Journal of Machine Learning Research 21 (2020) 1-45 Submitted 7/19; Revised 4/20; Published 4/20

High-dimensional Gaussian graphical models on network-linked data

Tianxi Li[*] TIANXILI@VIRGINIA.EDU
Department of Statistics
University of Virginia
Charlottesville, VA 22904, USA

Cheng Qian[*] QIANC@SEU.EDU.CN
School of Mathematics
Southeast University
Nanjing, Jiangsu 211189, China

Elizaveta Levina ELEVINA@UMICH.EDU

Ji Zhu JIZHU@UMICH.EDU
Department of Statistics
University of Michigan
Ann Arbor, MI 48109, USA

Editor: Jie Peng

Abstract

Graphical models are commonly used to represent conditional dependence relationships between variables. There are multiple methods available for exploring them from high-dimensional data, but almost all of them rely on the assumption that the observations are independent and identically distributed. At the same time, observations connected by a network are becoming increasingly common, and tend to violate these assumptions. Here we develop a Gaussian graphical model for observations connected by a network with potentially different mean vectors, varying smoothly over the network. We propose an efficient estimation algorithm and demonstrate its effectiveness on both simulated and real data, obtaining meaningful and interpretable results on a statistics coauthorship network. We also prove that our method estimates both the inverse covariance matrix and the corresponding graph structure correctly under the assumption of network "cohesion", which refers to the empirically observed phenomenon of network neighbors sharing similar traits.

Keywords: High-dimensional statistics, Gaussian graphical model, network analysis, network cohesion, statistical learning

1. Introduction

Network data represent information about relationships (edges) between units (nodes), such as friendships or collaborations, and are often collected together with more "traditional" covariates that describe one unit. In a social network, edges may represent friendships between people (nodes), and traditional covariates could be their demographic characteristics such as gender, race, age, and so on. Incorporating relational information in statistical

[*] Authors with equal contribution

modeling tasks focused on "traditional" node covariates should improve performance, since it offers additional information, but most traditional multivariate analysis methods are not designed to use such information. In fact, most such methods for regression, clustering, density estimation and so on tend to assume the sampled units are homogeneous, typically independent and identically distributed (i.i.d.), which is unlikely to be the case for units connected by a network. While there is a fair amount of work on incorporating such information into specific settings (Manski, 1993; Lee, 2007; Yang et al., 2011; Raducanu and Dornaika, 2012; Vural and Guillemot, 2016), work on extending standard statistical methods to network-linked data has only recently started appearing, for example, Li et al. (2019) for regression, Tang et al. (2013) for classification, and Yang et al. (2013), Binkiewicz et al. (2017) for clustering. Our goal in this paper is to develop an analog to the widely used Gaussian graphical models for network-linked data which takes advantage of this additional information to improve performance when possible.

Graphical models are commonly used to represent independence relationships between random variables, with each variable corresponding to a node, and edges representing conditional or marginal dependence between two random variables. Note that a graphical model is a graph connecting variables, as opposed to the networks discussed above, which are graphs connecting observations. Graphical models have been widely studied in statistics and machine learning and have applications in bioinformatics, text mining and causal inference, among others. The Gaussian graphical model belongs to the family of undirected graphical models, or Markov random fields, and assumes the variables are jointly Gaussian. Specifically, the conventional Gaussian graphical model for a data matrix $X \in \mathbb{R}^{n \times p}$ assumes that the rows $X_{i\cdot}$, $i = 1, \ldots, n$, are independently drawn from the same p-variate normal distribution $\mathcal{N}(\mu, \Sigma)$. This vastly simplifies analysis, since for the Gaussian distribution all marginal dependence information is contained in the covariance matrix, and all conditional independence information in its inverse. In particular, random variables j and j' are conditionally independent given the rest if and only if the (j, j')-th entry of the inverse covariance matrix Σ^{-1} (the precision matrix) is zero. Therefore estimating the graph for a Gaussian graphical model is equivalent to identifying zeros in the precision matrix, and this problem has been well studied, in both the low-dimensional and the high-dimensional settings. A pioneering paper by Meinshausen and Bühlmann (2006) proposed neighborhood selection, which learns edges by regressing each variable on all the others via lasso, and established good asymptotic properties in high dimensions. Many penalized likelihood methods have been proposed as well (Yuan and Lin, 2007; Banerjee et al., 2008; Rothman et al., 2008; d'Aspremont et al., 2008; Friedman et al., 2008). In particular, the graphical lasso (glasso) algorithm of Friedman et al. (2008) and its subsequent improvements (Witten et al., 2011; Hsieh et al., 2013b) are widely used to solve the problem efficiently.

The penalized likelihood approach to Gaussian graphical models assumes the observations are i.i.d., a restrictive assumption in many real-world situations. This assumption was relaxed in Zhou et al. (2010); Guo et al. (2011) and Danaher et al. (2014) by allowing the covariance matrix to vary smoothly over time or across groups, while the mean vector remains constant. A special case of modeling the mean vector on additional covariates associated with each observation has also been studied (Rothman et al., 2010; Yin and Li, 2011; Lee and Liu, 2012; Cai et al., 2013; Lin et al., 2016). Neither of these relaxations are easy to adapt to network data, and their assumptions are hard to verify in practice.

In this paper, we consider the problem of estimating a graphical model with heterogeneous mean vectors when a network connecting the observations is available. For example, in analyzing word frequencies in research papers, the conditional dependencies between words may represent certain common phrases used by all authors. However, since different authors also have different research topics and writing styles, there is individual variation in word frequencies themselves, and the coauthorship information is clearly directly relevant to modeling both the universal dependency graph and the individual means. We propose a generalization of the Gaussian graphical model to such a setting, where each data point can have its own mean vector but the data points share the same covariance structure. We further assume that a network connecting the observations is available, and that the mean vectors exhibit network "cohesion", a generic term describing the phenomenon of connected nodes behaving similarly, observed widely in empirical studies and experiments (Fujimoto and Valente, 2012; Haynie, 2001; Christakis and Fowler, 2007). We develop a computationally efficient algorithm to estimate the proposed Gaussian graphical model with network cohesion, and show that the method is consistent for estimating both the covariance matrix and the graph in high-dimensional settings under a network cohesion assumption. Simulation studies show that our method works as well as the standard Gaussian graphical model in the i.i.d. setting, and is effective in the setting of different means with network cohesion, while the standard Gaussian graphical model completely fails.

The rest of the paper is organized as follows. Section 2 introduces a Gaussian graphical model on network-linked observations and the corresponding two-stage model estimation procedure. An alternative estimation procedure based on joint likelihood is also introduced, although we will argue that the two-stage estimation is preferable from both the computational and the theoretical perspectives. Section 3 presents a formal definition of network cohesion and error bounds under the assumption of network cohesion and regularity conditions, showing we can consistently estimate the partial dependence graph and model parameters. Section 4 presents simulation studies comparing the proposed method to standard graphical lasso and the two-stage estimation algorithm to the joint likelihood approach. Section 5 applies the method to analyzing dependencies between terms from a collection of statistics papers' titles and the associated coauthorship network. Section 6 concludes with discussion.

2. Gaussian graphical model with network cohesion

2.1. Preliminaries

We start with setting up notation. For a matrix $X \in \mathbb{R}^{n\times p}$, let $X_{\cdot j}$ be the jth column and $X_{i\cdot}$ the ith row. By default, we treat all vectors as column vectors. Let $\|X\|_F = (\sum_{i,j} X_{ij}^2)^{1/2}$ be the Frobenius norm of X and $\|X\|$ the spectral norm, i.e., the largest singular value of X. Further, let $\|X\|_0 = \#\{(i,j) : X_{ij} \neq 0\}$ be the number of non-zero elements in X, $\|X\|_1 = \sum_{ij} |X_{ij}|$, and $\|X\|_{1,\text{off}} = \sum_{i\neq j} |X_{ij}|$. For a square matrix Σ, let $\text{tr}(\Sigma)$ and $\det(\Sigma)$ be the trace and the determinant of Σ, respectively, and assuming Σ is a covariance matrix, let $r(\Sigma) = \frac{\text{tr}(\Sigma)}{\|\Sigma\|}$ be its *stable rank*. It is clear that $1 \leq r(\Sigma) \leq p$ for any nonzero covariance matrix Σ.

While it is common, and not incorrect, to use the terms "network" and "graph" interchangeably, throughout this paper "*network*" is used to refer to the *observed* network connecting the n observations, and "*graph*" refers to the conditional dependence graph of p variables *to be estimated*. In a network or graph $\mathcal{G}$ of size n, if two nodes i and i' of $\mathcal{G}$ are connected, we write $i \sim_{\mathcal{G}} i'$, or $i \sim i'$ if $\mathcal{G}$ is clear from the context. The adjacency matrix of a graph $\mathcal{G}$ is an $n \times n$ matrix A defined by $A_{ii'} = 1$ if $i \sim_{\mathcal{G}} i'$ and 0 otherwise. We focus on undirected networks, which implies the adjacency matrix is symmetric. Given an adjacency matrix A, we define its Laplacian by $L = D - A$ where $D = \mathrm{diag}(d_1, d_2, \cdots, d_n)$ and $d_i = \sum_{i'=1}^{n} A_{ii'}$ is the degree of node i. A well-known property of the Laplacian matrix L is that, for any vector $\mu \in \mathbb{R}^n$,

$$\mu^T L \mu = \sum_{i \sim i'} (\mu_i - \mu_{i'})^2. \tag{1}$$

We also define a normalized Laplacian $\mathcal{L}_s = \frac{1}{\bar{d}} L$ where $\bar{d}$ is the average degree of the network $\mathcal{G}$, given by $\bar{d} = \frac{1}{n} \sum_i d_i$. We denote the eigenvalues of $\mathcal{L}_s$ by $\tau_1 \geq \tau_2 \geq \cdots \geq \tau_{n-1} \geq \tau_n = 0$, and the corresponding eigenvectors by $u_1, \ldots, u_n$.

2.2. Gaussian graphical model with network cohesion (GNC)

We now introduce the heterogeneous Gaussian graphical model, as a generalization of the standard Gaussian graphical model with i.i.d. observations. Assume the data matrix X contains n independent observations $X_{i\cdot} \in \mathbb{R}^p, i = 1, 2, \cdots, n$. Each $X_{i\cdot}$ is a random vector drawn from an individual multivariate Gaussian distribution

$$X_{i\cdot} \sim \mathcal{N}(\mu_i, \Sigma), i = 1, 2, \cdots, n. \tag{2}$$

where $\mu_i \in \mathbb{R}^p$ is a p-dimensional vector and Σ is a $p \times p$ symmetric positive definite matrix. Let $\Theta = \Sigma^{-1}$ be the corresponding precision matrix and $M = (\mu_1, \mu_2, \cdots, \mu_n)^T$ be the mean matrix, which will eventually incorporate cohesion. Recall that in the Gaussian graphical model, $\Theta_{jj'} = 0$ corresponds to the conditional independence relationship $x_j \perp x_j' | \{x_k, k \neq j, j'\}$ (Lauritzen, 1996). Therefore a typical assumption, especially in high-dimensional problems, is that Θ is a sparse matrix; this both allows us to estimate Θ when $p > n$, and produces a sparse conditional dependence graph.

Model (2) is much more flexible than the i.i.d. graphical model, and it separates covariation caused by individual preference (cohesion in the mean) from universal co-occurrence (covariance). The price we pay for this flexibility is the much larger number of parameters, and model (2) cannot be fitted without additional assumptions on the mean, since we only have one observation to estimate each vector μ_i. The structural assumption we make in this paper is *network cohesion*, a phenomenon of connected individuals in a social network tending to exhibit similar traits. It has been widely observed in many empirical studies such as health-related behaviors or academic performance (Michell and West, 1996; Haynie, 2001; Pearson and West, 2003). Specifically, in our Gaussian graphical model (2), we assume that connected nodes in the observed network have similar mean vectors. This assumption is reasonable and interpretable in many applications. For instance, in the coauthorship network example, cohesion indicates coauthors tend to have similar word preferences, which is reasonable since they work on similar topics and share at least some publications.

2.3. Fitting the GNC model

The log-likelihood of the data under model (2) is, up to a constant,

$$\ell(M, \Theta) = \log\det(\Theta) - \frac{1}{n}\mathrm{tr}(\Theta(X-M)^T(X-M)). \tag{3}$$

A sparse inverse covariance matrix Θ and a cohesive mean matrix M are naturally incorporated into the following two-stage procedure, which we call Gaussian graphical model estimation with Network Cohesion and lasso penalty (GNC-lasso).

Algorithm 1 (Two-stage GNC-lasso algorithm) *Input: a standardized data matrix X, network adjacency matrix A, tuning parameters λ and α.*

1. *Mean estimation. Let L_s be the standardized Laplacian of A. Estimated the mean matrix by*
$$\hat{M} = \arg\min_M \|X - M\|_F^2 + \alpha\,\mathrm{tr}(M^T\mathcal{L}_sM). \tag{4}$$
2. *Covariance estimation. Let $\hat{S} = \frac{1}{n}(X-\hat{M})^T(X-\hat{M})$ be the sample covariance matrix of X based on $\hat{M}$. Estimate the precision matrix by*
$$\hat{\Theta} = \arg\min_{\Theta\in\mathbb{S}_+^n} \log\det(\Theta) - \mathrm{tr}(\Theta\hat{S}) - \lambda\|\Theta\|_{1,\mathrm{off}}. \tag{5}$$

The first step is a penalized least squares problem, where the penalty can be written as

$$\mathrm{tr}(M^T\mathcal{L}_sM) = \sum_{i\sim i'}\|\mu_i - \mu_{i'}\|^2. \tag{6}$$

This can be viewed as a vector version of the Laplacian penalty used in variable selection (Li and Li, 2008, 2010; Zhao and Shojaie, 2016) and regression problems (Li et al., 2019) with network information. It penalizes the difference between mean vectors of connected nodes, encouraging cohesion in the estimated mean matrix. Both terms in (4) are separable in the p coordinates and the least squares problem has a closed form solution,

$$\hat{M}_{\cdot j} = (I_n + \alpha\mathcal{L}_s)^{-1}X_{\cdot j}, \quad j = 1, 2, \cdots, p. \tag{7}$$

In practice, we usually need to compute the estimate for a sequence of α values, so we first calculate the eigen-decomposition of $\mathcal{L}_s$ and then obtain each $(I + \alpha\mathcal{L}_s)^{-1}$ in linear time. In most applications, networks are very sparse, and taking advantage of sparsity and the symmetrically diagonal dominance of $\mathcal{L}_s$ allows to compute the eigen-decomposition very efficiently (Cohen et al., 2014). Given $\hat{M}$, criterion (5) is a graphical lasso problem that uses the lasso penalty (Tibshirani, 1996) to encourage sparsity in the estimated precision matrix, and can be solved by the glasso algorithm (Friedman et al., 2008) efficiently or any of its variants, later significantly improved further by Witten et al. (2011) and Hsieh et al. (2014, 2013a).

2.4. An alternative: penalized joint likelihood

An alternative and seemingly more natural approach is to maximize a penalized log-likelihood to estimate both M and Θ jointly as

$$(\hat{\Theta}, \hat{M}) = \arg\max_{\Theta, M} \ \log\det(\Theta) - \frac{1}{n}\mathrm{tr}(\Theta(X-M)^T(X-M)) - \lambda\|\Theta\|_{1,\mathrm{off}} - \frac{\alpha}{n}\mathrm{tr}(M^T\mathcal{L}_s M). \tag{8}$$

The objective function is bi-convex and the optimization problem can be solved by alternately optimizing over M with fixed Θ and then optimizing over Θ with fixed M until convergence. We refer to this method as iterative GNC-lasso. Though this strategy seems more principled in a sense, we implement our method with the two-stage algorithm, for the following reasons.

First, the computational complexity of the iterative method based on joint likelihood is significantly higher, and it does not scale well in either n or p. This is because when Θ is fixed and we need to maximize over M, all p coordinates are coupled in the objective function, so the scale of the problem is $np \times np$. Even for moderate n and p, solving this problem requires either a large amount of memory or applying Gauss-Seidel type algorithms that further increase the number of iterations. This problem is exacerbated by the need to select two tuning parameters λ and α jointly, because, as we will discuss later, they are also coupled.

More importantly, our empirical results show that the iterative estimation method does not improve on the two-stage method (if it does not slightly hurt it). The same phenomenon was observed empirically by Yin and Li (2013) and Lin et al. (2016), who used a completely different approach of applying sparse regression to adjust the Gaussian graphical model, though those papers did not offer an explanation. We conjecture that this phenomenon of maximizing penalized joint likelihood failing to improve on a two-stage method may be general. An intuitive explanation might lie in the fact that the two parameters M and Θ are only connected through the penalty: the Gaussian log-likelihood (3) without a penalty is maximized over M by $\hat{M} = X$, which does not depend on Θ. Thus the likelihood itself does not pool information from different observations to estimate the mean (nor should it, since we assumed they are different), while the cohesion penalty is separable in the p variables and does not pool information between them either. An indirect justification of this conjecture follows from a property of the two-stage estimator stated in Proposition 2 in Appendix B, and the numerical results in Section 4 provide empirical support.

2.5. Model selection

There are two tuning parameters, λ and α, in the two-stage GNC-lasso algorithm. The parameter α controls the amount of cohesion over the network in the estimated mean and can be easily tuned based on its predictive performance. In subsequent numerical examples, we always choose α from a sequence of candidate values by 10-fold cross-validation. In each fold, the sum of squared prediction errors on the validation set $\sum(X_{ij} - \hat{\mu}_{ij})^2$ is computed and the α value is chosen to minimize the average prediction error. If the problem is too large for cross-validation, we can also use the generalized cross-validation (GCV) statistic as an alternative, which was shown to be effective in theory for ridge-type regularization

(Golub et al., 1979; Li, 1986). The GCV statistic for α is defined by

$$\mathrm{GCV}(\alpha) = \frac{1}{np}\|X - \hat{M}(\alpha)\|_F^2/[1 - \frac{1}{n}\mathrm{tr}((I + \alpha\mathcal{L}_s)^{-1})]^2 = \frac{\|X - \hat{M}(\alpha)\|_F^2}{np[1 - \frac{1}{n}\sum_{i=1}^n \frac{1}{1+\alpha\tau_i}]^2}$$

where we write $\hat{M}(\alpha)$ to emphasize that the estimate depends on α. The parameter α should be selected to minimize GCV. Empirically, we observe running the true cross-validation is typically more accurate than using GCV. So the GCV is only recommended for problems that are too large to run cross-validation.

