Bayesian Analysis (2015) **10**, Number 2, pp. 351–377

Scaling It Up: Stochastic Search Structure Learning in Graphical Models

Hao Wang*

Abstract. Gaussian concentration graph models and covariance graph models are two classes of graphical models that are useful for uncovering latent dependence structures among multivariate variables. In the Bayesian literature, graphs are often determined through the use of priors over the space of positive definite matrices with fixed zeros, but these methods present daunting computational burdens in large problems. Motivated by the superior computational efficiency of continuous shrinkage priors for regression analysis, we propose a new framework for structure learning that is based on continuous spike and slab priors and uses latent variables to identify graphs. We discuss model specification, computation, and inference for both concentration and covariance graph models. The new approach produces reliable estimates of graphs and efficiently handles problems with hundreds of variables.

Keywords: Bayesian inference, Bi-directed graph, Block Gibbs, Concentration graph models, Covariance graph models, Credit default swap, Undirected graph, Structural learning.

1 Introduction

Graphical models use graph structures for modeling and making statistical inferences regarding complex relationships among many variables. Two types of commonly used graphs are undirected graphs, which represent conditional dependence relationships among variables, and bi-directed graphs, which encode marginal dependence among variables. Structure learning refers to the problem of estimating unknown graphs from the data and is usually carried out by sparsely estimating the covariance matrix of the variables by assuming that the data follow a multivariate Gaussian distribution. Under the Gaussian assumption, undirected graphs are determined by zeros in the concentration matrix and their structure learning problems are thus referred to as concentration graph models; bi-directed graphs are determined by zeros in the covariance matrix and their structure learning problems are thus referred to as covariance graph models. This work concerns structure learning in both concentration and covariance graph models.

Classical methods for inducing sparsity often rely on penalized likelihood approaches (Banerjee et al., 2008; Yuan and Lin, 2007; Bien and Tibshirani, 2011; Wang, 2014). Model fitting then uses deterministic optimization procedures such as coordinate descents. Thresholding is another popular method for the sparse estimation of covariance matrices for covariance graph models (Bickel and Levina, 2008; Rothman et al., 2009); however, there is no guarantee that the resulting estimator is always positive definite.

*Department of Epidemiology and Biostatistics, Michigan State University, East Lansing, Michigan 48824, U.S.A., haowang@sc.edu

 DOI: 10.1214/14-BA916

Bayesian methods for imposing sparsity require the specification of priors over the space of positive definite matrices constrained by fixed zeros. Under such priors, model determination is then carried out through stochastic search algorithms to explore a discrete graphical model space. The inherent probabilistic nature of the Bayesian framework permits estimation via decision-theoretical principles, addresses parameter and model uncertainty, and provides a global characterization of the parameter space. It also encourages the development of modular structures that can be integrated with more complex systems.

A major challenge in Bayesian graphical models is their computation. Although substantial progress in computation for these two graphical models has been made in recent years, scalability with dimensions remains a significant issue, hindering the ability to adapt these models to the growing demand for higher dimensional problems. Recently published statistical papers on these two graphical models either focus on small problems or report long computing times. In concentration graph models, Dobra et al. (2011) report that it takes approximately 1.5 days to fit a problem of 48 nodes on a dual-core 2.8 Ghz computer under C; Wang and Li (2012) report approximately two days for a 100 node problem under MATLAB; and Cheng and Lenkoski (2012) report a computing time of 1–20 seconds per one-edge update for a 150 node problem using a 400 core server with 3.2 GHz CPU under R and C++. In covariance graph models, Silva and Ghahramani (2009) fit problems up to 13 nodes and conclude that "further improvements are necessary for larger problems."

To scale up with dimension, this paper develops a new approach called stochastic search structure learning (SSSL) to efficiently determine covariance and concentration graph models. The central idea behind SSSL is the use of continuous shrinkage priors characterized by binary latent indicators in order to avoid the normalizing constant approximation and to allow block updates of graphs. The use of continuous shrinkage priors contrasts point-mass priors at zeros that are used essentially by all existing methods for Bayesian structure learning in these two graphical models.

