

Penalty Decomposition Methods for l_0 -Norm Minimization ^{*}

Zhaosong Lu[†]

Yong Zhang[‡]

September 8, 2010

Abstract

In this paper we consider general l_0 -norm minimization problems, that is, the problems with l_0 -norm appearing in either objective function or constraint. In particular, we first reformulate the l_0 -norm constrained problem as an equivalent rank minimization problem and then apply the penalty decomposition (PD) method proposed in [33] to solve the latter problem. By utilizing the special structures, we then transform all matrix operations of this method to vector operations and obtain a PD method that only involves vector operations. Under some suitable assumptions, we establish that any accumulation point of the sequence generated by the PD method satisfies a first-order optimality condition that is generally stronger than one natural optimality condition. We further extend the PD method to solve the problem with the l_0 -norm appearing in objective function. Finally, we test the performance of our PD methods by applying them to compressed sensing, sparse logistic regression and sparse inverse covariance selection. The computational results demonstrate that our methods generally outperform the existing methods in terms of solution quality and/or speed.

Key words: l_0 -norm minimization, penalty decomposition methods, compressed sensing, sparse logistic regression, sparse inverse covariance selection

1 Introduction

Nowadays, there are numerous applications in which sparse solutions are concerned. For example, in compressed sensing, a large sparse signal is decoded by using a relatively small number of linear measurements, which can be formulated as finding a sparse solution to a system of linear equalities and/or inequalities. The similar ideas have also been widely used in linear regression. Recently, sparse inverse covariance selection becomes an important tool in discovering the conditional independence in graphical model. One popular approach for sparse inverse covariance selection is to find an approximate sparse inverse covariance while maximizing the log-likelihood (see, for example, [12]). Similarly, sparse logistic regression has been proposed as a promising method for feature selection in classification problems in which a sparse solution is sought to minimize the average logistic loss (see, for example, [36]). Mathematically, all these applications can be formulated into the following l_0 -norm minimization problems:

$$\min_x \{f(x) : \|x_J\|_0 \leq r, x \in \mathcal{X}\}, \quad (1)$$

$$\min_x \{f(x) + \nu \|x_J\|_0 : x \in \mathcal{X}\} \quad (2)$$

^{*}This work was supported in part by NSERC Discovery Grant.

[†]Department of Mathematics, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada. (email: zhaosong@sfu.ca).

[‡]Department of Mathematics, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada. (email: yza30@sfu.ca).

for some integer $r \geq 0$ and $\nu \geq 0$ controlling the sparsity of the solution, where \mathcal{X} is a closed convex set in \mathfrak{R}^n , $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$ is a continuously differentiable function, and $\|x_J\|_0$ denotes the cardinality of the subvector formed by the entries of x indexed by J . Given that l_0 -norm $\|\cdot\|_0$ is an integer-valued, discontinuous and nonconvex function, it is generally hard to solve problems (1) and (2). One common approach in literature for these two problems is to replace $\|\cdot\|_0$ by l_1 -norm $\|\cdot\|_1$ and solve the resulting relaxation problems instead (see, for example, [10, 36, 6, 45]). For some applications such as compressed sensing, it has been shown in [4] that under some suitable assumptions this approach is capable of solving (1) and (2). Recently, another relaxation approach has been proposed to solve problems (1) and (2) in which $\|\cdot\|_0$ is replaced by l_p -norm $\|\cdot\|_p$ for some $p \in (0, 1)$ (see, for example, [5, 7, 8]). In general, it is not clear about the quality of the solutions given by these approaches. Indeed, for the example given in the Appendix, the l_p -norm relaxation approaches for $p \in (0, 1]$ fail to recover the sparse solution.

In this paper we propose penalty decomposition (PD) methods for solving problems (1) and (7). In particular, we first reformulate problem (1) as an equivalent rank minimization problem and then apply the PD method proposed in [33] to solve the latter problem. By utilizing the special structures of the problem, we then transform all matrix operations of the PD method to vector operations and obtain a PD method that only involves vector operations. Under some suitable assumptions, we establish that any accumulation point of the sequence generated by the PD method satisfies a first-order optimality condition of problem (1) that is generally stronger than one natural optimality condition. We further extend the PD method to solve problem (2). Finally, we test the performance of our PD methods by applying them to compressed sensing, sparse logistic regression and sparse inverse covariance selection. The computational results demonstrate that our methods generally outperform the existing methods in terms of solution quality and/or speed.

The rest of this paper is organized as follows. In Subsection 1.1, we introduce the notation that is used throughout the paper. In Section 2, we establish some technical results on a class of l_0 -norm minimization problems that are used to develop the PD methods for problems (1) and (2) in Section 3. The convergence of our PD method for problem (1) is also established. In Section 4, we conduct numerical experiments to test the performance of our PD methods for solving compressed sensing, sparse logistic regression, and sparse inverse covariance selection. Finally, we present some concluding remarks in section 5.

1.1 Notation

In this paper, the symbol \mathfrak{R}^n denotes the n -dimensional Euclidean space, and the set of all $m \times n$ matrices with real entries is denoted by $\mathfrak{R}^{m \times n}$. The space of symmetric $n \times n$ matrices will be denoted by \mathcal{S}^n . If $X \in \mathcal{S}^n$ is positive semidefinite, we write $X \succeq 0$. The cone of positive semidefinite (resp., definite) matrices is denoted by \mathcal{S}_+^n (resp., \mathcal{S}_{++}^n). Given matrices X and Y in $\mathfrak{R}^{m \times n}$, the standard inner product is defined by $\langle X, Y \rangle := \text{Tr}(XY^T)$, where $\text{Tr}(\cdot)$ denotes the trace of a matrix. The Frobenius norm of a real matrix X is defined as $\|X\|_F := \sqrt{\text{Tr}(XX^T)}$. The rank of a matrix X is denoted by $\text{rank}(X)$. We denote by I the identity matrix, whose dimension should be clear from the context. We define the operator $\mathcal{D} : \mathfrak{R}^n \rightarrow \mathfrak{R}^{n \times n}$ as follows:

$$\mathcal{D}(x)_{ij} = \begin{cases} x_i & \text{if } i = j; \\ 0 & \text{otherwise} \end{cases} \quad \forall x \in \mathfrak{R}^n, \quad (3)$$

and \mathcal{D}^* denotes its adjoint operator, that is, $\mathcal{D}^*(X) \in \mathfrak{R}^n$ is the vector extracted from the diagonal of X for any $X \in \mathfrak{R}^{n \times n}$. Given an $n \times n$ matrix X , $\mathcal{D}(X)$ denotes a diagonal matrix whose i th diagonal element is X_{ii} for $i = 1, \dots, n$. Given an index set $J \subseteq \{1, \dots, n\}$, $|J|$ denotes the size of J , X_J denotes

the submatrix formed by the columns of X indexed by J . Likewise, x_J denotes the subvector formed by the entries of x indexed by J . For any real vector, $\|\cdot\|_0$ and $\|\cdot\|_2$ denote the cardinality (i.e., the number of nonzero entries) and the Euclidean norm of the vector, respectively. Given a real vector space \mathcal{U} and a closed set $C \subseteq \mathcal{U}$, let $\text{dist}(\cdot, C) : \mathcal{U} \rightarrow \mathfrak{R}_+$ denote the distance function to C measured in terms of $\|\cdot\|$, that is,

$$\text{dist}(u, C) := \inf_{\tilde{u} \in C} \|u - \tilde{u}\| \quad \forall u \in \mathcal{U}.$$

Finally, $\mathcal{N}_C(x)$ and $\mathcal{T}_C(x)$ denote the normal and tangent cones of C at any $x \in C$, respectively.

2 Technical results on special l_0 -norm minimization

In this section we show that a class of l_0 -norm minimization problems have closed form solutions, which will be used to develop penalty decomposition methods for solving problems (1) and (2) in Sections 3.

Proposition 2.1 *Let $\mathcal{X}_i \subseteq \mathfrak{R}$ and $\phi_i : \mathfrak{R} \rightarrow \mathfrak{R}$ for $i = 1, \dots, n$ be given. Suppose that r is a positive integer and $0 \in \mathcal{X}_i$ for all i . Consider the following l_0 -norm minimization problem:*

$$\min \left\{ \phi(x) = \sum_{i=1}^n \phi_i(x_i) : \|x\|_0 \leq r, x \in \mathcal{X}_1 \times \dots \times \mathcal{X}_n \right\}. \quad (4)$$

Let $\tilde{x}_i^* \in \text{Arg min}\{\phi_i(x_i) : x_i \in \mathcal{X}_i\}$ and $I^* \subseteq \{1, \dots, n\}$ be the index set corresponding to r largest values of $\{v_i^*\}_{i=1}^n$, where $v_i^* = \phi_i(0) - \phi_i(\tilde{x}_i^*)$ for $i = 1, \dots, n$. Then, x^* is an optimal solution of problem (4), where x^* is defined as follows:

$$x_i^* = \begin{cases} \tilde{x}_i^* & \text{if } i \in I^*; \\ 0 & \text{otherwise,} \end{cases} \quad i = 1, \dots, n.$$

Proof. By the assumption $0 \in \mathcal{X}_i$ for all i , and the definitions of x^* , \tilde{x}^* and I^* , we clearly see that $x^* \in \mathcal{X}_1 \times \dots \times \mathcal{X}_n$ and $\|x^*\|_0 \leq r$. Hence, x^* is a feasible solution of (4). It remains to show that $\phi(x) \geq \phi(x^*)$ for any feasible point x of (4). Indeed, let x be arbitrarily chosen such that $\|x\|_0 \leq r$ and $x \in \mathcal{X}_1 \times \dots \times \mathcal{X}_n$, and let $J = \{i | x_i \neq 0\}$. Clearly, $|J| \leq r = |I^*|$. Let \bar{I}^* and \bar{J} denote the complement of I^* and J , respectively. It then follows that

$$|\bar{J} \cap I^*| = |I^*| - |I^* \cap J| \geq |J| - |I^* \cap J| = |J \cap \bar{I}^*|.$$

In view of the definitions of x^* , I^* , \bar{I}^* , J and \bar{J} , we further have

$$\begin{aligned} \phi(x) - \phi(x^*) &= \sum_{i \in J \cap I^*} (\phi_i(x_i) - \phi_i(x_i^*)) + \sum_{i \in \bar{J} \cap \bar{I}^*} (\phi_i(x_i) - \phi_i(x_i^*)) \\ &\quad + \sum_{i \in \bar{J} \cap I^*} (\phi_i(x_i) - \phi_i(x_i^*)) + \sum_{i \in J \cap \bar{I}^*} (\phi_i(x_i) - \phi_i(x_i)), \\ &\geq \sum_{i \in \bar{J} \cap I^*} (\phi_i(0) - \phi_i(x_i^*)) + \sum_{i \in J \cap \bar{I}^*} (\phi_i(x_i^*) - \phi_i(0)), \\ &= \sum_{i \in \bar{J} \cap I^*} (\phi_i(0) - \phi_i(x_i^*)) - \sum_{i \in J \cap \bar{I}^*} (\phi_i(0) - \phi_i(x_i^*)) \geq 0, \end{aligned}$$

where the last inequality follows from the definition of I^* and the relation $|\bar{J} \cap I^*| \geq |J \cap \bar{I}^*|$. Thus, we see that $\phi(x) \geq \phi(x^*)$ for any feasible point x of (4), which implies that the conclusion holds. \blacksquare

It is straightforward to show that the following result holds.

Proposition 2.2 Let $\mathcal{X}_i \subseteq \Re$ and $\phi_i : \Re \rightarrow \Re$ for $i = 1, \dots, n$ be given. Suppose that $\nu \geq 0$ and $0 \in \mathcal{X}_i$ for all i . Consider the following l_0 -norm minimization problem:

$$\min \left\{ \nu \|x\|_0 + \sum_{i=1}^n \phi_i(x_i) : x \in \mathcal{X}_1 \times \dots \times \mathcal{X}_n \right\}. \quad (5)$$

Let $\tilde{x}_i^* \in \text{Arg min}\{\phi_i(x_i) : x_i \in \mathcal{X}_i\}$ and $v_i^* = \phi_i(0) - \nu - \phi_i(\tilde{x}_i^*)$ for $i = 1, \dots, n$. Then, x^* is an optimal solution of problem (5), where x^* is defined as follows:

$$x_i^* = \begin{cases} \tilde{x}_i^* & \text{if } v_i^* \geq 0; \\ 0 & \text{otherwise,} \end{cases} \quad i = 1, \dots, n.$$

3 Penalty decomposition methods

In this section we propose penalty decomposition (PD) methods for solving problems (1) and (2). Throughout this section, we make the following assumption for problems (1) and (2).

