Machine Learning (2019) 108:2061–2086
https://doi.org/10.1007/s10994-019-05810-5

A scalable sparse Cholesky based approach for learning high-dimensional covariance matrices in ordered data

Kshitij Khare[1] · Sang-Yun Oh[2] · Syed Rahman[1] · Bala Rajaratnam[3]

Received: 3 May 2018 / Revised: 28 April 2019 / Accepted: 8 May 2019 / Published online: 4 June 2019

Abstract

Covariance estimation for high-dimensional datasets is a fundamental problem in machine learning, and has numerous applications. In these high-dimensional settings the number of features or variables p is typically larger than the sample size n. A popular way of tackling this challenge is to induce sparsity in the covariance matrix, its inverse or a relevant transformation. In many applications, the data come with a natural ordering. In such settings, methods inducing sparsity in the Cholesky parameter of the inverse covariance matrix can be quite useful. Such methods are also better positioned to yield a positive definite estimate of the covariance matrix, a critical requirement for several downstream applications. Despite some important advances in this area, a principled approach to general sparse-Cholesky based covariance estimation with both statistical and algorithmic convergence safeguards has been elusive. In particular, the two popular likelihood based methods proposed in the literature either do not lead to a well-defined estimator in high-dimensional settings, or consider only a restrictive class of models. In this paper, we propose a principled and general method for sparse-Cholesky based covariance estimation that aims to overcome some of the shortcomings of current methods, but retains their respective strengths. We obtain a *jointly convex* formulation for our objective function, and show that it leads to rigorous convergence guarantees and well-defined estimators, even when $p > n$. Very importantly, the approach always leads to a positive definite and symmetric estimator of the covariance matrix. We establish both high-dimensional estimation and selection consistency, and also demonstrate excellent finite sample performance on simulated/real data.

Keywords Covariance estimation · High-dimensional data · Sparse Cholesky · Penalized likelihood

Editor: Jean-Phillippe Vert.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s10994-019-05810-5) contains supplementary material, which is available to authorized users.

✉ Kshitij Khare
kdkhare@stat.ufl.edu

Extended author information available on the last page of the article

1 Introduction

In modern day applications, datasets where the number of variables is much higher than the number of samples are more pervasive than they have ever been. One of the major challenges in this setting is to formulate models and develop learning procedures to understand the complex relationships and multivariate dependencies present in these datasets. The covariance matrix is perhaps the most fundamental object that quantifies associations between the variables in multivariate datasets. Hence, learning the covariance matrix in a principled way is crucial in high-dimensional problems and enables the detection of the most important relationships. For many applications, estimating the covariance matrix is an important first step of a deeper, more nuanced analysis. Hence, it is critical to develop machine learning methods which guarantee the positive definiteness of the resulting covariance estimate.

In particular, suppose we have i.i.d. observations $\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n$ from a p-variate normal distribution with mean vector $\mathbf{0}$ and covariance matrix Σ. Note that $\Sigma \in \mathbb{P}_p^+$, the space of positive definite matrices of dimension p. In many modern applications, the number of observations n is much fewer than the number of variables p. In such situations, parsimonious models which restrict Σ to a lower dimensional subspace of $\mathbb{P}_p^+$ are required for meaningful statistical estimation. Let $\Sigma^{-1} = T^t D^{-1} T$ denote the modified Cholesky decomposition of $\Omega = \Sigma^{-1}$. Here T is a lower triangular matrix with diagonal entries equal to 1 (we will refer to T as the Cholesky parameter), and D is a diagonal matrix with positive diagonal entries. The entries of T and D have a very natural interpretation. In particular, the (nonredundant) entries in each row of T are precisely the regression coefficients of the corresponding variable on the preceding variables. Similarly, each diagonal entry of D is the residual variance of the corresponding variable regressed on the preceding variables. Note that the discussion above implicitly assumes a given ordering of the variables in the dataset. The Cholesky factor of a positive definite matrix is not invariant to a reordering of the variables, and if we impose sparsity in the Cholesky factor estimate, the resulting (inverse) covariance estimate can in general be different for two different orderings. *For a variety of applications however, a natural ordering, such as time based or location based ordering, of the variables is available* (see Sects. 3.2, 3.4, for example) [1].

Owing to the interpretation of T and D discussed in the last paragraph, various authors in the literature have considered sparse estimation of T as a means of inducing parsimony in high-dimensional settings. Smith and Kohn (2002) develop a hierarchical Bayesian approach which allows for sparsity in the Cholesky parameter. Wu and Pourahmadi (2003) develop a non-parametric smoothing approach which provides a sparse estimate of the Cholesky parameter, with a banded sparsity pattern. Huang et al. (2006) introduce a penalized likelihood method to find a regularized estimate of Ω with a sparse Cholesky parameter. Rothman et al. (2010) develop penalized likelihood approaches to provide a sparse banded estimator for T^{-1} (which can be regarded as the Cholesky parameter for the covariance matrix Σ). Shojaie and Michailidis (2010) motivate sparsity in the Cholesky parameter T as a way of estimating the skeleton graph for a Gaussian Directed Acyclic Graph (DAG) model. In recent parallel work, Yu and Bien (2016) develop a penalized likelihood approach to obtain a tapered/banded estimator of T (with possibly different bandwiths for each row).

To the best of our knowledge, the methods in Huang et al. (2006) and Shojaie and Michailidis (2010) are the only (non-Bayesian) methods which induce a general or unrestricted sparsity pattern in the Cholesky parameter T of the inverse covariance matrix Ω. Both these

[1] See the end of the introduction and Sect. 2.6 for a discussion regarding Cholesky based methods for unordered data.

methods represent important advances in the area. Regardless, the proposed methodologies either do not lead to well-defined estimators in high-dimensional settings (when $n < p$) or consider a restrictive class of models for estimation purposes. We on the other hand parameterize in terms of the classical Cholesky parameter (as opposed to the modified Cholesky parameter), and construct an objective function which leads to well-defined covariance estimators in a general setting. We elaborate on this below.

Huang et al. (2006) obtain a sparse estimate of T by minimizing the objective function

$$Q_{Chol}(T, D) = tr\left(T^t D^{-1} T S\right) + \log|D| + \lambda \sum_{1 \leq i < j \leq p} |T_{ij}|. \tag{1.1}$$

with respect to T and D, where $S = \frac{1}{n}\sum_{i=1}^{n} \mathbf{Y}_i \mathbf{Y}_i^T$ is the sample covariance matrix (note that $\mathbf{Y}_i's$ have mean zero). In other words, their proposed estimator for the covariance matrix is $\hat{T}^{-1}\hat{D}(\hat{T}^{-1})^t$, where

$$(\hat{T}, \hat{D}) = \text{argmin}_{(T,D)} Q_{Chol}(T, D).$$

Let $\boldsymbol{\phi}^i := (T_{ij})_{j=1}^{i-1}$ and $S_{\cdot i} := (S_{ij})_{j=1}^{i-1}$ respectively denote the vector of lower triangular entries in the ith row of T and S for $i = 2, 3, \ldots, p$. Let S_i denote the $i \times i$ submatrix of S starting from the first row (column) to the ith row (column), for $i = 1, 2, \ldots, p$. It can be established after some simplification (see Huang et al. 2006) that

$$Q_{Chol}(T, D) = \left\{\frac{S_{11}}{D_{11}} + \log D_{11}\right\} + \sum_{i=2}^{p}\left\{\frac{(\boldsymbol{\phi}^i)^t S_{i-1}\boldsymbol{\phi}^i + 2(\boldsymbol{\phi}^i)^t S_{\cdot i} + S_{ii}}{D_{ii}} + \log D_{ii} + \lambda\|\boldsymbol{\phi}^i\|_1\right\},$$

where $\|\mathbf{x}\|_1$ denotes the sum of absolute values of the entries of a vector $\mathbf{x}$. It follows that minimizing $Q_{Chol}(T, D)$ with respect to L and D, is equivalent to minimizing

$$Q_{Chol,i}(\boldsymbol{\phi}^i, D_{ii}) = \frac{(\boldsymbol{\phi}^i)^t S_{i-1}\boldsymbol{\phi}_i + 2(\boldsymbol{\phi}^i)^t S_{\cdot i} + S_{ii}}{D_{ii}} + \log D_{ii} + \lambda\|\boldsymbol{\phi}^i\|_1 \tag{1.2}$$

with respect to $(\boldsymbol{\phi}^i, D_{ii})$ for $i = 2, 3, \ldots, p$, and setting $D_{11} = S_{11}$. Huang et al. (2006) propose minimizing each $Q_{Chol,i}(\boldsymbol{\phi}^i, D_{ii})$ using cyclic block coordinatewise minimization, where each iteration consists of minimizing $Q_{Chol,i}$ with respect to $\boldsymbol{\phi}^i$ (fixing D_{ii} at its current value), and then with respect to D_{ii} (fixing $\boldsymbol{\phi}^i$ at its current value). In particular, each row i can be minimized separately from the other rows, i.e., there are p separate minimizations. For ease of exposition, we will refer to the algorithm in Huang et al. (2006) as the Sparse Cholesky algorithm. However, this Sparse Cholesky approach based on minimizing $Q_{Chol,i}$ encounters a problem in high-dimensional settings (when $n < p$) for $i > n$. In the lemma below, we show that when $n < p$, the global minimum value of $Q_{Chol,i}$ is $-\infty$, and this minimum value is attained at $D_{ii} = 0$, which is undesirable as it leads to a singular estimate for the covariance matrix Σ.

Lemma 1.1 *The function $Q_{Chol,i}(\boldsymbol{\phi}^i, D_{ii})$ is not jointly convex or bi-convex for $1 \leq i \leq p$. Moreover, if $n < p$, then*

$$\inf_{\boldsymbol{\phi}_i \in \mathbb{R}^{i-1}, D_{ii} > 0} Q_{Chol,i}(\boldsymbol{\phi}_i, D_{ii}) = -\infty,$$

for $i > n$. Moreover, this minimum is attained if and only if $D_{ii} = 0$.

The proof of the lemma is provided in the supplemental document.

Let $\mathcal{T}_p$ denote the space of $p \times p$ lower triangular matrices with unit diagonal entries, and $\mathcal{D}_p$ denote the space of $p \times p$ diagonal matrices with positive diagonal entries. Since

$\{(\boldsymbol{\phi}^i, D_{ii})\}_{i=1}^p$ forms a disjoint partition of (T, D), it follows from Lemma 1.1 that if $n < p$, then

$$\inf_{T \in \mathcal{T}_p, D \in \mathcal{D}_p} Q_{Chol}(T, D) = \inf_{D_{11}>0} Q_{Chol,1}(D_{11}) + \sum_{i=2}^{p} \inf_{\boldsymbol{\phi}^i \in \mathbb{R}^{i-1}, D_{ii}>0} Q_{Chol,i}(\boldsymbol{\phi}^i, D_{ii}) = -\infty,$$

and the infimum is achieved only if one of the D_{ii}'s takes the value zero (*which is unacceptable as it corresponds to a singular estimate* Σ). Another issue with the approach in Huang et al. (2006) is that since the function $Q_{Chol,i}$ is not a jointly convex or even bi-convex in $(\boldsymbol{\phi}_i, D_{ii})$, existing results in the literature do not provide a theoretical guarantee that the sequence of iterates generated by the block coordinatewise minimization algorithm of Huang et al. (2006) (which alternates between minimizing with respect to $\boldsymbol{\phi}_i$ and D_{ii}) will converge. If the sequence of iterates does converge, it is not clear whether the limit is a global minimum or a local minimum. Of course, convergence to a local minimum is not desirable as the resulting estimate is not in general meaningful, and as described above, convergence to a global minimum will imply that the limit lies outside the range of acceptable parameter values.