The motivation for the SSSL comes from the successful developments of continuous shrinkage priors in several related problems. In regression analysis, continuous shrinkage priors were used in the seminal paper by George and McCulloch (1993) in the form of a two component normal mixture for selecting important predictors and these priors have recently garnered substantial research attention as a computationally attractive alternative for regularizing many regression coefficients (e.g., Park and Casella 2008; Griffin and Brown 2010; Armagan et al. 2013). In estimation of covariance matrices, they are used for regularizing concentration elements and have been shown to provide fast and accurate estimates of covariance matrices (Wang, 2012; Khondker et al., 2013). In factor analysis, they are used instead of point-mass priors (Carvalho et al., 2008) for modeling factor loading matrices, efficiently handling hundreds of variables (Bhattacharya and Dunson, 2011).

Nevertheless, the current work is fundamentally different from the aforementioned works. The research focus here is the structure learning of graphs, which is distinct from regression analysis, factor analysis, and the pure covariance estimation problem

that solely performs parameter estimation without the structure learning of graphs. Although continuous shrinkage priors generally perform very well in these problems, little is known about their performance in problems of structure learning. Because graphs are directly determined by covariance matrices, the positive definiteness of any covariance matrices poses methodological challenges to investigating prior properties, as well as to the construction of efficient stochastic search algorithms. The paper's main contributions are the development and exploration of two classes of continuous shrinkage priors for learning undirected and bi-directed graphs, as well as two efficient block Gibbs samplers for carrying out the corresponding structure learning tasks that fit problems of one or two hundred variables within a few minutes.

2 Background on graphical models

Assume that $\mathbf{y} = (y_1, y_2, \ldots, y_p)'$ is a p-dimensional random vector following a multivariate normal distribution $\mathrm{N}(0, \boldsymbol{\Sigma})$ with mean of zero and covariance matrix $\boldsymbol{\Sigma} \equiv (\sigma_{ij})$. Let $\boldsymbol{\Omega} \equiv (\omega_{ij}) = \boldsymbol{\Sigma}^{-1}$ be the concentration matrix. Covariance and concentration graph models are immediately related to $\boldsymbol{\Sigma}$ and $\boldsymbol{\Omega}$, respectively. Let $\mathbf{Y}$ be the $n \times p$ data matrix consisting of n independent samples of $\mathbf{y}$ and let $\mathbf{S} = \mathbf{Y}'\mathbf{Y}$. The theory and existing methods for structure learning are briefly reviewed in the next two sections.

2.1 Concentration graph models

Concentration graph models (Dempster, 1972) consider the concentration matrix $\boldsymbol{\Omega}$ and encode conditional dependence using an undirected graph $G = (V, E)$, where $V = \{1, 2, \ldots, p\}$ is a non-empty set of vertices and $E \subseteq \{(i, j) : i < j\}$ is a set of edges representing unordered pairs of vertices. The graph G can also be indexed by a set of $p(p-1)/2$ binary variables $\mathbf{Z} = (z_{ij})_{i<j}$, where $z_{ij} = 1$ or 0 according to whether edge (i, j) belongs to E and not. Theoretically, the following properties are equivalent:

$$z_{ij} = 0 \quad \Leftrightarrow \quad (i, j) \notin E \quad \Leftrightarrow \quad y_i \perp\!\!\!\perp y_j \mid \mathbf{y}_{-(ij)} \quad \Leftrightarrow \quad \omega_{ij} = 0,$$

where $\mathbf{y}_{-(ij)}$ is the random vector containing all elements in $\mathbf{y}$ except for y_i and y_j, and "$\Leftrightarrow$" reads as "if and only if".

In the Bayesian paradigm, concentration graph models are usually modeled through hierarchical priors consisting of the following: *(i)* the conjugate G-Wishart prior $\boldsymbol{\Omega} \sim \mathrm{W}_G(b, \mathbf{D})$ (Dawid and Lauritzen, 1993; Roverato, 2002) for $\boldsymbol{\Omega}$ given the graph $\mathbf{Z}$; and *(ii)* independent Bernoulli priors for each edge-inclusion indicator z_{ij} with inclusion probability π:

$$p(\boldsymbol{\Omega} \mid \mathbf{Z}) = I_{GW}(b, \mathbf{D}, \mathbf{Z})^{-1} |\boldsymbol{\Omega}|^{\frac{b-2}{2}} \exp\{-\frac{1}{2}\mathrm{tr}(\mathbf{D}\boldsymbol{\Omega})\} 1_{\{\boldsymbol{\Omega} \in M^+(\mathbf{Z})\}}, \tag{1}$$