Assumption 1 Problems (1) and (2) are feasible, and moreover, at least a feasible solution, denoted by x^{feas} , is known.

For convenience of presentation, we first consider the case $J = \{1, \dots, n\}$, and problems (1) and (2) accordingly become

$$\min_x \{f(x) : \|x\|_0 \leq r, x \in \mathcal{X}\}, \quad (6)$$

$$\min_x \{f(x) + \nu \|x\|_0 : x \in \mathcal{X}\}, \quad (7)$$

respectively. The subsequent results developed for problems (6) and (7) can be straightforwardly extended to problems (1) and (2). We will address such an extension at the end of this section.

We next propose PD methods for solving problems (6) and (7). In particular, we first reformulate problem (6) as an equivalent rank minimization problem and then apply the PD method proposed in [33] to solve the latter problem. By utilizing the special structures of the problem, we then transform all matrix operations of the PD method to vector operations and obtain a PD method that only involves vector operations. Finally, we extend the PD method to solve problem (7).

We now define the set \mathcal{X}_M and the function f_M as follows:

$$\mathcal{X}_M = \{\mathcal{D}(x) : x \in \mathcal{X}\}, \quad f_M(X) = f(\mathcal{D}^*(X)) \quad \forall X \in \mathcal{D}^n. \quad (8)$$

It is easy to see that problem (6) is equivalent to

$$\min_X \{f_M(X) : \text{rank}(X) \leq r, X \in \mathcal{X}_M\}, \quad (9)$$

which can be suitably solved by the PD method proposed in [33]. Before reviewing this method, we introduce some notations as follows:

$$\mathcal{Y}_M := \{Y \in \mathcal{S}^n \mid \text{rank}(Y) \leq r\}, \quad (10)$$

$$Q_\varrho(X, Y) := f_M(X) + \frac{\varrho}{2} \|X - Y\|_F^2, \quad (11)$$

$$\tilde{Q}_\varrho(X, U, D) := Q_\varrho(X, UDU^T) \quad \forall X \in \mathcal{D}^n, U \in \Re^{n \times r}, D \in \mathcal{D}^r. \quad (12)$$

We are now ready to present the PD method proposed in Section 4.1 of [33] for solving problem (9) (or, equivalently, (6)).

Penalty decomposition method for (9):

Let $\{\epsilon_k\}$ be a positive decreasing sequence. Let $\varrho_0 > 0$, $\sigma > 1$ be given. Choose an arbitrary $Y_0^0 \in \mathcal{Y}_M$ and a constant $\Upsilon \geq \max\{f(x^{\text{feas}}), \min_{X \in \mathcal{X}_M} Q_{\varrho_0}(X, Y_0^0)\}$. Set $k = 0$.

- 1) Set $l = 0$ and apply the block coordinate descent (BCD) method to find an approximate solution $(X^k, Y^k) \in \mathcal{X}_M \times \mathcal{Y}_M$ for the penalty subproblem

$$\min\{Q_{\varrho_k}(X, Y) : X \in \mathcal{X}_M, Y \in \mathcal{Y}_M\} \quad (13)$$

by performing steps 1a)-1d):

- 1a) Solve $X_{l+1}^k \in \text{Arg} \min_{X \in \mathcal{X}_M} Q_{\varrho_k}(X, Y_l^k)$. $X_{l+1}^k \in \text{Arg} \min_{X \in \mathcal{X}_M} Q_{\varrho_k}(X, Y_l^k)$.

- 1b) Solve $Y_{l+1}^k \in \text{Arg} \min_{Y \in \mathcal{Y}_M} Q_{\varrho_k}(X_{l+1}^k, Y)$.

- 1c) Set $(X^k, Y^k) := (X_{l+1}^k, Y_{l+1}^k)$. If (X^k, Y^k) satisfies

$$\begin{aligned} \text{dist}(-\nabla_X Q_{\varrho_k}(X^k, Y^k), \mathcal{N}_{\mathcal{X}_M}(X^k)) &\leq \epsilon_k, \\ \|\nabla_U \tilde{Q}_{\varrho_k}(X^k, U^k, D^k)\|_F &\leq \epsilon_k, \\ \|\nabla_D \tilde{Q}_{\varrho_k}(X^k, U^k, D^k)\|_F &\leq \epsilon_k \end{aligned} \quad (14)$$

for some $U^k \in \mathbb{R}^{n \times r}$, $D^k \in \mathcal{D}^r$ such that

$$(U^k)^T U^k = I, \quad Y^k = U^k D^k (U^k)^T, \quad (15)$$

then go to step 2).

- 1d) Set $l \leftarrow l + 1$ and go to step 1a).

- 2) Set $\varrho_{k+1} := \sigma \varrho_k$.

- 3) If $\min_{X \in \mathcal{X}_M} Q_{\varrho_{k+1}}(X, Y^k) > \Upsilon$, set $Y_0^{k+1} := \mathcal{D}(x^{\text{feas}})$. Otherwise, set $Y_0^{k+1} := Y^k$.

- 4) Set $k \leftarrow k + 1$ and go to step 1).

end

By letting $\Omega = \mathcal{S}^n$, it follows from Corollary 4.3 of [33] that the approximate solution $(X^k, Y^k) \in \mathcal{X}_M \times \mathcal{Y}_M$ for problem (13) satisfying (14) can be found by the BCD method described in steps 1a)-1d) within a finite number of iterations. In addition, we observe from step 1a) that $X_l^k \in \mathcal{D}^n$. It follows from this relation and Corollary 2.9 of [33] that there exists a diagonal optimal solution for the problem $\min\{Q_{\varrho_k}(X_{l+1}^k, Y) : Y \in \mathcal{Y}_M\}$. Throughout the remainder of this section, we thus assume that Y_l^k defined in step 1b) is a diagonal matrix, namely, $Y_l^k \in \mathcal{D}^n$. It then implies that $(X^k, Y^k) \in \mathcal{D}^n \times \mathcal{D}^n$ for all k .

Using the special structure of problem (9), we next establish a convergence result for the above PD method under a weaker constraint qualification condition than the one stated in Theorem 4.1 of [33].

Theorem 3.1 Assume that $\epsilon_k \rightarrow 0$. Let $\{(X^k, Y^k, U^k, D^k)\}$ be the sequence generated by the above PD method satisfying (14) and (15). Suppose that the level set $\mathcal{X}_\Upsilon := \{X \in \mathcal{X}_M | f_M(X) \leq \Upsilon\}$ is compact. Then, the following statements hold:

- (a) The sequence $\{(X^k, Y^k, U^k, D^k)\}$ is bounded;
- (b) Suppose that a subsequence $\{(X^k, Y^k, U^k, D^k)\}_{k \in K}$ converges to (X^*, Y^*, U^*, D^*) . Then, $X^* = Y^*$ and X^* is a feasible point of problem (9). Moreover, if the following condition

$$\left\{ \begin{array}{l} d_X - d_U D^*(U^*)^T - U^* d_D (U^*)^T - U^* D^* d_U^T : \\ d_X \in \mathcal{T}_{\mathcal{X}_M}(X^*), \\ d_U \in \mathfrak{R}^{n \times r}, d_D \in \mathcal{D}^r \end{array} \right\} \supseteq \mathcal{D}^n \quad (16)$$

holds, then the subsequence $\{Z^k\}_{k \in K}$ is bounded, where $Z^k := \varrho_k(X^k - Y^k)$, and each accumulation point Z^* of $\{Z^k\}_{k \in K}$ together with (X^*, U^*, D^*) satisfies

$$\begin{aligned} -\nabla f_M(X^*) - Z^* &\in \mathcal{N}_{\mathcal{X}_M}(X^*), \\ Z^* U^* D^* &= 0, \\ \tilde{\mathcal{G}}((U^*)^T Z^* U^*) &= 0, \\ X^* - U^* D^* (U^*)^T &= 0. \end{aligned} \quad (17)$$

Proof. The statement (a) and the first part of statement (b) immediately follow from Theorem 4.1 of [33] by letting $\Omega = \mathcal{S}^n$. We next prove the second part of statement (b). In view of (11), (12), (14) and the definition of Z^k , we have

$$\begin{aligned} \text{dist}(-\nabla f_M(X^k) - Z^k, \mathcal{N}_{\mathcal{X}_M}(X^k)) &\leq \epsilon_k, \\ 2\|Z^k U^k D^k\|_F &\leq \epsilon_k, \\ \|\tilde{\mathcal{G}}((U^k)^T Z^k U^k)\|_F &\leq \epsilon_k. \end{aligned} \quad (18)$$

We now claim that the subsequence $\{Z^k\}_{k \in K}$ is bounded. Suppose not, by passing to a subsequence if necessary, we can assume that $\{Z^k\}_{k \in K} \rightarrow \infty$. Let $\bar{Z}^k = Z^k / \|Z^k\|_F$ for all k . Without loss of generality, assume that $\{\bar{Z}^k\}_{k \in K} \rightarrow \bar{Z}$ (otherwise, one can consider its convergent subsequence). Clearly, $\|\bar{Z}\|_F = 1$. Moreover, $\bar{Z} \in \mathcal{D}^n$ due to $(X^k, Y^k) \in \mathcal{D}^n \times \mathcal{D}^n$. Recall that $\{(X^k, U^k, D^k)\}_{k \in K} \rightarrow (X^*, U^*, D^*)$. Dividing both sides of the inequalities in (18) by $\|Z^k\|_F$, taking limits as $k \in K \rightarrow \infty$, and using the semicontinuity of $\mathcal{N}_{\mathcal{X}_M}(\cdot)$ (see Lemma 2.42 of [41]) we obtain that

$$-\bar{Z} \in \mathcal{N}_{\mathcal{X}_M}(X^*), \quad \bar{Z} U^* D^* = 0, \quad \tilde{\mathcal{G}}((U^*)^T \bar{Z} U^*) = 0. \quad (19)$$

By (16) and the fact $\bar{Z} \in \mathcal{D}^n$, there exist $d_X \in \mathcal{T}_{\mathcal{X}_M}(X^*)$, $d_U \in \mathfrak{R}^{n \times r}$, $d_D \in \mathcal{D}^r$ such that

$$-\bar{Z} = d_X - d_U D^*(U^*)^T - U^* d_D (U^*)^T - U^* D^* d_U^T.$$

It then follows from this equality, $\bar{Z} \in \mathcal{D}^n$ and $d_D \in \mathcal{D}^r$ that

$$\begin{aligned} \|\bar{Z}\|_F^2 &= -\langle \bar{Z}, d_X - d_U D^*(U^*)^T - U^* d_D (U^*)^T - U^* D^* d_U^T \rangle, \\ &= 2\langle \bar{Z}, U^* D^* d_U^T \rangle + \langle d_D, \tilde{\mathcal{G}}((U^*)^T \bar{Z} U^*) \rangle - \langle \bar{Z}, d_X \rangle, \end{aligned}$$

which together with (19) and the relation $d_X \in \mathcal{T}_{\mathcal{X}_M}(X^*)$ implies that $\|\bar{Z}\|_F^2 \leq 0$, which contradicts the identity $\|\bar{Z}\|_F = 1$. Thus, $\{Z^k\}_{k \in K}$ is bounded. Now let Z^* be an accumulation point of

$\{Z^k\}_{k \in K}$. By passing to a subsequence if necessary, we can assume that $\{Z^k\}_{k \in K} \rightarrow Z^*$. Recall that $\{(X^k, U^k, D^k)\}_{k \in K} \rightarrow (X^*, U^*, D^*)$. Taking limits on both sides of the inequalities in (18) as $k \in K \rightarrow \infty$, and using the semicontinuity of $\mathcal{N}_{\mathcal{X}_M}(\cdot)$, we immediately see that the first three relations of (17) hold. In addition, the last relation of (17) holds due to the identities $Y^k = U^k D^k (U^k)^T$ and $Y^* = X^*$. \blacksquare