We provide a simple illustration of this convergence issue in a setting with $p = 8$. We first generate a positive definite matrix $\Omega_0 = T_0^t D_0^{-1} T_0$ in the following manner. Sixty percent of the lower triangular entries of T_0 are randomly set to zero. The remaining 40% entries are chosen from a uniform distribution on [0.3, 0.7] and then assigned a positive/negative sign with probability 0.5. Now, a $p \times p$ diagonal matrix D_0 is generated with diagonal entries chosen uniformly from [2, 5]. We then set $n = p - 1$ and generate data from the multivariate normal distribution with mean $\mathbf{0}$ and inverse covariance matrix Ω_0. We initialize T and D to be I_8, and run the algorithm. After 4 interations, D_{77} jumps to 0 and stays there, as shown in Fig. 1. This leads to a degenerate covariance matrix estimate.

Note that the sparsity patterns in T can be associated with a directed acyclic graph $G = (V, E)$, where $V = \{1, 2, \ldots, p\}$ and $E = \{i \rightarrow j : i < j, T_{ij} \neq 0\}$. Based on this association, assuming that $D_{ii} = 1 \ \forall 1 \leq i \leq p$, Shojaie and Michailidis (2010) develop a 'graph selection' approach for obtaining a sparse estimate of T by minimizing the objective function

$$Q_{Chol}(T, I_p) = tr\left(T^t T S\right) + \lambda \sum_{1 \leq i < j \leq p} |T_{ij}|, \tag{1.3}$$

where I_p denotes the identity matrix of order p (an adaptive lasso version of the above objective function is also considered in Shojaie and Michailidis (2010)). It follows from (1.1) and (1.3) that from an optimization point of view, the approach in Shojaie and Michailidis (2010) is a special case of the approach in Huang et al. (2006). Note that fixing $D = I_p$ and only minimizing with respect to T significantly simplifies the optimization problem in Huang et al. (2006). Moreover, the resulting function is now convex in T with a quadratic term and an ℓ_1 penalty term. The authors in Shojaie and Michailidis (2010) provide a detailed evaluation of the asymptotic properties of their estimator in an appropriate high-dimensional setting (assuming that $D = I_p$). Owing to the interpretation of $\{T_{ij}\}_{j=1}^{i-1}$ as the regression coefficients of Y_i on $\{Y_j\}_{j=1}^{i-1}$, this can be regarded as a least squares approach with lasso penalty for sparsity selection in T. Hence, regardless of whether the true D_{ii}'s are all equal to one or not, this is a valid approach for sparsity selection/graph selection, which is precisely the goal in Shojaie and Michailidis (2010). Also, by substituting $D = I_p$ in (1.2) it follows that the approach in Shojaie and Michailidis (2010) is equivalent to performing p separate

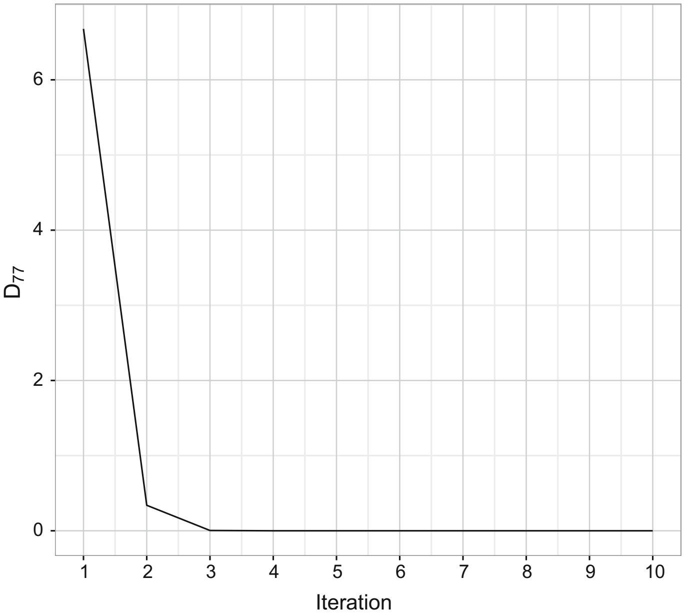


Fig. 1 Plot of the iterates for D_{77} for Sparse Cholesky in a setting with $p = 8$. It shows how the value jumps to 0 (and stays there)

lasso regressions to obtain a sparse estimate of T. For ease of exposition, we will refer to the algorithm developed in Shojaie and Michailidis (2010) as the Sparse Graph algorithm.

However, in many applications, the end goal is to obtain an accurate estimate of the covariance matrix, which requires estimating *both* T and D. We now point out some issues with making the assumption $D_{ii} = 1 \ \forall 1 \le i \le p$ when the goal is estimation of $\Sigma = T^{-1}D(T^t)^{-1}$. Note that if $cov(\mathbf{Y}) = \Sigma$, and if we define the vector of "latent variables" $\mathbf{Z} = T\mathbf{Y}$, then $cov(\mathbf{Z}) = D$. Hence, assuming that $D_{ii} = 1$ implies that the latent variables in $\mathbf{Z}$ have unit variance, NOT the variables in $\mathbf{Y}$. An assumption of unit variance for $\mathbf{Y}$ can be dealt with by scaling the observations in the data. *But scaling the data does not justify the assumption that the latent variables in* $\mathbf{Z}$ *have unit variances*. This is illustrated in the simulation example in Sect. 3.1. Also, it is not clear if an assumption of unit variances for the latent variables in $\mathbf{Z}$ can be dealt with by preprocessing the data another way. Hence, assuming that the diagonal entries of D are 1 can be restrictive, especially for estimation purposes. Indeed, the class of covariance matrices with $D_{ii} = 1 \ \forall 1 \le i \le p$ has probability zero under any probability measure on $\mathbb{P}^+$ which is mutually absolutely continuous with respect to Lebesgue measure.

One could propose an approach where estimates of T are obtained by minimizing (1.3), and estimates of D are obtained directly from the Cholesky decomposition of the sample covariance matrix S. However, this approach will not work when $n < p$ as S is a singular matrix in this case. *To summarize, the approach in* Shojaie and Michailidis (2010) *is useful for the purposes of sparsity selection/graph selection, but the assumption* $D_{ii} = 1 \ \forall 1 \le i \le p$ *can cause problems in obtaining accurate estimates of the entries of the covariance matrix* $\Sigma = T^{-1}D(T^t)^{-1}$.

In this paper, we develop an ℓ_1 penalized approach, called Convex Sparse Cholesky Selection (CSCS) which provides estimates for (T, D) (and consequently Σ and Ω) while

Table 1 Comparison of methods inducing sparsity in the Chloesky parameter of the inverse covariance matrix

Property	Method		
	Sparse Cholesky	Sparse Graph	CSCS
No constraints on sparsity pattern	+	+	+
No constraints on D (for estimation)	+		+
Convergence guarantee to acceptable global minimum when $n < p$		+	+
Asymptotic consistency ($n, p \to \infty$)		+	+

A "+" indicates that a specified method has the given property. A blank space indicates the absence of a property

inducing sparsity in T. This approach overcomes the drawbacks of the methods in Huang et al. (2006) and Shojaie and Michailidis (2010) while preserving the attractive properties of these approaches. The key is to reparameterize in terms of the *classical* Cholesky parameter for Ω, given by $\Omega = L^t L$. In particular, It can be shown that the CSCS objective function is *jointly convex* in the (nonredundant) entries of L, is bounded away from $-\infty$ even if $n < p$, and that the sparsity in the classical Cholesky parameter L is exactly reflected in the (modified) Cholesky parameter T. Furthermore, we provide a cyclic coordinatewise minimization algorithm to minimize this objective function, and show that the minimizer with respect to each coordinate is unique and can be evaluated in closed form. When $n < p$, our objective function is not strictly convex, and convergence of the cyclic coordinatewise minimization algorithm does not immediately follow from existing results in the literature. We show that recent results in Khare and Rajaratnam (2014) can be adapted in the current context to establish convergence to a global minimum for the cyclic coordinatewise minimization algorithm. We show that any global minimum lies in the acceptable range of parameter values, i.e., it leads to a positive definite estimate of the covariance matrix. We also establish high-dimensional asymptotic graph selection and estimation consistency of the resulting estimator under standard regularity assumptions. As explained in Sect. 4, proving consistency in the current setting is non-trivially different than the consistency arguments considered in Khare et al. (2015), Peng et al. (2009), Shojaie and Michailidis (2010) because the diagonal entries of L are not assumed to be known in this paper.

A comparison of the relevant properties of the estimators developed in Huang et al. (2006) (Sparse Cholesky), Shojaie and Michailidis (2010) (Sparse Graph) and this paper (CSCS) is provided in Table 1. For ease of exposition, we refer to the algorithm in Huang et al. (2006) as the Sparse Cholesky algorithm, and the one in Shojaie and Michailidis (2010) as the Sparse Graph algorithm. Through experiments based on simulated and real datasets, we demonstrate that CSCS can have significantly better graph selection as well as estimation performance than Sparse Cholesky when $n < p$. These experiments also demonstrate that CSCS can improve on the graph selection performance of Sparse Graph, and can lead to significant improvements in covariance estimation performance.

As mentioned previously, our approach as well as that of Huang et al. (2006), Shojaie and Michailidis (2010), Yu and Bien (2016) assumes the existence of a domain-specific ordering of the the variables, which is available in a large variety of applications (see these papers and Sects. 3.2 and 3.4 and for examples). However, in other applications, such an ordering may not be apparent. In such settings, a possible solution is to consider the ordering as an additional parameter in the objective function and optimize over it. This problem has been studied in the recent literature, see Wagaman and Levina (2009), van de Geer and

Buhlmann (2013), Aragam and Zhou (2015), Aragam et al. (2016); Zheng et al. (2017)) and the references therein. While such an approach is necessary in situations when a natural ordering does not exist, the corresponding optimization problem also becomes significantly complicated, and in general the resulting estimate can only be shown to be a local minimum of the corresponding objective functions. The contribution of our paper is to show that when a domain-specific ordering is available, one can develop faster, parallelizable algorithms, and stronger algorithmic convergence (convergence to a global minimum, even when $n < p$) and statistical consistency properties can be established. See Sect. 2.6 for a detailed discussion.

The remainder of the paper is organized as follows. Section 2 introduces the CSCS method, and then studies relevant properties such as convergence, computational complexity. In Sect. 2.5, we compare and contrast the CSCS method (which induces sparsity in T) with penalized methods which induce sparsity in Ω. Section 3 illustrates the performance of the CSCS method on simulated and real data. Section 4 establishes finite sample (non-asymptotic) bounds for the accuracy of the CSCS estimator (Theorem 4.1). These bounds are then used to establish high-dimensional asymptotic consistency (both estimation and model selection) of the CSCS method (Theorem 4.2). The supplementary document contains proofs of some of the results in the paper.