Given α, we obtain $\hat{M}$ and use $\hat{S} = \frac{1}{n}(X - \hat{M})^T(X - \hat{M})$ as the input of the glasso problem in (5); therefore λ can be selected by standard glasso tuning methods, which may depend on the application. For example, we can tune λ according to some data-driven goodness-of-fit criterion such as BIC, or via stability selection. Alternatively, if the graphical model is being fitted as an exploratory tool to obtain an interpretable dependence between variables, λ can be selected to achieve a pre-defined sparsity level of the graph, or chosen subjectively with the goal of interpretability. Tuning illustrates another important advantage of the two-stage estimation over the iterative method: when estimating the parameters jointly, due to the coupling of α and λ the tuning must be done on a grid of their values and using the same tuning criteria. The de-coupling of tuning parameters in the two-stage estimation algorithm is both more flexible, since we can use different tuning criteria for each if desired, and more computationally tractable since we only need to do two line searches instead of a two-dimensional grid search.

2.6. Related work and alternative penalties

The Laplacian smoothness penalty of the form (1) or (6) was originally used in machine learning for embedding and kernel learning (Belkin and Niyogi, 2003; Smola and Kondor, 2003; Zhou et al., 2005). More recently, this idea has been employed in graph-constrained estimation for variable selection in regression (Li and Li, 2008, 2010; Slawski et al., 2010; Pan et al., 2010; Shen et al., 2012; Zhu et al., 2013; Sun et al., 2014; Liu et al., 2019), principal component analysis (Shojaie and Michailidis, 2010), and regression inference (Zhao and Shojaie, 2016). In these problems, a network is assumed to connect a set of random variables or predictors and is used to achieve more effective variable selection or dimension reduction in high-dimensional settings. A generalization to potentially unknown network or group structure was studied by Witten et al. (2014). Though Step 1 of Algorithm 1 has multiple connections to graph constrained estimation, there are a few key differences. In our setting, the network is connecting observations, not variables. We only rely on smoothness across the network for accurate estimation without additional structural assumptions such as sparsity on M. In graph-constrained estimation literature, in addition to the Laplacian penalty, other penalties are proposed in special contexts (Slawski et al., 2010; Pan et al., 2010; Shen et al., 2012). We believe similar extensions can also be made in our problem for special applications and we will leave such extensions for future investigation.

An alternative penalty we can impose on M instead of $\sum_{i\sim i'} \|\mu_i - \mu_{i'}\|^2$ is

$$\sum_{i\sim i'} \|\mu_i - \mu_{i'}\|. \tag{9}$$

This penalty is called the network lasso penalty (Hallac et al., 2015) and can be viewed as a generalization of the fused lasso (Tibshirani et al., 2005) and the group lasso (Yuan and Lin, 2006). The penalty and its variants were studied recently by Wang et al. (2014); Jung et al. (2018); Tran et al. (2018). This penalty is also associated with convex clustering (Hocking et al., 2011; Lindsten et al., 2011), because it typically produces piecewise constant estimates which can be interpreted as clusters. Properties of convex clustering have been studied by Hallac et al. (2015); Chi and Lange (2015); Tan and Witten (2015). However, in our setting there are two clear reasons for using the Laplacian penalty and not the network lasso. First, piecewise constant individual effects within latent clusters of the network is a special case of the general cohesive individual effects, so our assumption is strictly weaker, and there is no reason to impose piecewise constant clusters in the mean unless there is prior knowledge. Second, solving the optimization in the network lasso problem is computationally challenging and not scalable to the best our knowledge: current state of art algorithms (Hallac et al., 2015; Chi and Lange, 2015) hardly handle more than about 200 nodes on a single core. In contrast, the Laplacian penalty in (6) admits a closed-form solution and can be efficiently solved for thousands of observations even with a naive implementation on a single machine. Moreover, there are many ways to improve the naive algorithm based on the special properties of the linear system (Spielman, 2010; Koutis et al., 2010; Cohen et al., 2014; Sadhanala et al., 2016; Li et al., 2019). Therefore, (6) is a better choice than (9) for this problem, both computationally and conceptually.

3. Theoretical properties

In this section, we investigate the theoretical properties of the two-stage GNC-lasso estimator. Throughout this section, we assume the observation network A is connected which implies that $\mathcal{L}_s$ has exactly one zero eigenvalue. The results can be trivially extended to a network consisting of several connected components, either by assuming the same conditions for each component or regularizing A to be connected as in Amini et al. (2013). Recall that $\tau_1 \geq \tau_2 \geq \cdots \geq \tau_{n-1} > \tau_n = 0$ are the eigenvalues of $\mathcal{L}_s$ corresponding to eigenvectors $u_1, \cdots, u_n$. For a connected network, we know τ_n is the only zero eigenvalue. Moreover,τ_{n-1} is known as *algebraic connectivity* that measure the connectivity of the network.

3.1. Cohesion assumptions on the observation network

The first question we have to address is how to formalize the intuitive notion of cohesion. We will start with the most intuitive definition of network cohesion for a vector, extend it to a matrix, and then give examples satisfying the cohesion assumptions.

Intuitively, we can think of a vector $v \in \mathbb{R}^n$ as cohesive on a network A if $v^T\mathcal{L}_s v$ is small in some sense, or equivalently, $\|\mathcal{L}_s v\|_2$ is small, since $\mathcal{L}_s v$ is the gradient of $v^T\mathcal{L}_s v$ up to a constant and

$$\|\mathcal{L}_s v\|_2 \to 0 \iff v^T\mathcal{L}_s v \to 0.$$

It will be convenient to define cohesion in terms of $\mathcal{L}_s v$, which also leads to a nice interpretation. The ith coordinate of $\mathcal{L}_s v$ can be written as

$$\frac{d_i}{\bar{d}}\left(v_i - \frac{1}{d_i}\sum_{i' \sim_A i} v_i'\right),$$

which is the difference between the value at node i and the average value of its neighbors, weighed by the degree of node i. Let $\mathcal{L}_s = U\Lambda U^T$ be the eigen-decomposition of $\mathcal{L}_s$, with Λ the diagonal matrix with the eigenvalues $\tau_1 \geq \cdots \geq \tau_n$ on the diagonal. The vector v can be expanded in this basis as $v = U\beta = \sum_{i=1}^n \beta_i v_i$ where $\beta \in \mathbb{R}^n$. Under cohesion, we would expect $\|\mathcal{L}_s v\|_2^2 = \sum_i \tau_i^2 \beta_i^2$ to be much smaller than $\|v\|_2^2 = \|\beta\|_2^2$. We formalize this in the following definition.

Definition 1 (A network-cohesive vector) *Given a network A and a vector v, let $v = \sum_{i=1}^n \beta_i u_i$ be the expansion of v in the basis of eigenvectors of $\mathcal{L}_s$. We say v is cohesive on A with rate $\delta > 0$ if for all $i = 1, \ldots, n$,*

$$\frac{\tau_i^2 |\beta_i|^2}{\|\beta\|_2^2} \leq n^{-\frac{2(1+\delta)}{3}-1}, \tag{10}$$

which implies

$$\frac{\|\mathcal{L}_s v\|_2^2}{\|v\|_2^2} \leq n^{-\frac{2(1+\delta)}{3}}.$$

Now we can easily define a network-cohesive matrix M.

Definition 2 (A network-cohesive matrix) *A matrix $M \in \mathbb{R}^{n\times p}$ is cohesive on a network A if all of its columns are cohesive on A.*

An obvious but trivial example of a cohesive vector is a constant vector, which corresponds to $\delta = \infty$. More generally, we define the class of *trivially cohesive* vectors as follows.

Definition 3 (A trivially cohesive vector) *We say vector v is trivially cohesive if*

$$\widehat{\mathrm{Var}}(v) = o(\bar{v}^2)$$

where $\bar{v} = \sum_{i=1}^n v_i/n$ is the sample mean of v, and $\widehat{\mathrm{Var}}(v) = \sum_{i=1}^n (v_i - \bar{v})^2/(n-1)$ is the sample variance of v.

Trivial cohesion does not involve a specific network A, because such vectors are essentially constant. We say v is nontrivially cohesive if it is cohesive but not trivially cohesive. Similarly, we will say a matrix is trivially cohesive if all its columns are trivially cohesive, and nontrivially cohesive if it is cohesive but not trivially cohesive.

For obtaining theoretical guarantees, we will need to make an additional assumption about the network, which essentially quantifies how much network structure can be used to control model complexity under nontrivial cohesion. This will be quantified through the concept of effective dimension of the network defined below.

Definition 4 *Given a connected network adjacency matrix A of size $n \times n$ and eigenvalues of its standardized Laplacian $\tau_1 \geq \dots \tau_{n-1} > \tau_n = 0$, define the* ***effective dimension*** *of the network as*

$$m_A = \inf\{m : 0 \leq m \leq n-1, \tau_{n-m} \geq \frac{1}{\sqrt{m}}\}.$$

Note that spectral graph theory (Brouwer and Haemers, 2011) implies $\tau_1 \geq c$ for some constant c, and thus for sufficiently large n, we always have $m_A \leq n-1$. For many sparse and/or structured networks the effective dimension is much smaller than $n-1$, and then we can show nontrivially cohesive vectors/matrices exist.

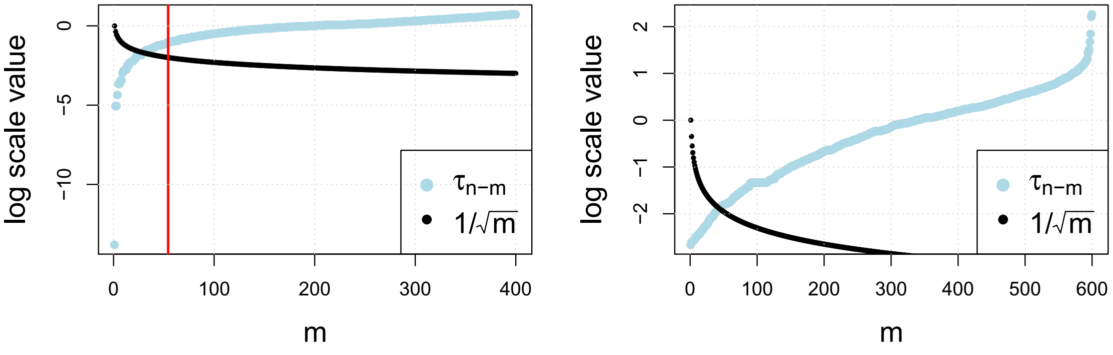
Our first example of a network with a small effective dimension is a lattice network. Assume $\sqrt{n}$ is an integer and define the lattice network of n nodes by arranging them on a $\sqrt{n} \times \sqrt{n}$ spatial grid and connecting grid neighbors (the four corner nodes have degree 2, nodes along the edges of the lattice have degree 3, and all internal nodes have degree 4).

Proposition 1 (Cohesion on a lattice network) *Assume A is a lattice network on n nodes, and $\sqrt{n}$ is an integer. Then for a sufficiently large n,*

1. *The effective dimension $m_A \leq n^{2/3}$.*

2. *There exist nontrivially cohesive vectors on the lattice network with rate $\delta = 1/2$.*

Figure 1 shows the eigenvalues and the function $1/\sqrt{m}$ for reference of a 20×20 lattice and of a coauthorship network we analyze in Section 5. For both networks, the effective dimension is much smaller than the number of nodes: for the lattice, $n = 400$, while $m_A = 30$ and for the coauthorship network with $n = 635$ nodes, $m_A = 66$.



(a) 20×20 lattice

(b) Coauthorship network

Figure 1: Eigenvalues and effective dimensions of a 20×20 lattice and the coauthorship networks from Section 5. The red vertical line in the left panel is $n^{2/3}$, the theoretical upper bound from Proposition 1.

3.2. Mean estimation error bound

Our goal here is to obtain a bound on the difference between M and the estimated $\hat{M}$ obtained by Algorithm 1, under the following cohesion assumption.

Assumption 1 *The mean matrix M is cohesive over the network A with rate δ where δ is a positive constant. Moreover, $\|M_{\cdot j}\|_2^2 \le b^2 n$ for every $j \in [p]$ for some positive constant b.*

Theorem 1 (Mean error bound) *Assume model* (2) *and Assumption 1 are true. Write $\sigma^2 = \max_j \Sigma_{jj}$ and $\Delta = n^{\frac{1+\delta}{3}}\tau_{n-1}$, where τ_{n-1} is the smallest nonzero eigenvalue of $\mathcal{L}_s$. Then $\hat{M}$ estimated by* (4) *with $\alpha = n^{\frac{1+\delta}{3}}$ satisfies*

$$\frac{\|\hat{M} - M\|_F^2}{np} \le \frac{(b^2 + 2\sigma^2)[1 + m_A(\frac{1}{(1+\Delta)^2} + n^{\frac{1-2\delta}{3}})]}{n} \tag{11}$$

with probability at least $1 - \exp(-c(n - m_A)r(\Sigma)) - \exp(-cm_A r(\Sigma)) - \exp(-c\frac{p\sigma^2}{\phi_{max}(\Sigma)})$ for some positive constant c, where m_A is the effective dimension of network A in Definition 4 and $r(\Sigma)$ is the stable rank of Σ.

The theorem shows that the average estimation error is vanishing with high probability as long as the cohesive dimension $m_A = o(n^{\frac{2(1+\delta)}{3}})$ while $m_A r(\Sigma)$ and $p/\phi_{max}(\Sigma)$ grow with n. Except for degenerate situations, we would expect $r(\Sigma)$ and $p/\phi_{max}(\Sigma)$ to grow with p, which in turn grows with n. In (11), the term $\Delta = n^{(1+\delta)/3}\tau_{n-1}$ involves both the cohesion rate of the mean matrix and the algebraic connectivity of the network. In trivially cohesive settings, $\delta \to \infty$ and $\Delta \to \infty$ so the bound does not depend on the network, and the error bound becomes the standard mean estimation error bound. General lower bounds for τ_{n-1} are available (Fiedler, 1973), but we prefer not to introduce additional algebraic definitions at this point.

Finally, note that the value of α depends on the cohesive rate δ of M. Therefore, the theorem is not adaptive to the unknown cohesive rate. In practice, as we discussed, one has to use cross-validation to tune δ.

3.3. Inverse covariance estimation error bounds

Our next step is to show that $\hat{M}$ is a sufficiently accurate estimate of M to guarantee good properties of the estimated precision matrix Θ in step 2 of the two-stage GNC-lasso algorithm. We will need some additional assumptions, the same ones needed for the glasso performance guarantees under the standard Gaussian graphical model (Rothman et al., 2008; Ravikumar et al., 2011).

Let $\Gamma = \Sigma \otimes \Sigma$ be the Fisher information matrix of the model, where $\otimes$ denotes the Kronecker product. In particular, under the multivariate Gaussian distribution, we have $\Gamma_{(j,k),(\ell,m)} = \mathrm{Cov}(X_j X_k, X_\ell X_m)$. Define the set of nonzero entries of Θ as

$$S(\Theta) = \{(j, j') \in [n] \times [n] : \Theta_{jj'} \ne 0\}. \tag{12}$$

We use $S^c(\Theta)$ to denote the complement of $S(\Theta)$. Let $s = |S(\Theta)|$ be the number of nonzero elements in Θ. Recall that we assume all diagonals of Θ are nonzero. For any two sets $T_1, T_2 \subset [n]$, let Γ_{T_1,T_2} denote the submatrix with rows and columns indexed by T_1, T_2, respectively. When the context is clear, we may simply write S for $S(\Theta)$. Define

$$\begin{aligned} \psi &= \max_j \|\Theta_{j\cdot}\|_0, \\ \kappa_\Sigma &= \|\Sigma\|_{\infty,\infty}, \\ \kappa_\Gamma &= \|(\Gamma_{SS})^{-1}\|_{\infty,\infty} \end{aligned}$$

where the vector operator $\|\cdot\|_0$ gives the number of nonzeros in the vector while the matrix norm $\|\cdot\|_{\infty,\infty}$ gives the maximum L_∞ norm of the rows.

Finally, by analogy to the well-known irrepresentability condition for the lasso, which is necessary and sufficient for the lasso to recover support (Wainwright, 2009), we need an edge-level irrepresentability condition.

Assumption 2 *There exists some $0 < \rho \le 1$ such that*

$$\max_{e \in S^c} \|\Gamma_{eS}(\Gamma_{SS})^{-1}\|_1 \le 1 - \rho.$$

If we only want to obtain a Frobenius norm error bound, the following much weaker assumption is sufficient, without conditions on $\psi, \kappa_\Sigma, \kappa_\Gamma$ and Assumption 2:

Assumption 3 *Let $\eta_{\min}(\Sigma)$ and $\eta_{\max}(\Sigma)$ be the minimum and maximum eigenvalues of Σ, respectively. There exists a constant $\bar{k}$ such that*

$$\frac{1}{\bar{k}} \le \eta_{\min}(\Sigma) \le \eta_{\max}(\Sigma) \le \bar{k}.$$

Let $\hat{S} = \frac{1}{n}(X - \hat{M})^T(X - \hat{M})$. We use $\hat{S}$ as input for the glasso estimator (5). We would expect that if $\hat{M}$ is an accurate estimate of M, then Θ can be accurately estimated by glasso. The following theorem formalizes this intuition, using concentration properties of $\hat{S}$ around Σ and the proof strategy of Ravikumar et al. (2011). We present the high-dimensional regime result here, with $p \ge n^{c_0}$ for some positive constant c_0, and state the more general result which includes the lower-dimensional regime in Theorem 3 in the Appendix, because the general form is more involved.

