# Covariate Adjusted Precision Matrix Estimation via Nonconvex Optimization

**Jinghui Chen** [1] **Pan Xu** [2] **Lingxiao Wang** [2] **Jian Ma** [3] **Quanquan Gu** [2]

## Abstract

We propose a nonconvex estimator for the covariate adjusted precision matrix estimation problem in the high dimensional regime, under sparsity constraints. To solve this estimator, we propose an alternating gradient descent algorithm with hard thresholding. Compared with existing methods along this line of research, which lack theoretical guarantees in optimization error and/or statistical error, the proposed algorithm not only is computationally much more efficient with a linear rate of convergence, but also attains the optimal statistical rate up to a logarithmic factor. Thorough experiments on both synthetic and real data support our theory.

## 1 Introduction

Gaussian graphical models (Lauritzen, 1996) (GGM) have been widely used in the field of statistical machine learning. The goal is to estimate the precision matrix, which captures the conditional dependency relationship among marginal variables of high dimensional random vectors. One typical application of Gaussian graphical models is to study the conditional independence among the genes at the expression level in genomics, and to estimate the gene regulatory network. In recent years, it has been noticed that one can elaborate a GGM model with additional side information for better estimation accuracy. For example, genetic variants have been shown to have great potential influence on gene expression (Brem & Kruglyak, 2005; Cheung & Spielman, 2002), yet directly applying Gaussian graphical model to gene expression data would neglect such a fact, hence may hinder us from revealing the intrinsic gene regulation relationships. On the other hand, utilizing such genetic variant as side information to adjust the estimation of precision matrix could lead to better accuracy.

In fact, this adjusted precision matrix estimation problem can be reformulated as a problem of jointly estimating multivariate regression matrix and the precision matrix. Cai et al. (2012a) proposed a two-stage covariate-adjusted precision matrix estimation method which first estimates the regression coefficient matrix and then estimates the precision matrix based on the estimated regression coefficient matrix. Some other work targets at simultaneously estimating both the regression coefficient matrix and the precision matrix (Yin & Li, 2011; Lee & Liu, 2012; Rothman et al., 2010). Following the literature of this line of research, we briefly introduce the model as follows: given data vectors $\{\mathbf{y}_i\}_{i=1}^n \in \mathbb{R}^m$ and side information vectors $\{\mathbf{x}_i\}_{i=1}^n \in \mathbb{R}^d$, assume that

$$\mathbf{y}_i = \mathbf{\Gamma}^* \mathbf{x}_i + \boldsymbol{\epsilon}_i, \tag{1.1}$$

where $\mathbf{\Gamma}^* \in \mathbb{R}^{d\times m}$ is the unknown regression coefficient matrix, and $\boldsymbol{\epsilon}_i \in \mathbb{R}^m$ is the error vector. We assume $\{\mathbf{x}_i\}_{i=1}^n$ and $\{\boldsymbol{\epsilon}_i\}_{i=1}^n$ are independent from each other and the error vector $\{\boldsymbol{\epsilon}_i\}_{i=1}^n$ follows a multivariate normal distribution with zero mean and covariance $\mathbf{\Sigma}^*$. Therefore, given $\mathbf{x}_i$, we have $\mathbf{y}_i|\mathbf{x}_i \sim N(\mathbf{\Gamma}^*\mathbf{x}_i, \mathbf{\Sigma}^*)$. Let $\mathbf{\Omega}^* = \mathbf{\Sigma}^{*-1}$ be the corresponding precision matrix that characterizes the conditional dependency structure among the data vectors $\{\mathbf{y}_i\}_{i=1}^n$. More specifically, $\Omega^*_{ij} = 0$ implies that $i$-th and $j$-th variables are conditionally independent given the covariates and other response variables.

To estimate $\mathbf{\Omega}^*$, we construct the following conditional likelihood function corresponding to our model (1.1): $\ell(\mathbf{\Gamma}, \mathbf{\Omega}) =$

$$\prod_{i=1}^n (2\pi)^{-\frac{m}{2}} \left|\mathbf{\Omega}\right|^{\frac{1}{2}} \exp\bigg[-\frac{(\mathbf{y}_i - \mathbf{\Gamma}\mathbf{x}_i)^\top \mathbf{\Omega}(\mathbf{y}_i - \mathbf{\Gamma}\mathbf{x}_i)}{2}\bigg],$$

where $|\mathbf{\Omega}|$ denotes the determinant of $\mathbf{\Omega}$. The corresponding negative log-likelihood function can be written as (neglect the constants):

$$f_n(\mathbf{\Gamma}, \mathbf{\Omega}) = -\log|\mathbf{\Omega}| + \frac{1}{n}\operatorname{tr}\big[(\mathbf{Y} - \mathbf{X}\mathbf{\Gamma})\mathbf{\Omega}(\mathbf{Y} - \mathbf{X}\mathbf{\Gamma})^\top\big], \tag{1.2}$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n\times d}$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]^\top \in \mathbb{R}^{n\times m}$. In many applications such as eco-

[1]Department of Computer Science, University of Virginia, Charlottesville, VA 22904, USA [2]Department of Computer Science, University of California, Los Angeles, CA 90095, USA [3]School of Computer Science, Carnegie Mellon University University, Pittsburgh, PA 15213, USA. Correspondence to: Quanquan Gu <qgu@cs.ucla.edu>.

*Proceedings of the $35^{th}$ International Conference on Machine Learning*, Stockholm, Sweden, PMLR 80, 2018. 

nomics and genomics, the number of parameters $dm + \binom{m}{2}$ is often much larger than the number of observations $n$, which imposes great challenges on the model estimation. Thus one common assumption is that both $\mathbf{\Gamma}^*$ and $\mathbf{\Omega}^*$ have certain structures. In this paper, without loss of generality, we assume both $\mathbf{\Gamma}^*$ and $\mathbf{\Omega}^*$ are row sparse. Specifically, we assume $\mathbf{\Gamma}^*$ and $\mathbf{\Omega}^*$ belong to the following classes respectively:

$$\mathcal{V}(s_1^*) := \Big\{\mathbf{\Gamma} \in \mathbb{R}^{d\times m} : \max_{1\le i\le d}\sum_{j=1}^{m}\mathbb{1}\{\Gamma_{ij}\neq 0\}\le s_1^*\Big\},$$

$$\mathcal{U}(s_2^*) := \Big\{\mathbf{\Omega} \in \mathbb{R}^{m\times m} : \|\mathbf{\Omega}\|_1 \le M, \max_{1\le i\le m}\sum_{j=1}^{m}\mathbb{1}\{\Omega_{ij}\neq 0\}\le s_2^*\Big\}.$$

Therefore, $\|\mathbf{\Gamma}^*\|_{0,0} = ds_1^*$ and $\|\mathbf{\Omega}^*\|_{0,0} = ms_2^*$, where $\|\cdot\|_{0,0}$ denotes the number of nonzero entries in a matrix. Note that the constraint that $\|\mathbf{\Omega}\|_1 \le M$ is a common condition in the literature on precision matrix estimation (Cai et al., 2012a;b). Under this assumption, we propose a cardinality constrained maximum likelihood estimator as follows:

$$\begin{aligned}&\min_{\mathbf{\Gamma},\mathbf{\Omega}} -\log|\mathbf{\Omega}| + \frac{1}{n}\operatorname{tr}\big((\mathbf{Y}-\mathbf{X}\mathbf{\Gamma})\mathbf{\Omega}(\mathbf{Y}-\mathbf{X}\mathbf{\Gamma})^\top\big)\\ &\text{subject to}\quad \|\mathbf{\Gamma}\|_{0,0}\le s_1, \|\mathbf{\Omega}\|_{0,0}\le s_2,\end{aligned} \tag{1.3}$$

where $s_1$ and $s_2$ are tuning parameters which control the sparsity of $\mathbf{\Gamma}$ and $\mathbf{\Omega}$ respectively. Note that even though our ultimate goal is to estimate $\mathbf{\Omega}^*$, since the likelihood function is also related to $\mathbf{\Gamma}^*$, it is thus also important to ensure that $\mathbf{\Gamma}$ is close enough to $\mathbf{\Gamma}^*$, so that it would not downgrade the accuracy in estimating $\mathbf{\Omega}^*$.