$$p(\mathbf{Z}) = \prod_{i<j} \left\{\pi^{z_{ij}} (1-\pi)^{1-z_{ij}}\right\}, \tag{2}$$

where b is the degrees-of-freedom parameter, $\mathbf{D}$ is the location parameter, $I_{GW}(b, \mathbf{D}, \mathbf{Z})$ is the normalizing constant, and $M^+(\mathbf{Z})$ is the cone of symmetric positive definite

matrices with off-diagonal entries $\omega_{ij} = 0$ whenever $z_{ij} = 0$. As for the hyperparameters, common choices are $b = 3, \mathbf{D} = \mathbf{I}_p$ with $\mathbf{I}_p$ being the $p \times p$ identity matrix and $\pi = 2/(p-1)$ (Jones et al., 2005). Under (1)–(2), some methods (e.g., Jones et al. 2005; Scott and Carvalho 2008; Lenkoski and Dobra 2011) learn $\mathbf{Z}$ directly through its posterior distribution over the model space $p(\mathbf{Z} \mid \mathbf{Y}) \propto p(\mathbf{Y} \mid \mathbf{Z})p(\mathbf{Z})$. Other methods learn $\mathbf{Z}$ indirectly through sampling over the joint space of graphs and concentration matrices $p(\mathbf{\Omega}, \mathbf{Z} \mid \mathbf{Y})$ (Giudici and Green, 1999; Dobra et al., 2011; Wang and Li, 2012). Regardless of the types of algorithms, two shared features of these methods cause the framework (1)–(2) to be inefficient for larger p problems. The first of these features is that graphs are updated in a one-edge-at-a-time manner, meaning that sweeping through all possible edges requires a loop of $O(p^2)$ iterations. The second feature is that the normalizing constant $I_{GW}(b, \mathbf{D}, \mathbf{Z})$ for non-decomposable graphs requires approximation. The commonly used Monte Carlo approximation proposed by Atay-Kayis and Massam (2005) is not only unstable in some situations but also requires a matrix completion step of time complexity $O(Mp^4)$ for M Monte Carlo samples, making these methods unacceptably slow in large graphs. Recent works by Wang and Li (2012) and Cheng and Lenkoski (2012) propose the use of exchange algorithms to avoid the Monte Carlo approximation. However, the computational burden remains daunting; empirical experiments in these papers suggest it would take several days to complete the fitting for problems of $p \approx 100$ on a desktop computer.

In the classical formulation, concentration graphs are induced by imposing a graphical lasso penalty on $\mathbf{\Omega}$ in order to encourage zeros in the penalized maximum likelihood estimates of $\mathbf{\Omega}$ (e.g., Yuan and Lin 2007; Friedman et al. 2008; Rothman et al. 2008). In particular, the standard graphical lasso problem is to maximize the penalized log-likelihood

$$\log(\det \mathbf{\Omega}) - \operatorname{tr}(\frac{\mathbf{S}}{n}\mathbf{\Omega}) - \rho||\mathbf{\Omega}||_1,$$

over the space of positive definite matrices M^+, with $\rho \geq 0$ as the shrinkage parameter and $||\mathbf{\Omega}||_1 = \sum_{1 \leq i,j \leq p} |\omega_{ij}|$ as the L_1-norm of $\mathbf{\Omega}$. The graphical lasso problem has a Bayesian interpretation (Wang, 2012). Its estimator is equivalent to the maximum a posteriori estimation under the following prior for $\mathbf{\Omega}$:

$$p(\mathbf{\Omega}) \quad = \quad C^{-1} \prod_{1 \leq i,j \leq p} \left\{ \exp(-\lambda|\omega_{ij}|) \right\} 1_{\mathbf{\Omega} \in M^+}, \tag{3}$$

where C is the normalizing constant. By exploiting the scale mixture of normal representation, Wang (2012) shows that fitting (3) is very efficient using block Gibbs samplers for up to the lower hundreds of variables.