Theorem 2 *Under the conditions of Theorem 1 and Assumption 2, suppose there exists some positive constant c_0 such that $p \ge n^{c_0}$. If $\log p = o(n)$ and $m_A = o(n)$, there exist some positive constants C, c, c', c'' that only depend on c_0, b and σ, such that if $\hat{\Theta}$ is the output of Algorithm 1 with $\alpha = n^{\frac{1+\delta}{3}}$, $\lambda = \frac{8}{\rho}\nu(n,p)$ where*

$$\nu(n,p) := C\sqrt{\frac{\log p}{n}} \max\left(1, m_A n^{-\frac{1+4\delta}{6}}, \sqrt{m_A} n^{\frac{1-2\delta}{6}}, \sqrt{\frac{\log p}{n}}(\frac{m_A}{\Delta+1}+1)(\sqrt{m_A}n^{-\frac{1+\delta}{3}}+1)\right) \tag{13}$$

and n sufficiently large so that

$$\nu(n,p) < \frac{1}{6(1+8/\rho)\psi \max\{\kappa_\Sigma\kappa_\Gamma, (1+8/\rho)\kappa_\Sigma^3\kappa_\Gamma^2\}},$$

then with probability at least $1 - \exp(-c\log(p(n-m_A))) - \exp(-c'\log(pm_A)) - \exp(-c''\log p)$, then the estimate $\hat{\Theta}$ has the following properties:

1. *Error bounds:*

$$\begin{aligned}\|\hat{\Theta}-\Theta\|_\infty &\leq 2(1+8/\rho)\kappa_\Gamma \nu(n,p)\\ \|\hat{\Theta}-\Theta\|_F &\leq 2(1+8/\rho)\kappa_\Gamma \nu(n,p)\sqrt{s+p}.\\ \|\hat{\Theta}-\Theta\| &\leq 2(1+8/\rho)\kappa_\Gamma \nu(n,p)\min(\sqrt{s+p},\psi).\end{aligned}$$

2. *Support recovery:*

$$S(\hat{\Theta}) \subset S(\Theta),$$

and if additionally $\min_{(j,j')\in S(\Theta)} |\Theta_{jj'}| > 2(1+8/\rho)\kappa_\Gamma \nu(n,p)$, *then*

$$S(\hat{\Theta}) = S(\Theta).$$

Remark 1 *As commonly assumed in literature, such as Ravikumar et al. (2011), we will treat* κ_Γ, κ_Σ *and* ρ *to be constants or bounded.*

Remark 2 *The Frobenius norm bound does not need the strong irrepresentability assumption and does not depend on* κ_Γ *and* κ_Σ. *Following the proof strategy in Rothman et al. (2008), this bound can be obtained under the much weaker Assumption 3 instead.*

The quantity in (13) involves four terms. The first term is from the inverse covariance estimation with a known M (a standard glasso problem), and the other three terms come from having to estimate a cohesive M. These three terms depend on both the cohesion rate and the effective dimension of the network. As expected, they all increase with m_A and decrease with δ. The last term also involves Δ, which depends on both δ and the algebraic connectivity τ_{n-1}. To illustrate these trade-offs, we consider the implications of Theorem 2 in a few special settings.

First, consider the setting of trivial cohesion, with $\delta = \infty$. In this case, the last three terms in (13) vanish.

Corollary 1 *Under the assumptions of Theorem 2, if* M *is trivially cohesive and* $\delta = \infty$, *then all results of Theorem 2 hold with*

$$\nu(n,p) = C\sqrt{\frac{\log p}{n}},$$

and the estimated $\hat{\Theta}$ *is consistent as long as* $\log p = o(n)$.

This result coincides with the standard glasso error bound from Ravikumar et al. (2011). Thus when M does not vary, we do not lose anything in the rate by using GNC-lasso instead of glasso.

Another illustrative setting is the case of bounded effective dimension m_A. Then the third term in (13) dominates.

Corollary 2 *Under the assumptions of Theorem 2, if the network has a bounded effective dimension* m_A, *then all the results of Theorem 2 hold with*

$$\nu(n,p) = C\sqrt{\frac{\log p}{n} n^{\frac{\max(1-2\delta,0)}{6}}}.$$

In particular, if $\delta \geq 1/2$, $\hat{\Theta}$ *is consistent as long as* $\log p = o(n)$.

This corollary indicates that if the network structure is favorable to cohesion, the GNC-lasso does not sacrifice anything in the rate up to a certain level of nontrivial cohesion.

Finally, consider a less favorable example in which $\log p = o(n)$ may no longer be enough for consistency. Recall Proposition 1 indicates $m_A = O(n^{2/3})$ for lattice networks, and suppose the cohesive can be highly nontrivial.

Corollary 3 (Consistency on a $\sqrt{n} \times \sqrt{n}$ lattice) *Suppose the conditions of Theorem 2 hold and $m_A \leq n^{2/3}$. The GNC-lasso estimate $\hat{\Theta}$ is consistent if $\delta > 3/8$ and*

$$\log p = o(n^{\min(1, 8\delta - 3)/3}).$$

In particular, if $\delta = 1/2$, it is necessary to have $\log p = o(n^{1/3})$ for consistency.

The corollary suggests that consistency under some regimes of nontrivial cohesion requires strictly stronger conditions than $\log p = o(n)$. Moreover, if cohesion is too weak (say, $\delta \leq 3/8$), consistency cannot be guaranteed by these results.

4. Simulation studies

We evaluate the new GNC-lasso method and compare it to some baseline alternative methods in simulations based on both synthetic and real networks. The synthetic network we use is a 20×20 lattice network with $n = 400$ nodes and a vector with dimension $p = 500$ observed at each node; this setting satisfies the assumptions made in our theoretical analysis. We also test our method on the coauthorship network shown in Figure 8, which will be described in Section 5. This network has $n = 635$ nodes at $p = 800$ observed features at each node.

Noise settings: The conditional dependence graph $\mathcal{G}$ in the Gaussian graphical model is generated as an Erdös-Renyi graph on p nodes, with each node pair connecting independently with probability 0.01. The Gaussian noise is then drawn from $\mathcal{N}(0, \Sigma)$ where $\Theta = \Sigma^{-1} = a(0.3A_{\mathcal{G}} + (0.3e_{\mathcal{G}} + 0.1)I)$, where $A_{\mathcal{G}}$ is the adjacency matrix of $\mathcal{G}$,$e_{\mathcal{G}}$ is the absolute value of the smallest eigenvalue of $A_{\mathcal{G}}$ and the scalar a is set to ensure the resulting Σ has all diagonal elements equal to 1. This procedure is implemented in Zhao et al. (2012).

Mean settings: We set up the mean to allow for varying degrees of cohesion. each row $M_{\cdot j}, j = 1, 2, \cdots, p$ as

$$M_{\cdot,j} = \sqrt{t}\sqrt{n}u^{(j)} + \sqrt{1-t}\mathbf{1} \tag{14}$$

where $u^{(j)}$ is randomly sampled with replacement from the eigenvectors of the Laplacian $u_{n-1}, n_{n-2}, \cdots, u_{n-k}$ for some integer k and t is the mixing proportion. We then rescale M so the signal-to-noise ratio becomes 1.6, so that the problem remains solvable to good accuracy by proper methods but is not too easy to solve by naive methods. In a connected network, the constant vector is trivially cohesive. The cohesion becomes increasingly nontrivial as one increases k and t. For example, $t = 0$ gives identical mean vectors for all observations and as t increases, the means become more different. The integer k is chosen to give a reasonably eigen-gap in eigenvalues, with details in subsequent paragraphs.

We evaluate performance on recovering the true underlying graph by the receiver operating characteristic (ROC) curve, along a graph estimation path obtained by varying λ.

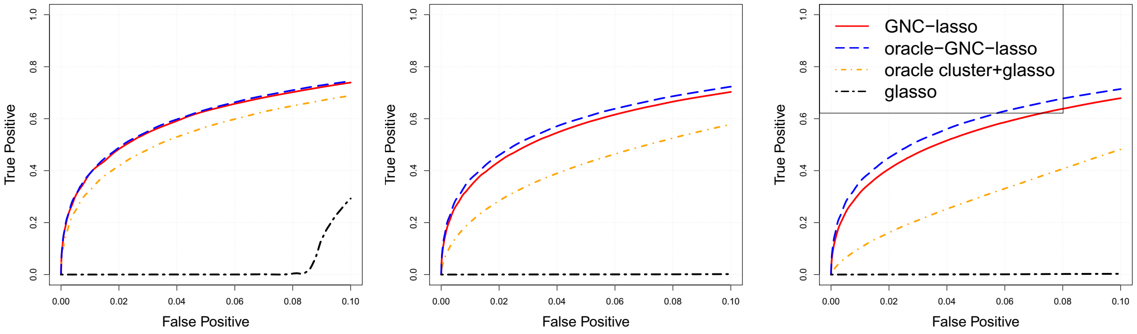
An ROC curve illustrates the tradeoff between the true positive rate (TPR) and the false positive rate (FPR), defined as

$$\text{TPR} = \frac{\#\{(j,j') : j \neq j', \Theta_{jj'} \neq 0, \hat{\Theta}_{jj'} \neq 0\}}{\#\{(j,j') : j \neq j', \Theta_{jj'} \neq 0\}}$$
$$\text{FPR} = \frac{\#\{(j,j') : j \neq j', \Theta_{jj'} = 0, \hat{\Theta}_{jj'} \neq 0\}}{\#\{(j,j') : j \neq j', \Theta_{jj'} = 0\}}.$$

We also evaluate the methods on the estimation error of M, measured as $\|\hat{M} - M\|_\infty = \max_{ij} |\hat{M}_{ij} - M_{ij}|$ for the worst-case entry-wise recovery and $\|\hat{M} - M\|_{2,\infty} = \max_i \|\hat{M}_{i\cdot} - M_{i\cdot}\|$ for the worst-case mean vector error for each observation.

As a baseline comparison, we include the standard glasso which does not use the network information at all. We also compare to a natural approach to incorporating heterogeneity without using the network; we do this by applying K-means clustering to group observations into clusters, estimating a common mean for each cluster, and applying glasso after centering each group with its own mean. This approach requires estimating the number of clusters. However, the widely used gap method (Tibshirani et al., 2001) always suggests only one cluster in our experiments, which defaults back to glasso. Instead, we picked the number of clusters to give the highest area under the ROC curve; we call this result "oracle cluster+glasso" to emphasize that it will not be feasible in practice. For GNC-lasso, we report both the oracle tuning (the highest AUC, not available in practice) and 10-fold cross-validation based tuning, which we recommend in practice. The oracle methods serve as benchmarks for the best possible performance available from each method.

4.1. Performance as a function of cohesion

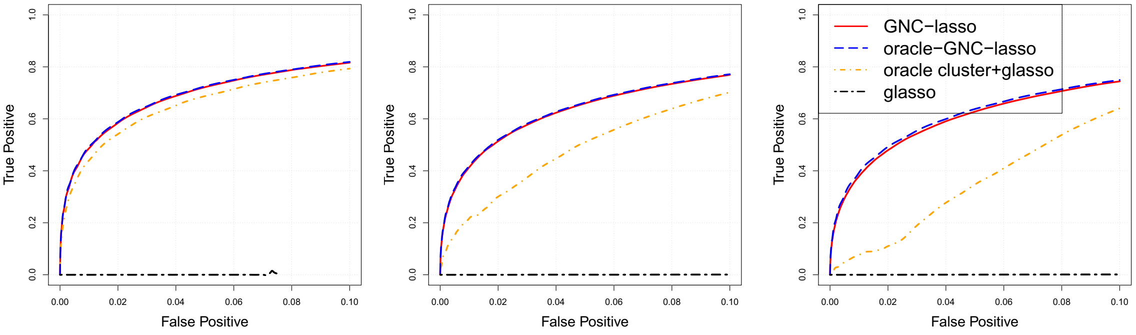


(a) $t = 0.1$ (b) $t = 0.5$ (c) $t = 1$

Figure 2: Graph recovery ROC curves under three different levels of cohesion corresponding to $t = 0.1, 0.5, 1$, for the lattice network ($n = 400$, $p = 500$).

First, we vary the level of cohesion in the mean, by setting t to 0.1, 0.5, or 1, corresponding to strong, moderate, or weak cohesion. Figure 2 shows the ROC curves of the four methods obtained from 100 independent replications for the lattice network. Glasso fails completely even when the model has only a slight amount of heterogeneity ($t = 0.1$). Numerically, we also observed that heterogeneity slows down convergence for glasso. The oracle cluster+glasso improves on glasso as it can accommodate some heterogeneity, but is not comparable to GNC-lasso. As t increases, the GNC-lasso maintains similar levels of performance by adapting to varying heterogeneity, while the oracle cluster+glasso degrades quickly, since for more heterogeneous means the network provides much more reliable information than K-means clustering on the observations. We also observed that cross-validation is similar to oracle tuning for GNC-lasso, giving it another advantage. Figure 3 shows the results for the same setting but on the real coauthorship network instead of the lattice. The results are very similar to what we obtained on the lattice, giving further support to GNC-lasso practical relevance.

We also compare estimation errors in $\hat{M}$ in Table 1. The oracle GNC-lasso is almost always the best, except for one setting where it is inferior to the CV-tuned GNC-lasso (note that the "oracle" is defined by the AUC and is thus not guaranteed to produce the lowest error in estimating M). For the lattice network, cluster + glasso does comparably to GNC-lasso (sometimes better, and sometimes worse). For the coauthorship network, a more realistic setting, GNC-lasso is always comparable to the oracle and substantially better than both alternatives that do not use the network information.



(a) $t = 0.1$

(b) $t = 0.5$

(c) $t = 1$

Figure 3: Graph recovery ROC curves under three different levels of cohesion corresponding to $t = 0.1, 0.5, 1$, for the coauthorship network ($n = 635$, $p = 800$).

4.2. Performance as a function of sparsity

A potential challenge for GNC-lasso is a sparse network that does not provide much information, and in particular a network with multiple connected components. As a simple test of what happens when a network has multiple components, we split the 20×20 lattice into either four disconnected 10×10 lattice subnetworks, or 16 disconnected 5×5 lattice subnetworks, by removing all edges between these subnetworks. The data size ($n = 400, p = 500$) and the data generating mechanism remain the same; we set $t = 0.5$ for a moderate degree of cohesion. The only difference here is when there are K connected components in the network, the last K eigenvectors of the Laplacian $u_n, \cdots, u_{n-K+1}$ are all constant within each

Table 1: Mean estimation errors for the four methods, averaged over 100 replications, with the lowest error in each configuration indicated in bold.

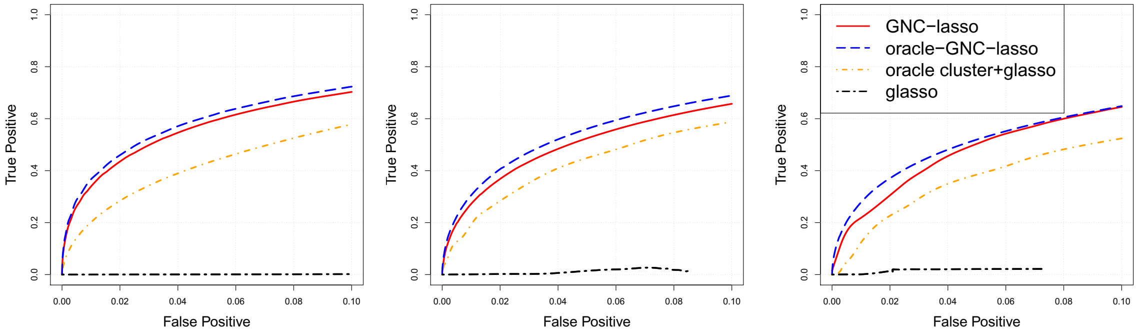
		$\|\hat{M} - M\|_\infty$			$\|\hat{M} - M\|_{2,\infty}$		
network	method	$t = 0.1$	0.5	1	$t = 0.1$	0.5	1
lattice	glasso	0.358	0.746	1.037	5.819	12.985	18.357
	oracle cluster+glasso	0.493	0.539	0.565	3.293	3.639	4.118
	oracle GNC-lasso	**0.328**	**0.520**	**0.526**	**2.054**	**3.247**	**3.287**
	GNC-lasso	0.419	0.669	0.820	2.619	4.105	4.874
coauthorship	glasso	1.540	3.401	4.795	25.072	57.655	78.426
	oracle cluster+glasso	0.724	1.077	1.342	7.051	13.078	16.962
	oracle GNC-lasso	**0.710**	**0.860**	**0.917**	**6.400**	7.404	**7.420**
	GNC-lasso	0.717	0.878	0.942	6.430	**7.037**	7.436

connected component (and thus trivially cohesive). Therefore, in the case of 4 disconnected subnetworks, we randomly sample the last $k = 12$ eigenvectors to generate M in (14) while in the case of 16 disconnected subnetworks, we set $k = 48$. The effective dimensions m_A are 30, 32, and 48, respectively.

Similarly, we also split the coauthorship network into two or four subnetworks by applying hierarchical clustering in Li et al. (2018), which is designed to separate high-level network communities (if they exist). We then remove all edges between the communities found by clustering to produce a network with either two or four connected components. To generate M from (14), we use $k = 6$ for two components and $k = 12$ for four components, and again set $t = 0.5$ for moderate cohesion. The effective dimension m_A becomes 66, 74, and 78, respectively.

Figure 4 shows the ROC curves and Table 2 shows the mean estimation errors for the three versions of the lattice network. Overall, all methods get worse as the network is split, but the drop in performance is fairly small for the oracle GNC-lasso. Cross-validated GNC lasso suffers slightly more from splitting (the connected components in the last case only have 25 nodes each, which can produce isolated nodes and hurt cross-validation performance). Again, both GNS methods are much more accurate than the two benchmarks (glasso completely fails, and oracle cluster+glasso performance substantially worse).

Figure 5 and Table 3 give the results for the three versions of the coauthorship network. The network remains well connected in all configurations and both the oracle and the cross-validated GNC-lasso perform well in all three cases, without deterioration. The oracle cluster+glasso performs well in this case as well, but GNC-lasso still does better on both graph recovery and estimating the mean. Glasso fails completely once again.