From Theorem 3.1 (b), we see that under condition (16), any accumulation point (X^*, U^*, D^*, Z^*) of $\{(X^k, U^k, D^k, Z^k)\}_{k \in K}$ satisfies (17). Thus, (X^*, U^*, V^*) together with Z^* satisfies the first-order optimality (i.e., KKT) conditions of the following reformulation of (9) (or, equivalently, (6)).

$$\min_{X, U, D} \{f_M(X) : X - UDU^T = 0, X \in \mathcal{X}_M, U \in \mathfrak{R}^{n \times r}, D \in \mathcal{D}^r\}.$$

We observe that almost all operations of the above PD method are matrix operations, which appear to be inefficient compared to vector operations. By utilizing the special structures, we next show that the above PD method can be transformed into the one only involving vector operations. Before proceeding, we define

$$\mathcal{Y} = \{y \in \mathfrak{R}^n : \|y\|_0 \leq r\}, \quad q_\varrho(x, y) = f(x) + \frac{\varrho}{2} \|x - y\|_2^2 \quad \forall x, y \in \mathfrak{R}^n. \quad (20)$$

Letting $y_l^k = \mathcal{D}^*(Y_l^k)$ and using (11) and Corollary 2.9 of [33], we can observe that the solutions Y_{l+1}^k and X_{l+1}^k of the subproblems in steps 1a) and 1b) above are given by $X_{l+1}^k = \mathcal{D}(x_{l+1}^k)$ and $Y_{l+1}^k = \mathcal{D}(y_{l+1}^k)$, respectively, where

$$x_{l+1}^k \in \text{Arg min}_{x \in \mathcal{X}} q_{\varrho_k}(x, y_l^k), \quad y_{l+1}^k \in \text{Arg min}_{y \in \mathcal{Y}} q_{\varrho_k}(x_{l+1}^k, y).$$

In addition, the inner termination conditions (14) can be similarly transformed into the one only involving vector operations. We are now ready to present the resulting PD method for solving problem (6).

Penalty decomposition method for (6):

Let $\{\epsilon_k\}$ be a positive decreasing sequence. Let $\varrho_0 > 0$, $\sigma > 1$ be given. Choose an arbitrary $y_0^0 \in \mathcal{Y}$ and a constant $\Upsilon \geq \max\{f(x^{\text{feas}}), \min_{x \in \mathcal{X}} q_{\varrho_0}(x, y_0^0)\}$. Set $k = 0$.

- 1) Set $l = 0$ and apply the BCD method to find an approximate solution $(x^k, y^k) \in \mathcal{X} \times \mathcal{Y}$ for the penalty subproblem

$$\min\{q_{\varrho_k}(x, y) : x \in \mathcal{X}, y \in \mathcal{Y}\} \quad (21)$$

by performing steps 1a)-1d):

- 1a) Solve $x_{l+1}^k \in \text{Arg min}_{x \in \mathcal{X}} q_{\varrho_k}(x, y_l^k)$.
- 1b) Solve $y_{l+1}^k \in \text{Arg min}_{y \in \mathcal{Y}} q_{\varrho_k}(x_{l+1}^k, y)$.
- 1c) Set $(x^k, y^k) := (x_{l+1}^k, y_{l+1}^k)$. If (x^k, y^k) satisfies

$$\text{dist}(-\nabla_x q_{\varrho_k}(x^k, y^k), \mathcal{N}_{\mathcal{X}}(x^k)) \leq \epsilon_k, \quad (22)$$

then go to step 2).

- 1d) Set $l \leftarrow l + 1$ and go to step 1a).

- 2) Set $\varrho_{k+1} := \sigma \varrho_k$.

- 3) If $\min_{x \in \mathcal{X}} q_{\varrho_{k+1}}(x, y^k) > \Upsilon$, set $y_0^{k+1} := x^{\text{feas}}$. Otherwise, set $y_0^{k+1} := y^k$.
- 4) Set $k \leftarrow k + 1$ and go to step 1).

end

Remark. The condition (22) is mainly used to establish global convergence for the above method. Nevertheless, it may be hard to verify (22) practically unless \mathcal{X} is simple. On the other hand, we observe that the sequence $\{q_{\varrho_k}(x_l^k, y_l^k)\}$ is non-increasing for any fixed k . In practical implementation, it is thus reasonable to terminate the BCD method based on the progress of $\{q_{\varrho_k}(x_l^k, y_l^k)\}$. Another reasonable termination criterion for the BCD method is

$$\max \left\{ \frac{\|x_l^k - x_{l-1}^k\|_\infty}{\max(\|x_l^k\|_\infty, 1)}, \frac{\|y_l^k - y_{l-1}^k\|_\infty}{\max(\|y_l^k\|_\infty, 1)} \right\} \leq \epsilon_I \quad (23)$$

for some $\epsilon_I > 0$. Similarly, we can terminate the outer iterations of the above method once

$$\|x^k - y^k\|_\infty \leq \epsilon_O \quad (24)$$

for some $\epsilon_O > 0$. In addition, given that problem (21) is nonconvex, the BCD method may converge to a stationary point. To enhance the quality of approximate solutions, one may execute the BCD method multiple times starting from a suitable perturbation of the current approximate solution. In detail, at the k th outer iteration, let (x^k, y^k) be a current approximate solution of (21) obtained by the BCD method, and let $r_k = \|y^k\|_0$. Assume that $r_k > 1$. Before starting the $(k + 1)$ th outer iteration, one can apply the BCD method again starting from $y_0^k \in \text{Arg min}\{\|y - y^k\|_2 : \|y\|_0 \leq r_k - 1\}$ and obtain a new approximate solution $(\tilde{x}^k, \tilde{y}^k)$ of (21). If $q_{\varrho_k}(\tilde{x}^k, \tilde{y}^k)$ is ‘‘sufficiently’’ smaller than $q_{\varrho_k}(x^k, y^k)$, one can set $(x^k, y^k) := (\tilde{x}^k, \tilde{y}^k)$ and repeat the above process. Otherwise, one can terminate the k th outer iteration and start the next outer iteration. Finally, it follows from Proposition 2.1 that the subproblem in step 1a) has closed form solution. \blacksquare

Theorem 3.2 *Assume that $\epsilon_k \rightarrow 0$. Let $\{(x^k, y^k)\}$ be the sequence generated by the above PD method and $J_k = \{j_1^k, \dots, j_r^k\}$ a set of r distinct indices such that $(y^k)_j = 0$ for all $j \notin J_k$. Suppose that the level set $\mathcal{X}_\Upsilon := \{x \in \mathcal{X} | f(x) \leq \Upsilon\}$ is compact. Then, the following statements hold:*

- (a) *The sequence $\{(x^k, y^k)\}$ is bounded;*
- (b) *Suppose (x^*, y^*) is an accumulation point of $\{(x^k, y^k)\}$. Then, $x^* = y^*$ and x^* is a feasible point of problem (6). Moreover, there exists a subsequence K such that $\{(x^k, y^k)\}_{k \in K} \rightarrow (x^*, y^*)$ and $J_k = J^*$ for some index set J^* when $k \in K$ is sufficiently large. Furthermore, if the following condition*

$$\{d_x + d_d : d_x \in \mathcal{T}_{\mathcal{X}}(x^*), d_d \in \mathfrak{R}^n, (d_d)_j = 0 \ \forall j \notin J^*\} = \mathfrak{R}^n \quad (25)$$

holds, then $\{z^k := \varrho_k(x^k - y^k)\}_{k \in K}$ is bounded and each accumulation point z^ of $\{z^k\}_{k \in K}$ together with x^* satisfies*

$$-\nabla f(x^*) - z^* \in \mathcal{N}_{\mathcal{X}}(x^*), \quad z_j^* = 0 \ \forall j \in J^*. \quad (26)$$

Proof. Let $X^k = \mathcal{D}(x^k)$, $Y^k = \mathcal{D}(y^k)$, $U^k = I_{J_k}$, $D^k = \mathcal{D}(y_{J_k}^k)$. We now show that $\{(X^k, Y^k, D^k, U^k)\}$ satisfies (14) and (15). Clearly, $U^k \in \mathfrak{R}^{n \times r}$, $D^k \in \mathcal{D}^r$, $(U^k)^T U^k = I$, $Y^k = U^k D^k (U^k)^T$, and hence (15) holds. Further, in view of (8), (11), (12) and (20), we have

$$\nabla_X Q_{\varrho_k}(X^k, Y^k) = \mathcal{D}(\nabla_x q_{\varrho_k}(x^k, y^k)), \quad \mathcal{N}_{\mathcal{X}_M}(X^k) = \{\mathcal{D}(x) : x \in \mathcal{N}_{\mathcal{X}}(x^k)\},$$

which together with (22) implies that the first relation of (14) holds. In addition, by the definition of y^k , we know that

$$y^k \in \text{Arg min}_{y \in \mathcal{Y}} q_{\varrho_k}(x^k, y),$$

which together with the definitions of X^k , Y^k , \mathcal{Y}_M and $Q_{\varrho_k}(\cdot)$ implies that

$$Y^k \in \text{Arg min}_{Y \in \mathcal{Y}_M} Q_{\varrho_k}(X^k, Y).$$

Using this relation and the definitions of \mathcal{Y}_M , D^k , U^k and $\tilde{Q}_{\varrho_k}(\cdot)$, we further obtain that

$$(U^k, D^k) \in \text{Arg min}_{U, D} \tilde{Q}_{\varrho_k}(X^k, U, D),$$

which yields

$$\nabla_U \tilde{Q}_{\varrho_k}(X^k, U^k, D^k) = 0, \quad \nabla_D \tilde{Q}_{\varrho_k}(X^k, U^k, D^k) = 0.$$

Hence, $\{(X^k, Y^k, U^k, D^k)\}$ satisfies (14). It then follows from Theorem 3.1 (a) that $\{(X^k, Y^k, U^k, D^k)\}$ is bounded, which together with the definitions of X^k and Y^k implies that statement (a) holds.

We next show that statement (b) also holds. Since (x^*, y^*) is an accumulation point of $\{(x^k, y^k)\}$, there exists a subsequence $\{(x^k, y^k)\}_{k \in \bar{K}} \rightarrow (x^*, y^*)$. Clearly, $\{(j_1^k, \dots, j_r^k)\}_{k \in \bar{K}}$ is bounded since J_k is an index set for all k . Thus there exists a subsequence $K \subseteq \bar{K}$ such that $\{(j_1^k, \dots, j_r^k)\}_{k \in K} \rightarrow (j_1^*, \dots, j_r^*)$ for some r distinct indices j_1^*, \dots, j_r^* . Since j_1^k, \dots, j_r^k are r distinct integers, one can easily conclude that $(j_1^k, \dots, j_r^k) = (j_1^*, \dots, j_r^*)$ for sufficiently large $k \in K$. Let $J^* = \{j_1^*, \dots, j_r^*\}$. It then follows that $J_k = J^*$ when $k \in K$ is sufficiently large, and moreover, $\{(x^k, y^k)\}_{k \in K} \rightarrow (x^*, y^*)$. Now, let $X^* = \mathcal{D}(x^*)$, $Y^* = \mathcal{D}(y^*)$. Since $\{D^k\}_{k \in K}$ is bounded, by passing to a subsequence if necessary, assume that $\{D^k\}_{k \in K} \rightarrow D^*$. In addition, we know that $U^k = I_{J_k} = I_{J^*}$ for sufficiently large $k \in K$. Let $U^* = I_{J^*}$. One has $\{(X^k, Y^k, U^k, D^k)\}_{k \in K} \rightarrow (X^*, Y^*, U^*, D^*)$. It then follows from Theorem 3.1 (b) that $X^* = Y^*$ and X^* is a feasible point of problem (9), which together with (8) and the definitions of X^* and Y^* implies that $x^* = y^*$ and x^* is a feasible solution of problem (6). Using the relation $U^* = I_{J^*}$, (25) and the definitions of \mathcal{X}_M and X^* , we have

$$\{d_X - U^* d_D (U^*)^T : d_X \in \mathcal{T}_{\mathcal{X}_M}(X^*), d_D \in \mathcal{D}^r\} = \mathcal{D}^n.$$