2 A convex approach for sparse Cholesky estimation

As pointed out in Lemma 1.1, if $n < p$, the infimum of $Q_{Chol,n+1}(\boldsymbol{\phi}_{n+1}, D_{n+1,n+1})$ over the range of acceptable values of $(\boldsymbol{\phi}_{n+1}, D_{n+1,n+1})$ is $-\infty$. However, the infimum is attained only if $D_{n+1,n+1} = 0$, which is outside the range of acceptable values of $D_{n+1,n+1}$. Also, since $Q_{Chol}(T, D)$ is not jointly convex in (L, D), their are no convergence guarantees for the block coordinate-wise minimization algorithm proposed in Huang et al. (2006). Given the attractive properties of convex functions and the rich theory for convex optimization, a natural approach to address these issues is to develop a convex objective function for this problem. Such an approach will also potentially lead to a deeper theoretical analysis of the properties of the solution and corresponding algorithm. The objective function $Q_{Chol}(T, I_p)$ used in Shojaie and Michailidis (2010) is jointly convex in T, but we want to avoid any restrictive constraints on D.

2.1 The CSCS objective function

We will now show that all the goals mentioned above can be achieved by reparametrizing in terms of the classical Cholesky parameter. Recall that the classical Cholesky decomposition of Ω is given by $\Omega = L^t L$, where L (which we will refer to as the classical Cholesky parameter) is a lower triangular matrix with positive diagonal entries. We introduce the following objective function

$$Q_{CSCS}(L) = tr\left(L^t L S\right) - 2\log|L| + \lambda \sum_{1\le j<i\le p} |L_{ij}|. \tag{2.1}$$

The first two terms in Q_{CSCS} correspond to the (negative) Gaussian log-likelihood, and the last term is an ℓ_1-penalty on the off-diagonal entries of L. To see the connection (and contrast) between Q_{CSCS} and Q_{Chol} (used for Sparse Cholesky), note that

$$L_{ij} = T_{ij}/\sqrt{D_{jj}} \text{ for every } i \le j. \tag{2.2}$$

Hence, $L_{ij} = 0$ if and only if $T_{ij} = 0$, i.e., *sparsity in T is equivalent to sparsity in L*. After reparametrizing Q_{Chol} in terms of L (as opposed to (T, D)) and some simple manipulations, we obtain the following objective function.

$$Q_{Chol}(L) = tr\left(LL^t S\right) - 2\log|L| + \lambda \sum_{1 \leq j < i \leq p} |L_{ij}| L_{jj}. \tag{2.3}$$

Note that the first term in (2.3) is a quadratic form in the entries of L, and hence is jointly convex in the entries of L. Since L is a lower triangular matrix, it follows that $-\log|L| = \sum_{i=1}^{p} -\log L_{ii}$, and hence the second term in (2.3) is also jointly convex in entries of L. However, terms of the form $|L_{ij}|L_{jj}$ are not jointly convex, and hence the penalty term in (2.3) is not jointly convex either. On the other hand, the term $\lambda \sum_{1\leq j<i\leq p} |L_{ij}|$ in (2.1) is jointly convex in the entries of L. The following lemma immediately follows from (2.2) and the discussion above.

Lemma 2.1 (Joint convexity) *$Q_{CSCS}(L)$ is jointly convex in the entries of L. Also, a global minimizer of Q_{CSCS} will be sparse in L (and hence sparse in T).*

Let $\boldsymbol{\eta}^i = (L_{ij})_{j=1}^{i}$ denote the vector of lower triangular and diagonal entries in the ith row of L for $1 \leq i \leq p$. Recall that S_i denotes the $i \times i$ sub matrix of S starting from the first row (column) to the ith row (column). Let $L_{i\cdot}$ denote the ith row of L, for $1 \leq i \leq p$. It follows from (2.1), the lower triangular nature of L, and the definition of $\boldsymbol{\eta}^i$ that

$$\begin{aligned} Q_{CSCS}(L) &= tr\left(LSL^t\right) - 2\sum_{i=1}^{p} \log L_{ii} + \lambda \sum_{1\leq j<i\leq p} |L_{ij}| \\ &= \sum_{i=1}^{p} L_{i\cdot} S L_{i\cdot}^t - 2\sum_{i=1}^{p} \log \eta_i^i + \lambda \sum_{i=2}^{p}\sum_{j=1}^{i-1} |\eta_j^i| \\ &= \sum_{i=1}^{p} (\boldsymbol{\eta}^i)^T S_i \boldsymbol{\eta}^i - 2\sum_{i=1}^{p} \log \eta_i^i + \lambda \sum_{i=2}^{p}\sum_{j=1}^{i-1} |\eta_j^i| \\ &= \sum_{i=1}^{p} Q_{CSCS,i}(\boldsymbol{\eta}^i), \end{aligned} \tag{2.4}$$

where

$$Q_{CSCS,i}(\boldsymbol{\eta}^i) = (\boldsymbol{\eta}^i)^T S_i \boldsymbol{\eta}^i - 2\log \eta_i^i + \lambda \sum_{j=1}^{i-1} |\eta_j^i| \tag{2.5}$$

for $2 \leq i \leq p$, and

$$Q_{CSCS,1}(L_{11}) = L_{11}^2 S_{11} - 2\log L_{11}. \tag{2.6}$$

Equation (2.4) demonstrates that Q_{CSCS} decomposes into a sum of p functions, and the functions in the sum depend on disjoint sets of parameters. We will use this decomposition crucially for establishing attractive theoretical and computational properties such as high-dimensional consistency and ability to parallelize.

Our next goal is to establish that the function $Q_{CSCS}(L)$ is uniformly bounded below over $\mathcal{L}_p$, the space of $p \times p$ lower triangular matrices with positive diagonal entries. We will assume that the diagonal entries of the sample covariance matrix S are strictly positive. This basically means that none of the underlying p marginal distributions are degenerate.

We now state a lemma from Khare and Rajaratnam (2014) which will play a crucial role in this exercise.

Lemma 2.2 (Khare and Rajaratnam 2014) *Let A be a $k \times k$ positive semi-definite matrix with $A_{kk} > 0$, and λ be a positive constant. Consider the function*

$$h(\mathbf{x}) = -\log x_k + \mathbf{x}^T A\mathbf{x} + \lambda \sum_{j=1}^{k-1} |x_j|$$

defined on $\mathbb{R}^{k-1} \times \mathbb{R}_+$. Then, there exist positive constants a_1 and a_2 (depending only on λ and A), such that

$$h(\mathbf{x}) \geq a_1 x_k - a_2$$

for every $\mathbf{x} \in \mathbb{R}^{k-1} \times \mathbb{R}_+$.

Using (2.5), (2.6) along with the facts that S_i is positive semi-definite and $S_{ii} > 0$, it follows from Lemma 2.2 that for every $1 \leq i \leq p$, there exist positive constants a_i and b_i such that

$$\begin{aligned} Q_{CSCS,i}(\boldsymbol{\eta}^i) &= (\boldsymbol{\eta}^i)^T S_i \boldsymbol{\eta}^i - 2\log \eta^i_i + \frac{\lambda}{2}\sum_{j=1}^{i-1} |\eta^i_j| + \frac{\lambda}{2}\sum_{j=1}^{i-1} |\eta^i_j| \\ &\geq a_i \eta^i_i - b_i + \frac{\lambda}{2}\sum_{j=1}^{i-1} |\eta^i_j| \end{aligned} \tag{2.7}$$

for every $\boldsymbol{\eta}^i \in \mathbb{R}^{i-1} \times \mathbb{R}_+$. The following lemma now follows immediately from (2.4), (2.7) and the fact that $\{\boldsymbol{\eta}^i\}_{i=1}^p$ forms a disjoint partition of L.

Lemma 2.3 *For every n and p,*

$$\inf_{L \in \mathcal{L}_p} Q_{CSCS}(L) = \sum_{i=1}^p \inf_{\eta^i \in \mathcal{R}^{i-1} \times \mathcal{R}_+} Q_{CSCS,i}(\boldsymbol{\eta}^i) \geq -\sum_{i=1}^p b_i > -\infty,$$

and $Q_{CSCS}(L) \to \infty$ as $|\eta^i_j| = |L_{ij}| \to \infty$ for any $j < i$, or as $\eta^i_i = L_{ii} \to 0$. Hence, any global minimum of $Q_{CSCS,i}$ has a strictly positive value for $\eta^i_i = L_{ii}$, and hence **any global minimum of Q_{CSCS} over the open set $\mathcal{L}_p$ lies in $\mathcal{L}_p$.**

2.2 A minimization algorithm for Q_{CSCS}

We now provide an algorithm to minimize the convex objective function $Q_{CSCS}(L)$. Since $\{\boldsymbol{\eta}^i\}_{i=1}^p$ form a disjoint partition of the (nonredundant) parameters in L, it follows that optimizing $Q_{CSCS}(L)$ is equivalent to separately optimizing $Q_{CSCS,i}(\boldsymbol{\eta}^i)$ for $1 \leq i \leq p$.

Consider, similar to Lemma 2.2, a generic function of the form

$$h_{k,A,\lambda}(\mathbf{x}) = -2\log x_k + \mathbf{x}^T A\mathbf{x} + \lambda \sum_{i=1}^{k-1} |x_j| \tag{2.8}$$

from $\mathbb{R}^{k-1} \times \mathbb{R}_+$ to $\mathbb{R}$. Here k is a positive integer, $\lambda > 0$, and A is a positive semi-definite matrix with positive diagonal entries. It follows from (2.5) and (2.6) that $Q_{CSCS,i}(\boldsymbol{\eta}^i) = h_{i,S_i,\lambda}(\boldsymbol{\eta}^i)$ for every $1 \leq i \leq p$. It therefore suffices to develop an algorithm to minimize a function of the form $h_{k,A,\lambda}$ as specified in (2.8). Note that without the logarithmic term and the restriction that $x_k > 0$, the optimization problem for $h_{k,A,\lambda}$ would have been equivalent to the lasso optimization problem for which several approaches have been developed in the literature, such as the shooting algorithm in Fu (1998), or the pathwise coordinate optimization approach in Friedman et al. (2008), for example. However, these algorithms do not apply in the current situation due to the presence of the logarithmic term and the condition $x_k > 0$.

We will now derive a cyclic coordinatewise minimization algorithm for $h_{k,A,\lambda}$. For every $1 \leq j \leq k$, define the function $M_j : \mathbb{R}^{k-1} \times \mathbb{R}_+ \to \mathbb{R}^{k-1} \times \mathbb{R}_+$ by

$$M_j(\mathbf{x}) = \inf_{\mathbf{y} \in \mathbb{R}^{k-1} \times \mathbb{R}_+ : y_l = x_l \forall l \neq j} h_{k,A,\lambda}(\mathbf{x}). \tag{2.9}$$

The following lemma (proof provided in the supplemental document) shows that the functions $\{M_j\}_{j=1}^k$ can be computed in closed form.