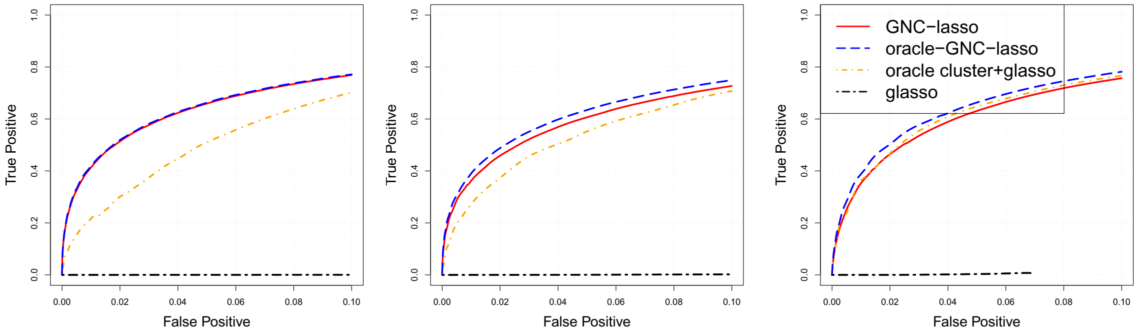


(a) Original 20 × 20 lattice. (b) Four connected components. (c) 16 connected components.

Figure 4: Graph recovery ROC curves for the lattice network and two of its sparsified variants. Here $n = 400, p = 500$ and we set $t = 0.5$ in generating M.

Table 2: Mean estimation errors for the four methods, averaged over 100 replications, with the lowest error in each configuration indicated in bold, for the lattice networks with one, four, and 16 connected components.

	$\|\hat{M} - M\|_\infty$			$\|\hat{M} - M\|_{2,\infty}$		
method	original	4 comp.	16 comp.	original	4 comp.	16 comp.
glasso	0.746	2.801	2.942	5.819	40.25	25.16
oracle cluster+glasso	0.539	1.091	1.099	3.293	12.20	8.22
oracle GNC-lasso	**0.520**	**0.866**	**0.785**	**2.054**	**6.46**	**5.13**
GNC-lasso	0.669	0.983	0.838	2.619	6.73	5.79



(a) Original coauthor-network (b) 2 connected components. (c) 4 connected components.

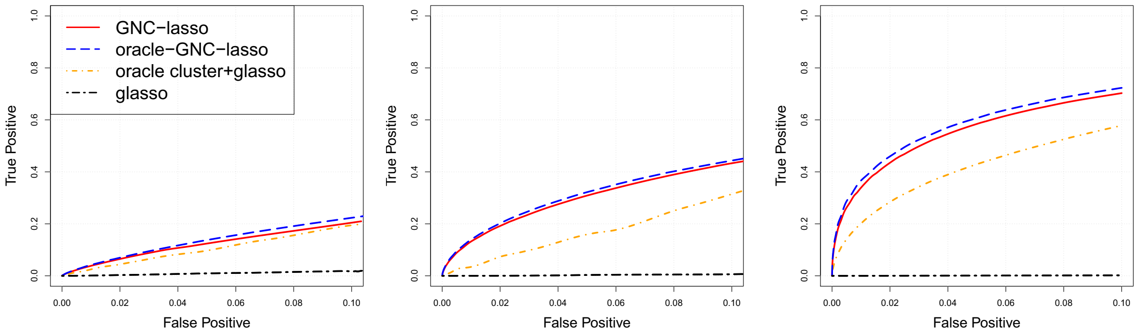
Figure 5: Graph recovery ROC curves for the coauthorship network and two of its sparsified variants. Here $n = 635, p = 800$ and we set $t = 0.5$ in generating M.

Table 3: Mean estimation errors for the four methods, averaged over 100 replications, with the lowest error in each configuration indicated in bold, for the coauthorship networks with one, two, or four components.

	$\|\hat{M} - M\|_\infty$			$\|\hat{M} - M\|_{2,\infty}$		
method	original	2 comp.	4 comp.	original	2 comp.	4 comp.
glasso	0.746	1.949	4.019	5.819	21.214	32.307
oracle cluster+glasso	0.539	0.936	1.676	3.293	6.033	7.208
oracle GNC-lasso	**0.520**	**0.659**	**0.958**	**2.054**	**3.860**	**4.843**
GNC-lasso	0.669	0.852	1.289	2.619	4.947	5.102

4.3. Performance as a function of the sample size

Here we compare the methods when the sample size n changes while p remains fixed. Specifically, we compare 10×10, 15×15, and 20×20 lattices, corresponding to $n = 100$, 225, and 400, respectively. The dimension $p = 500$, the data generating mechanism, and $t = 0.5$ remain the same as in Section 4.1. When $n = 100$, the sample size is too small for 10-fold cross-validation to be stable, and thus we use leave-one-out cross-validation instead. Figure 6 shows the ROC curves while Table 4 shows errors in the mean. Clearly, the problem is more difficult for smaller sample sizes, but both versions of GNC-lasso still work better than the other two baseline methods, even though for $n = 100$, the problem is essentially too difficult for all the methods. Results on estimating the mean do not favor any one method clearly, but the differences between the methods are not very large in most cases.



(a) 10×10 lattice.

(b) 15×15 lattice.

(c) 20×20 lattice.

Figure 6: Graph recovery ROC curves for the three lattice networks with $n = 100$ (10×10), 225 (15×15) and 400 (20×20). We fix $p = 500$, $t = 0.5$.

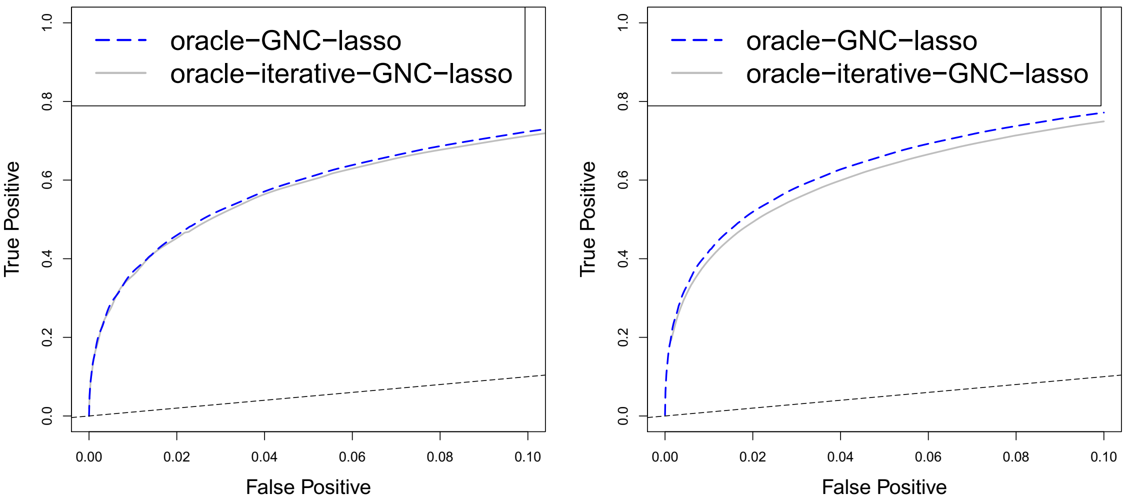
4.4. Comparing with the iterative GNC-lasso

Finally, we compare the estimator obtained by iteratively optimizing Θ and M in (8) (iterative GNC-lasso) to the proposed two-stage estimator (GNC-lasso). As mentioned in Section 2.4, the iterative method is too computationally intensive to tune by cross-validation, so we only compare the oracle versions of both methods, on the synthetic data used in Sec-

Table 4: The estimation errors of M from the four methods on three connected lattice networks with varying sample size, averaged over 100 independent replications. The network sizes are 100, 225 and 400, corresponding to lattice dimension 10 × 10, 15 × 15 and 20 × 20, respectively.

	$\|\hat{M}-M\|_\infty$			$\|\hat{M}-M\|_{2,\infty}$		
method	$n = 100$	225	400	100	225	400
glasso	1.009	1.030	0.746	15.02	15.69	12.985
oracle cluster+glasso	0.991	**0.716**	0.539	6.62	**4.72**	3.639
oracle GNC-lasso	0.911	0.794	**0.520**	5.52	4.83	**3.247**
GNC-lasso	**0.874**	0.988	0.669	**5.36**	5.67	4.105

tion 4.1 with moderate cohesion level $t = 0.5$. The results are shown in Figure 7 and Table 5. The methods are essentially identically on the lattice network and the two-stage method is in fact slightly better on the co-author network, indicating that there is no empirical reason to invest in the computationally intensive iterative method.



(a) 20 × 20 lattice

(b) Coauthorship network

Figure 7: Graph recovery ROC curves for the proposed two-stage GNC-lasso and the joint GNC-lasso. The network cohesion corresponding to $t = 0.5$ for the 20 × 20 lattice and the coauthorship network.

5. Data analysis: learning associations between statistical terms

Here we apply the proposed method to the dataset of papers from 2003-2012 from four statistical journals collected by Ji and Jin (2016). The dataset contains full bibliographical information for each paper and was curated for disambiguation of author names when necessary. Our goal is to learn a conditional dependence graph between terms in paper titles, with the aid of the coauthorship network.

Table 5: The estimation errors of M from the iterative and two-stage oracle GNC-lasso methods, averaged over 100 independent replications.

network	method	$\|\hat{M} - M\|_\infty$	$\|\hat{M} - M\|_{2,\infty}$
lattice	two-stage	0.520	3.247
	iterative	0.587	3.639
coauthorship	two-stage	0.860	7.404
	iterative	1.21	9.92

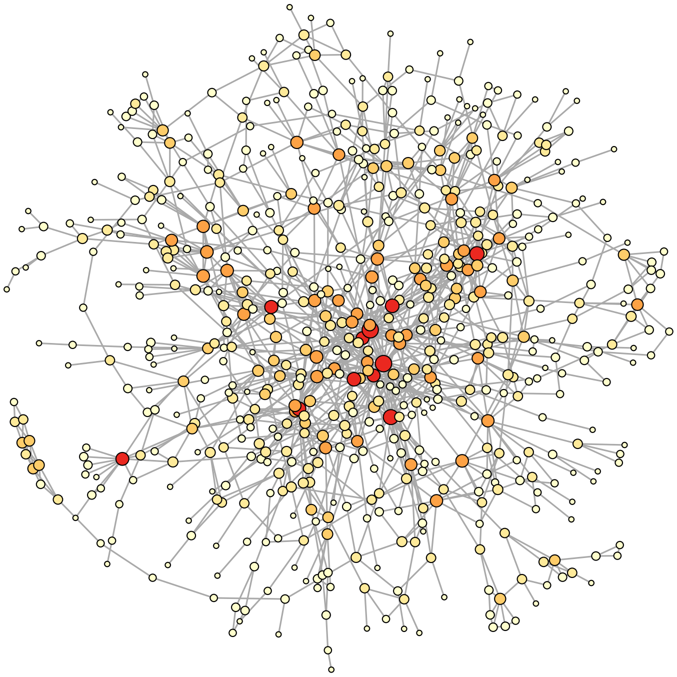


Figure 8: The coauthorship network of 635 statisticians (after pre-processing). The size and the color of each node correspond to the degree (larger and darker circles have more connections.

We pre-processed the data by removing authors who have only one paper in the data set, and filtering out common stop words ("and", "the", etc) as well as terms that appear in fewer than 10 paper titles. We then calculate each author's average term frequency across all papers for which he/she is a coauthor. Two authors are connected in the coauthorship network if they have co-authored at least one paper, and we focus on the largest connected component of the network. Finally, we sort the terms according to their term frequency-inverse document frequency score (tf-idf), one of the most commonly used measures in natural language processing to assess how informative a term is (Leskovec et al., 2014), and keep 300 terms with the highest tf-idf scores. After all pre-processing, we have $n = 635$ authors and $p = 300$ terms. The observations are 300-dimensional vectors recording the average frequency of term usage for a specific author. The coauthorship network is shown in Figure 8.

The interpretation in this setting is very natural; taking coauthorship into account makes sense in estimating the conditional graph, since the terms come from the shared paper title. We can expect that there will be standard phrases that are fairly universal (e.g., "confidence intervals"), as well as phrases specific to relatively small groups of authors with multiple connections, corresponding to specific research area (e.g., “principal components”), which is exactly the scenario where our model should be especially useful relative to the standard Gaussian graphical model. To ensure comparable scales for both columns and rows, we standardize the data using the successive normalization procedure introduced by Olshen and Rajaratnam (2010). If we select α using 10-fold cross-validation, as before, the graphs from GNC-lasso and glasso recover 4 and 6 edges, respectively, which are very sparse graphs. To keep the graphs comparable and to allow for more interpretable results, we instead set the number of edges to 25 for both methods, and compare resulting graphs, shown in = Figure 9 (glasso) and Figure 10 (GNC-glasso). For visualization purposes, we only plot the 55 terms that have at least one edge in at least one of the graphs.

Overall, most edges recovered by both methods represent common phrases in the statistics literature, including “exponential families”, “confidence intervals”, “measurement error”, “least absolute” (deviation), “probabilistic forecasting”, and “false discovery”. There are many more common phrases that are recovered by GNC-lasso but missed by Glasso, for example, “high dimension(al/s)”, “gene expression”, “covariance matri(x/ces)”, “partially linear”, “maximum likelihood”, “empirical likelihood”, “estimating equations”, “confidence bands”, “accelerated failure” (time model),“principal components” and “proportional hazards”. There are also a few that are found by Glasso but missed by GNC-lasso, for example, “moving average” and “computer experiments”. Some edges also seem like potential false positives, for example, the links between “computer experiments” and “orthogonal construction”, or the edge between “moving average” and “least absolute”, both found by glasso but not GNC-lasso.

Additional insights about the data can be drawn from the $\hat{M}$ matrix estimated by GNC-lasso; glasso does not provide any information about the means. Each $\hat{M}_{\cdot j}$ can be viewed as the vector of authors’ preferences for the term j, we can visualize the relative distances between terms as reflected in their popularity. Figure 11 shows the 55 terms from Figure 9, projected down from $\hat{M}$ to R^2 for visualization purposes by multidimensional scaling (MDS) (Mardia, 1978). The visualization shows a clearly outlying cluster, consisting of the terms “computer”, “experiments”, “construction”, and “orthogonal”, and to a lesser extent the cluster “Markov Chain Monte Carlo” is also further away from all the other terms. The clearly outlying group can be traced back to a single paper, with the title “Optimal and orthogonal Latin hypercube designs for computer experiments” (Butler, 2001), which is the only title where the words “orthogonal” and “experiments” appear together. Note that glasso estimated them as a connected component in the graph, whereas GNC-lasso did not, since it was able to separate a one-off combination occurring in a single paper from a common phrase. This illustrates the advantage of GNC-lasso’s ability to distinguish between individual variation in the mean vector and the overall dependence patterns, which glasso lacks.

6. Discussion

We have extended the standard graphical lasso problem and the corresponding estimation algorithm to the more general setting in which each observation can have its own mean vector. We studied the case of observations connected by a network and leveraged the empirically known phenomenon of network cohesion to share information across observations, so that we can still estimate the means in spite of having np mean parameters instead of just p in the standard setting. The main object of interest is the inverse covariance matrix, which is shared across observations and represents universal dependencies in the population. while all observations share the same covariance matrix under the assumption of network cohesion. The method is computationally efficient with theoretical guarantees on the estimated inverse covariance matrix and the corresponding graph. Both simulations and an application to a citation network show that GNC-lasso is more accurate and gives more insight into the structure of the data than the standard glasso when observations are connected by a network. One possible avenue for future work is obtaining inference results for the estimated model. This might be done by incorporating the inference idea of Zhao and Shojaie (2016) and Ren et al. (2015) with additional structural assumptions on the mean vectors. The absolute deviation penalty (Hallac et al., 2015) between connected nodes is a possible alternative, if the computational cost issue can be resolved through some efficient optimization approach. Another direction is to consider the case where the partial dependence graphs themselves differ for individuals over the network, but in a cohesive fashion; the case of jointly estimating several related graphs has been studied by Guo et al. (2011); Danaher et al. (2014). As always, in making the model more general there will be a trade-off between goodness of fit and parsimony, which may be elucidated by obtaining convergence rates in this setting.