It then follows that (16) holds. Thus, by Theorem 3.1 (b), we know that $\{Z^k := \varrho_k(X^k - Y^k)\}_{k \in K}$ is bounded and each accumulation point Z^* of $\{Z^k\}_{k \in K}$ together with (X^*, U^*, D^*) satisfies (17). Then, by the definitions of X^k and Y^k , we conclude that $\{z^k\}_{k \in K}$ is bounded. In addition, recall that $\{y^k\}_{k \in K} \rightarrow y^*$, $\{D^k\}_{k \in K} \rightarrow D^*$, $D^k = \mathcal{D}(y_{J_k}^k)$, and $J_k = J^*$ for sufficiently large $k \in K$. In view of these facts and the relation $x^* = y^*$, we have $D^* = \mathcal{D}(x_{J^*}^*)$. Also, we easily observe that $Z^* = \mathcal{D}(z^*)$ and $\mathcal{N}_{\mathcal{X}_M}(X^*) = \{\mathcal{D}(x) : x \in \mathcal{N}_{\mathcal{X}}(x^*)\}$. Using these results, the relation $U^* = I_{J^*}$ and the definitions of $f_M(\cdot)$ and X^* , we can easily conclude from (17) that (26) holds. \blacksquare

Remark. Let x^* and J^* be defined in Theorem 3.2. It is easy to observe that $I^* = \{j : x_j^* \neq 0\} \subseteq J^*$, but they may not be equal each other. In addition, when $I^* \neq J^*$, (26) is generally stronger than the following natural first-order optimality condition (29) for problem (6). Indeed, suppose further that x^* is a local minimum of (6). Then x^* is clearly a local minimum of

$$\min_{x \in \mathcal{X}} \{f(x) : x_j = 0 \ \forall j \notin I^*\}. \quad (27)$$

We now assume that the constraint qualification

$$\{d_x + d_d : d_x \in \mathcal{T}_{\mathcal{X}}(x^*), (d_d)_j = 0 \ \forall j \notin I^*\} = \mathfrak{R}^n \quad (28)$$

holds at x^* for problem (27). It then follows from Theorem 3.38 on page 134 of [41] that there exists $z^* \in \mathfrak{R}^n$ such that

$$-\nabla f(x^*) - z^* \in \mathcal{N}_{\mathcal{X}}(x^*), \quad z_j^* = 0 \quad \forall j \in I^*. \quad (29)$$

On the other hand, we can see that (28) implies (25) holds. It then follows from Theorem 3.2 that the optimality condition (26) holds. Clearly, when $I^* \neq J^*$, (26) is generally stronger than (29). For example, when $r = n$ and $J = \{1, \dots, n\}$, problem (1) reduces to

$$\min_x \{f(x) : x \in \mathcal{X}\}$$

and (26) becomes the standard first-order optimality condition for the above problem. But (29) clearly not when $I^* \neq \{1, \dots, n\}$. \blacksquare

We next extend the PD method proposed above to solve problem (7). Clearly, (7) can be equivalently reformulated as

$$\min_{x,y} \{f(x) + \nu \|y\|_0 : x - y = 0, x \in \mathcal{X}\}. \quad (30)$$

Given a penalty parameter $\varrho > 0$, the associated quadratic penalty function for (30) is defined as

$$p_\varrho(x, y) := f(x) + \nu \|y\|_0 + \frac{\varrho}{2} \|x - y\|_2^2. \quad (31)$$

We are now ready to present the PD method for solving (30) (or, equivalently, (7)) in which each penalty subproblem is approximately solved by a BCD method.

Penalty decomposition method for (7):

Let $\varrho_0 > 0$, $\sigma > 1$ be given. Choose an arbitrary $y_0^0 \in \mathfrak{R}^n$ and a constant Υ such that $\Upsilon \geq \max\{f(x^{\text{feas}}) + \nu \|x^{\text{feas}}\|_0, \min_{x \in \mathcal{X}} p_{\varrho_0}(x, y_0^0)\}$. Set $k = 0$.

- 1) Set $l = 0$ and apply the BCD method to find an approximate solution $(x^k, y^k) \in \mathcal{X} \times \mathfrak{R}^n$ for the penalty subproblem

$$\min\{p_{\varrho_k}(x, y) : x \in \mathcal{X}, y \in \mathfrak{R}^n\} \quad (32)$$

by performing steps 1a)-1c):

- 1a) Solve $x_{l+1}^k \in \text{Arg} \min_{x \in \mathcal{X}} p_{\varrho_k}(x, y_l^k)$.

- 1b) Solve $y_{l+1}^k \in \text{Arg} \min_{y \in \mathfrak{R}^n} p_{\varrho_k}(x_{l+1}^k, y)$.

- 1c) Set $(x^k, y^k) := (x_{l+1}^k, y_{l+1}^k)$.

- 2) Set $\varrho_{k+1} := \sigma \varrho_k$.

- 3) If $\min_{x \in \mathcal{X}} p_{\varrho_{k+1}}(x, y^k) > \Upsilon$, set $y_0^{k+1} := x^{\text{feas}}$. Otherwise, set $y_0^{k+1} := y^k$.

- 4) Set $k \leftarrow k + 1$ and go to step 1).

end

Remark. In view of Proposition 2.2, the BCD subproblem in step 1a) has closed form solution. In addition, the practical termination criteria proposed for the previous PD method can be suitably applied

to this method. Moreover, given that problem (32) is nonconvex, the BCD method may converge to a stationary point. To enhance the quality of approximate solutions, one may apply a similar strategy as mentioned above by executing the BCD method multiple times starting from a suitable perturbation of the current approximate solution. In addition, by a similar argument as in the proof of Theorem 3.2, we can show that every accumulation point of the sequence $\{(x^k, y^k)\}$ is a feasible point of (30). Nevertheless, it is not clear whether a similar convergence result as in Theorem 3.2 (b) can be established due to the discontinuity and nonconvexity of the objective function of (7). ■

Before ending this section we remark that the above PD methods for problems (6) and (7) can be easily extended to solve (1) and (2) simply by replacing the set \mathcal{Y} appearing in these methods by

$$\mathcal{Y} = \{y \in \mathbb{R}^n : \|y_J\|_0 \leq r\}. \quad (33)$$

Moreover, using a similar argument as in the proof Theorem 3.2, we can establish the following convergence result for the resulting PD method when applied to solve problem (1).

Theorem 3.3 *Assume that $\epsilon_k \rightarrow 0$. Let $\{(x^k, y^k)\}$ be the sequence generated by the above PD method with \mathcal{Y} given in (33), and $J_k = \{j_1^k, \dots, j_r^k\}$ a set of r distinct indices in J such that $(y^k)_j = 0$ for all $j \in J \setminus J_k$. Suppose that the level set $\mathcal{X}_\Upsilon := \{x \in \mathcal{X} \mid f(x) \leq \Upsilon\}$ is compact. Then, the following statements hold:*

- (a) *The sequence $\{(x^k, y^k)\}$ is bounded;*
- (b) *Suppose (x^*, y^*) is an accumulation point of $\{(x^k, y^k)\}$. Then, $x^* = y^*$ and x^* is a feasible point of problem (6). Moreover, there exists a subsequence K such that $\{(x^k, y^k)\}_{k \in K} \rightarrow (x^*, y^*)$ and $J_k = J^*$ for some index set $J^* \subseteq J$ when $k \in K$ is sufficiently large. Furthermore, if the following condition*

$$\{d_x + d_d : d_x \in \mathcal{T}_{\mathcal{X}}(x^*), d_d \in \mathbb{R}^n, (d_d)_j = 0 \forall j \in J \setminus J^*\} = \mathbb{R}^n$$

holds, then $\{z^k := \varrho_k(x^k - y^k)\}_{k \in K}$ is bounded and each accumulation point z^ of $\{z^k\}_{k \in K}$ together with x^* satisfies*

$$-\nabla f(x^*) - z^* \in \mathcal{N}_{\mathcal{X}}(x^*), \quad z_j^* = 0 \quad \forall j \in \bar{J} \cup J^*, \quad (34)$$

where \bar{J} is the complement of J in $\{1, \dots, n\}$.

Remark. By a similar argument as in an early remark, we can observe that (34) is generally stronger than the following natural first-order optimality condition for problem (1):

$$-\nabla f(x^*) - z^* \in \mathcal{N}_{\mathcal{X}}(x^*), \quad z_j^* = 0 \quad \forall j \in \bar{J} \cup I^*,$$

where $I^* = \{j \in J : x_j^* \neq 0\}$. ■

4 Numerical results

In this section, we conduct numerical experiments to test the performance of our PD methods proposed in Section 3 by applying them to solve sparse logistic regression, sparse inverse covariance selection, and compressed sensing problems. All computations below are performed on an Intel Xeon E5410 CPU (2.33GHz) and 8GB RAM running Red Hat Enterprise Linux (kernel 2.6.18).

4.1 Sparse logistic regression problem

In this subsection, we apply the PD method studied in Section 3 to solve sparse logistic regression problem, which has numerous applications in machine learning, computer vision, data mining, bioinformatics and neural signal processing (see, for example, [3, 47, 29, 38, 20, 39]).

Given n samples $\{z^1, \dots, z^n\}$ with p features, and n binary outcomes b_1, \dots, b_n , let $a^i = b_i z^i$ for $i = 1, \dots, n$. The *average logistic loss* function is defined as

$$l_{\text{avg}}(v, w) := \sum_{i=1}^n \theta(w^T a^i + v b_i) / n$$

for some model variables $v \in \Re$ and $w \in \Re^p$, where θ is the *logistic loss* function

$$\theta(t) := \log(1 + \exp(-t)).$$

Then the *sparse logistic regression* problem can be formulated as

$$\min_{v, w} \{l_{\text{avg}}(v, w) : \|w\|_0 \leq r\}, \quad (35)$$

where $r \in [1, p]$ is some integer for controlling the sparsity of the solution. Given that problem (35) is typically hard to solve, one common approach in literature is to solve the l_1 -norm regularization of (35) instead, namely,

$$\min_{v, w} l_{\text{avg}}(v, w) + \lambda \|w\|_1, \quad (36)$$

where $\lambda \geq 0$ is a regularization parameter (see, for example, [25, 15, 37, 43, 27]). Our aim below is to apply the PD method studied in Section 3 to solve (35) directly.

Letting $x = (v, w)$, $J = \{2, \dots, p+1\}$ and $f(x) = l_{\text{avg}}(x_1, x_J)$, we can see that problem (35) is in the form of (1). Thus the PD method studied in Section 3 can be suitably applied to solve (35). Moreover, we observe that the main computational part of the PD method when applied to (35) lies in solving the subproblem arising in step 1a), which is in the form of

$$\min_x \{l_{\text{avg}}(x_1, x_J) + \frac{\varrho}{2} \|x - c\|_2^2 : x \in \Re^{p+1}\} \quad (37)$$

for some $\varrho > 0$ and $c \in \Re^{p+1}$. Due to the similarity between (36) and (37), the interior point method (IPM) studied in [25] can be properly modified to solve problem (37) in which Newton's search direction is approximately computed by a preconditioned conjugate gradient method.

We now address the initialization and termination criteria for our PD method when applied to (35). In particular, we randomly generate $z \in \Re^{p+1}$ such that $\|z_J\|_0 \leq r$ and set the initial point $y_0^0 = z$. In addition, we choose the initial penalty parameter ϱ_0 to be 0.1, and set the parameter $\sigma = \sqrt{10}$. We use (23) and (24) as the inner and outer termination criteria for the PD method and set their associated accuracy parameters ϵ_I and ϵ_O to be 10^{-4} .

Next we conduct numerical experiments to test the performance of our PD method for solving the sparse logistic regression problem (35) on some real and random data. We also compare the results of our method with the IPM [25] which solves problem (36). The code of our PD method is written in Matlab while the code of the IPM is written in C that is downloaded from

http://www.stanford.edu/~boyd/l1_Logreg.