The proposed estimator in (1.3) poses great challenges for both optimization and statistical analysis. The sample loss function in (1.2) is not jointly convex in $\mathbf{\Gamma}$ and $\mathbf{\Omega}$. This together with the nonconvex cardinality constraints make our estimator a highly non-convex optimization problem. Moreover, statistical analysis becomes quite challenging for such a non-convex estimator in such a finite sample scenario. Many previous studies along this line of research (Yin & Li, 2011; Rothman et al., 2010; Lee & Liu, 2012) are only able to characterize the asymptotic performance of their estimators. To the best of our knowledge, Cai et al. (2012a) is the only work with non-asymptotic performance guarantee for such a nonconvex model, yet it only analyzes the statistical error and does not include any optimization analysis or guarantees on the statistical estimators, which makes it less practical. To overcome these challenges, we propose an alternating gradient descent algorithm for solving the nonconvex optimization problem in (1.3). We summarize our contributions as follows:

- We propose a practical algorithm which is easy to implement and fast to compute under strict run-time analysis. Therefore, our algorithm is much faster and closer to the real world situations than existing estimator based algorithms.
- We show that the proposed algorithm is guaranteed to converge to the true precision matrix $\mathbf{\Omega}^*$ at a linear rate. In particular, the statistical rate of the estimator from our algorithm actually matches the minimax optimal up to a logarithmic factor.
- To the best of our knowledge, this is the first work to analyze the non-asymptotic optimization performance guarantee of the covariate-adjusted precision matrix estimation model. This sheds some light on how this model works in real world scenarios.

The remainder of this paper is organized as follows: in Section 2, we briefly review existing work that is relevant to our study. We present the algorithm in Section 3, and the main theory in Section 4. In Section 5, we compare the proposed algorithm with existing algorithms on both synthetic data and real datasets. Finally, we conclude this paper in Section 6.

**Notation.** Let $[n]$ denote the set of $\{1,\ldots,n\}$. For random variable $X$, we define the sub-Gaussian norm as $\|X\|_{\psi_2} = \sup_{p\ge 1} p^{-1/2}(\mathbb{E}|X|^p)^{1/p}$. For a vector $\mathbf{x}\in\mathbb{R}^d$, define $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^d x_i^2}$. For a matrix $\mathbf{A}\in\mathbb{R}^{m_1\times m_2}$, we denote by $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ the largest and smallest eigenvalue of $\mathbf{A}$ respectively. We define $\operatorname{supp}(\mathbf{A})$ as the index set of nonzero entries of $\mathbf{A}$, and $\operatorname{supp}(\mathbf{A},s)$ as the index set of the top $s$ entries of $\mathbf{A}$ in terms of magnitude. We use $\mathbf{A}_{*k}$ to denote the $k$-th column of matrix $\mathbf{A}$ and $A_{jk}$ the $(j,k)$-th element of $\mathbf{A}$. For a pair of matrices $\mathbf{A},\mathbf{B}$ with commensurate dimensions, $\langle\mathbf{A},\mathbf{B}\rangle$ denotes the trace inner product on matrix space that $\langle\mathbf{A},\mathbf{B}\rangle := \operatorname{trace}(\mathbf{A}^\top\mathbf{B})$. We also use various norms for matrices, including spectral norm $\|\mathbf{A}\|_2 = \max_{\|\mathbf{u}\|_2=1}\|\mathbf{A}\mathbf{u}\|_2$, Frobenius norm $\|\mathbf{A}\|_F = \sqrt{\sum_{j=1}^{m_1}\sum_{k=1}^{m_2}A_{jk}^2}$, infinity norm $\|\mathbf{A}\|_{\infty,\infty} = \max_{1\le j\le m_1,1\le k\le m_2}|A_{jk}|$, $\|\mathbf{A}\|_{1,1} = \sum_{j=1}^{m_1}\sum_{k=1}^{m_2}|A_{jk}|$, $\|\mathbf{A}\|_1 = \max_{1\le j\le m_1}\sum_{k=1}^{m_2}|A_{jk}|$ and $|\mathbf{A}\|_\infty = \max_{1\le k\le m_2}\sum_{j=1}^{m_1}|A_{jk}|$. $\|\mathbf{A}\|_{0,0} = \sum_{j=1}^{m_1}\sum_{k=1}^{m_2}\mathbb{1}\{A_{jk}\neq 0\}$ denotes the number of nonzero entries in $\mathbf{A}$. For $S\in[m_1]$ and $T\in[m_2]$, we define $\mathbf{A}_{ST}$ to be the submatrix of $\mathbf{A}$, which is obtained by extracting the appropriate rows and columns in $S$ and $T$ respectively. Also $C, C', \ldots$ represent absolute constants which could be of different values in different places.

## 2 Related Work

A large body of literature has been devoted to the precision matrix estimation in Gaussian graphical models (GGM)

such as $\ell_{1,1}$ norm constrained sparse GGM (GLasso) (Friedman et al., 2008; Ravikumar et al., 2011; Rothman et al., 2008), distributed GGM (Xu et al., 2016), robust GGM (Wang & Gu, 2017), sparse tensor-variate GGM (Xu et al., 2017b) and latent variable GGM (Chandrasekaran et al., 2010; Ma et al., 2013; Xu et al., 2017a). In addition, Wang et al. (2016a); Xu & Gu (2016) studied the faster rate of GGM and its variants with noncovex penalties.

The approach of utilizing additional side information for GGM is first proposed by Yin & Li (2011), in which the model is called conditional Gaussian graphical model (CGGM). Along this line of research, there exist two families of methods in general. The first family of methods (Rothman et al., 2010; Lee & Liu, 2012; Yin & Li, 2011) simultaneously estimates the regression coefficient matrix and precision matrix using alternating optimization algorithms. However, all the theoretical results (Rothman et al., 2010; Lee & Liu, 2012) for alternating minimization algorithms are based on the assumption that there exists a local minimizer that possesses certain good properties, while neither of them is guaranteed to find such a satisfying local minimizer. The second family of methods is identified by their two-step optimization procedure (Cai et al., 2012a), in which the regression coefficient matrix is estimated first and the precision matrix estimation is built based on the estimated regression coefficient matrix. However, the two-step approaches cannot fully utilize the interdependency between the regression coefficient matrix and the precision matrix in the estimation process, which often leads to a sub-optimal solution.

Another line of research which is closely related to ours is call conditional Gaussian random fields (Wytock & Kolter, 2013), sometimes also been called conditional Gaussian graphical model (CGGM) (Sohn & Kim, 2012; McCarter & Kim, 2016; Zhang & Kim, 2014) or partial Gaussian graphical models (pGGM) (Yuan & Zhang, 2014). This model may seem similar to ours, yet we want to emphasize that it is fundamentally different from ours. The key difference is that their model does not make the sparsity assumption on $\mathbf{\Gamma}$ [1]. Due to this different sparsity assumption, their model is convex while ours is not. Yet the sparsity assumption on $\mathbf{\Gamma}$ is beneficial in modeling real world data.