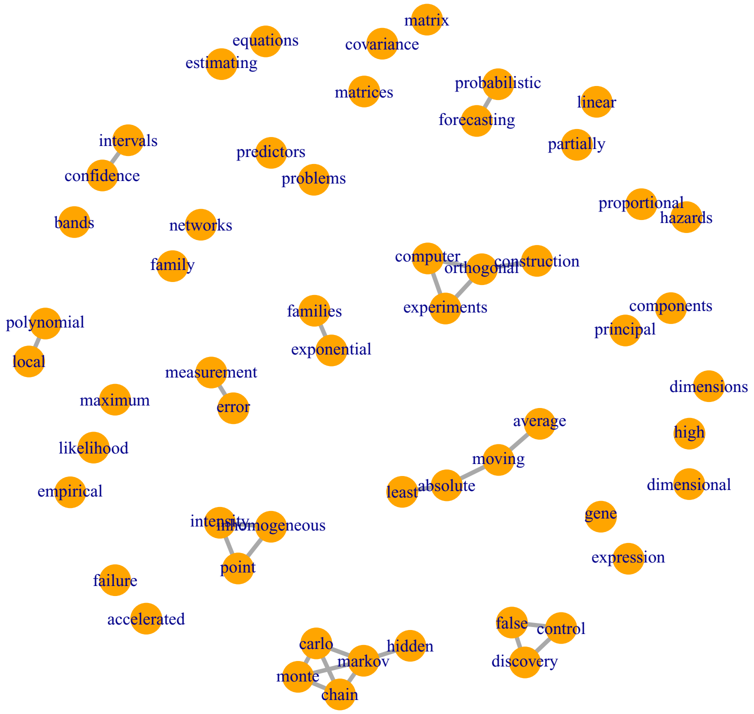


Figure 9: Partial correlation graphs estimated by Glasso

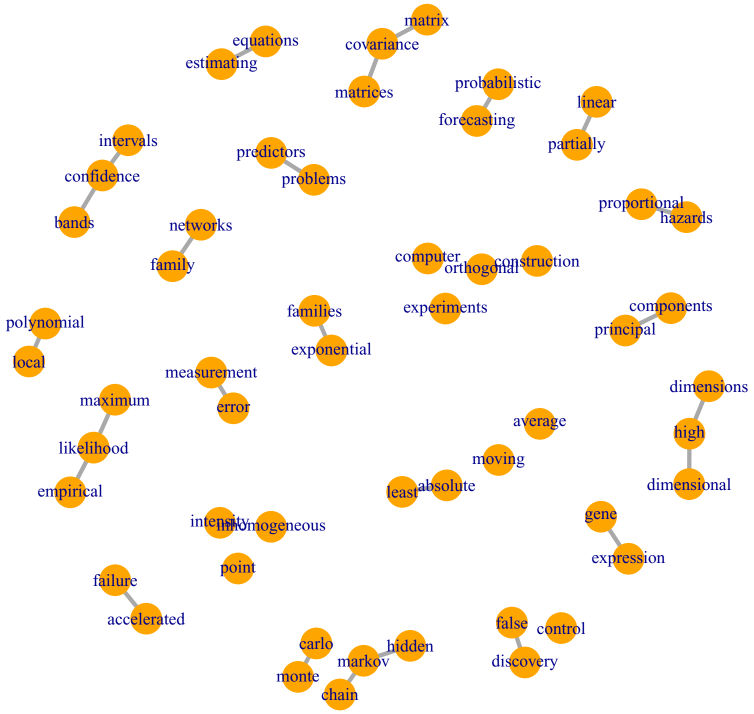


Figure 10: Partial correlation graphs estimated by GNC-lasso

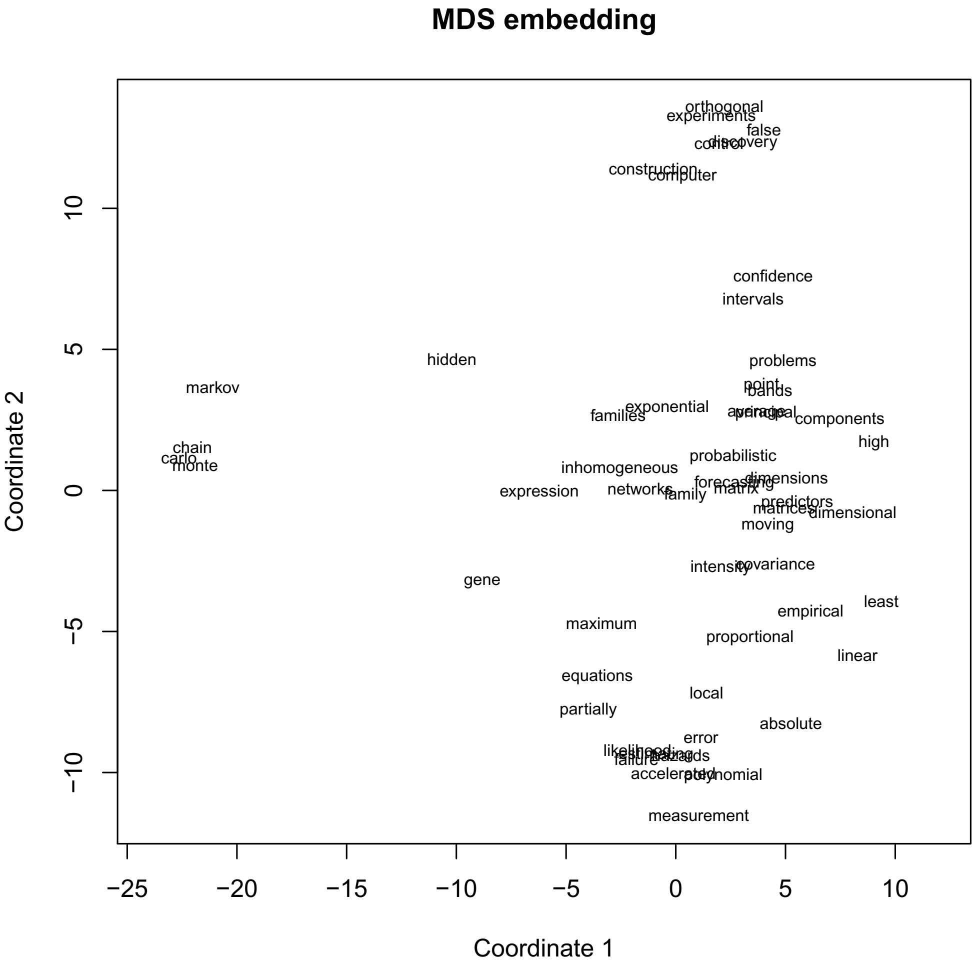


Figure 11: Projection of 55 terms by using the 2-D MDS.

Acknowledgments

T. Li was partially supported by the Quantitative Collaborative grant from the University of Viriginia. E. Levina and T. Li (while a PhD student at the University of Michigan) were supported in part by an ONR grant (N000141612910) and NSF grants (DMS-1521551 and DMS-1916222). J. Zhu and T. Li (while a PhD student at the University of Michigan) were supported in part by NSF grants (DMS-1407698 and DMS-1821243). C. Qian was supported in part by grants from National Key R&D Program of China (2018YFF0215500) and Science Foundation of Jiangsu Province for Young Scholars (SBK2019041494). We want to thank the action editor and reviewers for their valuable suggestions.

References

Arash A Amini, Aiyou Chen, Peter J Bickel, and Elizaveta Levina. Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics*, 41 (4):2097–2122, 2013.

Onureena Banerjee, Laurent El Ghaoui, and Alexandre d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research*, 9(Mar):485–516, 2008.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

N. Binkiewicz, J. T. Vogelstein, and K. Rohe. Covariate-assisted spectral clustering. *Biometrika*, 104(2):361–377, 2017. doi: 10.1093/biomet/asx008. URL `+http://dx.doi.org/10.1093/biomet/asx008`.

Andries E Brouwer and Willem H Haemers. *Spectra of graphs*. Springer Science & Business Media, 2011.

Neil A Butler. Optimal and orthogonal latin hypercube designs for computer experiments. *Biometrika*, pages 847–857, 2001.

T Tony Cai, Hongzhe Li, Weidong Liu, and Jichun Xie. Covariate-adjusted precision matrix estimation with an application in genetical genomics. *Biometrika*, 100(1):139–156, 2013.

Eric C Chi and Kenneth Lange. Splitting methods for convex clustering. *Journal of Computational and Graphical Statistics*, 24(4):994–1013, 2015.

Nicholas A Christakis and James H Fowler. The spread of obesity in a large social network over 32 years. *New England Journal of Medicine*, 357(4):370–379, 2007.

Michael B Cohen, Rasmus Kyng, Gary L Miller, Jakub W Pachocki, Richard Peng, Anup B Rao, and Shen Chen Xu. Solving sdd linear systems in nearly m log 1/2 n time. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 343–352. ACM, 2014.

Patrick Danaher, Pei Wang, and Daniela M Witten. The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(2):373–397, 2014.

Alexandre d'Aspremont, Onureena Banerjee, and Laurent El Ghaoui. First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and Applications*, 30 (1):56–66, 2008.

Thomas Edwards. The discrete laplacian of a rectangular grid, 2013.

Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23 (2):298–305, 1973.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008. doi: 10.1093/biostatistics/kxm045. URL `http://biostatistics.oxfordjournals.org/content/9/3/432.abstract`.

Kayo Fujimoto and Thomas W Valente. Social network influences on adolescent substance use: disentangling structural equivalence from cohesion. *Social Science & Medicine*, 74 (12):1952–1960, 2012.

Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

Jian Guo, Elizaveta Levina, George Michailidis, and Ji Zhu. Joint estimation of multiple graphical models. *Biometrika*, page asq060, 2011.

David Hallac, Jure Leskovec, and Stephen Boyd. Network lasso: Clustering and optimization in large graphs. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 387–396. ACM, 2015.

Dana L Haynie. Delinquent peers revisited: Does network structure matter? *American Journal of Sociology*, 106(4):1013–1057, 2001.

Toby Dylan Hocking, Armand Joulin, Francis Bach, and Jean-Philippe Vert. Clusterpath an algorithm for clustering using convex fusion penalties. 2011.

Cho-Jui Hsieh, Mátyás A Sustik, Inderjit S Dhillon, Pradeep K Ravikumar, and Russell Poldrack. Big & quic: Sparse inverse covariance estimation for a million variables. In *Advances in neural information processing systems*, pages 3165–3173, 2013a.

Cho-Jui Hsieh, Mtys Sustik, Inderjit S. Dhillon, Pradeep Ravikumar, and Russell A. Poldrack. Big & quic: Sparse inverse covariance estimation for a million variables. In *Neural Information Processing Systems (NIPS)*, dec 2013b.

Cho-Jui Hsieh, Mátyás A Sustik, Inderjit S Dhillon, and Pradeep Ravikumar. Quic: quadratic approximation for sparse inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1):2911–2947, 2014.

Pengsheng Ji and Jiashun Jin. Coauthorship and citation networks for statisticians. *The Annals of Applied Statistics*, 10(4):1779–1812, 2016.

Alexander Jung, Nguyen Tran, and Alexandru Mara. When is network lasso accurate? *Frontiers in Applied Mathematics and Statistics*, 3:28, 2018.

Ioannis Koutis, Gary L Miller, and Richard Peng. Approaching optimality for solving sdd linear systems. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 235–244. IEEE, 2010.

Steffen L Lauritzen. *Graphical models*, volume 17. Clarendon Press, 1996.

Lung-fei Lee. Identification and estimation of econometric models with group interactions, contextual factors and fixed effects. *Journal of Econometrics*, 140(2):333–374, 2007.

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

Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive datasets.* Cambridge University Press, 2014.

Caiyan Li and Hongzhe Li. Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9):1175–1182, 2008.

Caiyan Li and Hongzhe Li. Variable selection and regression analysis for graph-structured covariates with an application to genomics. *The Annals of Applied Statistics*, 4(3):1498, 2010.

Ker-Chau Li. Asymptotic optimality of cl and generalized cross-validation in ridge regression with application to spline smoothing. *The Annals of Statistics*, pages 1101–1112, 1986.

Tianxi Li, Sharmodeep Bhattacharyya, Purnamrita Sarkar, Peter J Bickel, and Elizaveta Levina. Hierarchical community detection by recursive bi-partitioning. *arXiv preprint arXiv:1810.01509*, 2018.

Tianxi Li, Elizaveta Levina, and Ji Zhu. Prediction models for network-linked data. *The Annals of Applied Statistics*, 13(1):132–164, 2019.

Jiahe Lin, Sumanta Basu, Moulinath Banerjee, and George Michailidis. Penalized maximum likelihood estimation of multi-layered gaussian graphical models. *J. Mach. Learn. Res.*, 17(1):5097–5147, January 2016. ISSN 1532-4435.

Fredrik Lindsten, Henrik Ohlsson, and Lennart Ljung. *Just relax and come clustering!: A convexification of k-means clustering.* Linköping University Electronic Press, 2011.

Jianyu Liu, Guan Yu, and Yufeng Liu. Graph-based sparse linear discriminant analysis for high-dimensional classification. *Journal of Multivariate Analysis*, 171:250–269, 2019.

Charles F Manski. Identification of endogenous social effects: The reflection problem. *The Review of Economic Studies*, 60(3):531–542, 1993.

Kanti V Mardia. Some properties of clasical multi-dimesional scaling. *Communications in Statistics-Theory and Methods*, 7(13):1233–1241, 1978.

Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, pages 1436–1462, 2006.

Lynn Michell and Patrick West. Peer pressure to smoke: the meaning depends on the method. *Health Education Research*, 11(1):39–49, 1996.

Richard A Olshen and Bala Rajaratnam. Successive normalization of rectangular arrays. *Annals of statistics*, 38(3):1638, 2010.

Wei Pan, Benhuai Xie, and Xiaotong Shen. Incorporating predictor network in penalized regression with application to microarray data. *Biometrics*, 66(2):474–484, 2010.

Michael Pearson and Patrick West. Drifting smoke rings. *Connections*, 25(2):59–76, 2003.

Bogdan Raducanu and Fadi Dornaika. A supervised non-linear dimensionality reduction approach for manifold learning. *Pattern Recognition*, 45(6):2432–2444, 2012.

Pradeep Ravikumar, Martin J Wainwright, Garvesh Raskutti, and Bin Yu. High-dimensional covariance estimation by minimizing ℓ1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980, 2011.

Zhao Ren, Tingni Sun, Cun-Hui Zhang, Harrison H Zhou, et al. Asymptotic normality and optimalities in estimation of large gaussian graphical models. *The Annals of Statistics*, 43(3):991–1026, 2015.

Adam J Rothman, Peter J Bickel, Elizaveta Levina, and Ji Zhu. Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515, 2008.

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

Veeranjaneyulu Sadhanala, Yu-Xiang Wang, and Ryan J Tibshirani. Graph sparsification approaches for laplacian smoothing. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 1250–1259, 2016.

Xiaotong Shen, Hsin-Cheng Huang, and Wei Pan. Simultaneous supervised clustering and feature selection over a graph. *Biometrika*, 99(4):899–914, 2012.

Ali Shojaie and George Michailidis. Penalized principal component regression on graphs for analysis of subnetworks. In *Advances in neural information processing systems*, pages 2155–2163, 2010.

Martin Slawski, Wolfgang zu Castell, Gerhard Tutz, et al. Feature selection guided by structural information. *The Annals of Applied Statistics*, 4(2):1056–1080, 2010.

Alexander J Smola and Risi Kondor. Kernels and regularization on graphs. In *Learning theory and kernel machines*, pages 144–158. Springer, 2003.

Daniel A Spielman. Algorithms, graph theory, and linear equations in laplacian matrices. In *Proceedings of the International Congress of Mathematicians*, volume 4, pages 2698–2722, 2010.

Hokeun Sun, Wei Lin, Rui Feng, and Hongzhe Li. Network-regularized high-dimensional cox regression for analysis of genomic data. *Statistica Sinica*, 24(3):1433, 2014.

Kean Ming Tan and Daniela Witten. Statistical properties of convex clustering. *Electronic journal of statistics*, 9(2):2324, 2015.

Minh Tang, Daniel L Sussman, and Carey E Priebe. Universally consistent vertex classification for latent positions graphs. *The Annals of Statistics*, 41(3):1406–1430, 2013.

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

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

Nguyen Tran, Henrik Ambos, and Alexander Jung. A network compatibility condition for compressed sensing over complex networks. In *2018 IEEE Statistical Signal Processing Workshop (SSP)*, pages 50–54. IEEE, 2018.

Elif Vural and Christine Guillemot. Out-of-sample generalizations for supervised manifold learning for classification. *IEEE Transactions on Image Processing*, 25(3):1410–1424, 2016.

Martin J Wainwright. Sharp thresholds for high-dimensional and noisy recovery of sparsity using l1-constrained quadratic programming. *IEEE Transactions on Information Theory*, 2009.

Yu-Xiang Wang, James Sharpnack, Alex Smola, and Ryan J Tibshirani. Trend filtering on graphs. *arXiv preprint arXiv:1410.7690*, 2014.

Daniela M Witten, Jerome H Friedman, and Noah Simon. New insights and faster computations for the graphical lasso. *Journal of Computational and Graphical Statistics*, 20(4): 892–900, 2011.

Daniela M Witten, Ali Shojaie, and Fan Zhang. The cluster elastic net for high-dimensional regression with unknown variable grouping. *Technometrics*, 56(1):112–122, 2014.

Jaewon Yang, Julian McAuley, and Jure Leskovec. Community detection in networks with node attributes. In *2013 IEEE 13th International Conference on Data Mining*, pages 1151–1156. IEEE, 2013.

Wankou Yang, Changyin Sun, and Lei Zhang. A multi-manifold discriminant analysis method for image feature extraction. *Pattern Recognition*, 44(8):1649–1657, 2011.

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

Jianxin Yin and Hongzhe Li. Adjusting for high-dimensional covariates in sparse precision matrix estimation by $\ell 1$-penalization. *Journal of multivariate analysis*, 116:365–381, 2013.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Ming Yuan and Yi Lin. Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.

Sen Zhao and Ali Shojaie. A significance test for graph-constrained estimation. *Biometrics*, 72(2):484–493, 2016.

Tuo Zhao, Han Liu, Kathryn Roeder, John Lafferty, and Larry Wasserman. The huge package for high-dimensional undirected graph estimation in r. *Journal of Machine Learning Research*, 13(Apr):1059–1062, 2012.

Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd international conference on Machine learning*, pages 1036–1043. ACM, 2005.

Shuheng Zhou, John Lafferty, and Larry Wasserman. Time varying undirected graphs. *Machine Learning*, 80(2-3):295–319, 2010.

Yunzhang Zhu, Xiaotong Shen, and Wei Pan. Simultaneous grouping pursuit and feature selection over an undirected graph. *Journal of the American Statistical Association*, 108 (502):713–725, 2013.

Appendix A. Proofs

First, recall the following matrix norm definitions we'll need: for any matrix M, $\|M\|_\infty = \max_{ij}|M_{ij}|$, $\|M\|_{1,1} = \max_j \|M_{\cdot j}\|_1$, and $\|M\|_{\infty,\infty} = \max_i \|M_{i\cdot}\|_1$.

The following lemma summarizes a few concentration inequalities that we will need.