In the first experiment, we compare the quality of the solution of our PD method with the IPM on three small- or medium-sized benchmark data sets which are from the UCI machine learning benchmark repository [35] and other sources [21]. The first data set is the colon tumor gene expression data [21] with more features than samples, the second is the ionosphere data [35] with less features than samples, and the third is the Internet advertisements data [35] with roughly same magnitude of features as samples. We discard the samples with missing data, and standardize each data set so that the sample mean is zero and the sample variance is one. For each data set, we first apply the IPM to solve problem (36) with four values of regularization parameter λ , which are $0.5\lambda_{\max}$, $0.1\lambda_{\max}$, $0.05\lambda_{\max}$, and $0.01\lambda_{\max}$, where λ_{\max} is the upper bound on the useful range of λ that is defined in [25]. For each such λ , let w_λ^* be the approximate optimal w obtained by the IPM. Then we apply our PD method to solve problem (35) with $r = \|w_\lambda^*\|_0$ so that the resulting approximate optimal w is at least as sparse as w_λ^* .

In order to compare the quality of the solutions given by both methods, we now introduce a criterion, that is, *error rate*. Given any model variables (v, w) and a sample vector $z \in \mathfrak{R}^p$, the outcome predicted by (v, w) for z is given by

$$\phi(z) = \text{sgn}(w^T z + v),$$

where

$$\text{sgn}(t) = \begin{cases} +1 & \text{if } t > 0, \\ -1 & \text{otherwise.} \end{cases}$$

Recall that z^i and b_i are the given samples and outcomes for $i = 1, \dots, n$. The *error rate* of (v, w) for predicting the outcomes b_1, \dots, b_n is defined as

$$\text{Error} := \left\{ \sum_{i=1}^n \|\phi(z^i) - b_i\|_0 / n \right\} \times 100\%.$$

The computational results are presented in Table 1. In detail, the name and dimensions of each data set are given in the first three columns. The fourth column gives the ratio between the regularization parameter λ and its upper bound λ_{\max} that is mentioned above. The fifth column lists the value of r , that is, the cardinality of w_λ^* which is defined above. In addition, the average logistic loss and error rate for the IPM and PD are reported in columns six to nine. We can observe that our PD method substantially outperforms the IPM as it generally achieves lower average logistic loss and error rate while the sparsity of both solutions is same. As the IPM and our PD are coded in different programming languages, we choose not to compare the speed of these two methods. But we shall mention that the PD method is capable of solving large-scale problems efficiently as shown in the test on some random data sets below.

In this experiment, we test our PD method on the random problems of six different sizes. For each size, we generate 100 instances. In particular, the first two groups of instances have more features than samples, the second two groups of instances have more samples than features, and the last two groups of instances have the same number of features as samples. The samples $\{z^1, \dots, z^n\}$ with p features and their corresponding outcomes b_1, \dots, b_n are generated in the same manner as described in [25]. In detail, for each instance we choose an equal number of positive and negative samples, that is, $m_+ = m_- = m/2$, where m_+ (resp., m_-) is the number of samples with outcome 1 (resp., -1). The features of positive (resp., negative) samples are independent and identically distributed, drawn from a normal distribution $N(\mu, 1)$, where μ is in turn drawn from a uniform distribution on $[0, 1]$ (resp., $[-1, 0]$). For each such instance, we apply the PD method to solve problem (35) with $r = 0.1p, 0.3p, 0.5p, 0.7p$ and $0.9p$, respectively. The average CPU time of the PD method of each group of 100 instances is reported in

Table 1: Computational results on three real data sets

Data	Features	Samples	IPM				PD	
	p	n	λ/λ_{\max}	r	l_{avg}	Error (%)	l_{avg}	Error (%)
Colon	2000	62	0.5	7	0.4398	17.74	0.3588	17.74
			0.1	22	0.1326	1.61	0.0003	0
			0.05	25	0.0664	0	0.0003	0
			0.01	28	0.0134	0	0.0003	0
Ionosphere	34	351	0.5	3	0.4804	17.38	0.3389	12.25
			0.1	11	0.3062	11.40	0.2393	9.69
			0.05	14	0.2505	9.12	0.2055	8.83
			0.01	24	0.1846	6.55	0.1707	6.27
Advertisements	1430	2359	0.5	3	0.2915	12.04	0.2242	6.06
			0.1	36	0.1399	4.11	0.1068	4.24
			0.05	67	0.1042	2.92	0.0613	2.25
			0.01	197	0.0475	1.10	0.0238	0.76

Table 2: Average computational time on six random problems

Size $n \times p$	Time				
	$r = 0.1p$	$r = 0.3p$	$r = 0.5p$	$r = 0.7p$	$r = 0.9p$
100×1000	1.2	0.5	0.2	0.2	0.2
500×5000	25.4	12.1	7.6	5.8	4.4
1000×100	3.8	1.3	1.0	0.7	0.6
5000×500	73.0	25.5	18.8	16.7	13.1
1000×1000	21.3	6.7	4.1	3.2	3.0
5000×5000	403.5	161.4	117.9	88.8	79.7

Table 2. We see that our PD method is capable of solving all the problems in a reasonable amount of time. Moreover, the CPU time of our PD method grows gradually as r decreases.

4.2 Sparse inverse covariance selection problem

In this subsection, we apply the PD method proposed in Section 3 to solve the sparse inverse covariance selection problem, which has numerous real-world applications such as speech recognition and gene network analysis (see, for example, [2, 14]).

Given a sample covariance matrix $\Sigma \in \mathcal{S}_{++}^p$ and a set Ω consisting of pairs of known conditionally independent nodes, the sparse inverse covariance selection problem can be formulated as

$$\begin{aligned}
& \max_{X \succeq 0} \log \det X - \langle \Sigma, X \rangle \\
& \text{s.t.} \quad \sum_{(i,j) \in \bar{\Omega}} \|X_{ij}\|_0 \leq r, \\
& \quad \quad X_{ij} = 0 \quad \forall (i,j) \in \Omega,
\end{aligned} \tag{38}$$

where $\bar{\Omega} = \{(i,j) : (i,j) \notin \Omega, i \neq j\}$, and $r \in [1, |\bar{\Omega}|]$ is some integer for controlling the sparsity of the solution. Given that problem (38) is typically hard to solve, one common approach in literature is to solve the l_1 -norm regularization of (38) instead, namely,

$$\begin{aligned}
& \max_{X \succeq 0} \log \det X - \langle \Sigma, X \rangle - \sum_{(i,j) \in \bar{\Omega}} \rho_{ij} |X_{ij}| \\
& \text{s.t.} \quad X_{ij} = 0 \quad \forall (i,j) \in \Omega,
\end{aligned} \tag{39}$$

where $\{\rho_{ij}\}_{(i,j) \in \bar{\Omega}}$ is a set of regularization parameters (see, for example, [10, 11, 1, 31, 32, 19, 48, 30]). Our aim below is to apply the PD method studied in Section 3 to solve (38) directly.

Letting $\mathcal{X} = \{X \in \mathcal{S}_+^p : X_{ij} = 0, (i, j) \in \Omega\}$ and $J = \bar{\Omega}$, we clearly see that problem (38) is in the form of (1) and thus it can be suitably solved by the PD method studied in Section 3 with

$$\mathcal{Y} = \left\{ Y \in \mathcal{S}^p : \sum_{(i,j) \in \bar{\Omega}} \|Y_{ij}\|_0 \leq r \right\}.$$

Notice that the main computational parts of the PD method when applied to (38) lies in solving the subproblem arising in step 1a), which is in the form of

$$\min_{X \succeq 0} \{-\log \det X + \frac{\varrho}{2} \|X - C\|_F^2 : X_{ij} = 0 \forall (i, j) \in \Omega\} \quad (40)$$

for some $\varrho > 0$ and $C \in \mathcal{S}^p$. Given that problem (40) generally does not have closed-form solution, we now slightly modify the above sets \mathcal{X} and \mathcal{Y} by replacing them by

$$\mathcal{X} = \mathcal{S}_+^p, \quad \mathcal{Y} = \left\{ Y \in \mathcal{S}^p : \sum_{(i,j) \in \bar{\Omega}} \|Y_{ij}\|_0 \leq r, Y_{ij} = 0, (i, j) \in \Omega \right\},$$

respectively, and then apply the PD method presented in Section 3 to solve (38) with such \mathcal{X} and \mathcal{Y} . For the resulting PD method, the subproblem arising in step 1a) is in the form of

$$\min_X \{-\log \det X + \frac{\varrho}{2} \|X - C\|_F^2 : X \succeq 0\} \quad (41)$$

for some $\varrho > 0$ and $C \in \mathcal{S}^p$. It can be easily shown that problem (41) has closed-form solution given by $V\mathcal{D}(x^*)V^T$, where $x_i^* = (\lambda_i + \sqrt{\lambda_i^2 + 4/\varrho})/2$ for all i and $V\mathcal{D}(\lambda)V^T$ is the eigenvalue decomposition of C for some $\lambda \in \mathfrak{R}^p$ (see, for example, Proposition 2.7 of [33]). In addition, it follows from Proposition 2.1 that the subproblem arising in step 1b) has closed-form solution. A similar convergence result as in Theorem 3.2 can also be established for such a PD method.

We now address the initialization and termination criteria for the above PD method. In particular, we choose the initial point $Y_0^0 = (\tilde{\mathcal{D}}(\Sigma))^{-1}$. Moreover, we choose the initial penalty parameter ϱ_0 to be 1, and set the parameter $\sigma = \sqrt{10}$. We use (23) and (24) as the inner and outer termination criteria for the PD method and set their associated accuracy parameters $\epsilon_O = 10^{-4}$ and $\epsilon_I = 10^{-4}, 10^{-3}$ for the random and real data below, respectively.

Next we conduct numerical experiments to test the performance of our PD method for solving sparse inverse covariance selection problem (38) on some random and real data. We also compare the results of our method with the proximal point algorithm (PPA) [48] which solves problem (39). The codes of both methods are written in Matlab. In addition, both methods call the LAPACK routine `dsyevd.f` [26] for computing the full eigenvalue decomposition of a symmetric matrix, which is faster than the Matlab's `eig` routine when p is larger than 500.

In the first experiment, we compare the performance of our PD method with the PPA on a set of instances which are randomly generated in a similar manner as described in [10, 31, 32, 48, 30]. In particular, we first generate a true covariance matrix $\Sigma^t \in \mathcal{S}_{++}^p$ such that its inverse $(\Sigma^t)^{-1}$ is with the density prescribed by δ , and set

$$\Omega = \{(i, j) : (\Sigma^t)^{-1}_{ij} = 0, |i - j| \geq \lfloor p/2 \rfloor\}.$$

We then generate the matrix $B \in \mathcal{S}^p$ by letting

$$B = \Sigma^t + \tau V,$$

where $V \in \mathcal{S}^p$ contains pseudo-random values drawn from a uniform distribution on the interval $[-1, 1]$, and τ is a small positive number. Finally, we obtain the following sample covariance matrix:

$$\Sigma = B - \min\{\lambda_{\min}(B) - \vartheta, 0\}I,$$

where ϑ is a small positive number. Specifically, we choose $\tau = 0.15$, $\vartheta = 1.0e - 4$ and randomly generate the instances with $\delta = 10\%$, 50% and 100% , respectively. It is clear that for $\delta = 100\%$, the set Ω is an empty set. In addition, for all $(i, j) \in \bar{\Omega}$, we set $\rho_{ij} = \rho_{\bar{\Omega}}$ for some $\rho_{\bar{\Omega}} > 0$. Also, we set $\text{Tol} = 10^{-6}$ for the PPA. For each instance, we first apply the PPA to solve problem (39) with four values of regularization parameter $\rho_{\bar{\Omega}}$, which are 0.01, 0.05, 0.1 and 0.5. For each $\rho_{\bar{\Omega}}$, let \tilde{X}^* be the solution obtained by the PPA. Then we apply our PD method to solve problem (38) with $r = \sum_{(i,j) \in \bar{\Omega}} \|\tilde{X}_{ij}^*\|_0$ so that the resulting solution is at least as sparse as \tilde{X}^* .