In terms of nonconvex optimization technique, Yuan et al. (2013); Jain et al. (2014); Chen & Gu (2016) proposed and analyzed the gradient descent algorithm with hard thresholding for cardinality constrained optimization problems. However, these algorithms are limited to single optimization variable case. Jain & Tewari (2015) proposed an alternating minimization algorithm for two regression models (i.e., pooled model and seemingly unrelated regression model) in classical regime. Chen & Banerjee (2017) follows the same model and analyze the statistical guarantee in terms of Gaussian width. Yet their proof technique is specific to the particular regression models they studied, and cannot be extended to our model. In addition, alternating minimization has also been analyzed for other models such as matrix factorization (Jain et al., 2013; Arora et al., 2015; Zhao et al., 2015; Zheng & Lafferty, 2015; Chen & Wainwright, 2015; Tu et al., 2015; Wang et al., 2016b; 2017), robust PCA (Gu et al., 2016; Zhang et al., 2018), phase retrieval (Candès et al., 2015; Chen et al., 2017) and latent variable models (Balakrishnan et al., 2014; Wang et al., 2015; Zhu et al., 2017). Yet none of these algorithms and theories can be directly extended to our problem.

[1] In their model, the sparsity assumptions are made on some other related matrices, please refer to their papers for more details

## 3 The Proposed Algorithm

In this section, we present a gradient descent based optimization algorithm for solving the proposed estimator in (1.3). The key motivation of the algorithm is that the objective function in (1.3) is bi-convex, i.e., it is convex with respect to $\mathbf{\Gamma}$ (resp. $\mathbf{\Omega}$) when the other variable is fixed. Therefore, we propose to optimize the target objective function by performing gradient descent with respect to $\mathbf{\Gamma}$ and $\mathbf{\Omega}$ alternatingly. The details about the proposed algorithm is shown in Algorithm 1.

**Algorithm 1** Alternating Gradient Descent with Hard Thresholding

1: **Input:** Number of iterations $T$, sparsity $s_1, s_2$, step size $\eta_1, \eta_2$.
2: **for** $t = 0$ to $T - 1$ **do**
3: **Update $\mathbf{\Gamma}$:**
$\mathbf{\Gamma}^{(t+0.5)} = \mathbf{\Gamma}^{(t)} - \eta_1 \nabla_1 f_n\big(\mathbf{\Gamma}^{(t)}, \mathbf{\Omega}^{(t)}\big)$,
$\mathbf{\Gamma}^{(t+1)} = \mathcal{HT}(\mathbf{\Gamma}^{(t+0.5)}, s_1)$
4: **Update $\mathbf{\Omega}$:**
$\mathbf{\Omega}^{(t+0.5)} = \mathbf{\Omega}^{(t)} - \eta_2 \nabla_2 f_n\big(\mathbf{\Gamma}^{(t)}, \mathbf{\Omega}^{(t)}\big)$,
$\mathbf{\Omega}^{(t+1)} = \mathcal{HT}(\mathbf{\Omega}^{(t+0.5)}, s_2)$
5: **end for**
6: **Output:** $\widehat{\mathbf{\Gamma}} = \mathbf{\Gamma}^{(T)}$, $\widehat{\mathbf{\Omega}} = \mathbf{\Omega}^{(T)}$

In Algorithm 1, $\mathbf{\Gamma}^{(t+0.5)}$ and $\mathbf{\Omega}^{(t+0.5)}$ are the outputs of gradient descent update. Note that in Algorithm 1, $\nabla_1 f_n$ denotes the gradient of $f_n$ with respect to $\mathbf{\Gamma}$, and $\nabla_2 f_n$ denotes its gradient with respect to $\mathbf{\Omega}$. The hard thresholding procedure (Yuan et al., 2013; Jain et al., 2014) right after gradient descent update is for ensuring the sparsity of the parameters after the gradient descent update. Specifically,

$$[\mathcal{HT}(\mathbf{A}, s)]_{ij} = \begin{cases} A_{ij}, & \text{if } (i,j) \in \text{supp}(\mathbf{A}, s) \\ 0, & \text{otherwise} \end{cases}. \tag{3.1}$$

In other words, the hard thresholding step preserves the largest $s_1$ and $s_2$ entries in $\mathbf{\Gamma}^{(t+0.5)}$ and $\mathbf{\Omega}^{(t+0.5)}$ respectively in terms of magnitudes and sets the rest to zero. This

gives rise to $\mathbf{\Gamma}^{(t+1)}$ and $\mathbf{\Omega}^{(t+1)}$. Recall that $s_1$ and $s_2$ are tuning parameters that control the sparsity level.

Algorithm 1 provides an efficient way to solve the nonconvex problem using gradient descent with hard thresholding. Yet it requires that the initial estimators $\mathbf{\Gamma}^{(0)}$ and $\mathbf{\Omega}^{(0)}$ to fall into the contraction region in order to work. In order to obtain a good pair of initial estimators $\mathbf{\Gamma}^{(0)}$ and $\mathbf{\Omega}^{(0)}$, we introduce the initialization algorithm in Algorithm 2.

**Algorithm 2** Initialization

1: **Input:** Regularization parameters $\lambda_\Gamma$, $\lambda_\Omega$ and $\lambda_u$
2: $\mathbf{\Gamma}^{\text{init}} = \mathcal{ST}\big((\mathbf{X}^\top\mathbf{X} + \epsilon_\Gamma\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y}, \lambda_\Gamma\big)$
3: $\mathbf{S} = (\mathbf{Y} - \mathbf{X\Gamma})^\top(\mathbf{Y} - \mathbf{X\Gamma})/n$
4: $\mathbf{\Omega}^{\text{init}} = \mathcal{ST}\big(\mathbf{S} + \epsilon_\Omega\mathbf{I})^{-1}, \lambda_\Omega\big)$
5: $\mathbf{\Gamma}^{(0)} = \mathcal{HT}(\mathbf{\Gamma}^{\text{init}}, s_1)$, $\mathbf{\Omega}^{(0)} = \mathcal{HT}(\mathbf{\Omega}^{\text{init}}, s_2)$

In Algorithm 2, we apply the closed form solution for the elementary estimator for linear regression (Yang et al., 2014a) and a revised version of elementary estimator for graphical models to obtain initial estimators $\mathbf{\Gamma}^{\text{init}}$ and $\mathbf{\Omega}^{\text{init}}$. Here is $\mathcal{ST}$ stands for the soft thresholding operator which is defined as follows:

$$[\mathcal{ST}(\mathbf{A}, \lambda)]_{ij} = \text{sign}(A_{ij}) \cdot \max(|A_{ij}| - \lambda, 0). \quad (3.2)$$

Algorithm 2 ensures that the initial estimators $\mathbf{\Gamma}^{(0)}$ and $\mathbf{\Omega}^{(0)}$ are sufficiently close to $\mathbf{\Gamma}^*$ and $\mathbf{\Omega}^*$ respectively and thus falls into the contraction region. While Algorithm 1 guarantees the model parameters' convergence inside the contraction region. By combining Algorithm 1 and Algorithm 2, we ensure that our nonconvex optimization algorithm will converge to the true parameters.

## 4 Main Theory

Before we present the main results, we first lay out a series of assumptions, which are essential for establishing our theory.

**Assumption 4.1.** There exist some $\nu \geq 1$ such that the maximum and minimum eigenvalues of $\mathbf{\Omega}^*$ satisfy

$$1/\nu \leq \lambda_{\min}(\mathbf{\Omega}^*) \leq \lambda_{\max}(\mathbf{\Omega}^*) \leq \nu,$$

where $\nu$ is an absolute constants which do not depend on $m$.