Lemma 1 (Concentration of norm of a multivariate Gaussian) *For a Gaussian random vector $x \sim \mathcal{N}(0,\Sigma)$, with $\Sigma \in \mathbb{R}^{p\times p}$ a positive definite matrix and $\phi_{\max}(\Sigma)$ the largest eigenvalue of Σ, we have,*

$$\mathbb{P}(|\|x\|_2 - \sqrt{\mathrm{tr}(\Sigma)}| > t) \leq 2\exp(-c\frac{t^2}{\phi_{\max}(\Sigma)}), \tag{15}$$

$$\mathbb{P}(|\|x\|_2^2 - \mathrm{tr}(\Sigma)| > t) \leq 2\exp(-c\frac{t}{\phi_{\max}(\Sigma)}), \tag{16}$$

$$\mathbb{P}(|\|x\|_1 - \frac{2}{\pi}\sum_{i=1}^{p}\sqrt{\Sigma_{ii}}| > t) \leq 2\exp(-c\frac{t^2}{p\phi_{\max}(\Sigma)}) \tag{17}$$

for some generic constant $c > 0$. Further, if $x_1, \cdots, x_n$ are i.i.d. observations from $\mathcal{N}(0,\Sigma)$, then

$$\mathbb{P}(\sum_{i}^{n}\|x_i\|_2^2 > 2n\mathrm{tr}(\Sigma)) \leq 2\exp(-cnr(\Sigma)) \tag{18}$$

where $r(\Sigma)$ is the stable rank of Σ.

Proof [Proof of Lemma 1] The first inequality (15) follows from concentration of a Lipschitz function of a sub-Gaussian random vector. Inequalities (16) and (17) follow from the definition of a sub-exponential random variable. Lastly, (18) follows from applying Bernstein's inequality to (16) with $t = n\mathrm{tr}(\Sigma)$. ■

Proof [Proof of Proposition 1] By Edwards (2013), the eigenvalues of A are given by

$$\frac{1}{\bar{d}}(4\sin^2(\frac{i\pi}{2\sqrt{n}}) + 4\sin^2(\frac{j\pi}{2\sqrt{n}})), i, j \in \{0, 1, \cdots, \sqrt{n}-1\}. \tag{19}$$

Since the average degree $2 \leq \bar{d} \leq 4$ for a lattice network, we ignore this constant. First, we show $m_A \leq n^{2/3}$, which by definition of m_A is equivalent to $\tau_{n-n^{2/3}} \geq n^{-1/3}$. Define the set of all eigenvalues satisfying this condition as

$$\mathcal{A}_n = \{(i,j) : i, j \in \mathbb{N} \cap [0, \sqrt{n}-1], 4\sin^2(\frac{i\pi}{2\sqrt{n}}) + 4\sin^2(\frac{j\pi}{2\sqrt{n}}) < n^{-1/3}\}.$$

Then it is sufficient to show $|\mathcal{A}_n| < n^{2/3}$. Applying the inequality $\sin(x) \geq \frac{2}{\pi}x$ for $x \in [0, \pi/2]$, we can see that it is sufficient to show $|\tilde{\mathcal{A}}_n| < n^{2/3}$, where

$$\tilde{\mathcal{A}}_n = \{(i,j) : i, j \in \mathbb{N} \cap [0, \sqrt{n}-1], \frac{4i^2}{n} + \frac{4j^2}{n} < n^{-1/3}\}.$$

The cardinality of $\tilde{\mathcal{A}}_n$ can be computed exactly by counting; for simplicity, we give an approximate calculation for when n is sufficiently large. In this case the proportion of pairs (i,j) out of the entire set of $(\mathbb{N} \cap [0, \sqrt{n}-1]) \times (\mathbb{N} \cap [0, \sqrt{n}-1])$ that satisfy the condition to be included in $\tilde{\mathcal{A}}_n$ can be upper bounded by twice the ratio betwen the area of the quarter circle with radius $\frac{n^{1/3}}{2}$ and the area of the $\sqrt{n} \times \sqrt{n}$ square. This gives

$$|\tilde{\mathcal{A}}_n| \leq 2\frac{\pi}{16}n^{2/3} < n^{2/3}.$$

To prove the second claim, consider the $\mu = U\beta$ such that all the inequalities in (10) hold as equalities and $\delta = 1/2$. Then, by noting that $P_{\mathbf{1}}u_n = u_n$, we have

$$\|\mu - P_{\mathbf{1}}\mu\|_2^2 = \sum_{i<n}\beta_i^2 = \|\mu\|_2^2 n^{-\frac{2(1+\delta)}{3}-1}\sum_{i<n}\frac{1}{\tau_i^2} = \|\mu\|_2^2 n^{-2}\sum_{i<n}\frac{1}{\tau_i^2} \ . \tag{20}$$

We need a lower bound for $\sum_{i<n}\frac{1}{\tau_i^2}$. By (19),

$$\begin{aligned}
\sum_{i<n}\frac{1}{\tau_i^2} &= \sum_{i,j\le\sqrt{n}-1,(i,j)\neq(0,0)} \frac{1}{(4\sin^2(\frac{i\pi}{2\sqrt{n}})+4\sin^2(\frac{j\pi}{2\sqrt{n}}))^2}\\
&> \sum_{1\le i,j\le\sqrt{n}-1} \frac{1}{(4\sin^2(\frac{i\pi}{2\sqrt{n}})+4\sin^2(\frac{j\pi}{2\sqrt{n}}))^2}\\
&= \frac{n}{\pi^2}\sum_{1\le i,j\le\sqrt{n}-1} \frac{1}{(4\sin^2(\frac{i\pi}{2\sqrt{n}})+4\sin^2(\frac{j\pi}{2\sqrt{n}}))^2}\frac{\pi}{\sqrt{n}}\frac{\pi}{\sqrt{n}}\\
&\ge \frac{n}{\pi^2}\sum_{1\le i,j\le\sqrt{n}-1} \frac{1}{(4\frac{\pi^2 i^2}{4n}+4\frac{\pi^2 j^2}{4n})^2}\frac{\pi}{\sqrt{n}}\frac{\pi}{\sqrt{n}} && \text{— applying} \sin^2(x)\le x^2\\
&> \frac{1}{2}\frac{n}{\pi^2}\int_{\frac{\pi}{\sqrt{n}}\le x,y\le\pi}\frac{1}{(x^2+y^2)^2}dxdy && \text{— sum lower bounded by 1/2 of the integral}\\
&> \frac{n}{2\pi^2}\int_{\pi/6}^{\pi/3}\int_{\frac{\pi}{\sqrt{n}/2}}^{\pi}\frac{1}{r^3}drd\theta \quad \text{— polar coordinates, } \{r\in[\frac{2\pi}{\sqrt{n}},\pi],\theta\in[\frac{\pi}{6},\frac{\pi}{3}]\}\subset[\frac{\pi}{\sqrt{n}},\pi]\times[\frac{\pi}{\sqrt{n}},\pi]\\
&= \frac{n}{24\pi^3}(\frac{n}{4}-1).
\end{aligned}$$

Substituting this lower bound for $\sum_{i<n}\frac{1}{\tau_i^2}$ in (20), for a sufficiently large n we have

$$\|\mu - P_{\mathbf{1}}\mu\|_2^2 = \|\mu\|_2^2 n^{-2}\sum_{i<n}\frac{1}{\tau_i^2} > c\|\mu\|_2^2.$$

Therefore, the μ we constructed is nontrivially cohesive.

■

We can represent each column of M by taking the basis expansion in U, obtaining the basis coefficient matrix $B = (B_{\cdot 1}, B_{\cdot 2}, \cdots, B_{\cdot p})$ such that $M = UB$. Let $\hat{B} = U^T\hat{M}$, where $\hat{M}$ is the estimate (4). We can view $\hat{B}$ as an estimate of B. We first state the error bound for $\hat{B}$ in Lemma 2, and the bound for $\hat{M}$ directly follows.

Lemma 2 *Under model* (2) *and Assumption 1, if* $\alpha = n^{\frac{1+\delta}{3}}$, *we have*

1. *In maximum norm,*

$$\|\hat{B} - B\|_\infty \le C\sigma\left((\sqrt{\log pn}\sqrt{m_A}n^{-\frac{1+\delta}{3}}) \vee \frac{\sqrt{\log(pm_A)}}{1+\Delta} \vee \sqrt{\log(p)}\right) \tag{21}$$

   with probability at least $1 - \exp(-c\log(p(n-m_A))) - \exp(-c'\log(pm_A))$ *for some constants* C, C', c, c', *and* c''.

2. *In Frobenius norm,*

$$\|\hat{B} - B\|_F \le \sqrt{(b^2+2\sigma^2)p((n-m_A)m_A n^{-\frac{2(1+\delta)}{3}} + \frac{m_A}{(1+\Delta)^2}+1)} \tag{22}$$

   with probability at least $1-\exp(-c''(n-m_A)r(\Sigma))-\exp(-c''m_A r(\Sigma))-\exp(-c''r(\Sigma))$.

3. if $\log p = o(n)$ *and* $m_A = o(n)$*, then*

$$\|\hat{B} - B\|_{1,1} \leq C'(b + 2\sigma)(\sqrt{m_A}n^{\frac{2-\delta}{3}} + \sqrt{\log p}(\frac{m_A}{\Delta + 1} + 1)). \tag{23}$$

with probability at least $1 - \exp(-cn) - \exp(-Cm_A \log p) - \exp(-C \log p)$*.*

Proof [Proof of Lemma 2] Solving (4), we can explicitly write out

$$\hat{B} = (I + \alpha\Lambda)^{-1}B + (I + \alpha\Lambda)^{-1}U^T E = (I + \alpha\Lambda)^{-1}B + (I + \alpha\Lambda)^{-1}\tilde{E}.$$

In particular, for each column $j \in [p]$, the estimate can be written as

$$\hat{B}_{\cdot j} = (I + \alpha\Lambda)^{-1}B_{\cdot j} + (I + \alpha\Lambda)^{-1}U^T E_{\cdot j} = (I + \alpha\Lambda)^{-1}B_{\cdot j} + (I + \alpha\Lambda)^{-1}\tilde{E}_{\cdot j},$$

where $\tilde{E}_{\cdot j} \sim \mathcal{N}(0, \sigma^2 I)$. Let $\mathcal{Q}^j$ and $\mathcal{R}^j$ be two n dimensional vectors such that the ith element of $\mathcal{Q}^j$ is given by $\frac{\alpha\tau_i}{1+\alpha\tau_i}B_{ij}$ while the ith element of $\mathcal{R}^j$ is given by $\frac{1}{1+\alpha\tau_i}\tilde{E}_{ij}$.

$$\hat{B}_{\cdot j} - B_{\cdot j} = \mathcal{Q}^j + \mathcal{R}^j.$$

For the element-wise L_∞ norm, we have

$$\|\mathcal{Q}^j\|_\infty \leq \max_{i<n} \frac{\alpha}{1 + \alpha\tau_i} \max_{i<n} |\tau_i B_{ij}| \leq \frac{\alpha}{1 + \alpha\tau_{n-1}} n^{-\frac{1+\delta}{3} - \frac{1}{2}} \|B_{\cdot j}\| \leq b \cdot \alpha n^{-\frac{1+\delta}{3}} = b. \tag{24}$$

where the second inequality is by Definition 1. The term $\mathcal{R}^j$ can be decomposed into two parts, the first $n - m_A$ elements and the last m_A elements. For the first $n - m_A$ elements, we have

$$\begin{aligned}
\max_{j\in[p]} \|\mathcal{R}^j_{1:n-m_A}\|_\infty &\leq \max_{j\in[p]} \max_{i\leq n-m_A} \frac{1}{1 + \alpha\tau_i} \max_{i\leq n-m_A} |\tilde{E}_{ij}| = \frac{1}{1 + \alpha\tau_{n-m_A}} \max_{i\leq n-m_A} \max_{j\in[p]} |\tilde{E}_{ij}| \\
&= \frac{1}{1 + \tau_{n-m_A} n^{\frac{1+\delta}{3}}} \max_{j\in[p]} \max_{i\leq n-m_A} |\tilde{E}_{ij}| \leq \frac{\sqrt{4\sigma^2 \log(p(n - m_A))}}{\tau_{n-m_A} n^{\frac{1+\delta}{3}}} \\
&\leq \sqrt{4\sigma^2 \log(p(n - m_A))} n^{-\frac{1+\delta}{3}} \sqrt{m_A}
\end{aligned} \tag{25}$$

by Definition 4, with probability at least $1 - \exp(-c \log(p(n - m_A)))$. For the remaining m_A elements, with probability at least $1 - \exp(-c' \log(pm_A))$, we have

$$\begin{aligned}
\max_{j\in[p]} \|\mathcal{R}^j_{n-m_A+1:n}\|_\infty &= \max_j \sum_{i>n-m_A} \frac{|\tilde{E}_{ij}|}{1 + \alpha\tau_i} \\
&= \max_j \sum_{n-m_A<i<n} \frac{|\tilde{E}_{ij}|}{1 + n^{\frac{1+\delta}{3}}\tau_i} |\tilde{E}_{ij}| + \max_j |\tilde{E}_{nj}| \\
&\leq \frac{\sqrt{4\sigma^2 \log(pm_A)}}{1 + \Delta} + \sqrt{4\sigma^2 \log(p)}.
\end{aligned} \tag{26}$$

Combining (24)–(26) leads to (21), since

$$\begin{aligned}\|\hat{B}-B\|_\infty &\leq \max_{j\in[p]}\|\mathcal{Q}^j\|_\infty + \max_{j\in[p]}\|\mathcal{R}^j_{1:n-m_A}\|_\infty + \max_{j\in[p]}\|\mathcal{R}^j_{n-m_A+1:n}\|_\infty \\ &\leq b + \sqrt{4\sigma^2\log p(n-m_A)}n^{-\frac{1+\delta}{3}}\sqrt{m_A} + \frac{\sqrt{4\sigma^2\log{(pm_A)}}}{1+\Delta} + \sqrt{4\sigma^2\log(p)} \\ &\leq (b+2\sigma)[(\sqrt{\log p(n-m_A)}n^{-\frac{1+\delta}{3}}\sqrt{m_A}) \vee \frac{\sqrt{\log(pm_A)}}{1+\Delta} \vee \log(p)]\end{aligned}$$

with probability at least $1-\exp(-c\log(p(n-m_A)))-\exp(-c'\log(pm_A))$ for sufficiently large n.

For the column-wise L_∞ norm, we have

$$\begin{aligned}\max_j\|\mathcal{Q}^j\|_1 &= \max_j\sum_i\frac{\alpha\tau_i|B_{ij}|}{1+\alpha\tau_i} \leq \max_j\Big(\sum_{i\leq n-m_A}|B_{ij}| + \sum_{i>n-m_A}\frac{\alpha\tau_i|B_{ij}|}{1+\alpha\tau_i}\Big) \\ &\leq \max_j\Big(b\frac{n-m_A}{\tau_{n-m_A}}n^{-\frac{1+\delta}{3}} + b\sum_{i>n-m_A}\frac{\alpha}{1+\alpha\tau_{n-1}}n^{-\frac{1+\delta}{3}}\Big) \quad \text{— by Assumption 1 —} \\ &\leq \max_j b\Big(\frac{n-m_A}{\tau_{n-m_A}}n^{-\frac{1+\delta}{3}} + \sum_{n-m_A<i<n}\frac{\alpha n^{-\frac{1+\delta}{3}}}{1+\Delta} + \alpha n^{-\frac{1+\delta}{3}}\Big) \\ &= b((n-m_A)\sqrt{m_A}n^{-\frac{1+\delta}{3}} + \frac{m_A}{1+\Delta}+1). \end{aligned} \tag{27}$$

For the second term,

$$\begin{aligned}\max_j\|\mathcal{R}^j\|_1 &\leq \max_j\sum_{i\leq n-m_A}\frac{1}{1+\alpha\tau_i}|\tilde{E}_{ij}| + \max_j\sum_{i>n-m_A}\frac{1}{1+\alpha\tau_i}|\tilde{E}_{ij}| \\ &\leq \frac{1}{1+\tau_{n-m_A}n^{\frac{1+\delta}{3}}}\max_j\sum_{i\leq n-m_A}|\tilde{E}_{ij}| + \max_j\sum_{n-m_A<i<n}\frac{|\tilde{E}_{ij}|}{1+\Delta} + \max_j|\tilde{E}_{nj}|. \end{aligned} \tag{28}$$

By Lemma 1, for each $j\in[p]$, $\mathbb{P}(\sum_{i\leq n-m_A}|\tilde{E}_{ij}|>2\sigma(n-m_A))\leq\exp(-2c(n-m_A))$ for some constant c; therefore

$$\mathbb{P}(\max_j\sum_{i\leq n-m_A}|\tilde{E}_{ij}|>2\sigma(n-m_A))\leq p\exp(-2c(n-m_A))\leq\exp(-cn),$$

as long as $\log p=o(n)$ and $m_A=o(n)$.

Assume $m_A\geq 2$, again by Lemma 1, for each $j\in[p]$,

$$\mathbb{P}(\sum_{n-m_A<i<n}|\tilde{E}_{ij}|>2\sigma(m_A-1)\sqrt{c'\log p})\leq\exp(-2Cm_A\log p)$$

for some constant $C,c'>0$ with $C>1$. Therefore,

$$\mathbb{P}(\max_j\sum_{n-m_A<i<n}|\tilde{E}_{ij}|>2\sigma m_A\sqrt{c'\log p})\leq p\exp(-2Cm_A\log p)\leq\exp(-C(m_A-1)\log p)$$

and

$$\mathbb{P}(\max_j |\tilde{E}_{nj}| > 2\sigma\sqrt{c'\log p}) \le \exp(-C\log p).$$

The above result is also trivially true if $m_A = 1$. Substituting these two inequalities into (28) gives

$$\begin{aligned}\max_j \|\mathcal{R}^j\|_1 &\le \frac{2\sigma(n-m_A)}{\tau_{n-m_A} n^{\frac{1+\delta}{3}}} + 2\sigma(\frac{m_A}{\Delta+1}+1)\sqrt{c'\log p}\\ &\le 2\sigma((n-m_A)\sqrt{m_A}n^{-\frac{1+\delta}{3}} + (\frac{m_A}{\Delta+1}+1)\sqrt{c'\log p}) \qquad (29)\end{aligned}$$

with probability at least $1-\exp(-cn)-\exp(-Cm_A\log p)-\exp(-C\log p)$. Now combining (27) and (29), we get

$$\begin{aligned}\|\hat{B}-B\|_{1,1} &\le \max_j \|\mathcal{Q}^j\|_1 + \max_j \|\mathcal{R}^j\|_1\\ &\le b((n-m_A)\sqrt{m_A}n^{-\frac{1+\delta}{3}} + (\frac{m_A}{\Delta+1}+1)) + 2\sigma((n-m_A)\sqrt{m_A}n^{-\frac{1+\delta}{3}} + (\frac{m_A}{\Delta+1}+1)\sqrt{c'\log p})\\ &\le (b+2\sigma)(\sqrt{m_A}n^{\frac{2-\delta}{3}} + (\frac{m_A}{\Delta+1}+1)(1\vee\sqrt{c'\log p}))\\ &\le (1\vee\sqrt{c'})(b+2\sigma)(\sqrt{m_A}n^{\frac{2-\delta}{3}} + (\frac{m_A}{\Delta+1}+1)\sqrt{\log p}).\end{aligned}$$

with probability at least $1-\exp(-cn)-\exp(-Cm_A\log p)-\exp(-C\log p)$ as long as $p\ge 3$.