Lemma 2.4 *The function $M_j(\mathbf{x})$ defined in (2.9) can be computed in closed form. In particular,*

$$\left(M_j(\mathbf{x})\right)_j = \frac{S_\lambda\left(-2\sum_{l \neq j} A_{lj} x_l\right)}{2A_{jj}} \tag{2.10}$$

for $1 \leq j \leq k-1$, and

$$(M_k(\mathbf{x}))_k = \frac{-\sum_{l \neq k} A_{lk} x_l + \sqrt{\left(\sum_{l \neq k} A_{lk} x_l\right)^2 + 4A_{kk}}}{2A_{kk}}. \tag{2.11}$$

Here S_λ is the soft-thresholding operator given by $S_\lambda(x) = sign(x)(|x| - \lambda)_+$. Lemma 2.4 provides the required ingredients to construct a cyclic coordinatewise minimization algorithm to minimize $h_{k,A,\lambda}$ (see Algorithm 1). Now, to minimize $Q_{CSCS}(L)$, we use Algorithm 1 to minimize $Q_{CSCS,i}(\boldsymbol{\eta}^i)$ for every $1 \leq i \leq p$, and combine the outputs to obtain the a matrix on $\mathcal{L}_p$ (see Algorithm 2). We refer to Algorithm 2 as the CSCS algorithm.

Note that although the function $Q_{CSCS,i}$ is jointly convex in the entries of $\boldsymbol{\eta}^i$, it is not in general strictly convex if $n < i$, and does not necessarily have a unique global minimum. Hence, it is not immediately clear if existing results in the literature imply the convergence of Algorithm 2 to a global minimum of Q_{CSCS}. The next theorem invokes results in Khare and Rajaratnam (2014) to establish convergence of Algorithm 2.

Theorem 2.1 *If $S_{ii} > 0$ for every $1 \leq i \leq p$, then Algorithm 2 converges to a global minimum of Q_{CSCS}.*

The proof of the above theorem is provided in the supplemental document.

2.3 Selection of tuning parameter

The tuning parameter λ can be selected using a "BIC"-like measure, defined as follows:

$$BIC(\lambda) = n\text{tr}(S\hat{\Omega}) - n \log|\hat{\Omega}| + \log n * E$$

where E denotes the number of non-zero entries in $\hat{L}$, n is the sample size, S is the sample covariance and $\hat{\Omega} = \hat{L}^t \hat{L}$. The value of λ minimizing the function $BIC(\lambda)$ can be chosen.

In Huang et al. (2006) and Shojaie and Michailidis (2010) the authors respectively propose tuning parameter choices based on cross-validation and scaled normal quantiles. These procedures are described briefly in Sects. 3.2 and 3.1 respectively.

Algorithm 1. (Cyclic coordinatewise algorithm for $h_{k,A,\lambda}$)

```
Input: $k$, $A$ and $\lambda$
Input: Fix maximum number of iterations: $r_{max}$
Input: Fix initial estimate: $\hat{\mathbf{x}}^{(0)}$
Input: Fix convergence threshold: $\epsilon$
Set $r \leftarrow 1$
converged = FALSE
Set $\hat{\mathbf{x}}^{\text{current}} \leftarrow \hat{\mathbf{x}}^{(0)}$
Repeat
$\hat{\mathbf{x}}^{\text{old}} \leftarrow \hat{\mathbf{x}}^{\text{current}}$
    For $j = 1, 2, \cdots, k-1$
        $\hat{x}_j^{\text{current}} \leftarrow (M_j(\mathbf{x}^{\text{current}}))_j$
    $\hat{x}_k^{\text{current}} \leftarrow (M_k(\mathbf{x}^{\text{current}}))_k$
$\hat{\mathbf{x}}^{(r)} \leftarrow \hat{\mathbf{x}}^{\text{current}}$
## Convergence checking
    If $\|\hat{\mathbf{x}}^{\text{current}} - \hat{\mathbf{x}}^{\text{old}}\|_\infty < \epsilon$
        converged = TRUE
    Else
        $r \leftarrow r + 1$
Until converged = TRUE or $r > r_{\max}$
Return final estimate: $\hat{\mathbf{x}}^{(r)}$
```

Algorithm 2. (CSCS algorithm: minimization algorithm for Q_{CSCS})

```
Input: Data $\mathbf{Y}_1, \mathbf{Y}_2, \cdots, \mathbf{Y}_n$ and $\lambda$
Input: Fix maximum number of iterations: $r_{max}$
Input: Fix initial estimate: $\hat{L}^{(0)}$
Input: Fix convergence threshold: $\epsilon$
    For $i = 1, 2, \cdots, p$
        $(\hat{\boldsymbol{\eta}^i})^{(0)} \leftarrow i^{th}$ row of $\hat{L}^{(0)}$ (up to the diagonal)
        Set $\hat{\boldsymbol{\eta}^i}$ to be minimizer of $Q_{CSCS,i}$ obtained by using Algorithm 1
        with $k = i$, $A = S_i$, $\lambda$, $r_{max}$, $\hat{\mathbf{x}}^{(0)} = (\hat{\boldsymbol{\eta}^i})^{(0)}$, $\epsilon$
Construct $\hat{L} \in \mathcal{L}_p$ by setting its $i^{th}$ row (up to the diagonal) as $\hat{\boldsymbol{\eta}^i}$
Return final estimate: $\hat{L}$
```

2.4 Computational complexity of the CSCS algorithm

We now proceed to evaluate the computational complexity of the CSCS algorithm. Note that the CSCS algorithm (Algorithm 2) involves p separate minimizations. In modern computing environments, all of these minimizations can be run in parallel. In a parallelizable setting, we define the computational complexity as a maximum number of computations among all processes running in parallel. Hence, keeping the modern computing environment in mind, the next lemma provides the computational complexity of CSCS in both parallel and sequential settings. The proof of this lemma is provided in the supplemental document.

Lemma 2.5 *(a) If all the p minimizations are run in parallel, the computational complexity per iteration for Algorithm 2 is* $\min(O(np), O(p^2))$.

(b) If all the p minimizations are run sequentially, the computational complexity per iteration for Algorithm 2 is $\min\left(O\left(n \sum_{i=1}^{p} i\right), O\left(\sum_{i=1}^{n} i^2\right)\right) = \min(O(np^2, p^3))$.

2.5 Comparison and connections with other methods—penalized sparse partial correlation methods

In this section we compare and contrast the CSCS method (which induces sparsity in the Cholesky factor of Ω) with sparse partial correlation methods, i.e., penalized methods which induce sparsity in the inverse covariance matrix Ω itself. The entries in the ith row of Ω (appropriately scaled) can be interpreted as regression coefficients of the ith variable against *all* other variables. Recall that the (non-redundant) entries in the ith row of T, on the other hand, are the regression coefficients of the ith variable against only the *preceding* variables. A natural question to ask is whether there is any connection between models which introduce sparsity in the Cholesky factor of Ω and models which induce sparsity in Ω itself. In general, the sparsity pattern in the Cholesky factor T of a positive definite matrix Ω is not the same as the sparsity pattern in Ω itself. Note that a given pattern of zeros in the lower triangle a $p \times p$ matrix uniquely corresponds to a graph with vertices $\{1, 2, \ldots, p\}$, where two vertices do not share an edge whenever the corresponding entry is included in the pattern of zeros. It is known that the sparsity pattern in Ω is exactly the same as its Cholesky factor if and only if the corresponding graph is chordal (decomposable) and the vertices are ordered based on a perfect vertex elimination scheme (see Paulsen et al. 1989).

We now summarize the relevant details of penalized methods which induce sparsity in Ω. Such methods can be roughly divided into four categories: penalized likelihood methods such as GLASSO (Banerjee et al. 2008; Friedman et al. 2008), penalized pseudo-likelihood methods such as CONCORD (Khare et al. 2015), SPACE (Peng et al. 2009) and SYMLASSO (Friedman et al. 2010), Dantzig-selector type methods such as in Yuan (2010), Cai et al. (2011), Liu and Luo (2015), and regularized score matching based methods such as in Zhang and Zou (2014), Lin et al. (2016). The GLASSO objective function is comprised of a log Gaussian likelihood term and an ℓ_1-penalty term for entries of Ω. Friedman et al. (2008) present an algorithm for minimizing this objective function with has computational complexity of $O(p^3)$ per iteration.[2] Pseudo-likelihood based objective functions used in CONCORD, SPACE and SYMLASSO are comprised of a log pseudo-likelihood trem which is based on the regression based interpretation of the entries of Ω, and an ℓ_1-penalty term for entries of Ω. These objective functions are typically minimized using cyclic coordinate-wise minimization with a computational complexity of $\min(O(np^2), O(p^3))$.[3] Owing to the regression based interpretation of the pseudo-likelihood, the minimization is done over all symmetric matrices with positive diagonal entries (as opposed to GLASSO, where the minimization is done over the set of positive definite matrices), and hence the minimizer is not guaranteed to be positive definite. In many applications, the main goal is selection of the sparsity pattern (network), and this does not pose a problem. In fact, getting rid of the positive definiteness constraint is helpful in improving the performance of such methods (as compared to GLASSO) in high-dimensional settings including increased robustness to heavier tailed data (see Khare et al. 2015). The CONCORD algorithm, unlike SPACE and SYMLASSO, provides crucial theoretical guarantees of convergence to a global minimum of the respective objective function (while preserving all the other attractive properties of SPACE and SYMLASSO).

[2] In recent years, several adaptations/alternatives to this algorithm have been proposed in order to improve its speed (see Hsieh et al. 2011; Mazumder and Hastie 2012 for instance). However, for these methods to provide substantial improvements over the graphical lasso, certain assumptions are required on the number and size of the connected components of the graph implied by the zeros in the minimizer.

[3] Recently, a much faster proximal gradient based optimization method for the CONCORD objective function has been developed in Oh et al. (2014).

There is, in fact, an interesting parallel between CONCORD and CSCS. The CONCORD objective function (scaled by $\frac{2}{n}$) is given by

$$Q_{\text{con}}(\Omega) = -\sum_{i=1}^{p} 2\log\omega_{ii} + tr\left(\Omega^t \Omega S\right) + \lambda \sum_{1\le j<i\le p} |\omega_{ij}|.$$

On the other hand, it follows from (2.1) that the CSCS objective function can be written as

$$Q_{CSCS}(L) = -\sum_{i=1}^{p} 2\log L_{ii} + tr\left(L^t L S\right) + \lambda \sum_{1\le j<i\le p} |L_{ij}|.$$

Hence, from a purely mathematical point of view, CONCORD and CSCS are both optimizing over the same objective function. The difference is that CONCORD optimizes the function over the set of symmetric matrices with positive diagonal entries, whereas CSCS optimizes the function over the set of lower triangular matrices with positive diagonal entries. Despite this very close connection between the objective functions for CONCORD and CSCS, the difference in the range of optimization leads to some critically important differences between the respective optimization algorithms and estimators.