Note that the same assumption has also been made in (Lee & Liu, 2012; Wang, 2013; Cai et al., 2012b).

**Assumption 4.2.** Let $\{\mathbf{x}_i\}_{i=1}^n$ be the rows in $\mathbf{X}$. Each $\mathbf{x}_i$ is a sub-Gaussian random vector. In addition, let $\mathbf{\Sigma}_X^* = n^{-1}\mathbb{E}[\mathbf{X}^\top\mathbf{X}]$. There exists $\tau \geq 1$ such that

$$1/\tau \leq \lambda_{\min}(\mathbf{\Sigma}_X^*) \leq \lambda_{\max}(\mathbf{\Sigma}_X^*) \leq \tau,$$

and

$$\|{\mathbf{\Sigma}_X^*}^{-\frac{1}{2}}\mathbf{x}_i\|_{\psi_2} \leq K, \quad \text{for all } i = 1, \ldots, n$$

where $\tau, K$ are absolute constants independent of $n, d$.

Assumption 4.2 states that the minimum eigenvalue of the population covariance matrix of the predictors is bounded away from zero. This assumption is mild and has been widely made in the literature of multivariate regression (Obozinski et al., 2011; Lounici et al., 2009; Negahban & Wainwright, 2011). Also since $\|{\mathbf{\Sigma}_X^*}^{-\frac{1}{2}}\mathbf{x}_i\|_{\psi_2}$ is bounded for all $i = 1, \ldots, n$, it immediately implies that $\|\mathbf{x}_i\|_{\psi_2} \leq \sqrt{\tau}\|{\mathbf{\Sigma}_X^*}^{-\frac{1}{2}}\mathbf{x}_i\|_{\psi_2}$ is also bounded.

Now we are going to present our main theorem. To simplify the technical analysis, we focus on the resampling version of Algorithm 1, which is illustrated in Algorithm 3 in the supplementary material. The key idea of resampling (or sample splitting) (Hansen, 2000; Balakrishnan et al., 2017) is to split the whole dataset into $T$ pieces and use a fresh piece of data in each iteration. The main propose for resampling is to remove the statistical dependencies between iterates.

**Theorem 4.3.** Under Assumptions 4.1 and 4.2, let $R := \min\big\{1/(\nu\tau^2), 1/(4\nu^2\tau), \sqrt{\nu/\tau}, M\big\}$. Suppose the initial value $\mathbf{\Gamma}^{(0)}$ and $\mathbf{\Omega}^{(0)}$ satisfy $\max\{\|\mathbf{\Gamma}^{(0)} - \mathbf{\Gamma}^*\|_F, \|\mathbf{\Omega}^{(0)} - \mathbf{\Omega}^*\|_F\} \leq R$. Let the sparsity parameters satisfy $s_1 \geq \big(1 + 4/(1/\rho - 1)^2\big)ds_1^*, s_2 \geq \big(1 + 4/(1/\rho - 1)^2\big)ms_2^*$, where

$$\rho = \max\left\{1 - \frac{2 - 2R\nu\tau^2}{\nu^2\tau^2 + 1}, 1 - \frac{2 - 8\tau\nu^2 R}{16\nu^4 + 1}\right\}.$$

And suppose the sample size $n$ satisfies that

$$n \geq \frac{CM^2T\max\{\nu\tau ds_1^*, ms_2^*\}\log(dmT)}{(1 - \sqrt{\rho})^2R^2}. \quad (4.1)$$

Let $\eta_1 = \nu\tau/(\nu^2\tau^2 + 1)$, $\eta_2 = 8\nu^2/(16\nu^4 + 1)$, for all $t \in [T]$, we have with probability at least $1 - 2/d - C'/m$ that

$$\max\big\{\|\mathbf{\Gamma}^{(t)} - \mathbf{\Gamma}^*\|_F, \|\mathbf{\Omega}^{(t)} - \mathbf{\Omega}^*\|_F\big\} \leq \underbrace{\frac{C''M\max\big\{\sqrt{\nu\tau ds_1^*}, \sqrt{ms_2^*}\big\}}{1 - \sqrt{\rho}}\sqrt{\frac{\log(dmT)}{n/T}}}_{\text{Statistical Error}} + \underbrace{R\cdot\rho^{t/2}}_{\text{Optimization Error}}, \quad (4.2)$$

where $C, C', C''$ are absolute constants.

**Remark 4.4.** Conditions in Theorem 4.3 imply that the sparsity parameters $s_1$ and $s_2$ should be chosen to be sufficiently large but meanwhile in the same order as the true sparsity level $ds_1^*$ and $ms_2^*$ respectively. This ensures that the extra error caused by hard thresholding step can be upper bounded. Moreover, we can observe that the definition of $R$ indeed guarantees that $\rho < 1$. Therefore our proposed algorithm indeed converges.

**Remark 4.5.** In Theorem 4.3, the result suggests that the estimation error is bounded by two terms: the optimization error term (i.e., the second term on the right hand side of (4.2)), which decays to zero at a linear rate, and the statistical error term (i.e., the first term on the right hand side of (4.2)), which characterizes the the unavoidable estimation error in Algorithm 3 when the optimization error term goes to zero as $T$ goes to infinity.

In the next corollary, we show the statistical error achieved by our proposed method matches the minimax lower bound.

**Corollary 4.6.** Under the same assumptions and conditions as in Theorem 4.3, suppose $s_1^*$ satisfies $s_1^* \leq ms_2^*/(d\nu\tau)$, and if we choose the number of iterations $T = C\log n$ for sufficiently large $C$ such that the optimization error term is dominated by the statistical error term, then we have

$$\|\widehat{\boldsymbol{\Omega}} - \boldsymbol{\Omega}^*\|_F \leq \frac{C' M\sqrt{\log n}}{1-\sqrt{\rho}}\sqrt{\frac{ms_2^*\log m}{n}},$$

where $C, C'$ are some absolute constants.

Comparing with the minimax lower bound, which is in the order of $O\big(M\sqrt{ms_2^*\log m/n}\big)$ (Cai et al., 2012b), our bounds on $\boldsymbol{\Omega}^*$ matches the minimax lower bounds aside from an additional logarithmic term $\sqrt{\log n}$. Such a logarithmic factor is introduced by the resampling step in Algorithm 3, since we only utilize $n/T$ samples within each iteration. We expect that it is an artifact of our proof technique, and such a logarithmic factor can be eliminated by directly analyzing Algorithm 1, which however requires extra technical effort for the analysis.

**Theorem 4.7.** Under Assumptions 4.1 and 4.2, let $R := \min\big\{1/(\nu\tau^2), 1/(4\nu^2\tau), \sqrt{\nu/\tau}, M\big\}$. Suppose the sample size $n$ satisfies

$$n \geq C\max\left\{\frac{\tau(ds_1^*)^2}{R^3}, \frac{M^2\nu^2 ms_2^*}{R^2}, \sqrt[5]{\frac{M^6\nu^3\tau^5(ds_1^*)^3(ms_2^*)^4}{R^6}}\right\}\log(dm),$$

then the output from Algorithm 2 satisfies

$$\max\{\|\boldsymbol{\Gamma}^{(0)} - \boldsymbol{\Gamma}^*\|_F, \|\boldsymbol{\Omega}^{(0)} - \boldsymbol{\Omega}^*\|_F\} \leq R.$$

**Remark 4.8.** Notice that for initialization we can also choose to adopt multivariate Lasso for estimating $\boldsymbol{\Gamma}^{\text{init}}$ and graphical Lasso for initializing $\boldsymbol{\Omega}^{\text{init}}$, which is also provable and could lead to a bit better sample complexity. The price to pay is higher computational complexity, since elementary estimators have closed-form solutions. Yet in experiments part, we observe that the current initialization mechanism can also provide accurate enough initial estimators which satisfy the requirements from Algorithm 1 under the same sample complexity constraint.