Finally, for the Frobenius norm we have

$$\begin{aligned}\sum_j \|\mathcal{Q}^j\|_2^2 &= \sum_j\sum_i \frac{\alpha^2\tau_i^2|B_{ij}|^2}{(1+\alpha\tau_i)^2} \le \sum_j\Big(\sum_{i\le n-m_A}|B_{ij}|^2 + \sum_{i>n-m_A}\frac{\alpha^2\tau_i^2|B_{ij}|^2}{(1+\alpha\tau_i)^2}\Big)\\ &\le b^2\sum_j\Big(\frac{n-m_A}{\tau_{n-m_A}^2}n^{-\frac{2(1+\delta)}{3}} + \sum_{i>n-m_A}(\frac{\alpha}{1+\alpha\tau_{n-1}})^2 n^{-\frac{2(1+\delta)}{3}}\Big) \quad \text{— by Assumption 1 —}\\ &\le b^2 p\Big((n-m_A)m_A n^{-\frac{2(1+\delta)}{3}} + \frac{m_A}{(1+\Delta)^2}+1\Big). \qquad (30)\end{aligned}$$

For the second term,

$$\begin{aligned}\sum_j \|\mathcal{R}^j\|_2^2 &= \sum_j\Big(\sum_{i\le n-m_A}(\frac{1}{1+\alpha\tau_i})^2|\tilde{E}_{ij}|^2 + \sum_{i>n-m_A}(\frac{1}{1+\alpha\tau_i})^2|\tilde{E}_{ij}|^2\Big)\\ &\le \frac{1}{\tau_{n-m_A}^2}n^{-\frac{2(1+\delta)}{3}}\sum_{i\le n-m_A}\sum_j|\tilde{E}_{ij}|^2 + \sum_{n-m_A<i<n}\sum_j|\tilde{E}_{ij}|^2/(1+\Delta)^2 + \sum_j|\tilde{E}_{nj}|^2\\ &\le m_A n^{-\frac{2(1+\delta)}{3}}\sum_{i\le n-m_A}\|\tilde{E}_{i\cdot}\|_2^2 + \sum_{n-m_A<i<n}\|\tilde{E}_{i\cdot}\|_2^2/(1+\Delta)^2 + \|\tilde{E}_{n\cdot}\|^2. \qquad (31)\end{aligned}$$

If $m_A\ge 2$, by (18) from Lemma 1, for a proper c, we have

$$\mathbb{P}(\sum_{i\le n-m_A}\|\tilde{E}_{i\cdot}\|_2^2 > 2(n-m_A)p\sigma^2) \le \mathbb{P}(\sum_{i\le n-m_A}\|\tilde{E}_{i\cdot}\|_2^2 > 2(n-m_A)\mathrm{tr}(\Sigma)) \le 2\exp(-c(n-m_A)r(\Sigma)),$$

$$\mathbb{P}(\sum_{n-m_A<i<n}\|\tilde{E}_{i\cdot}\|_2^2 > 2m_Ap\sigma^2) \le \mathbb{P}(\sum_{i>n-m_A}\|\tilde{E}_{i\cdot}\|_2^2 > 2m_A\mathrm{tr}(\Sigma)) \le 2\exp(-cm_Ar(\Sigma)),$$

$$\mathbb{P}(\|\tilde{E}_{n\cdot}\|^2 > 2p\sigma^2) \le 2\exp(-c\frac{p\sigma^2}{\phi_{max}(\Sigma)}).$$

Putting everything together,

$$\begin{aligned}
\|\hat{B} - B\|_F^2 &\leq \sum_j \|\mathcal{Q}^j\|_2^2 + \sum_j \|\mathcal{R}^j\|_2^2 \\
&\leq b^2 p\Big((n - m_A) m_A n^{-\frac{2(1+\delta)}{3}} + \frac{m_A}{(1+\Delta)^2} + 1\Big) \\
&\qquad + 2(n - m_A) p m_A n^{-\frac{2(1+\delta)}{3}} \sigma^2 + 2(\frac{m_A}{(1+\Delta)^2} + 1) p \sigma^2 \\
&= (b^2 + 2\sigma^2) p\Big((n - m_A) m_A n^{-\frac{2(1+\delta)}{3}} + \frac{m_A}{(1+\Delta)^2} + 1\Big)
\end{aligned}$$

with probability at least $1 - 2\exp(-c(n-m_A)r(\Sigma)) - 2\exp(-c m_A r(\Sigma)) - 2\exp(-c\frac{p\sigma^2}{\phi_{max}(\Sigma)})$. Notice that when $m_A = 1$, the above result is trivially true.

■

Proof [Proof of Theorem 1] By definition, we have $\|\hat{M} - M\|_F = \|U(\hat{B} - B)\|_F = \|\hat{B} - B\|_F$. Thus the theorem follows directly from Lemma 2 and the fact that $n - m_A \leq n$. Note that the Frobenius norm bound in Lemma 2 does not need $\log p = o(n)$ and $m_A = o(n)$. ■

Now we proceed to prove Theorem 2. Let

$$\begin{aligned}
\hat{S} &= \frac{1}{n}(X - \hat{M})^T (X - \hat{M}) \\
S &= \frac{1}{n}(X - M)^T (X - M)
\end{aligned}$$

S is the sample covariance matrix used by the glasso algorithm when the mean is assumed known (and without loss of generality set to 0). The success of glasso is dependent on S concentrating around the true covariance matrix Σ. If we can show $\hat{S}$ concentrates around Σ, we should be able to prove similar properties of GNC-lasso.

Lemma 3 *Under the conditions of Theorem 1 and assuming* $\log p = o(n), m_A = o(n)$*, we have*

$$\begin{aligned}
\|\hat{S} - \Sigma\|_\infty \leq C \max\Big(&\sqrt{\log{(pn)}} m_A n^{-\frac{2+2\delta}{3}}, \sqrt{\log{(pn)}}\sqrt{\log p} m_A^{3/2} n^{-\frac{4+\delta}{3}}, \\
&\sqrt{\log{(pn)}}\sqrt{m_A} n^{-\frac{1+\delta}{3}}, \sqrt{\log{(pn)}}\sqrt{\log p}\frac{m_A}{n}, \sqrt{\frac{\log p}{n}}\Big)
\end{aligned}$$

with probability at least $1 - \exp(-c\log(p(n-m_A))) - \exp(-c\log(pm_A)) - \exp(-c\log p)$ *for some constant* C *and* c *that only depend on* b *and* σ*.*

Proof [Proof of Lemma 3] We will be using C and c to denote generic constants whose value might change across different lines. Using the triangular inequality,

$$\|\hat{S} - \Sigma\|_\infty \leq \|\hat{S} - S\|_\infty + \|S - \Sigma\|_\infty.$$

we will prove concentration in two steps. Starting with the first term and writing $X = M + E$, we have

$$
\begin{aligned}
\hat{S} - S &= \frac{1}{n}(UB + E - U\hat{B})^T(UB + E - U\hat{B}) - \frac{1}{n}E^TE \\
&= \frac{1}{n}[(\hat{B} - B)^T(\hat{B} - B) - E^TU((\hat{B} - B)) - ((\hat{B} - B))^TU^TE + E^TE] - \frac{1}{n}E^TE \\
&= \frac{1}{n}(\hat{B} - B)^T(\hat{B} - B) - \frac{1}{n}E^TU(\hat{B} - B) - \frac{1}{n}(\hat{B} - B)^TU^TE. \qquad (32)
\end{aligned}
$$

By Lemma 2, for some constant C depending on b and σ,

$$
\begin{aligned}
\|\frac{1}{n}(\hat{B} - B)^T(\hat{B} - B)\|_\infty &\le \frac{1}{n}\|\hat{B} - B\|_\infty\|\hat{B} - B\|_{1,1} \\
&\le \frac{C}{n}[(\sqrt{\log p(n - m_A)}n^{-\frac{1+\delta}{3}}\sqrt{m_A}) \vee \frac{\sqrt{\log(pm_A)}}{1 + \Delta} \vee \sqrt{\log(p)}] \times \\
&\qquad [\sqrt{m_A}n^{\frac{2-\delta}{3}} \vee \sqrt{\log p}(\frac{m_A}{\Delta + 1} + 1)] \\
&\le C\max\Big(\sqrt{\log{(pn)}}m_An^{-\frac{2+2\delta}{3}}, \sqrt{\log{(pn)}}\sqrt{\log p}n^{-\frac{4+\delta}{3}}\sqrt{m_A}(\frac{m_A}{\Delta + 1} + 1), \\
&\qquad \frac{\sqrt{\log{(pm_A)}}\sqrt{m_A}n^{-\frac{1+\delta}{3}}}{1 + \Delta}, \frac{\sqrt{\log{(pm_A)}}\sqrt{\log p}}{(1 + \Delta)n}(\frac{m_A}{\Delta + 1} + 1), \\
&\qquad \sqrt{\log p}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\log p}{n}(\frac{m_A}{\Delta + 1} + 1)\Big) \qquad (33)
\end{aligned}
$$

with probability at least $1 - \exp(-c\log(p(n - m_A))) - \exp(-c\log(pm_A)) - \exp(-c\log(p))$.

On the other hand, note that $U^TE = (U_{1\cdot}E_{\cdot j})_{i,j=1}^n$ and $\|U_{i\cdot}\|_2 = 1$, so $(U^TE)_{ij} \sim \mathcal{N}(0, \sigma^2)$. Therefore

$$
\|U^TE\|_\infty \le \sqrt{2\sigma^2\log(np)}
$$

with probability at least $1 - \exp(-c\log(np))$. Hence the second and third terms in (32) satisfy

$$
\begin{aligned}
\|\frac{1}{n}E^TU(\hat{B} - B)\|_\infty &\le \frac{1}{n}\|U^TE\|_\infty\|\hat{B} - B\|_{1,1} \\
&\le C\frac{1}{n}\sqrt{\log(np)}[\sqrt{m_A}n^{\frac{2-\delta}{3}} \vee \sqrt{\log p}(\frac{m_A}{\Delta + 1} + 1)] \\
&= C[\sqrt{\log{(np)}}\sqrt{m_A}n^{-\frac{1+\delta}{3}} + \frac{\sqrt{\log{(np)}}\sqrt{\log p}}{n}(\frac{m_A}{\Delta + 1} + 1)] \qquad (34)
\end{aligned}
$$

with probability at least $1 - \exp(-c\log(p(n - m_A)) - \exp(-c\log(pm_A)) - \exp(-c\log(np))$. Note that both terms in (34) dominate the last two terms in (33). Thus substituting (33) and (34) into (32) leads to

$$
\begin{aligned}
\|\hat{S} - S\|_\infty \le C\max\Big(&\sqrt{\log{(pn)}}m_An^{-\frac{2+2\delta}{3}}, \sqrt{\log{(pn)}}\sqrt{\log p}n^{-\frac{4+\delta}{3}}\sqrt{m_A}(\frac{m_A}{\Delta + 1} + 1), \\
&\sqrt{\log{(np)}}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\sqrt{\log{(np)}}\sqrt{\log p}}{n}(\frac{m_A}{\Delta + 1} + 1), \\
&\sqrt{\log p}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\log p}{n}(\frac{m_A}{\Delta + 1} + 1)\Big)
\end{aligned}
$$

with probability at least $1 - \exp(-c\log(p(n-m_A))) - \exp(-c\log(pm_A)) - \exp(-c\log(p))$.

In addition, Lemma 1 of Ravikumar et al. (2011) implies that

$$\|S - \Sigma\|_\infty \le \sqrt{\frac{2c\log p}{n}}$$

with probability at least $1 - \exp(-c\log p)$. Therefore, we have

$$\begin{aligned}\|\hat{S} - \Sigma\|_\infty \le C\max\Big(&\sqrt{\log{(pn)}}m_A n^{-\frac{2+2\delta}{3}}, \sqrt{\log{(pn)}}\sqrt{\log p}n^{-\frac{4+\delta}{3}}\sqrt{m_A}(\frac{m_A}{\Delta+1}+1),\\ &\sqrt{\log{(np)}}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\sqrt{\log{(np)}}\sqrt{\log p}}{n}(\frac{m_A}{\Delta+1}+1),\\ &\sqrt{\log p}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\log p}{n}(\frac{m_A}{\Delta+1}+1), \sqrt{\frac{\log p}{n}}\Big)\end{aligned}$$

with probability at least $1 - \exp(-c\log(p(n-m_A))) - \exp(-c\log(pm_A)) - 2\exp(-c\log p)$.

■

For conciseness, we present Theorem 2 in the main text by assuming the more interesting situations $p \ge n^{c_0}$. This is not necessary in any sense, so here we prove a trivially more general version of Theorem 2 without the high-dimensional assumption. For completeness, we rewrite the theorem here.

Theorem 3 (Trivially generalized version of Theorem 2) *Under the conditions of Theorem 1 and Assumption 2, if* $\log p = o(n)$ *and* $m_A = o(n)$*, there exist some positive constants* C, c, c', c'' *that only depend on* b *and* σ*, such that if* $\hat{\Theta}$ *is the output of Algorithm 1 with* $\alpha = n^{\frac{1+\delta}{3}}$, $\lambda = \frac{8}{\rho}\nu(n,p)$ *where*

$$\begin{aligned}\nu(n,p) := C\max\Big(&\sqrt{\log{(pn)}}m_A n^{-\frac{2+2\delta}{3}}, \sqrt{\log{(pn)}}\sqrt{\log p}n^{-\frac{4+\delta}{3}}\sqrt{m_A}(\frac{m_A}{\Delta+1}+1),\\ &\sqrt{\log{(np)}}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\sqrt{\log{(np)}}\sqrt{\log p}}{n}(\frac{m_A}{\Delta+1}+1),\\ &\sqrt{\log p}\sqrt{m_A}n^{-\frac{1+\delta}{3}}, \frac{\log p}{n}(\frac{m_A}{\Delta+1}+1), \sqrt{\frac{\log p}{n}}\Big)\end{aligned} \tag{35}$$

and n *sufficiently large so that*

$$\nu(n,p) < \frac{1}{6(1+8/\rho)\psi\max\{\kappa_\Sigma\kappa_\Gamma, (1+8/\rho)\kappa_\Sigma^3\kappa_\Gamma^2\}},$$

then with probability at least $1-\exp(-c\log(p(n-m_A)))-\exp(-c'\log(pm_A))-\exp(-c''\log p)$*, then the estimate* $\hat{\Theta}$ *has the following properties:*

1. *Error bounds:*

$$\begin{aligned}\|\hat{\Theta} - \Theta\|_\infty &\le 2(1+8/\rho)\kappa_\Gamma\nu(n,p)\\ \|\hat{\Theta} - \Theta\|_F &\le 2(1+8/\rho)\kappa_\Gamma\nu(n,p)\sqrt{s+p}.\\ \|\hat{\Theta} - \Theta\| &\le 2(1+8/\rho)\kappa_\Gamma\nu(n,p)\min(\sqrt{s+p}, \psi).\end{aligned}$$

2. *Support recovery:*
$$S(\hat{\Theta}) \subset S(\Theta),$$
and if additionally $\min_{(j,j')\in S(\Theta)} |\Theta_{jj'}| > 2(1+8/\rho)\kappa_\Gamma \nu(n,p)$, *then*
$$S(\hat{\Theta}) = S(\Theta).$$

We will use the *primal-dual witness* strategy from Ravikumar et al. (2011) for proof. We show that even if $\hat{S}$ has a worse concentration around Σ than S, we can still achieve consistency and sparsistency under certain regularity conditions.

Proof [Proof of Theorem 3 and Theorem 2] The argument follows the proof of Theorem 1 in Ravikumar et al. (2011). In particular, for the event where the bound in Lemma 3 holds, we just have to show that the primal-dual witness construction succeeds. The choice of $\lambda = \frac{8}{\rho}\nu(n,p)$ ensures $\|\hat{S}-\Sigma\|_\infty \leq \frac{\rho}{8}\lambda$. With the requirement on the sample size, the assumptions of Lemma 5 and 6 in Ravikumar et al. (2011) hold, implying strict dual feasibility holds for the primal-dual witness, which shows the procedure succeeds. Then the first claim of the theorem is a direct result of Lemma 6 in Ravikumar et al. (2011) and the second claim is true by construction of the primal-dual witness procedure. The remaining bounds can be proved similarly.