As mentioned in [49], to evaluate how well the true inverse covariance matrix $(\Sigma^t)^{-1}$ is recovered by a matrix $X \in \mathcal{S}_{++}^p$, we can compute the *normalized entropy loss* which is defined as follows:

$$\text{Loss} := \frac{1}{p}(\langle \Sigma^t, X \rangle - \log \det(\Sigma^t X) - p).$$

The performance of the PPA and our PD method on these instances is presented in Tables 3-5, respectively. In each table, the row size p of Σ is given in column one. The size of Ω is given in column two. The values of $\rho_{\bar{\Omega}}$ and r are given in columns three and four. The log-likelihood defined in terms of the objective value of (38)), the normalized entropy loss and CPU time (in seconds) of the PPA and our PD method are given in the last six columns, respectively. We can observe that the CPU times of both methods are comparable when δ is small, but our PD method is substantially faster than the PPA when δ is large. Moreover, our PD method outperforms the PPA in terms of the quality of solutions as it achieves larger log-likelihood and smaller normalized entropy loss.

Our second experiment is similar to the experiment conducted in [10, 32]. We intend to compare the sparsity recovery ability of our PD method with the PPA. To this aim, we specialize $p = 30$ and $(\Sigma^t)^{-1} \in \mathcal{S}_{++}^p$ to be the matrix with diagonal entries around one and a few randomly chosen, nonzero off-diagonal entries equal to $+1$ or -1 . And the sample covariance matrix Σ is then generated by the aforementioned approach. In addition, we set $\Omega = \{(i, j) : (\Sigma^t)_{ij}^{-1} = 0, |i - j| \geq 15\}$ and $\rho_{ij} = \rho_{\bar{\Omega}}$ for all $(i, j \in \bar{\Omega})$, where $\rho_{\bar{\Omega}}$ is the smallest value such that the total number of nonzero off-diagonal entries of the approximate solution obtained by the PPA when applied to (39) equals $\sum_{(i,j) \in \bar{\Omega}} \|(\Sigma^t)_{ij}^{-1}\|_0$. For model (38), we choose $r = \sum_{(i,j) \in \bar{\Omega}} \|(\Sigma^t)_{ij}^{-1}\|_0$. Also, we set $\text{Tol} = 10^{-6}$ for the PPA. The PPA and PD methods are then applied to solve the models (39) and (38) with the aforementioned ρ_{ij} and r , respectively. In Figure 1, we plot the sparsity patterns of the original inverse covariance matrix $(\Sigma^t)^{-1}$, the noisy inverse sample covariance matrix Σ^{-1} , and the approximate solutions to (39) and (38) obtained by the PPA and PD methods, respectively. We first observe that the sparsity of the solutions of these methods is the same as $(\Sigma^t)^{-1}$. Moreover, the solution of our PD method completely recovers the sparsity patterns of $(\Sigma^t)^{-1}$, but the solution of the PPA misrecovers a few patterns. In addition, we compare the log-likelihood and the normalized entropy loss of these solutions in Table 6. One can clearly see that the solution of our PD method achieves much larger log-likelihood and smaller normalized entropy loss.

Table 3: Computational results for $\delta = 10\%$

Problem		$\rho_{\bar{\Omega}}$	r	PPA			PD		
p	$ \Omega $			Likelihood	Loss	Time	Likelihood	Loss	Time
500	56724	0.01	183876	-950.88	0.0982	31.9	-936.45	0.0694	7.3
		0.05	108418	-978.45	0.1534	34.6	-949.92	0.0963	11.5
		0.1	45018	-999.89	0.1962	40.8	-978.61	0.1537	16.1
		0.5	8602	-1020.56	0.2376	55.6	-1015.98	0.2284	58.7
1000	226702	0.01	745470	-2247.14	0.0883	147.8	-2220.47	0.0616	55.8
		0.05	459524	-2301.07	0.1422	151.9	-2245.66	0.0868	67.2
		0.1	186602	-2344.03	0.1852	155.4	-2301.11	0.1423	94.1
		0.5	42904	-2370.86	0.2120	247.3	-2358.41	0.1996	278.1
1500	509978	0.01	1686128	-3647.71	0.0941	372.2	-3607.23	0.0672	237.7
		0.05	1072854	-3731.47	0.1500	289.0	-3646.32	0.0932	249.2
		0.1	438146	-3799.02	0.1950	303.5	-3731.17	0.1498	311.8
		0.5	93578	-3832.95	0.2176	646.1	-3819.74	0.2088	718.9
2000	905240	0.01	3012206	-5177.80	0.0868	786.5	-5126.09	0.0609	484.4
		0.05	1974238	-5285.89	0.1408	565.2	-5171.87	0.0838	623.0
		0.1	822714	-5375.21	0.1855	667.1	-5282.37	0.1391	827.1
		0.5	188954	-5412.77	0.2043	1266.1	-5397.87	0.1968	1341.3

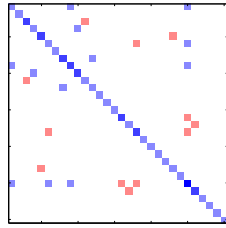
Table 4: Computational results for $\delta = 50\%$

Problem		$\rho_{\bar{\Omega}}$	r	PPA			PD		
p	$ \Omega $			Likelihood	Loss	Time	Likelihood	Loss	Time
500	37738	0.01	202226	-947.33	0.0829	34.0	-937.09	0.0624	11.2
		0.05	119536	-978.51	0.1453	31.4	-953.26	0.0948	12.9
		0.1	50118	-1001.23	0.1907	35.3	-981.95	0.1521	20.1
		0.5	16456	-1022.34	0.2329	58.3	-1005.92	0.2000	32.1
1000	152512	0.01	816070	-2225.74	0.0780	147.3	-2207.32	0.0596	76.4
		0.05	501248	-2288.28	0.1405	138.2	-2227.53	0.0875	91.0
		0.1	203686	-2335.81	0.1881	127.1	-2292.49	0.1447	93.1
		0.5	63646	-2362.87	0.2151	295.7	-2340.06	0.1923	168.0
1500	340656	0.01	1851266	-3649.78	0.0725	366.2	-3623.70	0.0551	217.2
		0.05	1178778	-3742.41	0.1342	267.2	-3660.38	0.0795	283.0
		0.1	475146	-3815.09	0.1827	308.5	-3745.00	0.1359	326.0
		0.5	76206	-3852.35	0.2075	816.7	-3845.43	0.2029	646.4
2000	605990	0.01	3301648	-5149.12	0.0729	823.2	-5116.42	0.0566	557.0
		0.05	2161490	-5272.67	0.1347	539.9	-5160.40	0.0786	595.1
		0.1	893410	-5371.26	0.1840	646.1	-5269.88	0.1333	620.3
		0.5	226194	-5409.58	0.2031	1204.3	-5382.19	0.1895	884.3

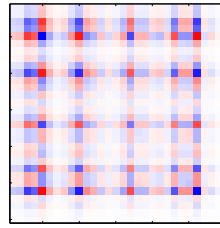
In the third experiment, we aim to compare the performance of our PD method with the PPA on two gene expression data sets that have been widely used in the model selection and classification literature (see, for example, [22, 40, 50, 13, 30]). We first pre-process the data by the same procedure as described in [30] to obtain a sample covariance matrix Σ , and set $\Omega = \emptyset$ and $\rho_{ij} = \rho_{\bar{\Omega}}$ for some $\rho_{\bar{\Omega}} > 0$. Also, we set $\text{Tol} = 10^{-6}$ for the PPA. Then we apply the PPA to solve problem (39) with $\rho_{\bar{\Omega}} = 0.01, 0.05, 0.1$ and 0.5 , respectively. For each $\rho_{\bar{\Omega}}$, we choose r in the same δ manner as above so that the solution given by the PD method when applied to (38) is at least as sparse as the one obtained by the PPA. As the true covariance matrix Σ^t is unknown for these data sets, we now modify the normalized entropy loss defined above by simply replacing Σ^t by Σ . The performance of the PPA and our PD method on these two data sets is presented in Table 7. In detail, the name and dimensions of each data set are given in the first three columns. The values of $\rho_{\bar{\Omega}}$ and r are listed in the fourth and fifth columns.

Table 5: Computational results for $\delta = 100\%$

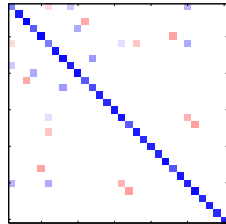
Problem		$\rho_{\bar{\Omega}}$	r	PPA			PD		
p	$ \Omega $			Likelihood	Loss	Time	Likelihood	Loss	Time
500	0	0.01	238232	-930.00	0.0445	35.9	-914.62	0.0138	1.7
		0.05	140056	-973.57	0.1317	30.8	-940.49	0.0655	4.2
		0.1	57064	-1000.78	0.1861	36.2	-978.64	0.1418	7.4
		0.5	20968	-1022.41	0.2294	49.1	-1002.64	0.1898	12.6
1000	0	0.01	963400	-2188.06	0.0406	154.3	-2155.70	0.0083	8.0
		0.05	590780	-2276.69	0.1292	140.1	-2210.76	0.0633	21.0
		0.1	231424	-2335.09	0.1876	119.1	-2285.20	0.1378	28.4
		0.5	65682	-2363.94	0.2165	240.7	-2340.02	0.1926	49.5
1500	0	0.01	2181060	-3585.21	0.0365	369.3	-3538.11	0.0051	20.7
		0.05	1385872	-3716.91	0.1243	252.9	-3615.58	0.0568	55.3
		0.1	551150	-3806.07	0.1837	289.1	-3723.29	0.1286	67.8
		0.5	144160	-3840.95	0.2070	670.1	-3811.37	0.1873	110.7
2000	0	0.01	3892952	-5075.44	0.0341	748.9	-5014.78	0.0037	41.8
		0.05	2543142	-5248.42	0.1206	515.2	-5112.48	0.0526	102.1
		0.1	1027584	-5367.86	0.1803	609.7	-5249.32	0.1210	142.9
		0.5	196448	-5410.37	0.2015	1399.0	-5390.21	0.1914	231.3



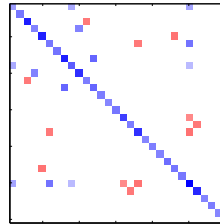
(a) Original inverse $(\Sigma^t)^{-1}$



(b) Noisy inverse Σ^{-1}



(c) Approximate solution of (39)



(d) Approximate solution of (38)

Figure 1: Sparsity recovery.

The log-likelihood, the normalized entropy loss and CPU time (in seconds) of the PPA and our PD method are given in the last six columns, respectively. We can observe that our PD method is generally faster than the PPA. Moreover, our PD method outperforms the PPA in terms of log-likelihood and normalized entropy loss.

Table 6: Numerical results for sparsity recovery

	nnz	Likelihood	Loss
PPA	24	-35.45	0.178
PD	24	-29.56	0.008

Table 7: Computational results on two real data sets

Data	Genes p	Samples n	$\rho_{\bar{\Omega}}$	r	PPA			PD		
					Likelihood	Loss	Time	Likelihood	Loss	Time
Lymph	587	148	0.01	144881	790.12	23.23	97.7	1034.99	22.95	44.2
			0.05	68061	174.86	24.35	80.9	724.16	23.27	38.2
			0.1	39091	-47.03	24.74	63.3	395.92	23.85	30.7
			0.5	5027	-561.37	25.52	27.8	-237.81	24.88	47.6
Leukemia	1255	72	0.01	250471	3229.75	0.678	681.6	4306.82	0.627	208.6
			0.05	170399	1308.38	0.769	483.5	3003.61	0.689	233.3
			0.1	108435	505.02	0.810	487.2	2541.73	0.710	251.7
			0.5	39169	-931.59	0.873	343.4	841.71	0.785	328.9

4.3 Compressed sensing

In this subsection, we apply the PD method proposed in Section 3 to solve the compressed sensing problem, which has important applications in signal processing (see, for example, [9, 44, 28, 42, 6, 34, 46]). It can be formulated as

$$\begin{aligned} \min_{x \in \mathfrak{R}^p} \quad & \frac{1}{2} \|Ax - b\|_2^2 \\ \text{s.t.} \quad & \|x\|_0 \leq r, \end{aligned} \quad (42)$$

where $A \in \mathfrak{R}^{n \times p}$ is a data matrix, $b \in \mathfrak{R}^n$ is an observation vector, and $r \in [1, p]$ is some integer for controlling the sparsity of the solution. Given that problem (42) is typically hard to solve, one popular approach in literature is to solve the l_1 -norm regularization of (42), namely,

$$\min_{x \in \mathfrak{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1, \quad (43)$$

where $\lambda \geq 0$ is a regularization parameter (see, for example, [18, 23, 24]). Our aim below is to apply the PD method studied in Section 3 to solve (42) directly.