*Table 1.* Comparison of run time complexity for our proposed algorithm and other baselines. $T_o$ denotes the number of outer iterations.

| Methods | Run-time Complexity | Linear Rate |
|---|---|---|
| MRCE | $O\big((m^2d^2+m^3)\cdot(1/\sqrt{\epsilon})\cdot T_o\big)$ | No |
| Capme | $O\big((d^3+m^3)\cdot(1/\epsilon)\big)$ | No |
| Alt-NCD | $O\big((m\cdot(1/\sqrt{\epsilon})+m^3+nm^2)\cdot T_o\big)$ | No |
| Ours | $O\big(m(d+m)(n+m)\cdot\log(1/\epsilon)\big)$ | Yes |

### 4.1 Runtime Complexity Analysis

We compare the run-time complexity in terms of the optimization error for all baseline algorithms in Table 1. Note that none of baseline algorithms actually proved the linear rate of convergence. As a consequence, their run-time complexity can only be written as per-iteration complexity times the outer iteration $T_o$. For MRCE, in each iteration, it requires to solve a coordinate descent sub-problem which will take at least $1/\sqrt{\epsilon}$ inner loops to converge. For Capme, it only performs one outer iteration, yet since it adopts Dantzig selector and CLIME estimator in their framework, it will take at least $1/\epsilon$ inner loops to converge, which is quite slow. Alt-NCD adopts a convex problem formulation and is claimed to be much scalable especially when their memory is limited. Yet in terms of run-time complexity, in each iteration it also requires to alternatingly solve a coordinate descent problem which takes at least $1/\sqrt{\epsilon}$ inner loops to converge. By securing a linear rate of convergence and only required to solve one step gradient update inside each iteration, our proposed algorithm clearly enjoys better run-time complexity comparing with all these baselines.

## 5 Experiments

In this section, we will present numerical results on both synthetic and real datasets to verify the performance of the proposed algorithm in Algorithm 1. We compare our algorithm with several state-of-the-art baseline algorithms:

- Multivariate regression with covariance estimation (MRCE) by Rothman et al. (2010);
- Covariate-adjusted precision matrix estimation (Capme) by Cai et al. (2012a)
- Alternating Newton coordinate descent algorithm (Alt-NCD) by McCarter & Kim (2016).

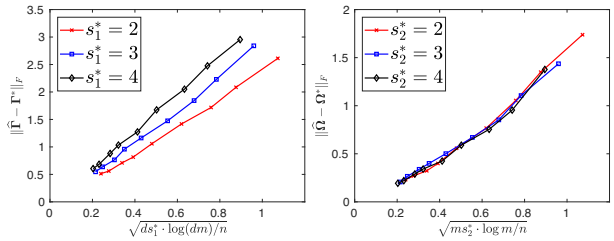
Note that McCarter & Kim (2016) adopted different problem formulations (convex rather than bi-convex) with the rest of the algorithms including ours. Also, in McCarter & Kim (2016) it presented a block coordinate descent version for solving limited memory case, which we do not compare with.

*Table 2.* Comparison of estimation error (in terms of $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ and $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$) and running time (seconds) on synthetic dataset over 10 replications. N/A means the algorithm cannot output the solution in an hour. $\mathbf{\Omega}^*$ is generated from Hub graph.

| | $n = 500, m = 500, d = 500$ | | | $n = 1000, m = 1000, d = 1000$ | | | $n = 1000, m = 1500, d = 1500$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Methods | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time |
| MRCE | 48.32±0.15 | 11.05±0.03 | 22.87 | 67.35±0.09 | 15.63±0.02 | 185.56 | 81.12±0.08 | 18.95±0.03 | 395.00 |
| Alt-NCD | 66.05±0.09 | 8.38±0.01 | 8.37 | 92.81±0.04 | 11.74±0.02 | 77.94 | 111.99±0.06 | 14.15±0.01 | 324.67 |
| Capme | 37.78±0.05 | 16.90±0.05 | 345.44 | 54.36±0.02 | 9.49±0.02 | 2679.38 | N/A | N/A | N/A |
| **Ours** | **14.79±0.18** | **6.69±0.46** | **2.99** | **16.01±0.17** | **5.42±0.03** | **16.72** | **21.63±0.16** | **6.84±0.02** | **49.48** |

*Table 3.* Comparison of estimation error (in terms of $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ and $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$) and running time (seconds) on synthetic dataset over 10 replications. N/A means the algorithm cannot output the solution in an hour. $\mathbf{\Omega}^*$ is generated from Band graph.

| | $n = 500, m = 500, d = 500$ | | | $n = 1000, m = 1000, d = 1000$ | | | $n = 1000, m = 1500, d = 1500$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Methods | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time |
| MRCE | 47.46±0.10 | 15.16±0.03 | 12.44 | 67.01±0.05 | 21.55±0.02 | 65.09 | 81.17±0.07 | 26.16±0.03 | 246.53 |
| Alt-NCD | 58.49±0.05 | 13.76±0.01 | 9.74 | 84.85±0.04 | 19.50±0.01 | 64.74 | 102.08±0.06 | 23.86±0.01 | 187.21 |
| Capme | 37.85±0.04 | 10.91±0.06 | 390.90 | 54.23±0.04 | 15.82±0.04 | 3415.07 | N/A | N/A | N/A |
| **Ours** | **15.64±0.19** | **8.24±0.13** | **2.48** | **13.89±0.12** | **5.51±0.14** | **14.07** | **18.51±0.17** | **6.64±0.09** | **35.71** |




(a) (b)

*Figure 1.* (a): Scaled error plot for $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ under different sparsity settings (b): Scaled error plot for $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ under different sparsity settings. $\mathbf{\Omega}^*$ is generated from Cluster graph.

### 5.1 Synthetic Data

In each replication of each model, we generate an $n \times d$ predictor matrix $\mathbf{X}$ with rows drawn independently from a multivariate normal distribution $N(\mathbf{0}, \mathbf{I})$ as in Cai et al. (2012a). On the other hand, the data matrix $\mathbf{Y}$ is generated from model (1.1), i.e., $\mathbf{y}_i|\mathbf{x}_i \sim N(\mathbf{\Gamma}^*\mathbf{x}_i, \mathbf{\Omega}^{*-1})$ where the precision matrix $\mathbf{\Omega}^*$ is generated using the following sparse models: (1) Hub graph; (2) Band graph; (3) Cluster graph.

We compare the performance on synthetic data with three different settings: (1) $n = 500, m = 500, d = 500$ (2) $n = 1000, m = 1000, d = 1000$ (3) $n = 1000, m = 1500, d = 1500$. Each setting is repeated for 10 times. The averaged estimation error for both $\mathbf{\Gamma}$ and $\mathbf{\Omega}$, and also the running time for all settings precision matrix types are reported in Tables 2, 3, 4. We can observe that in terms of the estimation error of both $\mathbf{\Gamma}$ and $\mathbf{\Omega}$, our proposed algorithm achieves the best accuracy and also the fastest running time comparing with the state-of-the-art algorithms. To be more specific, MRCE achieves the least accurate $\mathbf{\Omega}$ estimation with mediocre running time. Capme achieves relatively high accuracy on $\mathbf{\Omega}$ estimation, yet it is not quite scalable with a large amount of time needed. Even comparing with Alt-NCD, which adopts a much easier (convex) problem formulation, our algorithm still outperforms it with clear advantage. Also due to different assumptions on $\mathbf{\Gamma}^*$, the $\mathbf{\Gamma}$ estimation error for Alt-NCD is the least accurate.