Finally, if $p \geq n^{c_0}$, we have $\log(p) \geq c_0 \log(n)$, so in (35), the 3rd coincides with the 5th, and the 4th term coincides with the 6th term, by magnitude, resulting in the form

$$\begin{aligned}
&C\max\Big(m_A n^{-\frac{2+2\delta}{3}}\sqrt{\log p}, \sqrt{m_A} n^{-\frac{4+\delta}{3}}(\frac{m_A}{\Delta+1}+1)\log p,\\
&\qquad\qquad \sqrt{m_A} n^{-\frac{1+\delta}{3}}\sqrt{\log p}, (\frac{m_A}{\Delta+1}+1)\frac{\log p}{n}, \sqrt{\frac{\log p}{n}}\Big)\\
=&C\sqrt{\frac{\log p}{n}}\max\Big(1, m_A n^{-\frac{1+4\delta}{6}}, \sqrt{m_A} n^{-\frac{5+2\delta}{6}}(\frac{m_A}{\Delta+1}+1)\sqrt{\log p}, \sqrt{m_A} n^{\frac{1-2\delta}{6}}, (\frac{m_A}{\Delta+1}+1)\sqrt{\frac{\log p}{n}}\Big).
\end{aligned}$$

To further simplified the form, we apply another upper bound for the term by the fact that $\sqrt{m_A} \geq 1$, $\frac{m_A}{\Delta+1}+1 \geq 1$ and $\delta \geq 0$, which gives

$$\begin{aligned}
&C\sqrt{\frac{\log p}{n}}\max\Big(1, m_A n^{-\frac{1+4\delta}{6}}, \sqrt{m_A} n^{-\frac{5+2\delta}{6}}(\frac{m_A}{\Delta+1}+1)\sqrt{\log p}, \sqrt{m_A} n^{\frac{1-2\delta}{6}}, (\frac{m_A}{\Delta+1}+1)\sqrt{\frac{\log p}{n}}\Big)\\
&\leq C\sqrt{\frac{\log p}{n}}\max\Big(1, m_A n^{-\frac{1+4\delta}{6}}, \sqrt{m_A} n^{-\frac{5+2\delta}{6}}(\frac{m_A}{\Delta+1}+1)\sqrt{\log p} + (\frac{m_A}{\Delta+1}+1)\sqrt{\frac{\log p}{n}}, \sqrt{m_A} n^{\frac{1-2\delta}{6}}\Big)\\
&\leq C\sqrt{\frac{\log p}{n}}\max\Big(1, m_A n^{-\frac{1+4\delta}{6}}, \sqrt{m_A} n^{\frac{1-2\delta}{6}}, \sqrt{\frac{\log p}{n}}(\frac{m_A}{\Delta+1}+1)(\sqrt{m_A} n^{-\frac{1+\delta}{3}}+1)\Big).
\end{aligned}$$

■

Appendix B. Oracle mean estimation by GNC-lasso

In our setting, unlike in the classical glasso setting, the mean estimate is also of interest, and in this section we show that our estimate $\hat{M}$ enjoys a weak oracle property in a certain sense. We use the spectrum of $\mathcal{L}_s$ as a basis again, writing U for the matrix of eigenvectors of $\mathcal{L}_s$ and expanding a matrix $M \in \mathbb{R}^{n\times p}$ as $M = UB$. Since U is given and orthonormal, estimating M is equivalent to estimating B. In an ideal scenario, if the true value Θ is given to us by an oracle, we could estimate B by minimizing one of the two objective functions:

$$\min_{B\in\mathbb{R}^{n\times p}} \operatorname{tr}((X-UB)^T(X-UB)) + \alpha\operatorname{tr}(B^T\Lambda B), \tag{36}$$

$$\min_{B\in\mathbb{R}^{n\times p}} \operatorname{tr}(\Theta(X-UB)^T(X-UB)) + \alpha\operatorname{tr}(B^T\Lambda B), \tag{37}$$

where $\Lambda = \operatorname{diag}(\tau_1, \tau_2, \cdots, \tau_n)$ is the diagonal matrix of eigenvalues of $\mathcal{L}_s$. It is easy to verify that (36) is equivalent to the mean estimation step (4) in the two-stage procedure (up to U), while (37) is equivalent to estimating the mean by maximizing the joint penalized likelihood (8) with Θ fixed at the true value. We can then treat (37) as an oracle estimate in the sense that it uses the true value of the covariance matrix. It serves as a benchmark for the best performance one could expect in estimating B (or equivalently M). Let $\hat{B}_1$ and $\hat{B}_2$ be the estimates from (36) and (37), respectively, and let $W_k = B - \hat{B}_k$, $k = 1, 2$ be the corresponding estimation error matrices. We then have the following result.

Proposition 2 *Under model (2), assume W_1 and W_2 are the errors defined above with the same tuning parameter α. Under the Assumption 3, if Θ is diagonally dominant with $\max_j \frac{\sum_{j'\neq j}|\Theta_{j'j}|}{\Theta_{jj}} \leq \rho < 1$, then there exist a matrix $\tilde{W}$ such that*

$$(1-\rho)\frac{1}{\bar{k}} \leq \frac{\|\tilde{W}\|_\infty}{\|W_2\|_\infty} \leq (1+\rho)\bar{k}$$

for the constant $\bar{k}$ in Assumption 3 and

$$W_1 - \tilde{W} = (I + \alpha\Lambda)^{-1}U^TE(I-\Theta).$$

where each row E is i.i.d from multivariate Gaussian $\mathcal{N}(0,\Sigma)$.

Proposition 2 shows $\tilde{W}$ and W_1 are stochastically equivalent while $\tilde{W}$ and W_2 are roughly the same in $\|\cdot\|_\infty$. Therefore, (36) and (37) are essentially equivalent in the sense of entrywise error bound, implying that $\hat{M}$ computed by GNC-lasso cannot be non-trivially improved by the oracle estimator under the true model with known Θ.

Proposition 2 makes an additional assumption on diagonal dominance of Θ, which is a relatively mild assumption consistent with others in this context. To see this, consider a general multivariate Gaussian vector $y \sim \mathcal{N}(0,\Sigma)$. Then we can write

$$y_j = \sum_{j'\neq j} \zeta^j_{j'} y_{j'} + \xi_j$$

where the vector $\zeta^j \in \mathbb{R}^p$ satisfies $\zeta^j_{j'} = -\frac{\Theta_{jj'}}{\Theta_{j'j'}}$ for $j' \neq j$ and $\zeta^j_j = 0$, and ξ_j is a Gaussian random variable with zero mean and variance equal to the conditional variance of y_j given $\{y_{j'}\}_{j'\neq j}$. Thus the diagonal dominance assumption of Proposition 2 is essentially assuming

$$\max_j \|\zeta^j\|_1 = \max_j \sum_{j'\neq j} |\zeta^j_{j'}| < \rho < 1.$$

This has the same form as Assumption 4 of Meinshausen and Bühlmann (2006), who proposed node-wise regression to estimate the Gaussian graphical model. There $\rho < 1$ is needed for node-wise regression to consistently estimate the graph structure (see Proposition 4 of Meinshausen and Bühlmann (2006)).

Remark 3 (Implications for iterative estimation) *If the iterative algorithm is used to obtain $\tilde{M}$ and $\tilde{\Theta}$, we know $\tilde{M}$ is the solution of* (37) *with Θ replaced by $\tilde{\Theta}$. Since $\tilde{\Theta}$ is only an estimate of Θ, we would not expect this estimator to work as well as the oracle estimator* (37)*. Since $\hat{M}$ cannot be improved by the oracle estimator, intuitively we make the conjecture that $\tilde{M}$ cannot significantly improve on $\hat{M}$ either.*

To prove Proposition 2, we need a few properties of Kronecker products. Recall that given two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$, their Kronecker product is defined to be an $(mp) \times (nq)$ matrix such that

$$A \otimes B = \begin{pmatrix} A_{11}B & A_{12}B & \cdots & A_{1n}B \\ A_{21}B & A_{22}B & \cdots & A_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1}B & A_{m2}B & \cdots & A_{mn}B \end{pmatrix}.$$

For a matrix A, define $\text{vec}(A)$ to be the column vector stacking all columns of A, $\text{vec}(A) = (A_{\cdot 1}, A_{\cdot 2}, \cdots, A_{\cdot n})$. Some standard properties we'll need, assuming the matrix dimensions match appropriately, are stated next.

$$\begin{aligned} &\text{vec}(AB) = (I_q \otimes A)\text{vec}(B), A \in \mathbb{R}^{n\times p}, B \in \mathbb{R}^{p\times q} \\ &\text{vec}(B^T \otimes A)\text{vec}(C) = \text{vec}(ACB), A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{p\times q}, C \in \mathbb{R}^{n\times p} \\ &(A \otimes B)(C \otimes D) = (AC) \otimes (BD) \\ &\text{tr}(ABA^T) = \text{vec}(A)^T(B \otimes I_n)\text{vec}(A) \\ &\qquad\qquad\quad = \text{vec}(A^T)^T(I_n \otimes B)\text{vec}(A^T), A \in \mathbb{R}^{n\times p}, B \in \mathbb{R}^{p\times p}. \end{aligned}$$

Proposition 3 *For the estimates W_1 from* (36) *and W_2 from* (37)*, we have*

$$W_1 I_p + \alpha\Lambda W_1 = \alpha\Lambda B + \tilde{E}, \tag{38}$$

$$W_2\Theta + \alpha\Lambda W_2 = \alpha\Lambda B + \dot{E}, \tag{39}$$

where $\tilde{E} = (\tilde{\epsilon}_{1\cdot}, \tilde{\epsilon}_{2\cdot}, \cdots, \tilde{\epsilon}_{n\cdot})$ and $\tilde{\epsilon}_{i\cdot} \sim \mathcal{N}(0, \Sigma)$ are i.i.d., and $\dot{E} = (\dot{\epsilon}_{1\cdot}, \dot{\epsilon}_{2\cdot}, \cdots, \dot{\epsilon}_{n\cdot})$, and $\dot{\epsilon}_{i\cdot} \sim \mathcal{N}(0, \Theta)$ are i.i.d. In particular, $\tilde{E} = -U^TE$ and $\dot{E} = -U^TE\Theta$.

Proof [Proof of Proposition 3] We only prove (39); the proof of (38) is exactly the same, with Θ replaced by I_p. The conclusion follows directly from writing out the quadratic optimiation solution after vectorizing all matrices. Specifically, the objective function (37) can be written as

$$
\begin{aligned}
&\mathrm{tr}(\Theta(X-UB)^T(X-UB)) + \alpha \mathrm{tr}(B^T \Lambda B) = \\
&\quad = \mathrm{vec}(X-UB)^T(\Theta \otimes I_n)\mathrm{vec}(X-UB) + \alpha \mathrm{vec}(B)^T(I_p \otimes \Lambda)\mathrm{vec}(B) \\
&\quad = \mathrm{vec}(UB)^T(\Theta \otimes I_n)\mathrm{vec}(UB) - 2\mathrm{vec}(UB)^T(\Theta \otimes I_n)\mathrm{vec}(X) + \alpha \mathrm{vec}(B)^T(I_p \otimes \Lambda)\mathrm{vec}(B) + const \\
&\quad = \mathrm{vec}(B)^T(I_p \otimes U^T)(\Theta \otimes I_n)(I_p \otimes U)\mathrm{vec}(B) - 2\mathrm{vec}(X)^T(\Theta \otimes I_n)(I_p \otimes U)\mathrm{vec}(B) \\
&\qquad\qquad + \alpha \mathrm{vec}(B)^T(I_p \otimes \Lambda)\mathrm{vec}(B) + const \\
&\quad = \mathrm{vec}(B)^T[(\Theta \otimes I_n) + \alpha(I_p \otimes \Lambda)]\mathrm{vec}(B) - 2\mathrm{vec}(X)^T(\Theta \otimes U)\mathrm{vec}(B) + const.
\end{aligned}
$$

The minimizer of this quadratic function satisfies

$$[(\Theta \otimes I_n) + \alpha(I_p \otimes \Lambda)]\mathrm{vec}(\hat{B}) = (\Theta \otimes U^T)\mathrm{vec}(X).$$

Substituting $X = UB + E$ into the estimating equation gives

$$
\begin{aligned}
[(\Theta \otimes I_n) + \alpha(I_p \otimes \Lambda)]\mathrm{vec}(\hat{B}) &= (\Theta \otimes U^T)\mathrm{vec}(UB + E) \\
&= (\Theta \otimes U^T)(I_p \otimes U)\mathrm{vec}(B) + (\Theta \otimes U^T)\mathrm{vec}(E) \\
&= (\Theta \otimes I_n)\mathrm{vec}(B) + \mathrm{vec}(U^T E \Theta),
\end{aligned}
$$

and therefore

$$(\Theta \otimes I_n)\mathrm{vec}(W) + \alpha(I_p \otimes \Lambda)\mathrm{vec}(W) = \alpha(I_p \otimes \Lambda)\mathrm{vec}(B) - \mathrm{vec}(U^T E \Theta).$$

We then get

$$\mathrm{vec}(W\Theta) + \alpha \mathrm{vec}(\Lambda W) = \alpha \mathrm{vec}(\Lambda B) - \mathrm{vec}(U^T E \Theta).$$

This is equivalent to (39) by noting that $\dot{E} = -U^T E \Theta$. ■

Now we show W_1 and W_2 are essentially equivalent estimation errors. Define two additional estimating equations as below:

$$W_3 I_p + \alpha \Lambda W_3 = \alpha \Lambda B + \dot{E} \tag{40}$$

$$W_4 \mathrm{diag}(\Theta) + \alpha \Lambda W_4 = \alpha \Lambda B + \dot{E} \tag{41}$$

The error equation (40) corresponds to the situation when we carry p separate Laplacian smoothing estimations. The error equation (40) is also from p separate Laplacian smoothing but it adjusts the weight each variable to be proportional to $1/\Theta_{jj}$, which can be seen as W_2 approximation after ignoring off-diagonal elements of Θ. Intuitively, when the off-diagonal elements are small, W_2 should not be very different from W_4, and when the diagonal elements of Θ are similar, as in Assumption 3, W_3 and W_4 should also be similar. The following proposition formalizes this intuition under the assumption that Θ is diagonally dominant. We can then conclude that using the true Θ in (37) does not really bring improvement and W_1, W_2, W_3, and W_4 are all essentially equivalent.

Proposition 4 *Assume W_2, W_3, and W_4 are the estimation errors from* (39), (40) *and* (41), *respectively, with the same α. If Θ is diagonally dominant with $\max_j \frac{\sum_{j'\neq j}|\Theta_{j'j}|}{\Theta_{jj}} \leq \rho < 1$, then*

$$(1-\rho)\min(1,\min_j \Theta_{jj}) \leq \frac{\|W_3\|_\infty}{\|W_2\|_\infty} \leq (1+\rho)\max(1,\max_j \Theta_{jj}). \tag{42}$$

In particular, under Assumption 3,

$$(1-\rho)\frac{1}{\bar{k}} \leq \frac{\|W_3\|_\infty}{\|W_2\|_\infty} \leq (1+\rho)\bar{k}$$

for a constant $\bar{k}$.

Proof [Proof of Proposition 4] Directly from the definition, we have

$$\begin{aligned} W_{3,ij} &= \frac{1}{1+\alpha\tau_i}(\alpha\tau_i B_{ij} + \dot{E}_{ij})\,, \\ W_{4,ij} &= \frac{1}{\Theta_{jj}+\alpha\tau_i}(\alpha\tau_i B_{ij} + \dot{E}_{ij})\,. \end{aligned}$$

This implies that for any i, j and an arbitrary α,

$$\min(1,\min_j \Theta_{jj}) \leq \frac{W_{3,ij}}{W_{4,ij}} = \frac{\Theta_{jj}+\alpha\tau_i}{1+\alpha\tau_i} \leq \max(1,\max_j \Theta_{jj}). \tag{43}$$

We next show that under the assumption of diagonal dominance of Θ, even W_2 cannot do much better. For each $j = 1, 2, \cdots, p$, from (39),

$$W_2\Theta_{\cdot j} + \alpha W_{2,\cdot j} = (\Theta_{jj}I + \alpha\Lambda)W_{2,\cdot j} + \Theta_{jj}\sum_{i\neq j}\frac{\Theta_{ij}}{\Theta_{jj}}W_{2,\cdot i} = \alpha\Lambda B_{\cdot j} + \dot{E}_{\cdot j}\ .$$

Therefore, we have

$$W_{2,\cdot j} + (\Theta_{jj}I+\alpha\Lambda)^{-1}\Theta_{jj}\sum_{i\neq j}\frac{\Theta_{ij}}{\Theta_{jj}}W_{2,\cdot i} = \alpha(\Theta_{jj}I+\alpha\Lambda)^{-1}\Lambda B_{\cdot j} + (\Theta_{jj}I+\alpha\Lambda)^{-1}\dot{E}_{\cdot j} = W_{4,\cdot j} \tag{44}$$

in which the last equation comes from (41). By triangle inequality, (44) leads to

$$\|W_{2,\cdot j}\|_\infty \leq \|W_{4,\cdot j}\|_\infty + \|(\Theta_{jj}I+\alpha\Lambda)^{-1}\Theta_{jj}\sum_{i\neq j}\frac{\Theta_{ij}}{\Theta_{jj}}W_{2,\cdot i}\|_\infty \leq \|W_{4,\cdot j}\|_\infty + \sum_{i\neq j}\frac{|\Theta_{ij}|}{\Theta_{jj}}\max_i \|W_{2,\cdot i}\|_\infty. \tag{45}$$

Taking the maximum over j on both sides, we have

$$\|W_2\|_\infty \leq \|W_4\|_\infty + \rho\|W_2\|_\infty. \tag{46}$$

Similarly using triangle inequality in the other direction, we get

$$1-\rho \leq \frac{\|W_4\|_\infty}{\|W_2\|_\infty} \leq 1+\rho\,.$$

Combining this with (43), we get

$$(1-\rho)\min(1, \min_j \Theta_{jj}) \leq \frac{\|W_3\|_\infty}{\|W_2\|_\infty} \leq (1+\rho)\max(1, \max_j \Theta_{jj}).$$

Note that (45) holds if we replace $\|\cdot\|_\infty$ by other norms. For example, if we take the L_1 norm instead, we get a similar bound in $\|\cdot\|_{1,1}$. ■

Now we are ready to prove Proposition 2.

Proof [Proof of Proposition 2] By taking $\tilde{W} = W_3$ and using the conclusion of Proposition 4, the first half of Proposition 2 directly follows. Subtracting (38) from (40) leads to

$$(I_n + \alpha\Lambda)(\tilde{W} - W_1) = (I_n + \alpha\Lambda)(W_3 - W_1) = \dot{E} - \tilde{E} = U^T E(I - \Theta)\,,$$

and therefore

$$\tilde{W} - W_1 = (I_n + \alpha\Lambda)^{-1} U^T E(I - \Theta).$$

■