boldsymbol{\beta}) + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\beta}_k\|_2). \tag{12}$$

where $L(\boldsymbol{\beta}) = \sum_{i=1}^{n}\{-y_i(\mathbf{x}_i'\boldsymbol{\beta}) + b(\mathbf{x}_i'\boldsymbol{\beta})\}/n$ is the negative log-likelihood function. For example, the negative log-likelihood of the logistic regression is given as

$$L(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}\Big\{ - y_i(\mathbf{x}_i'\boldsymbol{\beta}) + \log(1 + \exp(\mathbf{x}_i'\boldsymbol{\beta}))\Big\}.$$

Note that the negative log-likelihood function is convex but not a quadratic function. Hence, the proposed algorithm can not be directly applied to generalized linear models. However, we can combine the proposed algorithm with the Newton-Raphson algorithm. Suppose the log-likelihood function is smooth and has the second derivative with respect to $\boldsymbol{\beta}$. For a given current solution $\tilde{\boldsymbol{\beta}}$, we can approximate the negative log-likelihood $L(\boldsymbol{\beta})$ as a quadratic function by Taylor expansion around the current solution. That is, $Q_\lambda(\boldsymbol{\beta})$ can be locally approximated by $\tilde{Q}_\lambda(\boldsymbol{\beta})$, where

$$\tilde{Q}_\lambda(\boldsymbol{\beta}) \approx (\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})'\nabla L(\tilde{\boldsymbol{\beta}}) + (\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})'\nabla^2 L(\tilde{\boldsymbol{\beta}})(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})/2 + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\beta}_k\|_2). \tag{13}$$

Here, $\nabla L(\tilde{\boldsymbol{\beta}})$ and $\nabla^2 L(\tilde{\boldsymbol{\beta}})$ are the first and second derivatives of $L(\boldsymbol{\beta})$, respectively. Then, we apply the proposed algorithm to minimize (13). Finally, we iterate these two steps until convergence. In general, this procedure is not guaranteed to converge. However, Lee *et al.* (2014) showed that adding a simple line search guarantees the convergence.

4 Algorithm efficiency

In this section, we investigate the efficiency of the proposed algorithms in terms of accuracy, computing time and sparseness through analysing simulated as well as real data sets. We compare the proposed algorithms with the GCD algorithms for group LASSO, group MCP and group SCAD. All algorithms are implemented by the **R** program and the computing time is measured using SYSTEM.TIME() in the **R** system with Intel Core i7-4790 3.60GHz with 16GB memory. In our experiments, all algorithms stopped when the relative change of the ℓ_2-norm of the coefficients is less than 10^{-6}.

4.1 Linear regression model

First, we consider the linear regression model with group structures of covariates in (1). Simulated data sets are generated as follows. The design matrix consists of 100 groups (block), each with 5 elements. The

coefficients for the first and second groups are equal to all 0.5 and -0.5, respectively; the coefficients in the other 98 groups are all zero. We set $\mathrm{Var}(\epsilon) = 1$ and $n = 100$. We generate 100 simulated data sets, and calculate the averages of the objective function values (Cost) in (5), and the averages of computing times (Time) as well as the averages of the number of selected groups (#Group) and variables (#Variable).

We consider the two cases, the orthogonal case ($\mathbf{X}_k'\mathbf{X}_k/n = \mathbf{I}_{p_k}, \forall k$) and non-orthogonal case. For the orthogonal case, we generate a covariate vector from the multivariate Gaussian distribution with mean 0, variance 1 and identity correlation matrix, and then orthogonalize covariates in each group. For the non-orthogonal case, we generate a covariate vector from the multivariate Gaussian distribution with mean 0, variance 1 and block-diagonal correlation structure that has within-block correlation $\rho = 0,\ 0.5$ and 0.9. Note that the proposed algorithm minimizes (5) while the GCD algorithm minimizes (9). The objective functions (5) and (9) are the same for the orthogonal case, but they differ for the nonorthogonal case. We use $\tilde{\boldsymbol{\beta}} = \mathbf{0}$ for the initial value.

Table 1: Comparison of the proposed and GCD algorithms for the orthogonal case in the linear regression model.

λ	Method	Cost	Time	#Group
	gLASSO_1	1.4680	0.0758	1.28
	gLASSO_2	1.4681	0.0707	1.28
$\lambda = 0.5$	gMCP_1	1.4595	0.1045	1.28
	gMCP_2	1.4595	0.0697	1.28
	gSCAD_1	1.4680	0.1099	1.28
	gSCAD_2	1.4681	0.0685	1.28
	gLASSO_1	0.8948	0.0873	8.55
	gLASSO_2	0.8949	0.0905	8.55
$\lambda = 0.1$	gMCP_1	0.6555	0.1361	5.75
	gMCP_2	0.6518	0.0983	5.45
	gSCAD_1	0.7304	0.1505	6.05
	gSCAD_2	0.7351	0.1005	6.03

Table 1 summarizes the results for the orthogonal cases. In the table, the subscript 1 and 2 in each method represent the proposed algorithm and existing GCD algorithm, respectively. All the results except the computing times are almost equal, which is because the two algorithms minimize the same objective function. The results indicate that the proposed algorithm is not unacceptably slow compared to the GCD algorithm.