Figure 1 illustrates the scaling of the estimation error for $\mathbf{\Gamma}^*$ and $\mathbf{\Omega}^*$ respectively. The $x$ axis of these graphs is the rescaled sample size. This result support our conclusion that our estimator by Algorithm 1 achieves $O\big(\sqrt{ds_1^* \log(dm)/n}\big)$ statistical estimation error for $\mathbf{\Gamma}^*$, and $O\big(\sqrt{ms_2^* \log m/n}\big)$ statistical estimation error for $\mathbf{\Omega}^*$.

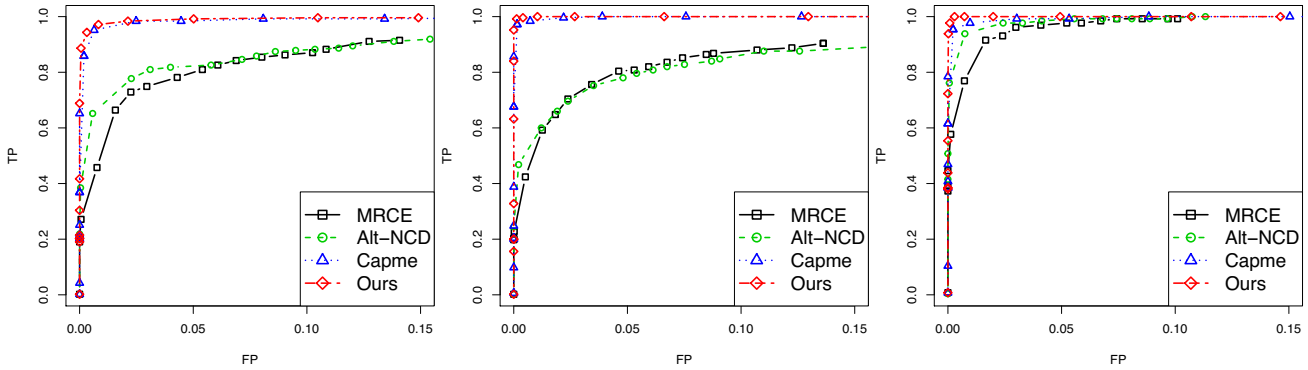
Figure 2 demonstrates the support recovery results of $\mathbf{\Omega}^*$ under three different graph structure of the precision matrix. We use receiver operating characteristic (ROC) curves to compare the support recovery performance of our proposed algorithm with other baselines algorithm. From figure 2 we can see that our proposed algorithm outperforms all other baselines.

### 5.2 eQTL analysis on Yeast Data

We demonstrate the effectiveness of our proposed method by applying it on an eQTL dataset (yeast) from Brem & Kruglyak (2005), which contains the expression measurements of 5,740 transcripts measured on 112 yeast segregants grown from two yeast parent strains: BY4716 (BY) and RM11-1a (RM), with dense genotype data on 2,956 markers. Our goal is to decode the gene regulatory relationships. Here we choose to analyze the gene set selected from yeast cell cycle *Saccharomyces Cerevisiae* pathway provided by the KEGG database (Ogata et al., 1999). We implemented our method to analyze the 92 genes involved in the cell cycle pathway, along with 787 markers. We utilize five-fold cross validation to select the optimal parameters. As a result, our

*Table 4.* Comparison of estimation error (in terms of $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ and $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$) and running time (seconds) on synthetic dataset over 10 replications. N/A means the algorithm cannot output the solution in an hour. $\mathbf{\Omega}^*$ is generated from Cluster graph.

| | $n = 500, m = 500, d = 500$ | | | $n = 1000, m = 1000, d = 1000$ | | | $n = 1000, m = 1500, d = 1500$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Methods | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time | $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^*\|_F$ | $\|\widehat{\mathbf{\Omega}} - \mathbf{\Omega}^*\|_F$ | Time |
| MRCE | 48.63±0.10 | 21.27±0.02 | 16.92 | 65.60±0.09 | 30.07±0.02 | 86.80 | 81.78±0.11 | 36.98±0.03 | 546.45 |
| Alt-NCD | 60.00±0.06 | 20.19±0.01 | 8.47 | 82.95±0.06 | 28.56±0.01 | 44.44 | 103.29±0.03 | 34.99±0.01 | 146.29 |
| Capme | 38.52±0.03 | 16.83±0.04 | 508.98 | 53.25±0.04 | 24.16±0.02 | 3585.11 | N/A | N/A | N/A |
| **Ours** | **15.07±0.24** | **6.01±0.38** | **3.23** | **13.28±0.11** | **4.31±0.04** | **17.36** | **18.86±0.16** | **5.31±0.04** | **48.30** |




(a) Band graph (b) Cluster graph (c) Hub graph

*Figure 2.* ROC plot for the support recovery of $\mathbf{\Omega}^*$ from three different graph structure of the precision matrix, where $n = 500$, $d = 100$, $m = 100$.

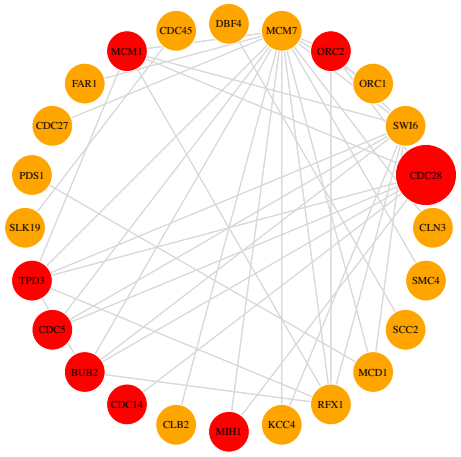



*Figure 3.* Gene network recovered by our proposed covariate adjusted method for the 92 genes on the cell-cycle yeast pathway.

proposed covariate adjusted precision estimation method identifies 102 links among 92 genes, and find around 1000 nonzero entries for coefficient matrix, suggesting that a lot of gene expression levels are affected by genetic variants.

Figure 3 displays the gene network recovered by our proposed method. Although the estimated gene network may not fully recover the cell-cycle pathway due to some inherent limitations, such as the lack of observations and the missing data for gene expression in our datasets, we find that our method reveals meaningful observations. For instance, gene CDC28 (the catalytic subunit of the cyclin-dependent kinase) is connected with genes MIH1, MCM1, ORC2, TPD3, CDC5, BUB2, implying a strong interaction mechanism between CDC28 and those genes.

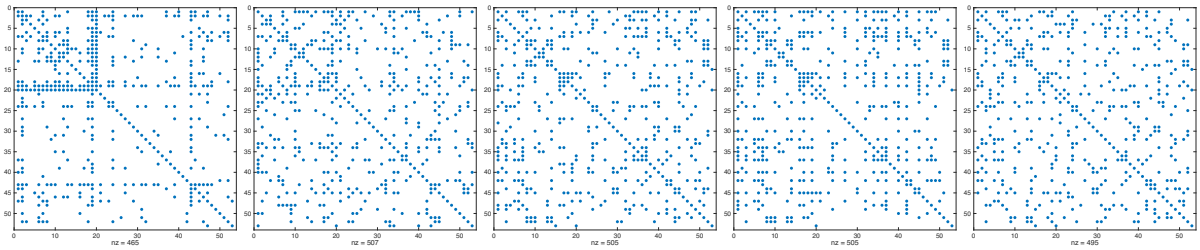
We also implement other baseline methods: MRCE, Alt-NCD, Campe on the same dataset for direct comparisons. In detail, we choose the optimal parameters for these methods by five-fold cross validation, and the gene networks obtained by the baseline methods can be found in Supplemental Materials. We found that other methods cannot recover meaningful results as compared to the known KEGG pathway.