(a) (Computational Complexity) The p minimizations in the CSCS algorithm can be run in parallel, giving it a distinct computational advantage over the CONCORD algorithm (Khare et al. 2015) (which does not share this property). Even in the worst case, when all the p minimizations for CSCS are implemented sequentially, the computational complexity is the same as CONCORD (by Lemma 2.5).

(b) (Positive definiteness of resulting estimator of Ω) As discussed above, the CONCORD estimator (and other pseudo-likelihood based estimators) for Ω is not guaranteed to be positive definite. However, the estimator for Ω constructed by taking the CSCS estimator and multiplying it by its transpose, is always positive definite.

We close this section by observing that as discussed above, the regression based interpretation for the entries of Ω leads to a different objective function than the log Gaussian likelihood for Ω. However, it can be easily shown that the objective function based on the regression based interpretation for the entries of the Cholesky parameter T (or equivalently L) exactly corresponds to the log Gaussian likelihood for T.

2.6 Comparison and connections with other methods—methods for DAG selection

As mentioned in the introduction, while a natural ordering of variables is available in several applications, in many applications such an ordering may not be apparent. In such a setting, the ordering too needs to be selected using the data. Methods to tackle this problem have been studied in recent literature (see van de Geer and Buhlmann 2013; Aragam and Zhou 2015; Aragam et al. 2016 and the references therein, for a review). The method/algorithm most relevant in the context of CSCS is the CCDr-ℓ_1 algorithm in Aragam and Zhou (2015).

Recall that element T_{kj} of lower triangular matrix T corresponds to coefficient for regularized regression of X_j onto preceding variables X_k, $1 \le k < j \le p$. Given an ordering of the variables, it is clear that coefficient for regressing X_k onto X_j is not meaningful. As such, setting $T_{jk} = 0$ is immediate in this context, and results in elements of upper triangular matrix to be set to zero.

However, when variable ordering is not specified, it is not clear if $T_{jk} = 0$ or $T_{kj} = 0$, and needs to be inferred. CCDr suggests heuristic-based selection between $\{T_{jk}, T_{kj}\}$ by imposing that an appropriate choice should (1) result in a larger increase of a given scoring function (2) assuming the choice does not create a cycle in resulting DAG. Hence, when a variable ordering is not given, the matrix of regression coefficients selected by CCDr is not necessarily lower triangular. However, due to the heuristic constraint that cycles are not created, it can be row and column permuted to be put into a lower triangular matrix T. Hence, CCDr can be thought of as performing a heuristic-based constrained optimization (based on the DAG condition) of the same objective function as CSCS when ℓ_1 regularization is choice of penalty.

Despite this very close connection between the objective functions for CCDr-ℓ_1 and CSCS the difference in the range of optimization leads to some important qualitative differences between the respective optimization algorithms and estimators.

(a) (Convexity) The minimization problem in the CCDr-ℓ_1 setting is not a convex problem, as the set of all matrices with sparsity pattern consistent with a DAG is not convex. In such settings, general results for coordinatewise minimization in the literature only guarantee convergence of the sequence of iterates to a local minimum of the objective function. On the other hand, for CSCS, the optimization problem is a convex problem (though not strictly convex when $n < p$), and we are able to leverage this convexity to establish the convergence of the sequence of iterates to a global minimum of the objective function.

(b) (Computational Complexity) The parallelizability of the p minimizations in the CSCS algorithm, gives it a distinct computational advantage over the CCDr-ℓ_1 algorithm (which is not parallelizable, as one needs to minimize over T_{kj} and T_{jk} simultaneously).

(c) (Asymptotic properties) To the best of our knowledge, high-dimensional consistency results in Aragam et al. (2016) (a follow-up work to Aragam and Zhou (2015)) are available only for a restricted version of the CCDr-ℓ_1 algorithm, where an estimate of the DAG is obtained by minimizing the objective function

$$tr\left(\tilde{T}^t\tilde{T}S\right) + \lambda \sum_{1\leq i<j\leq p} |\tilde{T}_{ij}| \tag{2.12}$$

as $\tilde{T}$ varies over the space of matrices with *unit diagonal entries* which can be converted to a lower triangular matrices by (a row and column) reordering. As in the case of the Sparse Graph algorithm in Shojaie and Michailidis (2010), an estimate of T obtained by minimizing the function in (2.12) will give the sparsity structure to recover the DAG, and also estimates of the appropriate regression coefficients of each variable on its predecessors, but does not provide estimates of the conditional variances. These estimates are required to construct the covariance matrix, which is the goal in many applications. In other words, estimation consistency for the entire covariance matrix produced by the general CCDr-ℓ_1 in Aragam and Zhou (2015) (which does provide estimates for the conditional variances) are not available. On the other hand, Theorem 4.1 leverages the parallelizability of CSCS to establish estimation consistency for the entire covariance matrix produced by the CSCS algorithm.

3 Experiments

3.1 Simulated data: graph selection and estimation

In this section, we perform a simulation study to compare the graph/model selection and estimation performance of CSCS, Sparse Cholesky and Sparse Graph.

Graph selection comparison

As stated in the introduction, the Sparse Graph algorithm is sensible and useful for model selection whether or not the true conditional variances D_{ii} are all equal to one or not (although as we demonstrate later in this section, the assumption can lead to inaccuracies in estimation when the true D_{ii}'s are not all equal to one). The goal of this experiment is to investigate whether CSCS can lead to improved graph selection performance as compared to Sparse Graph and Sparse Cholesky in high- dimensional settings.

For this purpose, we consider eight different settings with $p \in \{1000, 2000\}$ and $n \in \{p/8, p/4, p/2, 3p/2\}$. In particular, for each $p \in \{1000, 2000\}$, a $p \times p$ lower triangular matrix T_0 is generated as follows. We randomly choose 98% of the lower triangular entries, and set them to zero. The remaining 2% entries are chosen randomly from a uniform distribution on $[0.3, 0.7]$ and then assigned a positive/negative sign with probability 0.5. Now, a $p \times p$ diagonal matrix D_0 is generated with diagonal entries chosen uniformly from $[2, 5]$. For each sample size $n \in \{p/8, p/4, p/2, 3p/2\}$, 100 datasets, each having i.i.d. multivariate normal distribution with mean zero and inverse covariance matrix $\Omega_0 = T_0^t D_0^{-1} T_0$, are generated.

The model selection performance of the three algorithms, CSCS, Sparse Cholesky, Sparse Graph, is then compared using receiver operating characteristic (ROC) curves. These curves compare true positive rates (TPR) and false positive rates (FPR), and are obtained by varying the penalty parameter over roughly 40 possible values. In applications, FPR is typically controlled to be sufficiently small, and therefore we focus on comparing portion of ROC curves for which FPR is less than 0.15. In order to compare the ROC curves, Area-under-the-curve (AUC) is used (see Fawcett 2006; Friedman et al. 2010).

Tables 2 and 3show the mean and standard deviation (over 100 simulations) for the AUCs for CSCS, Sparse Cholesky and Sparse Graph when $p \in \{1000, 2000\}$ and $n \in \{p/8, p/4, p/2, 3p/2\}$. It is clear that CSCS has a better model selection performance as compared to Sparse Cholesky and Sparse Graph in all cases.

(a) As expected Sparse Cholesky performs significantly worse than other methods when $n < p$, but its absolute and relative performance improves with increasing sample size, especially when $n > p$.

(b) The tables also show that CSCS performs better model selection than Sparse Graph, although the difference in AUC is not as pronounced as with Sparse Cholesky. In should be noted that CSCS obtains higher AUC than Sparse Graph for each of the 800 datasets (100 each for $p \in \{1000, 2000\}$ and $n \in \{p/8, p/4, p/2, 3p/2\}$), and for all the eight settings displayed in Tables 2 and 3

$$\text{Mean AUC - } 3 \times \text{Std. Dev. for CSCS} > \text{Mean AUC + } 3 \times \text{Std. Dev. for Sparse Graph.}$$

We also note that the variability is much lower for CSCS than the other methods.

It is worth mentioning that for each of the 800 datasets, the data was centered to zero mean and scaled to unit variance before running each method. Firstly, this illustrates that scaling the data does not justify assuming that the latent variable conditional variances $\{D_{ii}\}_{i=1}^p$ are identically 1, evidenced by CSCS performing consistently better model selection as compared to Sparse Graph. Secondly, we observed that the three algorithms typically run much faster when the data is scaled; therefore, data standardization was performed in the interest of time given the extensive nature of our simulation study. Note that premultiplying a multivariate normal vector by a diagonal matrix does not affect the sparsity pattern of the Cholesky factor of the inverse covariance matrix. This approach helps accommodate the setting where the variables have different marginal variances.

Table 2 Mean and Standard Deviation of area-under-the-curve (AUC) for 100 simulations for $p = 1000$

Solver	**n = 125**		**n = 250**		**n = 500**		**n = 1500**	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
Sparse Cholesky	0.012796	0.000045	0.018461	0.000108	0.078832	0.000122	0.127916	0.000027
Sparse Graph	0.113955	0.000200	0.129142	0.000048	0.135271	0.000066	0.138633	0.000026
CSCS	**0.118440**	0.000111	**0.133958**	0.000036	**0.138492**	0.000023	**0.139891**	0.000001

Each simulation yields a ROC curve from which the AUC is computed for FPR in the interval [0.01, 0.15]. The best results for each sample size are given in bold. CSCS achieves the highest AUC in each column

Table 3 Mean and Standard Deviation of area-under-the-curve (AUC) for 100 simulations for $p = 2000$

Solver	**n = 250**		**n = 500**		**n = 1000**		**n = 3000**	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
Sparse Cholesky	0.015131	0.000050	0.032391	0.000105	0.124284	0.000058	0.142678	0.000012
Sparse Graph	0.141957	0.000044	0.146362	0.000009	0.147984	0.000005	0.148742	0.000001
CSCS	**0.144686**	0.000019	**0.147839**	0.000004	**0.148722**	0.000002	**0.148904**	0.000001