Table 2 shows the results for the non-orthogonal cases, in which the proposed algorithms yield smaller values of the objective function (5) than the GCD algorithms. Furthermore, the differences of the values of the objective functions between the proposed and GCD algorithms are getting larger as the within-block correlation ρ becomes larger. In contrast, by comparing computing times in Tables 1 and 2, we can

Table 2: Comparison of the proposed and GCD algorithms in the linear regression model.

		$\rho = 0$			$\rho = 0.5$			$\rho = 0.9$		
λ	Method	Cost	Time	#Group	Cost	Time	#Group	Cost	Time	#Group
0.5	gLASSO_1	1.5578	0.1148	1.52	2.1748	0.1292	2.02	2.2816	0.1282	2.02
	gLASSO_2	1.5665	0.1169	1.71	2.3550	0.1411	2.00	2.8055	0.1430	2.01
	gMCP_1	1.5382	0.1940	1.49	1.9934	0.2168	2.00	2.0586	0.2112	2.00
	gMCP_2	1.5470	0.1242	1.70	2.0530	0.1538	2.00	2.2656	0.1584	2.00
	gSCAD_1	1.5578	0.1726	1.52	2.1747	0.1920	2.02	2.2805	0.1985	2.02
	gSCAD_2	1.5665	0.1193	1.71	2.3100	0.1515	2.00	2.5999	0.1633	2.00
0.1	gLASSO_1	0.9155	0.1771	11.04	0.9678	0.2026	11.14	0.9833	0.2104	14.41
	gLASSO_2	0.9225	0.1576	10.84	1.0403	0.1850	11.08	1.2921	0.1864	11.12
	gMCP_1	0.6811	0.2981	7.31	0.6739	0.3085	9.44	0.6728	0.3275	13.05
	gMCP_2	0.6879	0.1897	6.61	0.7132	0.1913	6.81	0.7928	0.1970	6.82
	gSCAD_1	0.7565	0.2948	7.65	0.7474	0.3115	9.64	0.7461	0.3293	13.10
	gSCAD_2	0.7616	0.1811	7.50	0.7853	0.1840	7.67	0.8622	0.1796	7.56

see that differences of computing times are not sensitive to the within-block correlation.

4.2 Logistic regression model

Second, we consider the logistic regression model with the success probability π_i,

$$\pi_i = P(y_i = 1|\mathbf{x}_i) = \exp(\mathbf{x}_i'\boldsymbol{\beta})/\{1 + \exp(\mathbf{x}_i'\boldsymbol{\beta})\},$$

where the vector of covariates $\mathbf{x}_i$ consists of K groups. We generate 100 samples of size $n = 100$ with the same design matrices and true coefficients as in the linear regression model. For the logistic regression, the approximated quadratic objective function $\tilde{Q}_\lambda(\boldsymbol{\beta})$ at the current solution $\tilde{\boldsymbol{\beta}}$ in (13) can be expressed as

$$\tilde{Q}_\lambda(\boldsymbol{\beta}) \approx \frac{1}{2n}\big(\mathbf{z} - \mathbf{X}\boldsymbol{\beta}\big)'\mathbf{W}(\tilde{\boldsymbol{\beta}})\big(\mathbf{z} - \mathbf{X}\boldsymbol{\beta}\big) + \sum_{k=1}^{K} J_{\lambda_k}(\|\boldsymbol{\beta}_k\|_2),$$

where $\mathbf{z} = (z_1, \ldots, z_n)$ and $\mathbf{W}(\tilde{\boldsymbol{\beta}})$ are the vector of working responses and the diagonal matrix of weights with $z_i = \mathbf{x}_i'\tilde{\boldsymbol{\beta}} + (y_i - \pi_i)/(\pi_i(1 - \pi_i))$ and $w_i = \pi_i(1 - \pi_i)$. Note that both the proposed and GCD algorithms are iteratively applied to the approximated quadratic objective function $\tilde{Q}_\lambda(\boldsymbol{\beta})$. For each quadratic approximation step, all algorithms are required to orthogonalze $\mathbf{X}'\mathbf{W}(\tilde{\boldsymbol{\beta}})\mathbf{X}/n$ which depends on $\tilde{\boldsymbol{\beta}}$, not $\mathbf{X}'\mathbf{X}/n$. Hence, we do not consider the orthogonal cases for the logistic regression, and only consider the nonorthogonal cases with various within-block correlations.