*Table 5.* Comparison of F1 score for gene regulation network support recovery for different algorithms.

| Methods | F1 Score | #Non-zero Elements in $\mathbf{\Omega}$ |
|---|---|---|
| MRCE | 0.38 | 505 |
| Alt-NCD | 0.40 | 505 |
| Campe | 0.37 | 495 |
| **Ours** | **0.46** | 507 |

### 5.3 eQTL analysis on GTEx Data

The Genotype-Tissue Expression (GTEx) project generated RNA-seq expression data for a large number of human tissues (as of February 2018, there are 11688 samples in more than 53 tissues) (Lonsdale et al., 2013). By analyzing global RNA expression within individual tissues and treating the



(a) Groud Truth (b) Ours (c) MRCE (d) Alt-NCD (e) Capme

*Figure 4.* Support recovery result on GTEx data for different algorithms comparing with the ground truth.

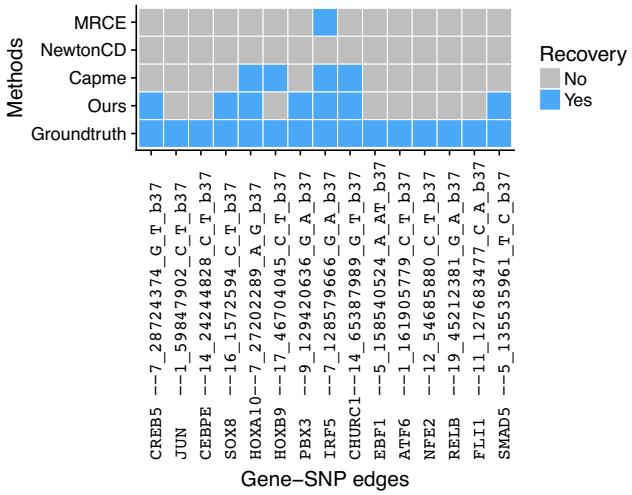



*Figure 5.* The gene-SNPs regulation support recovery result by different algorithms and comparison with ground truth.

expression levels of genes as quantitative traits, variations in gene expression that are highly correlated with genetic variation can be identified as expression quantitative trait loci (eQTLs). Here we choose to focus on the whole blood dataset due to its relatively large number of samples. We specifically look at 19 genes that are related to GATA1, which is a key TF regulator in blood cells, from the Erythroid (K562) network (Neph et al., 2012). In the dataset, we also randomly add 33 other genes from the Erythroid (K562) network. The total number of genes selected is 53. We also select 333 SNPs from the dataset, 15 of which are known to be significantly related to one of genes selected above.

Figure 4 describes the gene regulation network (GRN) that supports the results for different algorithms including ours, on GTEx dataset. Comparing with the ground truth, we can see that our proposed algorithm is the only one that nearly recovers the upper block structure. Note that GATA1 is numbered 20 in the plot, it is clear that our proposed algorithm achieves better performance in identifying the gene regulation relationship for the GATA1 regulatory network. In Table 5 we report the F1 score for recovering the GRN, and also the number of non-zero elements for a fair comparison. It also shows the clear advantage of our proposed algorithm over other baselines.

Figure 5 demonstrates the gene-SNP regulation relationship recovery results. We found that even though all the algorithms cannot fully recover all the significant gene-SNP pairs, our proposed algorithm still outperforms all other baselines. Note that Alt-NCD method does not successfully recover any significant gene-SNP pairs, possibly due to the fact that it adopts a different problem formulation (did not assume sparsity on $\mathbf{\Gamma}$), which is less intuitive towards this regression task. Taken together, these results demonstrate the potential of our proposed algorithm in identifying important gene regulatory relationships by jointly considering gene-gene interaction and variant-gene relationships.

## 6 Conclusions

In this paper, we presented a gradient descent algorithm with hard thresholding for joint multivariate regression and precision matrix estimation in the high dimensional regime, under cardinality constraints. It attains a linear convergence to the true regression coefficients and precision matrix simultaneously, up to a near optimal statistical error. Compared with existing methods along this line of research, the proposed algorithm out-performs the baseline algorithm in both accuracy and running time. Thorough experiments on synthetic datasets support our theory and the real world eQTL experiments on yeast and GTEx dataset shows the promising potential of applying our proposed algorithm in biological studies.

## Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. This research was sponsored in part by the National Science Foundation IIS-1618948, IIS-1652539 and IIS-1717206. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

## References


Arora, S., Ge, R., Ma, T., and Moitra, A. Simple, efficient, and neural algorithms for sparse coding. *arXiv preprint arXiv:1503.00778*, 2015.

Balakrishnan, S., Wainwright, M. J., and Yu, B. Statistical guarantees for the EM algorithm: From population to sample-based analysis. *arXiv preprint arXiv:1408.2156*, 2014.

Balakrishnan, S., Wainwright, M. J., Yu, B., et al. Statistical guarantees for the em algorithm: From population to sample-based analysis. *The Annals of Statistics*, 45(1):77–120, 2017.

Brem, R. B. and Kruglyak, L. The landscape of genetic complexity across 5,700 gene expression traits in yeast. *Proceedings of the National Academy of Sciences of the United States of America*, 102(5):1572–1577, 2005.

Cai, T. T., Li, H., Liu, W., and Xie, J. Covariate-adjusted precision matrix estimation with an application in genetical genomics. *Biometrika*, 100(1):139–156, 2012a.

Cai, T. T., Liu, W., and Zhou, H. H. Estimating sparse precision matrix: Optimal rates of convergence and adaptive estimation. *arXiv preprint arXiv:1212.2882*, 2012b.

Candès, E. J., Li, X., and Soltanolkotabi, M. Phase retrieval via wirtinger flow: Theory and algorithms. *Information Theory, IEEE Transactions on*, 61(4):1985–2007, 2015.

Chandrasekaran, V., Parrilo, P. A., and Willsky, A. S. Latent variable graphical model selection via convex optimization. In *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pp. 1610–1613. IEEE, 2010.

Chen, J. and Gu, Q. Accelerated stochastic block coordinate gradient descent for sparsity constrained nonconvex optimization. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, pp. 132–141. AUAI Press, 2016.

Chen, J., Wang, L., Zhang, X., and Gu, Q. Robust wirtinger flow for phase retrieval with arbitrary corruption. *arXiv preprint arXiv:1704.06256*, 2017.

Chen, S. and Banerjee, A. Alternating estimation for structured high-dimensional multi-response models. In *Advances in Neural Information Processing Systems*, pp. 2835–2844, 2017.

Chen, Y. and Wainwright, M. J. Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. *arXiv preprint arXiv:1509.03025*, 2015.

Cheung, V. G. and Spielman, R. S. The genetics of variation in gene expression. *Nature genetics*, 32:522, 2002.

Friedman, J., Hastie, T., and Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3): 432–441, 2008.

Gu, Q., Wang, Z. W., and Liu, H. Low-rank and sparse structure pursuit via alternating minimization. In *Artificial Intelligence and Statistics*, pp. 600–609, 2016.

Hansen, B. E. Sample splitting and threshold estimation. *Econometrica*, 68(3):575–603, 2000.