A comparison between the two Bayesian methods (1)–(2) and (3) helps to explain the intuition behind the proposed SSSL. Model (1)–(2) explicitly treats a graph $\mathbf{Z}$ as an unknown parameter and considers its posterior distribution, which leads to straightforward Bayesian inferences about graphs. However, it is slow to run due to the one-edge-at-a-time updating and the normalizing constant approximation. In contrast, Model (3) uses continuous priors, enabling a fast block Gibbs sampler that updates $\mathbf{\Omega}$ one column at a time and avoids normalizing constant evaluation. However, no graphs are used in the

formulation, and thus this approach does not constitute a formal Bayesian treatment of structure learning. Still, a better approach might be developed by using the best aspects of the two methods. That is, such a method would allow explicit structure learning, as in (1)–(2), while maintaining good scalability, as in (3). This possibility is exactly the key of SSSL. Similar insights also apply to the covariance graph models described below.

2.2 Covariance graph models

Covariance graph models (Cox and Wermuth, 1993) consider the covariance matrix $\mathbf{\Sigma}$ and encode the marginal dependence using a bi-directed graph $G = (V, E)$, where each edge has bi-directed arrows instead of the full line used by an undirected graph. Similar to concentration graph models, the covariance graph G can also be indexed by binary variables $\mathbf{Z} = (z_{ij})_{i<j}$. Theoretically, the following properties are equivalent:

$$z_{ij} = 0 \quad \Leftrightarrow \quad (i,j) \notin E \quad \Leftrightarrow \quad y_i \perp\!\!\!\perp y_j \quad \Leftrightarrow \quad \sigma_{ij} = 0.$$

In the Bayesian framework, structure learning again relies on the general hierarchical priors $p(\mathbf{\Sigma}, \mathbf{Z}) = p(\mathbf{\Omega} \mid \mathbf{Z})p(\mathbf{Z})$. For $p(\mathbf{\Sigma} \mid \mathbf{Z})$, Silva and Ghahramani (2009) propose the conjugate G-inverse Wishart prior $\mathbf{\Sigma} \sim \mathrm{IW}_G(b, \mathbf{D})$ with the density as:

$$p(\mathbf{\Sigma} \mid \mathbf{Z}) \quad = \quad I_{GIW}^{-1}(b, \mathbf{D}, \mathbf{Z})|\mathbf{\Sigma}|^{-\frac{b+2p}{2}} \exp\{-\frac{1}{2}\mathrm{tr}(\mathbf{D}\mathbf{\Sigma}^{-1})\}1_{\mathbf{\Sigma} \in M^+(\mathbf{Z})}, \tag{4}$$

where b specifies the degrees of freedom, $\mathbf{D}$ is the location parameter, and $I_{GIW}(b, \mathbf{D}, \mathbf{Z})$ is the normalizing constant. Structure learning is then carried out through the marginal likelihood function $p(\mathbf{Y} \mid \mathbf{Z}) = (2\pi)^{-np/2} I_{GIW}(b+n, \mathbf{D}+\mathbf{S}, \mathbf{Z}) / I_{GIW}(b, \mathbf{D}, \mathbf{Z})$. Unfortunately, the key quantity of the normalizing constant $I_{GIW}(b, \mathbf{D}, \mathbf{Z})$ is analytically intractable, even for decomposable graphs. Silva and Ghahramani (2009) propose a Monte Carlo approximation via an importance sampling algorithm, which becomes computationally infeasible for p beyond a few dozen. Their empirical experiments are thus limited to small problems (i.e., $p < 20$). Later, Khare and Rajaratnam (2011) investigate a broad class of priors for decomposable covariance graph models that embed (4) as a special case. They also derive closed-form normalizing constants for decomposable homogeneous graphs which account for only a tiny portion of the overall graph space. Despite these advances, the important question of scalability to higher dimensional problems remains almost untouched.