Clearly, problem (42) is in the form of (6) and thus the PD method studied in Section 3 can be suitably applied to solve (42). Moreover, the main computational parts of the PD method when applied to (42) lies in solving the subproblem arising in step 1a), which is an unconstrained quadratic programming problem that can be solved by the conjugate gradient method. We now address the initialization and termination criteria for the PD method. In particular, we randomly generate the initial point $y_0^0 \in \mathfrak{R}^p$ such that $\|y_0^0\|_0 \leq r$. In addition, we choose the initial penalty parameter ρ_0 to be 1, and set the parameter $\sigma = \sqrt{10}$. We use (23) and (24) as the inner and outer termination criteria for the PD method, and set their associated accuracy parameters $\epsilon_O = 10^{-4}$ and $\epsilon_I = 10^{-2}, 10^{-3}$ for the random and real data below, respectively. Next we conduct numerical experiments to test the performance of our PD method for solving problem (42) on some random and real data sets. We also compare the results of our method with the gradient projection method (GPSR) proposed in [18] which solves problem (43). The codes of both methods are written in Matlab.

In the first experiment, we consider a typical compressed sensing scenario (same as the one in [24, 18]), where the aim is to reconstruct a length- p sparse signal (in the canonical basis) from n observations with $n < p$. In particular, the $n \times p$ data matrix A is generated by first filling it with

independent samples of a standard Gaussian distribution and then orthonormalizing its rows. In our test, we choose $p = 4096$, $n = 1024$, and generate the original signal $x \in \mathbb{R}^p$ containing 160 randomly placed ± 1 spikes. In addition, the observation $b \in \mathbb{R}^n$ is generated according to

$$b = Ax + \xi,$$

where ξ is a white Gaussian noise of variance 10^{-4} . For model (43), we consider two values for λ : one is $0.1\|A^T b\|_\infty$ as suggested in [18] and another one is the smallest number such that the cardinality of the approximate solution obtained by the GPSR when applied to (43) is 160 (that is, the cardinality of the original signal x). For model (42), we choose $r = 160$. In addition, we set $StopCriterion = 0$, $ToleranceA = 10^{-8}$ and $Debias = 1$ for the GPSR as mentioned in [18]. The GPSR and PD methods are then applied to solve the models (43) and (42) with the aforementioned λ and r , respectively. As mentioned in [18], to evaluate how well the original signal x is recovered by an estimate \hat{x} , we can compute the mean squared error (MSE) according to the formula $MSE = \|\hat{x} - x\|_2^2/p$. The original signal and the estimates obtained by the GPSR and our PD method are shown in Figure 2. In detail, the top graph is the original signal. The middle two graphs are the estimates obtained from the GPSR for the above two values of λ . The bottom graph is the estimate obtained from our PD method. We observe that the first estimate given by the GPSR has small MSE, but it is quite noisy as it has 848 nonzeros which is much larger than the cardinality (160) of the original signal. In addition, the second estimate obtained by the GPSR has exactly the same number of nonzeros as the original signal, yet its MSE is fairly large. Compared to the GPSR, the estimate given by our PD method is not noisy and also has the smallest MSE.

In the second experiment, we compare the performance of our PD method with the GPSR on some random problems. In particular, we first randomly generate a data matrix $A \in \mathbb{R}^{n \times p}$ and an observation vector $b \in \mathbb{R}^n$ according to a standard Gaussian distribution and then orthonormalize the rows of A . We set $StopCriterion = 0$, $ToleranceA = 10^{-8}$ and $Debias = 1$ for the GPSR. Then we apply the GPSR to problem (43) with a set of p distinct λ 's so that the cardinality of the resulting approximate solutions gradually increases from 1 to p . Accordingly, we apply our PD method to problem (42) with $r = 1, \dots, p$. It shall be mentioned that the warm start strategy is applied for both methods. Indeed, the approximate solution of problem (42) or (43) with the preceding r or λ is used as the initial point for the PD or GPSR when applied to the corresponding problem with the current r or λ . In our test, we randomly generate two groups of 100 instances with $(n, p) = (256, 1024)$ and $(512, 512)$, respectively. The average computational results of both methods on each group of instances are reported in Figures 3 and 4. In each figure, we plot the average residual $\|Ax - b\|_2$ against the cardinality in the left graph and the average accumulated CPU time (in seconds) against the cardinality in the right graph. Clearly, we see that the residual curve of our PD method is below that of the GPSR, which implies that given the same sparsity, the solution of the PD method always outperforms the solution of the GPSR in terms of the residual. In addition, we can observe that the CPU times of both methods are comparable for $(n, p) = (512, 512)$, but our PD method is substantially faster than the GPSR for $(n, p) = (256, 1024)$.

In the last experiment, we consider three benchmark image deconvolution problems, all based on the well-known Cameraman image whose size is 256×256 . These problems have been widely studied in literature (see, for example, [16, 17, 18]). For each problem, we first blur the original image by one of three blur kernels described in the second column of Table 8 and then add to the blurred image a white Gaussian noise whose variance is listed in the third column of Table 8. The sample data matrix A of size $256^2 \times 256^2$ in an operator form and the observation vector b are similarly constructed as detailed in [16, 17, 18]. We next apply the GPSR and our PD method to models (43) and (42) to recover the original image, respectively. We hand-tune the parameters and choose $\lambda = 0.35$ for the

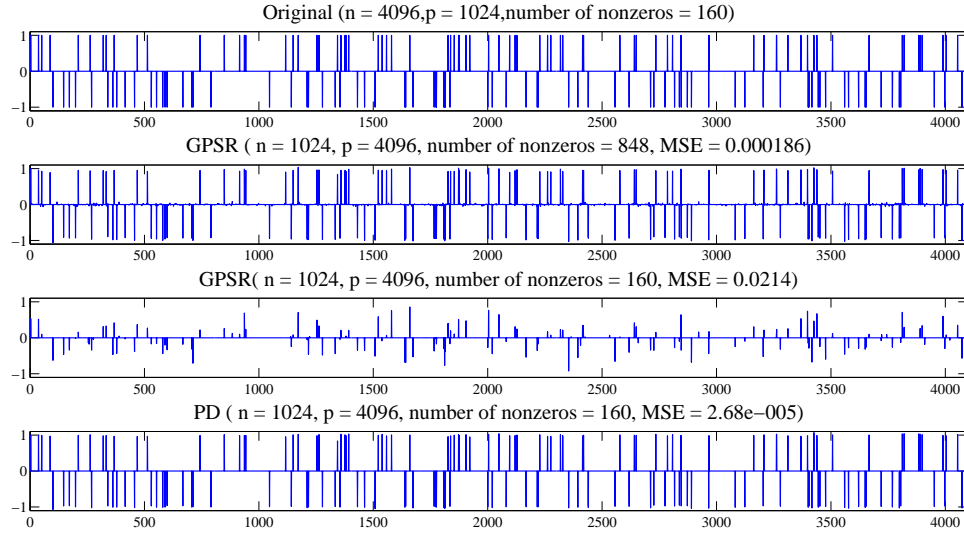
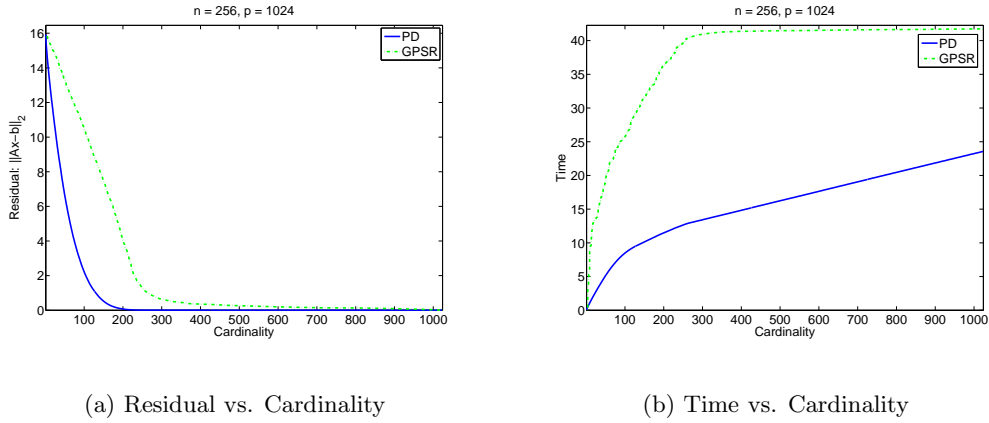


Figure 2: Signal reconstruction.

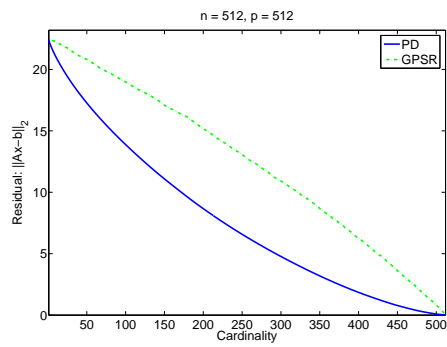


(a) Residual vs. Cardinality

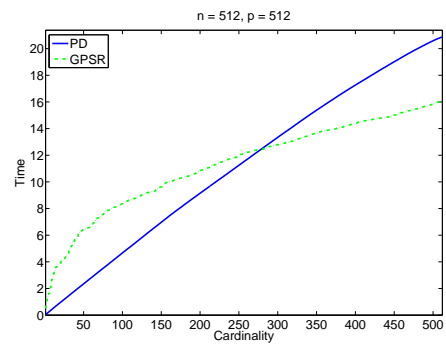
(b) Time vs. Cardinality

Figure 3: Trade-off curves.

GPSR and $r = 8000$ for our PD method. In addition, we set $StopCriterion = 1$, $ToleranceA = 10^{-3}$ and $Debias = 1$ for the GPSR. In Figure 5, we only display the images recovered by both methods for the third image deconvolution problem detailed in Table 8 as the results for the other two test problems are similar. In detail, the top left image is the original image and the top right is the blurred image. The bottom two images are the *deblurred* images by the GPSR and our PD method, respectively. We see that the original image has also been well recovered by the PD method compared to the one by the GPSR.



(a) Residual vs. Cardinality



(b) Time vs. Cardinality

Figure 4: Trade-off curves.

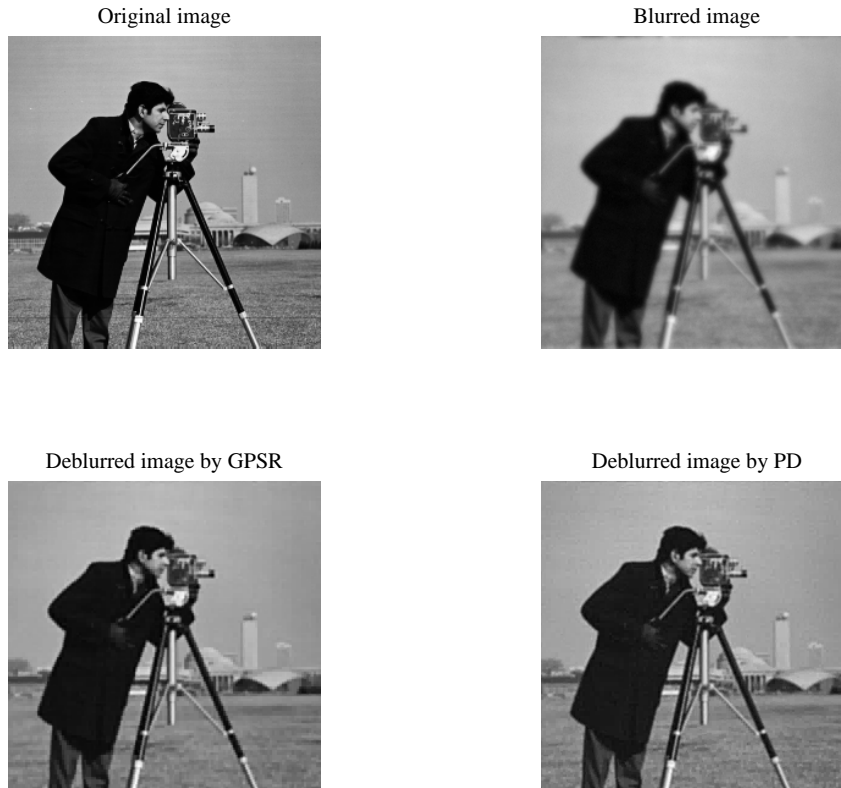


Figure 5: Image deconvolution.