Jain, P. and Tewari, A. Alternating minimization for regression problems with vector-valued outputs. In *Advances in Neural Information Processing Systems*, pp. 1126–1134, 2015.

Jain, P., Netrapalli, P., and Sanghavi, S. Low-rank matrix completion using alternating minimization. In *STOC*, pp. 665–674, 2013.

Jain, P., Tewari, A., and Kar, P. On iterative hard thresholding methods for high-dimensional m-estimation. In *NIPS*, pp. 685–693, 2014.

Lauritzen, S. L. *Graphical Models*. Clarendon Press, Oxford, 1996.

Lee, W. and Liu, Y. Simultaneous multiple response regression and inverse covariance matrix estimation via penalized gaussian maximum likelihood. *Journal of multivariate analysis*, 111: 241–255, 2012.

Li, X., Zhao, T., Arora, R., Liu, H., and Haupt, J. Stochastic variance reduced optimization for nonconvex sparse learning. In *International Conference on Machine Learning*, pp. 917–925, 2016.

Loh, P.-L. and Wainwright, M. J. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. In *NIPS*, pp. 476–484, 2013.

Lonsdale, J., Thomas, J., Salvatore, M., Phillips, R., Lo, E., Shad, S., Hasz, R., Walters, G., Garcia, F., Young, N., et al. The genotype-tissue expression (gtex) project. *Nature genetics*, 45 (6):580, 2013.

Lounici, K., Pontil, M., Tsybakov, A. B., and Van De Geer, S. Taking advantage of sparsity in multi-task learning. *Proc. Computational Learning Theory Conference*, 2009.

Ma, S., Xue, L., and Zou, H. Alternating direction methods for latent variable gaussian graphical model selection. *Neural computation*, 25(8):2172–2198, 2013.

McCarter, C. and Kim, S. Large-scale optimization algorithms for sparse conditional gaussian graphical models. In *Artificial Intelligence and Statistics*, pp. 528–537, 2016.

Negahban, S. N. and Wainwright, M. J. Simultaneous support recovery in high dimensions: Benefits and perils of block-regularization. *IEEE Transactions on Information Theory*, 57 (6):3841–3863, 2011.

Neph, S., Stergachis, A. B., Reynolds, A., Sandstrom, R., Borenstein, E., and Stamatoyannopoulos, J. A. Circuitry and dynamics of human transcription factor regulatory networks. *Cell*, 150(6): 1274–1286, 2012.

Nesterov, Y. *Introductory lectures on convex optimization*, volume 87. Springer Science & Business Media, 2004.

Obozinski, G., Wainwright, M. J., and Jordan, M. I. High-dimensional union support recovery in multivariate. 2011.

Ogata, H., Goto, S., Sato, K., Fujibuchi, W., Bono, H., and Kanehisa, M. Kegg: Kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 27(1):29–34, 1999.

Ravikumar, P., Wainwright, M. J., Raskutti, G., Yu, B., et al. High-dimensional covariance estimation by minimizing 1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5: 935–980, 2011.

Rothman, A. J., Bickel, P. J., Levina, E., Zhu, J., et al. Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515, 2008.

Rothman, A. J., Levina, E., and Zhu, J. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962, 2010.

Sohn, K.-A. and Kim, S. Joint estimation of structured sparsity and output structure in multiple-output regression via inverse-covariance regularization. In *International Conference on Artificial Intelligence and Statistics*, pp. 1081–1089, 2012.

Tu, S., Boczar, R., Soltanolkotabi, M., and Recht, B. Low-rank solutions of linear matrix equations via procrustes flow. *arXiv preprint arXiv:1507.03566*, 2015.

Vershynin, R. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Wang, J. Joint estimation of sparse multivariate regression and conditional graphical models. *arXiv preprint arXiv:1306.4410*, 2013.

Wang, L. and Gu, Q. Robust gaussian graphical model estimation with arbitrary corruption. In *International Conference on Machine Learning*, pp. 3617–3626, 2017.

Wang, L., Ren, X., and Gu, Q. Precision matrix estimation in high dimensional gaussian graphical models with faster rates. In *Artificial Intelligence and Statistics*, pp. 177–185, 2016a.

Wang, L., Zhang, X., and Gu, Q. A unified computational and statistical framework for nonconvex low-rank matrix estimation. *arXiv preprint arXiv:1610.05275*, 2016b.

Wang, L., Zhang, X., and Gu, Q. A unified variance reduction-based framework for nonconvex low-rank matrix recovery. In *International Conference on Machine Learning*, pp. 3712–3721, 2017.

Wang, Z., Gu, Q., Ning, Y., and Liu, H. High dimensional em algorithm: Statistical optimization and asymptotic normality. In *Advances in Neural Information Processing Systems*, pp. 2512–2520, 2015.

Wytock, M. and Kolter, Z. Sparse gaussian conditional random fields: Algorithms, theory, and application to energy forecasting. In *International conference on machine learning*, pp. 1265–1273, 2013.

Xu, P. and Gu, Q. Semiparametric differential graph models. In *Advances in Neural Information Processing Systems*, pp. 1064–1072, 2016.

Xu, P., Tian, L., and Gu, Q. Communication-efficient distributed estimation and inference for transelliptical graphical models. *arXiv preprint arXiv:1612.09297*, 2016.

Xu, P., Ma, J., and Gu, Q. Speeding up latent variable gaussian graphical model estimation via nonconvex optimizations. *Neural Information Processing Systems*, 2017a.

Xu, P., Zhang, T., and Gu, Q. Efficient algorithm for sparse tensor-variate gaussian graphical models via gradient descent. In *Artificial Intelligence and Statistics*, pp. 923–932, 2017b.

Yang, E., Lozano, A., and Ravikumar, P. Elementary estimators for high-dimensional linear regression. In *International Conference on Machine Learning*, pp. 388–396, 2014a.

Yang, E., Lozano, A. C., and Ravikumar, P. K. Elementary estimators for graphical models. In *Advances in neural information processing systems*, pp. 2159–2167, 2014b.

Yin, J. and Li, H. A sparse conditional gaussian graphical model for analysis of genetical genomics data. *The annals of applied statistics*, 5(4):2630, 2011.

Yuan, X.-T. and Zhang, T. Partial gaussian graphical model estimation. *IEEE Transactions on Information Theory*, 60(3): 1673–1687, 2014.

Yuan, X.-T., Li, P., and Zhang, T. Gradient hard thresholding pursuit for sparsity-constrained optimization. *arXiv preprint arXiv:1311.5750*, 2013.

Zhang, L. and Kim, S. Learning gene networks under snp perturbations using eqtl datasets. *PLoS computational biology*, 10(2): e1003420, 2014.

Zhang, X., Wang, L., and Gu, Q. A unified framework for nonconvex low-rank plus sparse matrix recovery. In *International Conference on Artificial Intelligence and Statistics*, pp. 1097–1107, 2018.

Zhao, T., Wang, Z., and Liu, H. A nonconvex optimization framework for low rank matrix estimation. In *Advances in Neural Information Processing Systems*, pp. 559–567, 2015.

Zheng, Q. and Lafferty, J. A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. In *Advances in Neural Information Processing Systems*, pp. 109–117, 2015.

Zhou, S. Restricted eigenvalue conditions on subgaussian random matrices. *arXiv preprint arXiv:0912.4045*, 2009.

Zhu, R., Wang, L., Zhai, C., and Gu, Q. High-dimensional variance-reduced stochastic gradient expectation-maximization algorithm. In *International Conference on Machine Learning*, pp. 4180–4188, 2017.