Table 3 shows the results for the logistic regression model. The proposed algorithms provide more accurate solutions than the GCD algorithms in terms of the values of the objective functions. The larger ρ is, the larger difference between the two algorithms is. The simulation results amply illustrate that the proposed algorithm minimizes the objective function (12) well without requiring too much computing time.

Table 3: Comparison of the proposed and GCD algorithms in the logistic regression model.

		$\rho = 0$			$\rho = 0.5$			$\rho = 0.9$		
λ	Method	Cost	Time	#Group	Cost	Time	#Group	Cost	Time	#Group
0.1	gLASSO_1	0.6833	0.3058	0.93	0.6096	0.4819	2.12	0.5606	0.5102	2.26
	gLASSO_2	0.6843	0.2351	1.09	0.6448	0.2789	1.90	0.6652	0.2937	1.98
	gMCP_1	0.6768	0.5175	0.84	0.5747	0.8074	1.99	0.5213	0.8643	2.00
	gMCP_2	0.6805	0.2529	1.06	0.6312	0.3300	1.89	0.6407	0.3470	1.96
	gSCAD_1	0.6822	0.4418	0.91	0.5988	0.8094	2.03	0.5475	0.8495	2.07
	gSCAD_2	0.6839	0.2353	1.09	0.6443	0.2820	1.90	0.6605	0.3054	1.98
0.05	gLASSO_1	0.6162	0.4011	10.73	0.5066	0.4421	7.49	0.4564	0.4619	8.42
	gLASSO_2	0.6206	0.3403	10.64	0.5490	0.3401	7.02	0.6116	0.3380	5.83
	gMCP_1	0.4987	0.7812	7.08	0.3585	0.9274	5.13	0.3221	0.9702	5.98
	gMCP_2	0.5475	0.5630	8.00	0.4360	0.5949	4.33	0.3840	0.6156	3.25
	gSCAD_1	0.5277	0.7846	7.22	0.3763	0.9526	4.90	0.3389	0.9841	5.93
	gSCAD_2	0.5868	0.4376	10.30	0.4915	0.6878	5.88	0.4679	0.7449	4.41

4.3 Real data analysis

The ozone data, which is available from the R library `mlbench`, has been analyzed by Breiman and Friedman (1985), Hastie and Tibshirani (1990), and Lin and Zhang (2006). It consists of the daily measurements of ozone concentration (maximum one hour average) and meteorological quantities, measured in the Los Angeles basin for 366 days of 1976. We exclude a variable with too many missing values and 36 observations including missing values. We consider ozone concentration as the response variable and the other eleven variables as the covariates. The covariates used in our study are:

`DoW`: Day of week.

`Month`: Month.

`DoM`: Day of month.

`vh`: 500 millibar pressure height (m) measured at Vandenberg AFB.

`wind`: Wind speed (mph) at Los Angeles International Airport (LAX).

`hum`: Humidity (%) at LAX.

`temp`: Temperature (degrees F) measured at Sandburg, CA.

`ibh`: Inversion base height (feet) at LAX.

`dpg`: Pressure gradient (mm Hg) from LAX to Daggett, CA.

`ibt`: Inversion base temperature (degrees F) at LAX.

`vis`: Visibility (miles) measured at LAX.

Among these variables, there are three categorical variables and eight continuous variables. Since `Month` and `DoM` have values of 12 and 31 different ordered levels, respectively, we regard these two covariates as being continuous. Although values of `DoW` are ordered, the day of week effect are related to human activity, which are supposed to change discretely. Hence, we keep `DoW` as a categorical variable, which

is expanded to 6 dummy covariates. We are interested in finding the relationship between ozone concentration and meteorological quantities. The Ozone data set is well known to have a nonlinear relation of covariates and response variable. To assess the nonlinear effects, we use the third order polynomial model. For each continuous variable, we consider the third polynomial expansion. Thus, the problem of selecting relevant variables becomes the problem of selecting groups of variables after the expansion. This problem is also considered by Lin and Zhang (2006) and Kim *et al.* (2006)

Table 4: Comparison of the proposed and GCD algorithms based on 100 random sampling of the Ozone data

Method	Cost	Time	#Group
gLASSO_1	7.495	0.067	10.40
gLASSO_2	8.027	0.061	10.97
gMCP_1	7.538	0.089	8.75
gMCP_2	7.983	0.081	8.64
gSCAD_1	7.772	0.085	8.96
gSCAD_2	8.339	0.081	9.52

Table 4 shows the efficiency of the proposed algorithms compared to GCD algorithms. The results are obtained by 100 random sampling of 2/3 observations from the data set. For each data set, the optimal values of regularization parameters are chosen by the BIC with the generalized degrees of freedom in Section 3.3. Similarly to the results of the simulations, the proposed algorithms provide smaller cost values and are not unacceptably slow compared to the GCD algorithms.