Table 8: Image deconvolution problems

Problem	blur kernel	Variance
1	9×9 uniform	0.56^2
2	$h_{ij} = 1/(i^2 + j^2)$	2
3	$h_{ij} = 1/(i^2 + j^2)$	8

5 Concluding remarks

In this paper we proposed penalty decomposition methods for general l_0 -norm minimization problems in which each subproblem is solved by a block coordinate descend method. Under some suitable assumptions, we establish that any accumulation point of the sequence generated by the PD method when applied to the l_0 -norm constrained minimization problem satisfies a first-order optimality condition, which is generally stronger than one natural optimality condition. The computational results on compressed sensing, sparse logistic regression and sparse inverse covariance selection problems demonstrate that our methods generally outperform the existing methods in terms of solution quality and/or speed.

We remark that the methods proposed in this paper can be straightforwardly extended to solve more general l_0 -norm minimization problems:

$$\begin{array}{ll}
\min_x & f(x) \\
\text{s.t.} & g_i(x) \leq 0, \quad i = 1, \dots, p, \\
& h_i(x) = 0, \quad i = 1, \dots, q, \\
& \|x_J\|_0 \leq r, \quad x \in \mathcal{X},
\end{array}
\qquad
\begin{array}{ll}
\min_x & f(x) + \nu \|x_J\|_0 \\
\text{s.t.} & g_i(x) \leq 0, \quad i = 1, \dots, p, \\
& h_i(x) = 0, \quad i = 1, \dots, q, \\
& x \in \mathcal{X}
\end{array}$$

for some $r, \nu \geq 0$, where $J \subseteq \{1, \dots, n\}$ is an index set, \mathcal{X} is a closed convex set, $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$, $g_i : \mathfrak{R}^n \rightarrow \mathfrak{R}$, $i = 1, \dots, p$, and $h_i : \mathfrak{R}^n \rightarrow \mathfrak{R}$, $i = 1, \dots, q$, are continuously differentiable functions. In addition, we can also develop augmented Lagrangian decomposition methods for solving these problems simply by replacing the quadratic penalty functions in the PD methods by augmented Lagrangian functions.

Appendix

In this appendix we provide an example to demonstrate that the l_p -norm relaxation approaches for $p \in (0, 1]$ may fail to recover the sparse solution.

Let $p \in (0, 1]$ be arbitrarily chosen. Given any $b^1, b^2 \in \mathfrak{R}^n$, let $b = b^1 + b^2$, $\alpha = \|(b^1; b^2)\|_p$ and $A = [b^1, b^2, \alpha I_n, \alpha I_n]$, where I_n denotes the $n \times n$ identity matrix and $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}$ for all $x \in \mathfrak{R}^n$. Consider the linear system $Ax = b$. It is easy to observe that this system has the sparse solution $x^s = (1, 1, 0, \dots, 0)^T$. However, x^s cannot be recovered by solving the l_p -norm regularization problem:

$$f^* = \min_x \left\{ f(x) := \frac{1}{2} \|Ax - b\|_2^2 + \nu \|x\|_p \right\}$$

for any $\nu > 0$. Indeed, let $\bar{x} = (0, 0, b^1/\alpha, b^2/\alpha)^T$. Then, we have $f(x^s) = 2^{1/p}\nu$ and $f(\bar{x}) = \nu$, which implies that $f(x^s) > f(\bar{x}) \geq f^*$. Thus, x^s cannot be an optimal solution of the above problem for any $\nu > 0$. Moreover, the relative error between $f(x^s)$ and f^* is fairly large since

$$(f(x^s) - f^*)/f^* \geq (f(x^s) - f(\bar{x}))/f(\bar{x}) = 2^{1/p} - 1 \geq 1.$$

Therefore, the true sparse solution x^s may not even be a ‘‘good’’ approximate solution to the l_p -norm regularization problem.

References

- [1] O. Banerjee, L. E. Ghaoui, and A. D’Aspremont. Model selection through sparse maximum likelihood estimation. *J. Mach. Learn. Res.*, 9:485-516, 2008.
- [2] J. A. Bilmes. Factored sparse inverse covariance matrices. *International Conference on Acoustics, Speech and Signal processing*, Washington, D.C., 1009-1012, 2000.
- [3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.
- [4] E. J. Candés, J. Romberg and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE T. Inform. Theory*, 52:489-509, 2006.
- [5] R. Chartrand. Exact reconstruction of sparse signals via nonconvex minimization. *IEEE Signal Proc. Let.*, 14:707-710, 2007.
- [6] S. Chen, D. Donoho and M. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20:33-61, 1998.
- [7] X. Chen, F. Xu and Y. Ye. Lower bound theory of nonzero entries in solutions of l_2 - l_p Minimization. Technical report, 2009.
- [8] X. Chen and W. Zhou. Convergence of reweighted l_1 minimization algorithms and unique solution of truncated l_p minimization. Technical report, 2010.
- [9] J. Claerbout and F. Muir. Robust modelling of erratic data. *Geophysics*, 38:826-844, 1973.
- [10] A. D’Aspremont, O. Banerjee and L. E. Ghaoui. First-order methods for sparse covariance selection. *SIAM J. Matrix Anal. A.*, 30(1):56-66, 2008.
- [11] J. Dahl, L. Vandenberghe and V. Roychowdhury. Covariance selection for nonchordal graphs via chordal embedding. *Optim. Method. Softw.*, 23(4):501-520, 2008.
- [12] A. Dempster. Covariance selection. *Biometrics*, 28:157-175, 1978.
- [13] A. Dobra. Dependency networks for genome-wide data. Technical Report No. 547, Department of Statistics, University of Washington, 2009.
- [14] A. Dobra, C. Hans, B. Jones, J. R. Nevins, G. Yao and M. West. Sparse graphical models for exploring gene expression data. *J. Multivariate Anal.*, 90:196-212, 2004.
- [15] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani. Least angle regression. *Ann. Stat.*, 32(2):407-499, 2004.
- [16] M. Figueiredo and R. Nowak. An EM algorithm for wavelet-based image restoration. *IEEE T. Image Process.*, 12:906-916, 2003.
- [17] M. Figueiredo and R. Nowak. A bound optimization approach to wavelet-based image deconvolution. *IEEE Image Proc.*, 2005.
- [18] M. A. T. Figueiredo, R. D. Nowak and S. J. Wright. Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. *IEEE J. Sel. Top. Signa.: Special Issue on Convex Optimization Methods for Signal Processing*, 1(4):586-598, 2007.

- [19] J. Friedman, T. Hastie and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostat.*, 9(3):432-441, 2008.
- [20] A. D. Gerson, L. C. Parra and P. Sajda. Cortical origins of response time variability during rapid discrimination of visual objects. *Neuroimage*, 28(2):342-353, 2005.
- [21] G. Golub and C. Van Loan. Matrix Computations, volume 13 of Studies in Applied Mathematics. John Hopkins University Press, third edition, 1996.
- [22] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: class discovery and class prediction by expression monitoring. *Science*, 286:531-537, 1999.
- [23] E. T. Hale, W. Yin and Y. Zhang. A fixed-point continuation method for l_1 -regularized minimization with applications to compressed sensing. Technical report, 2007.
- [24] S. J. Kim, K. Koh, M. Lustig, S. Boyd and D. Gorinevsky. An interior-point method for large-scale l_1 -regularized least squares. *IEEE J. Sel. Top. Signa.*, 1(4):606-617, December 2007.
- [25] K. Koh, S. J. Kim and S. Boyd. An interior-point method for large-scale l_1 -regularized logistic regression. *J. Mach. Learn. Res.*, 8:1519-1555, 2007.
- [26] Linear Algebra PACKage. Available at <http://www.netlib.org/lapack/index.html>.
- [27] S. Lee, H. Lee, P. Abbeel and A. Ng. Efficient l_1 -regularized logistic regression. In *21th National Conference on Artificial Intelligence (AAAI)*, 2006.
- [28] S. Levy and P. Fullagar. Reconstruction of a sparse spike train from a portion of its spectrum and application to high-resolution deconvolution. *Geophysics*, 46:1235-1243, 1981.
- [29] J. G. Liao and K. V. Chin. Logistic regression for disease classification using microarray data: model selection in a large p and small n case. *Bioinformatics*, 23(15):1945-1951, 2007.
- [30] L. Li and K. C. Toh. An inexact interior point method for l_1 -regularized sparse covariance selection. Technical report, National University of Singapore, 2010.
- [31] Z. Lu. Smooth optimization approach for sparse covariance selection. *SIAM J. Optimiz.*, 19(4):1807-1827, 2009.
- [32] Z. Lu. Adaptive first-order methods for general sparse inverse covariance selection. *SIAM J. Matrix Anal. A.*, 31(4):2000-2016, 2010.
- [33] Z. Lu and Y. Zhang. Penalty decomposition methods for rank minimization. Technical report, Department of Mathematics, Simon Fraser University, Canada, 2010.
- [34] A. Miller. Subset selection in regression. Chapman and Hall, London, 2002.
- [35] D. Newman, S. Hettich, C. Blake and C. Merz. UCI repository of machine learning databases, 1998. Available from www.ics.uci.edu/~mllearn/MLRepository.html.
- [36] A. Y. Ng. Feature selection, l_1 vs. l_2 regularization, and rotational invariance. In *Proceedings of the Twenty-First International Conference on Machine learning (ICML)*, 72-85, 2004.

- [37] M. Y. Park and T. Hastie. Regularization path algorithms for detecting gene interactions, 2006b. Technical report.
- [38] L. C. Parra, C. D. Spence, A. D. Gerson and P. Sajda. Recipes for the linear analysis of EEG. *Neuroimage*, 28(2):326-341, 2005.
- [39] M. G. Philiastides and P. Sajda. Temporal characterization of the neural correlates of perceptual decision making in the human brain. *Cereb. Cortex*, 16(4):509-518, 2006.
- [40] J. Pittman, E. Huang, H. Dressman, C. F. Horng, S. H. Cheng, M. H. Tsou, C. M. Chen, A. Bild, E. S. Iversen, A. T. Huang, J. R. Nevins and M. West. Integrated modeling of clinical and gene expression information for personalized prediction of disease outcomes. *P. Natl. Acad. Sci. USA*, 101(22):8431-8436, 2004.
- [41] A. Ruszczyński. *Nonlinear Optimization*. Princeton University Press, 2006.
- [42] F. Santosa and W. Symes. Linear inversion of band-limited reflection histograms. *SIAM J. Sci. Stat. Comp.*, 7:1307-1330, 1986.
- [43] J. Shi, W. Yin, S. Osher and P. Sajda. A fast hybrid algorithm for large-scale l_1 -regularized logistic regression. *J. Mach. Learn. Res.*, 11:713-741, 2010.
- [44] H. Taylor, S. Bank and J. McCoy. Deconvolution with the l_1 -norm. *Geophysics*, 44:39-52, 1979.
- [45] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B*, 58(1):267-288, 1996.
- [46] J. Tropp. Just relax: Convex programming methods for identifying sparse signals. *IEEE T. Inform. Theory*, 51:1030-1051, 2006.
- [47] Y. Tsuruoka, J. McNaught, J. Tsujii and S. Ananiadou. Learning string similarity measures for gene/protein name dictionary look-up using logistic regression. *Bioinformatics*, 23(20):2768-2774, 2007.
- [48] C. Wang, D. Sun and K. C. Toh. Solving log-determinant optimization problems by a Newton-CG proximal point algorithm. Technical report, National University of Singapore, 2009.
- [49] W. B. Wu and M. Pourahmadi. Nonparametric estimation of large covariance matrices of longitudinal data. *Biometrika*, 90:831-844, 2003.
- [50] K. Y. Yeung, R. E. Bumgarner and A. E. Raftery. Bayesian model averaging: development of an improved multi-class, gene selection and classification tool for microarray data. *Bioinformatics*, 21(10):2394-2402, 2005.