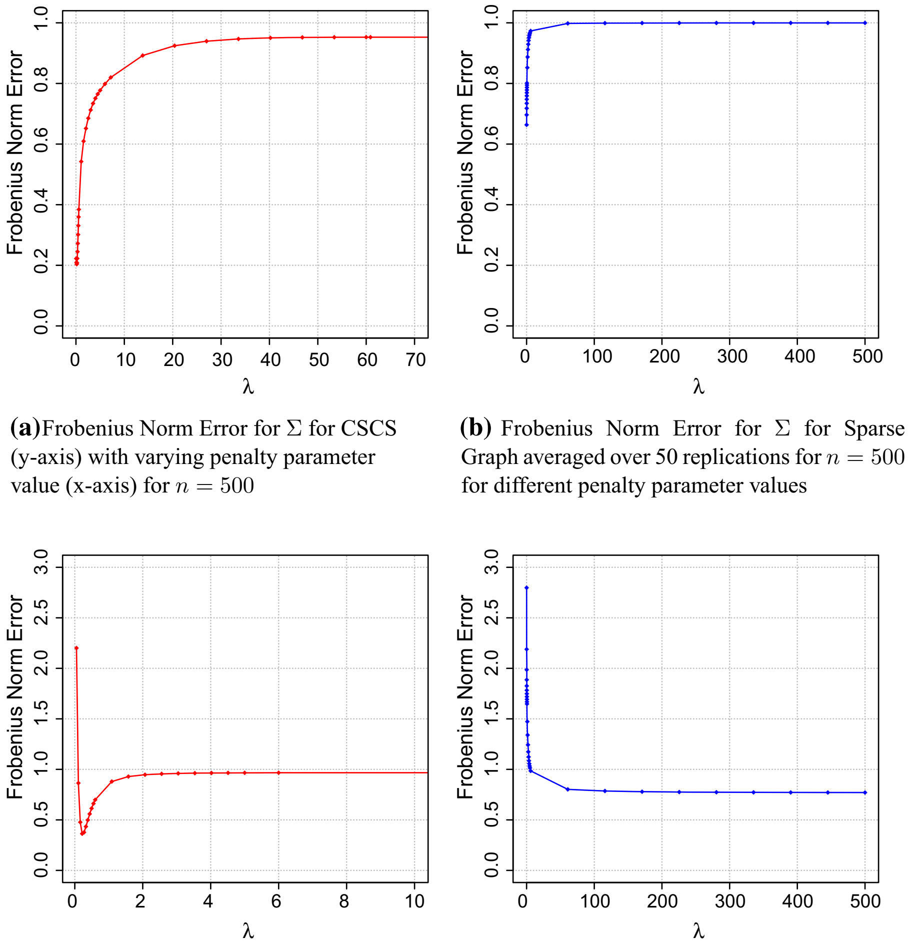
Each simulation yields a ROC curve from which the AUC is computed for FPR in the interval [0.001, 0.15]. The best results for each sample size are given in bold. CSCS achieves the highest AUC in each column

Covariance estimation comparison

Recall that Sparse Cholesky, Sparse Graph and CSCS all provide sparse estimates of (T, D), which can then be used to construct estimates of Ω. The goal of this experiment is to illustrate that the assumption $\{D_{ii}\}_{i=1}^{p}$ are identically 1 in the Sparse Graph approach can lead to inaccuracies in the estimation of the entries of the covariance matrix, and these can be improved by the CSCS approach, where no such assumption is required.

For this purpose, we consider the settings $p = 1000$ and $n \in \{p/2, 3p/2\}$ and generate 50 datasets for a range of λ values similar to the model selection experiment above. The true covariance matrix is generated using the same mechanism as in the "Graph selection comparison" part. Figure 2 show the Frobenius norm difference (averaged over 50 independent repetitions) between the true inverse covariance matrix and the estimate ($||\Omega - \hat{\Omega}||_F$), where $\hat{\Omega}$ is the estimated inverse covariance matrix for CSCS and Sparse Graph for a range on penalty parameter values for $n = 500$.

For each method (CSCS and Sparse Graph), we start with a penalty parameter value near zero (0.01) and increase it till the Frobenius norm error becomes constant, i.e., the penalty parameter is large enough so that all the off-diagonal entries of the Cholesky parameter are set to zero. That is why the range of penalty parameter values for the error curves is different in the various parts of Fig. 2. For $n = 500$, if the error is measured in terms of estimating the covariance matrix Σ, CSCS achieves a minimum error value of 0.2035 at $\lambda = 0.2$, and the maximum error value of 0.9526 is achieved at $\lambda = 60$ (or higher) when the resulting estimate of Σ is a diagonal matrix with the ith diagonal entry given by S_{ii} for $1 \leq i \leq p$. On the the other hand, Sparse Graph achieves a minimum error value of 0.6635 at $\lambda = 0.05$, and a maximum error value of 0.9996 at $\lambda = 70$ (or higher) when the resulting estimate of Ω is the identity matrix. If the error is measured in terms of estimating the inverse covariance matrix Ω, CSCS achieves a minimum error value of 0.363 at $\lambda = 0.25$, and the maximum error value of 2.2 is achieved at $\lambda = 0.05$. On the other hand, Sparse Graph achieves a minimum error value of 0.7725 at $\lambda = 70$ (or higher) when the resulting estimate of Ω is a diagonal matrix



(a) Frobenius Norm Error for Σ for CSCS (y-axis) with varying penalty parameter value (x-axis) for $n = 500$

(b) Frobenius Norm Error for Σ for Sparse Graph averaged over 50 replications for $n = 500$ for different penalty parameter values

(c) Frobenius Norm Error for Ω for CSCS (y-axis) with varying penalty parameter value (x-axis) for $n = 500$

(d) Frobenius Norm Error for Ω for Sparse Graph averaged over 50 replications for $n = 500$ for different penalty parameter values.

Fig. 2 **a** Frobenius Norm Error for Σ for CSCS (y-axis) with varying penalty parameter value (x-axis) for $n = 500$, **b** Frobenius Norm Error for Σ for Sparse Graph averaged over 50 replications for $n = 500$ for different penalty parameter values, **c** Frobenius Norm Error for Ω for CSCS (y-axis) with varying penalty parameter value (x-axis) for $n = 500$, **d** Frobenius Norm Error for Ω for Sparse Graph averaged over 50 replications for $n = 500$ for different penalty parameter values

with the ith diagonal entry given by $1/S_{ii}$ for $1 \leq i \leq p$, and achieves a maximum error value of 2.798 at $\lambda = 0.05$. If the penalty parameter is chosen by BIC (see Table 4) then CSCS has a Σ-error value of 0.2334 and an Ω-error value of 0.4054 (corresponding to $\lambda = 0.3$) and Sparse Graph has a Σ-error value of 0.7642 and an Ω-error value of 1.7658 (corresponding to $\lambda = 0.35$). A similar pattern is observed for the case $n = 1500$. When the true latent variable variances are not all equal to 1, it is clear that CSCS has a significantly superior overall estimation performance than Sparse Graph. This comparison of the two methods is to be interpreted in this non-equivariance setting.

Table 4 Frobenius Norm error for λ chosen by BIC for CSCS and Sparse Graph for $p = 1000$

	$n = 500(\Omega)$	$n = 1500(\Omega)$	$n = 500(\Sigma)$	$n = 1500(\Sigma)$
CSCS	**0.4054** (0.0018)	**0.4154** (0.0012)	**0.2334** (0.0107)	**0.2043** (0.0084)
Sparse Graph	1.7658 (0.0036)	1.9089 (0.0023)	0.7642 (0.0040)	0.7669 (0.0021)

The best results for each sample size are given in bold

3.2 Application to call center data

In this section we discuss the application of CSCS, Sparse Cholesky and Sparse Graph to the call center data from Huang et al. (2006). The data, coming from one call center in a major U.S. northeastern financial organization, contain the information about the time every call arrives at the service queue. For each day in 2002, except for 6 days when the data-collecting equipment was out of order, phone calls are recorded from 7:00am until midnight. The 17-hour period is divided into 102 10-minute intervals, and the number of calls arriving at the service queue during each interval are counted. Since the arrival patterns of weekdays and weekends differ, the focus is on weekdays here. In addition, after using singular value decomposition to screen out outliers that include holidays and days when the recording equipment was faulty (see Shen and Huang 2005), we are left with observations for 239 days.

The data were ordered by time period. Denote the data for day i by $N_i = (N_{i,1}, \dots, N_{i,102})'$, $i = 1, \dots, 239$ where $N_{i,t}$ is the number of calls arriving at the call centre for the tth 10-minute interval on day i. Let $y_{it} = \sqrt{N_{it} + 1/4}, i = 1, \dots, 239, t = 1, \dots, 102$. We apply the three penalized likelihood methods (CSCS, Sparse Graph, Sparse Cholesky) to estimate the 102×102 covariance matrix based on the residuals from a fit of the saturated mean model. That is, counts of each time period is centered by mean of that period. Following the analysis in Huang et al. (2006), the ℓ_1 penalty parameter for all three methods was picked using 5-fold cross validation on the training data set as follows. Randomly split the full dataset D into K subsets of about the same size, denoted by D_v, $v = 1, ..., K$. For each v, we use the data $D - D_v$ to estimate Ω_{-v} and D_v to validate. Then pick λ to minimize:

$$\mathrm{CV}(\lambda) = \frac{1}{K} \sum_{v=1}^{K} \left(d_v \log |\hat{\Omega}_{-v}^{-1}| + \sum_{i \in I_v} y_i' \hat{\Omega}_{-v} y_i \right)$$

where I_v is the index set of the data in D_v, d_v is the size of I_v, and $\hat{\Omega}_v$ is the inverse variance-covariance matrix estimated using the training data set $D - D_v$.

To assess the performance of different methods, we split the 239 days into training and test datasets. The data from the first T days ($T = 205, 150, 100, 75$), form the training dataset that is used to estimate the mean vector and the covariance matrix. The mean vector is estimated by the mean of the training data vectors. Four different methods, namely, CSCS, Sparse Cholesky, Sparse Graph and S (sample covariance matrix) are used to get an estimate of the covariance matrix. For each of the three penalized methods, the penalty parameter is chosen both by cross-validation and the BIC criterion. Hence, we have a total of seven estimators for the covariance matrix. The log-likelihood for the test dataset (consisting of the remaining $239 - T$ days) evaluated at all the above estimators is provided in Table 5. For all training data sizes, CSCS clearly demonstrates superior performance as compared to the other methods. Also, the comparative performance of CSCS with other methods improves significantly with decreasing training data size.

Table 5 Test data log-likelihood values for various estimation methods with training data size 205, 150, 100, 75

Method	Training data size			
	205	150	100	75
CSCS-CV	– 1090.447	– 1369.181	– 2225.907	**– 2841.348**
CSCS-BIC	**– 1072.75**	**– 1364.145**	**– 2214.729**	– 2849.931
Sparse graph-CV	– 1077.791	– 2237.298	– 3576.343	–4499.298
Sparse graph-BIC	– 1135.980	– 2421.950	– 3817.689	– 4846.118
Sparse Cholesky-CV	– 1500.094	– 2121.005	– 3579.932	– 496617558322
Sparse Cholesky-BIC	– 1523.409	– 2178.738	– 3584.160	– 5444.471
S	–1488.224	– 7696.740	Not pd	Not pd

The maximum of the likelihood values in each column is written in bold

Huang et al. (2006) additionally use the estimated mean and covariance matrix to forecast the number of arrivals in the later half of the day using arrival patterns in the earlier half of the day. Following their method, we compared the performance of all the four methods under consideration (details provided in Supplemental Section G). We found that all the three penalized methods outperform the sample covariance matrix estimator. However, as far as this specific forecasting task is concerned, the differences in their performance compared to each other are marginal. We suspect that the for the purposes of this forecasting task, the estimated mean (same for all three methods) has a much stronger effect than the estimated covariance matrix. Hence the difference in forecasting performance is much smaller than the difference in likelihood values. Nevertheless, Sparse Cholesky has the best performance for training data size $T = 205, 150$ (when the sample size is more than the number of variables) and CSCS has the best performance for training data sizes $T = 100, 75$ (when the sample size is less than the number of variables). See Supplemental Section G for more details.

3.3 Application to HapMap data

In this section, we analyze the HapMap phase 3 data from the International HapMap project (Consortium et al. 2010). The data consist of $n = 167$ humans from the YRI (Yoruba in Ibadan, Nigeria) population, and we focus on $p = 201$ consecutive tag SNPs on chromosome 22 (after filtering out infrequent sites with minor allele frequency ≤ 10 %).