In the classical framework, the earlier literature focuses on maximum likelihood estimates and likelihood ratio test procedures (e.g., Kauermann 1996; Wermuth et al. 2006; Chaudhuri et al. 2007). Later, two general types of approaches are proposed to estimate zeros in the covariance elements. The first is the thresholding procedure, which sets σ_{ij} to be zero if its sample estimate is below a threshold (Bickel and Levina, 2008; Rothman et al., 2009; Cai and Liu, 2011). Another approach is motivated by the lasso-type procedures. Bien and Tibshirani (2011) propose a covariance graphical lasso procedure for simultaneously estimating covariance matrix and marginal dependence structures. Their method is to minimize the following objective function:

$$\log(\det \mathbf{\Sigma}) + \mathrm{tr}(\frac{\mathbf{S}}{n}\mathbf{\Sigma}^{-1}) + \rho||\mathbf{\Sigma}||_1, \tag{5}$$

over the space of positive definite matrices M^+, with $\rho \geq 0$ as the shrinkage parameter. In comparison with thresholding and likelihood ratio testing methods, this approach has the advantage of guaranteeing the positive definiteness of the estimated $\mathbf{\Sigma}$. Although a Bayesian version of (5) has not been explored previously, its derivation is straightforward through the prior

$$p(\mathbf{\Sigma}) \quad = \quad C^{-1} \prod_{1 \leq i,j \leq p} \{ \exp(-\lambda|\sigma_{ij}|) \} 1_{\mathbf{\Sigma} \in M^+}, \tag{6}$$

In light of the excellent performance of the Bayesian concentration graphical lasso (3) reported in Wang (2012), we hypothesize that (6) shares similar performances. In fact, we have developed a block Gibbs sampler for (6) and found that it gives a shrinkage estimation of $\mathbf{\Sigma}$ and is computationally efficient, although it provides no explicit treatment of the graph $\mathbf{Z}$. The detailed algorithm and results are not reported in this paper but are available upon request. Comparing (4) and (6) suggests that again, the different strengths of (4) and (6) might be combined to provide a better approach for structure learning in covariance graph models.

3 Continuous spike and slab priors for positive definite matrices

Let $\mathbf{A} = (a_{ij})_{p \times p}$ denote a p-dimensional covariance or concentration matrix; that is, $\mathbf{A} = \mathbf{\Sigma}$ or $\mathbf{\Omega}$. SSSL uses the following new prior for $\mathbf{A}$:

$$p(\mathbf{A}) = \{C(\theta)\}^{-1} \prod_{i<j} \left\{ (1-\pi) \mathrm{N}(a_{ij} \mid 0, v_0^2) + \pi \mathrm{N}(a_{ij} \mid 0, v_1^2) \right\} \prod_i \mathrm{Exp}(a_{ii} \mid \frac{\lambda}{2}) 1_{\{\mathbf{A} \in M^+\}}, \tag{7}$$

where $\mathrm{N}(a \mid 0, v^2)$ is the density function of a normal random variable with mean 0 and variance v^2 evaluated at a, $\mathrm{Exp}(a \mid \lambda)$ represents the exponential density function of the form $p(a) = \lambda \exp(-\lambda x) 1_{a>0}$, and $1_{\{.\}}$ is the indicator function. The parameter spaces are $v_0 > 0$, $v_1 > 0$, $\lambda > 0$ and $\pi \in (0,1)$, and the set of all parameters is denoted as $\theta = \{v_0, v_1, \pi, \lambda\}$. The values of v_0 and v_1 are further set to be small and large, respectively. The term $C(\theta)$ represents the normalizing constant that ensures the integration of the density function $p(\mathbf{A})$ over the space M^+ is one, and it depends on θ. The first product symbol multiplies $p(p-1)/2$ terms of two-component normal mixture densities for the off-diagonal elements, connecting this prior to the classical and Bayesian graphical lasso methods through the familiar framework of normal mixture priors for a_{ij}. The second product symbol multiplies p terms of exponential densities for the diagonal elements. The two-component normal mixture density plays a critical role in structure learning, as will be clear below.

Prior (7) can be defined by introducing binary latent variables $\mathbf{Z} \equiv (z_{ij})_{i<j} \in \mathcal{Z} \equiv \{0,1\}^{p(p-1)/2}$ and a hierarchical model:

$$p(\mathbf{A} \mid \mathbf{Z}, \theta) = \{C(\mathbf{Z}, v_0, v_1, \lambda)\}^{-1} \prod_{i<j} \mathrm{N}(a_{ij} \mid 0, v_{z_{ij}}^2) \prod_i \mathrm{Exp}(a_{ii} \mid \frac{\lambda}{2}), \tag{8}$$