5 Concluding remarks

We proposed the optimization algorithm for nonconvex group penalties, which is a hybrid of the CCCP and group LASSO algorithm. The algorithm can be applied to a wide class of nonconvex group penalties regardless of the form of the design matrix.

We only consider the selection of groups, but not variable selection in the selected groups. It would be valuable to apply our algorithm for selecting groups and individual variables, simultaneously. We leave this problem as a future work.

References

Bach, F. R. (2008). Consistency of the group Lasso and multiple kernel learning. *Journal of Machine Learning Research*, **16**, 1369–1384.

Breheny, P. and Huang, J. (2012). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, 1–15.

Breiman, L. and Friedman, J. (1985). Estimating optimal transforms for multiple regres- sion and correlation. *Journal of the American Statistical Association*, **80**, 580–598.

Collobert, R., Sinz, F., Weston, J., and Bottou, L. (2006). Large-Scale Transductive SVMs. *Journal of Machine Learning Research*, **7**, 1687–1712.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, **96**, 1348–1360.

Foygel, R. and Drton, M. (2010). Exact block-wise optimization in group lasso and sparse group lasso for linear regression. *arXiv preprint arXiv:1010.3320*.

Friedman, J. and Hastie, T. and Höfling, H. and Tibshirani, R. (2007). Pathwise coordinate optimization. *Annals of Applied Statistics*, **1**, 302–332.

Friedman, J. and Hastie, T. and Tibshirani, R. (2010). A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736*.

Fu, W. J. (1998). Penalized regressions: the bridge versus the lasso. *Journal of Computational and Graphical Statistics*, **7**, 397–416.

Hastie, T. J. and Tibshirani, R. (1990). *Generalized Additive Models*. Chapman & Hall/CRC.

Huang, J. and Zhang, T. (2010). The benefit of group sparsity. *Annals of Statistics*, **38**, 1978–2004.

Huang, J., Wei, F. and Ma, S. (2012a). Semiparametric regression pursuit. *Statistica Sinica*, **22**, 1403–1426.

Huang, J. and Breheny, P. and Ma, S. (2012b). A selective review of group selection in high dimensional models. *Statistical Science*, **27**, 481–499.

Kim, Y. and Kim, J. and Kim, Y. (2006). Blockwise sparse regression. *Statistica Sinica*, **16**, 375–390.

Kim, Y., Choi, H. and Oh, H. S. (2008). Smoothly clipped absolute deviation on high dimensions. *Journal of the American Statistical Association*, **103**, 1665–1673.

Kim, Y. and Kwon, S. (2012). Global optimality of nonconvex penalized estimators. *Biometrika*, **99**, 315–325.

Lee, S. and Kim, Y. and Kwon, S. (2012). Quadratic approximation for nonconvex penalized estimations with a diverging number of parameters. *Statistics and Probability Letters*, **82**, 1710–1717.

Lee, S. and Kwon, S. and Kim, Y. (2014). Modified local quadratic approximation algorithm for ℓ_1-penalized convex optimization. *Submitted*.

Lin, Y. and Zhang, H. H. (2006). Component selection and smoothing in smoothing spline analysis of variance models. *Annals of Statistics*, **34**, 2272–2297.

McCullagh, P. and Nelder, J. A. (1989). *Generalized linear models*, New York: Chapman & Hall/CRC.

Qin, Z. and Scheinberg, K. and Goldfarb, D. (2010). Efficient block-coordinate descent algorithms for the group lasso. *Preprint*.

Shen, X., Tseng, G. C., Zhang, X., and Wong, W. H. (2003). On ψ-Learning. *Journal of the American Statistical Association*, **98**, 724–734.

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

Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*. **109**, 475–494.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with groped variables. *Journal of the Royal Statistical Society, Series B*, **68**, 49–67.

Yuille, A. L. and Rangarajan, A. (2003). The concave-convex procedure. *Neural Computation*, **15**, 915–936.

Wang, L., Chen, G. and Li, H. (2007). Group SCAD regression analysis for microarray time course gene expression data. *Bioinformatics*, **23**, 1486-1494.

Wei, F. and Huang, J. (2010). Consistent group selection in high-dimensional linear regression. *Bernoulli*, **16**, 1369–1384.

Wei, F. and Zhu, H. (2012). Group coordinate descent algorithms for nonconvex penalized regression. *Computational Statistics and Data Analysis*. **56**, 316–326.

Zhang, C. H. (2010). Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, **38**, 894–942.