To assess the performance of different methods, we split the 167 individuals into training and test datasets. The data from T randomly selected individuals (T = 100, 116, 133), form the training dataset that is used to estimate the mean vector and the covariance matrix. The mean vector is estimated by the mean of the training data vectors. CSCS, Sparse Cholesky and Sparse Graph are then used to get an estimate of the covariance matrix. For each of the penalized methods, the penalty parameter is chosen both by cross-validation and the BIC criterion. For the Sparse Cholesky algorithm, the resulting estimates have at least one zero in the diagonals of the D matrix, which results in a singular estimate of the covariance matrix with a log-likelihood of negative infinity. Hence, in Table 6, we report the log-likelihood for the test dataset (consisting of the remaining $167 - T$ individuals) evaluated at the four estimators (CSCS-CV, CSCS-BIC, Sparse Graph-CV, Sparse Graph-BIC). For all training data sizes, CSCS coupled with cross-validation clearly demonstrates the best performance as compared to the other methods.

Table 6 Test data log-likelihood values for various estimation methods with training data size $T = 100, 116, 133$

Training dataset	Method			
	CSCS-CV	Sparse graph-CV	CSCS-BIC	Sparse graph-BIC
100	**1718.216**	− 2784.134	582.6724	− 3477.754
116	**866.771**	− 2012.598	488.7348	− 2687.4
133	**860.0983**	− 1338.959	293.1705	− 1757.237

The maximum of the likelihood values in each row is written in bold

Table 7 TPR & FPR for cell signalling pathway data

Solver	$\lambda_i(\alpha) = 2n^{-\frac{1}{2}} Z^*_{\frac{\alpha}{2p(i-1)}}$		
	FP	TP	MCC
CSCS	0.5135	**0.9444**	**0.4848**
Sparse Cholesky	0.6216	**0.9444**	0.3916
Sparse graph	**0.4595**	0.7778	0.3277

The minimum FP rate value, the maximum TP rate value and the maximum MCC value in each column is written in bold

3.4 Application to flow cytometry data

In this section, we analyze a flow cytometry dataset on p = 11 proteins and n = 7466 cells, from Sachs et al. (2003). These authors fit a directed acyclic graph (DAG) to the data, producing the network in Figure 7a (in Supplemental section H). The ordering of the connections between pathway components were established based on perturbations in cells using molecular interventions and we consider the ordering to be known a priori. This dataset is analyzed in Friedman et al. (2008) and Shojaie and Michailidis (2010) using the GLASSO algorithm and the Sparse Graph algorithms, respectively.

In Shojaie and Michailidis (2010), the authors recommend using the following equation for penalty parameter selection: $\lambda_i(\alpha) = 2n^{-\frac{1}{2}} Z^*_{\frac{\alpha}{2p(i-1)}}$, where Z^*_q denotes the $(1-q)th$ quantile of the standard normal distribution. This choice uses a different penalty parameter for each row, and all the three penalized methods (Sparse Cholesky, Sparse Graph, CSCS) can be easily adapted to incorporate this. As shown in Table 7, using this method for Sparse Graph gives us a false positive rate of 0.46 and a true positive rate of 0.78, while Sparse Cholesky has a false positive rate of 0.62 and a true positive rate of 0.94. Hence, while Sparse Cholesky tends to find a lot of false edges, it fails to detect only one true edge. CSCS also fails to detect only one edge and thus has a true positive rate of 0.94. However, it does better overall compared to Sparse Cholesky as indicated by the lower false positive rate at 0.51. Figure 7 (in Supplemental section H) shows the true graph as well as the estimated graph using CSCS, Sparse Cholesky and Sparse Graph. Matthew's correlation coefficient (MCC) is defined as

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

where TP, TN, FP and FN correspond to true positive, true negative, false positive and false negative, respectively. The value of MCC ranges from -1 to 1 with larger values corresponding to better fits (-1 and 1 represent worst and best fits, respectively). CSCS attains the highest MCC value using the penalty parameter according to $\lambda_i(\alpha) = 2n^{-\frac{1}{2}} Z^*_{\frac{\alpha}{2p(i-1)}}$ indicating that it has the best overall performance.

4 Statistical properties

In this section, we will first examine statistical properties of the CSCS algorithm in a finite-sample (non-asymptotic) setting. Using these properties, we will establish estimation consistency and model selection consistency (oracle properties) for the CSCS algorithm in a high-dimensional setting, where the dimension p and the penalty parameter λ vary with n. Our approach is based on the strategy outlined in Meinshausen and Buhlmann (2006) and Massam et al. (2007). A similar approach was used by Peng et al. (2009) to establish asymptotic properties of SPACE, which is a penalized pseudo likelihood based algorithm for sparse estimation of Ω. Despite the similarity in the basic line of attack, there is an important structural difference between the asymptotic consistency arguments in Peng et al. (2009) and this section (apart from the fact that we are imposing sparsity in L, not Ω). For the purpose of establishing statistical properties, the authors in Peng et al. (2009) assume that diagonal entries of Ω are known, thereby reducing their objective function to the sum of a quadratic term and an ℓ_1 penalty term in Ω. The authors in Shojaie and Michailidis (2010) also establish graph selection consistency of the Sparse Graph approach under the assumption that the diagonal entries of L are 1. We do not make such an assumption for L, which leaves us with p additional non-zero parameters, and additional logarithmic terms in the objective function to work with. Nevertheless, we are able to adapt the basic argument in this challenging setting with an almost identical set of regularity assumptions as in Peng et al. (2009) (with assumptions on Ω replaced by the same assumptions on L). In particular, we only replace two assumptions in Peng et al. (2009) with a weaker and a stronger version respectively (see Assumption (A4) and Assumption (A5) below for more details).

Recall that n is the sample size, p is the number of variables, and λ is the penalty parameter in Q_{CSCS}. Let $\bar{\Omega} = \bar{L}^t \bar{L}$ denote the true inverse covariance matrix, and $\bar{\boldsymbol{\eta}}^r$ denote the lower triangular entries (including the diagonal) in the rth row of $\bar{L}$, for $1 \leq r \leq p$. Let $\mathcal{A}^r$ denote the set of indices corresponding to non-zero entries in rth row of $\bar{L}$ for $1 \leq r \leq p$, and let $q = \sum_{r=1}^{p} |\mathcal{A}^r|$. Let $\bar{\Sigma} = \bar{\Omega}^{-1}$ denote the true covariance matrix, and

$$s = \min_{1 \leq r \leq p} \min_{j \in \mathcal{A}^r} \left| \bar{\eta}_j^r \right|.$$

We first establish a non-asymptotic (finite sample) result which holds for every hexatuple (n, p, d, q, λ, s) satisfying certain algebraic constraints (see statement of Theorem 4.1) and the following standard assumptions regarding bounded eigenvalues, sub-Gaussianity, and incoherence.

- (A1 - Bounded eigenvalues) The eigenvalues of $\bar{\Omega}$ are bounded below by $\theta_{min} > 0$, and bounded above by $\theta_{max} < \infty$.
- (A2 - Sub Gaussianity) The random vectors $\mathbf{Y}^1, \ldots, \mathbf{Y}^n$ are *i.i.d.* sub-Gaussian i.e., there exists a constant $c > 0$ such that for every $\mathbf{x} \in \mathbb{R}^p$, $E\left[e^{\mathbf{x}'\mathbf{Y}^i}\right] \leq e^{c\mathbf{x}'\mathbf{x}}$. Along with Assumption A1, this in particular implies that the sub-Gaussian norm of $\boldsymbol{\alpha}^T \mathbf{Y}^i$, with $\boldsymbol{\alpha}^T \boldsymbol{\alpha} = 1$, is bounded by a constant κ.

- (A3 - Incoherence condition) There exists $\delta < 1$ such that for every $1 \leq r \leq p$ and $j \notin \mathcal{A}^r$,

  $$\left| \bar{\Sigma}^t_{j,\mathcal{A}^r} \left(\bar{\Sigma}_{\mathcal{A}^r\mathcal{A}^r} + \frac{2}{(\bar{\eta}^r_r)^2} \Delta_r \right)^{-1} \operatorname{sign}\left(\bar{\boldsymbol{\eta}}^r_{\mathcal{A}^r}\right) \right| \leq \delta.$$

  Here, Δ_r is a $|\mathcal{A}^r| \times |\mathcal{A}^r|$ matrix with

  $$(\Delta_r)_{jj'} = \begin{cases} 1 & \text{if } j = j' = |\mathcal{A}^r|, \\ 0 & \text{otherwise.} \end{cases}$$

  This assumption is a version of the standard mutual incoherence condition needed for establishing consistency for ℓ_1 penalized estimators, see for example (Peng et al. 2009; Khare et al. 2015; Yu and Bien 2016). Roughly speaking, this condition ensures that for each r, the correlation between the signal variables (all j such that $\bar{L}_{rj} \neq 0$) and the noise variables (all j such that $\bar{L}_{rj} = 0$) is not too strong.

Under these assumptions, the following non-asymptotic result can be established.

Theorem 4.1 *Suppose that (A1)–(A3) are satisfied, and the hexatuple (n, p, d, q, λ, s) is such that*

- $n > K_1 \log p$
- $s > (4\theta_{max} + 2)d\lambda$
- $q\lambda \leq K_2,$

where K_1, K_2 are known constants (see Supplemental Section E). Then, the following holds with probability at least $1 - \frac{64}{p}$.

(i) A solution of the minimization problem

$$\inf_{L \in \mathcal{L}_p} Q_{CSCS}(L) \tag{4.1}$$

exists.

(ii) Any solution $\hat{L}_n$ of the minimization problem in (4.1) satisfies

$$\|\hat{L}_n - \bar{L}_n\| \leq (2\theta_{max} + 1)q\lambda.$$

and

$$sign(\bar{L}_{p_n,ij}) = sign(\bar{\Omega}_{p_n,ij}),$$

for every $1 \leq j \leq i \leq p$.

Here $\|\cdot\|$ denotes the operator norm, and $\operatorname{sign}(x)$ takes the values $\{-1, 0, 1\}$ when $x < 0$, $x = 0$, and $x > 0$ respectively. A proof of the above result is provided in the Supplemental Section E.

The above non-asymptotic result can be easily converted into an asymptotic result establishing estimation and sign-consistency. In this setting, the dimension $p = p_n$ and the penalty parameter $\lambda = \lambda_n$ vary with n, and various quantities such as d, q, s defined above, now depend on n. In particular, $\{\bar{\Omega}_{p_n} = \bar{L}^t_{p_n}\bar{L}_{p_n}\}_{n \geq 1}$ denotes the sequence of true inverse covariance matrices, $\bar{\boldsymbol{\eta}}^r_n$ denotes the lower triangular entries (including the diagonal) in the rth row of $\bar{L}_{p_n}$, for $1 \leq r \leq p$, $\mathcal{A}^r_n$ denotes the set of indices corresponding to non-zero entries in

rth row of $\bar{L}_{p_n}$ for $1 \leq r \leq p_n$, and $q_n = \sum_{r=1}^{p_n} \left|\mathcal{A}_n^r\right|$. Also, $\bar{\Sigma}_{p_n} = \bar{\Omega}_{p_n}^{-1}$ denotes the true covariance matrix for every $n \geq 1$, and for every $n \geq 1$,

$$s_n = \min_{1 \leq r \leq p} \min_{j \in \mathcal{A}_n^r} \left|\bar{\eta}_{n,j}^r\right|.$$

In addition to assumptions (A1)–(A3) (with $p = p_n$ and the bounds holding uniformly in n), we need the following standard assumptions on the sequences s_n, p_n, d_n, q_n, and λ_n.