$$p(\mathbf{Z} \mid \theta) = \{C(\theta)\}^{-1} C(\mathbf{Z}, v_0, v_1, \lambda) \prod_{i<j} \{\pi^{z_{ij}} (1-\pi)^{1-z_{ij}}\}, \tag{9}$$

where $v_{z_{ij}} = v_0$ or v_1 if $z_{ij} = 0$ or 1. The intricacy here is the two terms of $C(\mathbf{Z}, v_0, v_1, \lambda)$. Note that $C(\mathbf{Z}, v_0, v_1, \lambda) \in (0, 1)$ because it is equal to the integration of the product of normal densities and exponential densities over a constrained space M^+. Thus, (8) and (9) are proper distributions. The joint distribution of $(\mathbf{A}, \mathbf{Z})$ acts to cancel out the two terms of $C(\mathbf{Z}, v_0, v_1, \lambda)$ and results in a marginal distribution of $\mathbf{A}$, as in (7).

The rationale behind using $\mathbf{Z}$ for structure learning is as follows. For an appropriately chosen small value of v_0, the event $z_{ij} = 0$ means that a_{ij} comes from the concentrated component $\mathrm{N}(0, v_0^2)$, and so a_{ij} is likely to be close to zero and can reasonably be estimated as zero. For an appropriately chosen large value of v_1, the event $z_{ij} = 1$ means that a_{ij} comes from the diffuse component $\mathrm{N}(0, v_1^2)$ and so a_{ij} can be estimated to be substantially different from zero. Because zeros in $\mathbf{A}$ determine missing edges in graphs, the latent binary variables $\mathbf{Z}$ can be viewed as edge-inclusion indicators. Given data $\mathbf{Y}$, the posterior distribution of $\mathbf{Z}$ provides information about graphical model structures. The remaining questions are then how to specify parameters θ and how to perform posterior computations.

3.1 Choice of π

From (9), the hyperparameter π controls the prior distribution of the edge-inclusion indicators in $\mathbf{Z}$. The choice of π should thus reflect the prior belief about what the graphs will be in the final model. In practice, such prior information is often summarized via the marginal prior edge-inclusion probability $\Pr(z_{ij} = 1)$. Specifically, a prior for $\mathbf{Z}$ is chosen such that the implied edge-inclusion probability of edge (i, j) meets the prior belief about the chance of the existence of edge (i, j). For example, the common choice $\Pr(z_{ij} = 1) = 2/(p-1)$ reflects the prior belief that the expected number of edges is approximately $\binom{p}{2} \Pr(z_{ij} = 1) = p$. Another important reason that $\Pr(z_{ij} = 1)$ is used for calibrating priors over $\mathbf{Z}$ is that the posterior inference about $\mathbf{Z}$ is usually based upon the marginal posterior probability of $\Pr(z_{ij} = 1 \mid \mathbf{Y})$. For example, the median probability graph, the graph consisting of those edges whose marginal edge-inclusion probability exceeds 0.5, is often used to estimate G (Wang, 2010). Focusing on the marginal edge-inclusion probability allows us to understand how the posterior truly responds to the data.

Calibrating π according to $\Pr(z_{ij} = 1)$ requires knowledge of the relation between $\Pr(z_{ij} = 1)$ and π. From (9), the explicit form of $\Pr(z_{ij} = 1)$ as a function of π is unavailable because of the intractable term $C(\mathbf{Z}, v_0, v_1, \lambda)$. A comparison between (9) and (2) helps illustrate the issue. Removing $C(\mathbf{Z}, v_0, v_1, \lambda)$ from (9) turns it into (2) but will not cancel out the term $C(\mathbf{Z}, v_0, v_1, \lambda)$ in (8) for the posterior distribution of $\mathbf{Z}$. Tedious and unstable numerical integration is then necessary to evaluate $C(\mathbf{Z}, v_0, v_1, \lambda)$ at each iteration of sampling $\mathbf{Z}$. Inserting $C(\mathbf{Z}, v_0, v_1, \lambda)$ into (9) cancels $C(\mathbf{Z}, v_0, v_1, \lambda)$ in (8) in the posterior, thus facilitating computation, yet concerns might be raised

about such a "fortunate" cancelation. For example, Murray (2007) notes that a prior that cancels out an intractable normalizing constant in the likelihood would depend on the number of data points and would also be so extreme that it would dominate posterior inferences. These two concerns appear to be not problematic in our case. The prior (9) is for the hyperparameter $\mathbf{Z}$, rather than for the parameter directly involved in the likelihood; thus it does not depend on sample size. Instead, the prior also only introduces mild bias without dominating the inferences, as shown below.