- (A4 - Signal size growth) $\frac{s_n}{\sqrt{d_n}\lambda_n} \to \infty$, where $d_n = \max_{1 \leq r \leq p_n} |\mathcal{A}^r|$. This assumption will be useful for establishing sign consistency. The signal size condition in Peng et al. (2009) is $\frac{s_n}{\sqrt{q_n}\lambda_n} \to \infty$, which is stronger than the signal size condition above, as $d_n \leq q_n$. The difference in Assumption A4 and the corresponding assumption in Peng et al. (2009) can be explained by the fact that the proposed optimization problem can be broken into p separate row-wise optimization problems. Hence, the overall theoretical and computational difficulty in estimating L lies in its densest row. This has also been pointed out in Yu and Bien (2016).
- (A5 - Growth of p_n, q_n and λ_n) The following conditions hold: $q_n = o\left(\sqrt{\frac{n}{\log p_n}}\right)$, $\sqrt{\frac{q_n \log p_n}{n}} = o(\lambda_n)$, $\lambda_n \sqrt{\frac{n}{\log p_n}} \to \infty$ and $q_n\lambda_n \to 0$ as $n \to \infty$. The growth conditions in Peng et al. (2009) are the same as above (with q_n denoting the sparsity in the true Ω in Peng et al. (2009)), expect that $q_n\lambda_n \to 0$ above is replaced by the weaker assumption $\sqrt{q_n}\lambda_n \to 0$.

The asymptotic consistency result stated below follows immediately from Theorem 4.1.

Theorem 4.2 *Under Assumptions (A1)–(A3) (with $p = p_n$ and the bounds holding uniformly in n), and (A4)–(A5), the following event happens with probability tending to 1 as $n \to \infty$: A solution to the minimization problem $\inf_{L \in \mathcal{L}_{p_n}} Q_{CSCS}(L)$ exists, and any such solution satisfies $sign(\hat{L}_{p_n,ij}) = sign(\bar{L}_{p_n,ij})$ for every $1 \leq i \leq j \leq p$, and $\|\hat{L}_{p_n} - \bar{L}_{p_n}\| \leq (2\theta_{max} + 1)q_n\lambda_n$.*

5 Discussion

This paper proposes a novel penalized likelihood based approach for sparse Cholesky based covariance estimation for multivariate Gaussian data, when a natural ordering of variables is available. The goal is to overcome some of the shortcomings of current methods, but at the same time retain their respective strengths. We start with the objective function for the highly useful Sparse Cholesky approach in Huang et al. (2006). Reparametrization of this objective function in terms of the inverse of the classical Cholesky factor of the covariance matrix, along with appropriate changes to the penalty term, leads us to the formulation of the CSCS objective function. It is then shown that the CSCS objective function is jointly convex in its arguments. A coordinate-wise minimization algorithm that minimizes this objective, via closed form iterates, is proposed, and subsequently analyzed. The convergence of this coordinate-wise minimization algorithm to a global minimum is established rigorously. It is also established that the estimate produced by the CSCS algorithm always leads to a positive definite estimate of the covariance matrix—thus ensuring that CSCS leads to well defined estimates that are always computable. Such a guarantee is not available with the Sparse Cholesky approach when $n < p$. Large sample properties of CSCS establish estimation and model selection consistency of the method as both the sample size and dimension tend to infinity. We also point out that the Sparse Graph approach in Shojaie and Michailidis (2010), while always useful for graph selection, may suffer for estimation purposes due the assumption that the conditional variances $\{D_{ii}\}_{i=1}^{p}$ are identically 1. The performance of CSCS compared to Sparse Cholesky and Sparse Graph is also illustrated via simulations and application to a call center dataset, HapMap dataset, and a flow cytometry dataset. These experiments complement and support the technical results in the paper by demonstrating the following.

(a) When $n < p$, it is easy to find examples where Sparse Cholesky converges to its global minimum which corresponds to a singular covariance matrix (Fig. 1).
(b) When $n < p$, the graph selection and estimation performance of CSCS is significantly better than Sparse Cholesky, due to the fact that Sparse Cholesky either converges to a global minimum with singularity issues, or to a local minimum (Sects. 3.1 and 3.2).
(c) For graph selection, CSCS is competitive with Sparse Graph and can have better performance as compared to Sparse Graph. Although the improvement may not sometimes be as significant as that over Sparse Cholesky, these results demonstrate that CSCS is a useful addition to the high-dimensional DAG selection toolbox (Sects. 3.1 and 3.4).
(d) For estimation purposes, CSCS can lead to significant improvements in performance over Sparse Graph (Sect. 3.1).

Acknowledgements The work of Kshitij Khare was partially supported by NSF Grant DMS-1106084. Sang-Yun Oh was supported in part by Laboratory Directed Research and Development (LDRD) funding from Berkeley Lab, provided by the Director, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. The work of Bala Rajaratnam was partially supported by US Air Force Office of Scientific Research grant award number FA9550-13-1-0043, US National Science Foundation under Grant Nos. DMS-CMG 1025465, AGS-1003823, DMS-1106642, DMS-CAREER-1352656, Defense Advanced Research Projects Agency DARPA- YFAN66001-111-4131 and SMC-DBNKY.

References

Aragam, B., Amini, A., & Zhou, Q. (2016). Learning directed acyclic graphs with penalized neighbourhood regression. arxiv.

Aragam, B., & Zhou, Q. (2015). Concave penalized estimation of sparse Gaussian Bayesian networks. *Journal of Machine Learning Research*, *16*, 2273–2328.

Banerjee, O., Ghaoui, L. E., & D'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *The Journal of Machine Learning Research*, *9*, 485–516.

Cai, T., Liu, W., & Luo, X. (2011). A constrained l1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, *106*, 594–607.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874.

Friedman, J., Hastie, T., & Tibshirani, R. (2008a). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*, 1–22.

Friedman, J., Hastie, T., & Tibshirani, R. (2008b). Sparse inverse covariance estimation with the graphical Lasso. *Biostatistics*, *9*, 432–441.

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Applications of the lasso and grouped Lasso to the estimation of sparse graphical models. Technical Report, Department of Statistics, Stanford University.

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

Hsieh, C.-J., Sustik, M. A., Dhillon, I. S., & Ravikumar, P. (2011). Sparse inverse covariance matrix estimation using quadratic approximation. In *Advances in Neural Information Processing Systems 24 (NIPS 2011)*

Huang, J., Liu, N., Pourahmadi, M., & Liu, L. (2006). Covariance selection and estimation via penalised normal likelihoode. *Biometrika*, *93*, 85–98.

International HapMap 3 Consortium et al. (2010). Integrating common and rare genetic variation in diverse human populations. *Nature*, *467*(7311), 52–58.

Khare, K., Oh, S., & Rajaratnam, B. (2015). A convex pseudo-likelihood framework for high dimensional partial correlation estimation with convergence guarantees. *Journal of the Royal Statistical Society B*, *77*, 803–825.

Khare, K. & Rajaratnam, B. (2014). Convergence of cyclic coordinatewise l1 minimization. arxiv.

Lin, L., Drton, M., & Shojaie, A. (2016). Estimation of high-dimensional graphical models using regularized score matching. *Electronic Journal of Statistics*, *10*, 806–854.

Liu, W., & Luo, X. (2015). Fast and adaptive sparse precision matrix estimation in high dimensions. *Journal of Multivariate Analysis*, *135*, 153–162.

Massam, H., Paul, D., & Rajaratnam, B. (2007). Penalized empirical risk minimization using a convex loss function and ℓ_1 penalty. (**unpublished manuscript**).

Mazumder, R., & Hastie, T. (2012). Exact covariance thresholding into connected components for large-scale graphical lasso. *The Journal of Machine Learning Research*, *13*, 781–794.

Meinshausen, N., & Buhlmann, P. (2006). High dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, *34*, 1436–1462.

Oh, S., Dalal, O., Khare, K., & Rajaratnam, B. (2014). Optimization methods for sparse pseudo-likelihood graphical model selection. In *Proceedings of neural information processing systems*.

Paulsen, V. I., Power, S. C., & Smith, R. R. (1989). Schur products and matrix completions. *Journal of Functional Analysis*, *85*, 151–178.

Peng, J., Wang, P., Zhou, N., & Zhu, J. (2009). Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association*, *104*, 735–746.

Rothman, A., Levina, E., & Zhu, J. (2010). A new approach to cholesky-based covariance regularization in high dimensions. *Biometrika*, *97*, 539–550.

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D., & Nolan, G. (2003). Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, *308*(5721), 504–6.

Shen, H., & Huang, J. Z. (2005). Analysis of call center arrival data using singular value decomposition. *Applied Stochastic Models in Business and Industry*, *21*, 251–63.

Shojaie, A., & Michailidis, G. (2010). Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs. *Biometrika*, *97*, 519–538.

Smith, M., & Kohn, R. (2002). Parsimonious covariance matrix estimation for longitudinal data. *Journal of the American Statistical Association*, *97*, 1141–1153.

van de Geer, S., & Buhlmann, P. (2013). l0-penalized maximum likelihood for sparse directed acyclic graphs. *Annals of Statistics*, *41*, 536–567.

Wagaman, A., & Levina, E. (2009). Discovering sparse covariance structures with the isomap. *Journal of Computational and Graphical Statistics*, *18*, 551–572.

Wu, W. B., & Pourahmadi, M. (2003). Nonparametric estimation of large covariance matrices of longitudinal data. *Biometrika*, *90*, 831–844.

Yu, G., & Bien, J. (2016). Learning local dependence in ordered data. arXiv:1604.07451.

Yuan, M. (2010). High dimensional inverse covariance matrix estimation via linear programming. *Journal of Machine Learning Research*, *11*, 2261–2286.

Zhang, T., & Zou, H. (2014). High dimensional inverse covariance matrix estimation via linear programming. *Biometrika*, *101*, 103–120.

Zheng, H., Tsui, K. W., Kang, X., & Deng, X. (2017). Cholesky-based model averaging for covariance matrix estimation. *Statistical Theory and Related Fields*, *1*, 48–58.

Affiliations

Kshitij Khare[1] · Sang-Yun Oh[2] · Syed Rahman[1] · Bala Rajaratnam[3]

Sang-Yun Oh
syoh@pstat.ucsb.edu

Syed Rahman
shr264@ufl.edu

Bala Rajaratnam
brajaratnam01@gmail.com

[1] University of Florida, Gainesville, USA

[2] University of California, Santa Barbara, USA

[3] University of California, Davis, USA