To investigate whether the cancelation of $C(\mathbf{Z}, v_0, v_1, \lambda)$ causes the prior to be too extreme, we compute $\Pr(z_{ij} = 1)$ numerically from Monte Carlo samples generated by the algorithm in Section 4.1. In (8)–(9), we first fix $\pi = 2/(p-1), v_0 = 0.05$, and $\lambda = 1$, and then vary the dimension $p \in \{50, 100, 150, 200, 250\}$ and $v_1 = hv_0$ with $h \in \{10, 50, 100\}$. Panel (a) of Figure 1 displays these estimated $\Pr(z_{ij} = 1)$ as a function of p for different h values. As a reference, the curve $\Pr(z_{ij} = 1) = 2/(p-1)$ is also plotted. The most noticeable pattern is that all three curves representing the implied $\Pr(z_{ij} = 1)$ from (9) are below the reference curve, suggesting that there is a downward bias introduced by $C(\mathbf{Z}, v_0, v_1, \lambda)$ on $\Pr(z_{ij} = 1)$. The bias is introduced by the fact that the positive definite constraint on $\mathbf{A}$ favors a small v_0, specified by $z_{ij} = 0$, over a large v_1, specified by $z_{ij} = 1$. We also see that the bias is larger at larger values of h, at which the impact of positive definite constraints is more significant. Next, we fix $p = 150, h = 50$, and $\lambda = 1$ and vary $v_0 \in \{0.02, 0.05, 0.1\}$ and $\pi \in \{2/(p-1), 4/(p-1), 6/(p-1), 8/(p-1), 10/(p-1)\}$. Panel (b) of Figure 1 displays these implied $\Pr(z_{ij} = 1)$ as a function of π for different v_0 values. Again, as a reference, the curve $\Pr(z_{ij} = 1) = \pi$ is plotted. The downward bias of $\Pr(z_{ij} = 1)$ relative to π continues to exist and is larger at larger values of v_0 or π because the positive definite constraint on $\mathbf{A}$ forces $z_{ij} = 0$ to be chosen more often when v_0 or π is large.

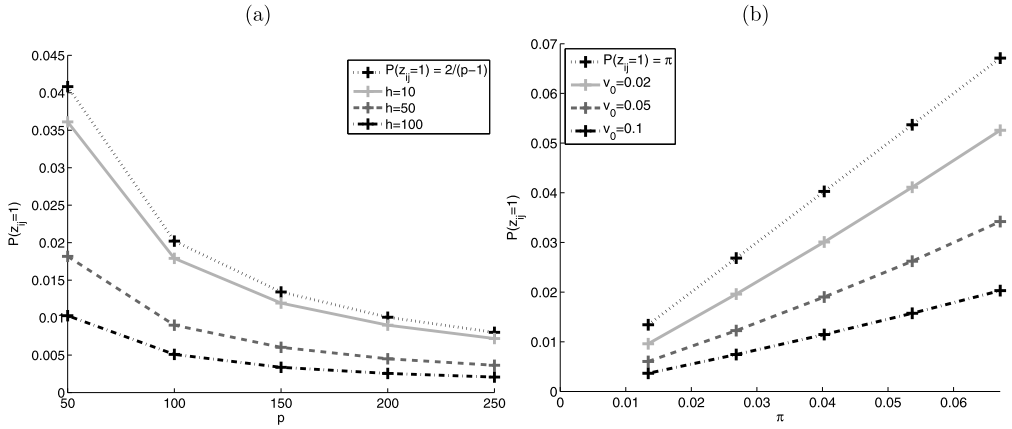

Figure 1: The implied prior marginal edge-inclusion probability $\Pr(z_{ij} = 1)$ from the prior (8)–(9) as a function of p at different h (left) and a function of π at different v_0 (right), together with two reference curves of $\Pr(z_{ij} = 1) = 2/(p-1)$ (left) and of $\Pr(z_{ij} = 1) = \pi$ (right).

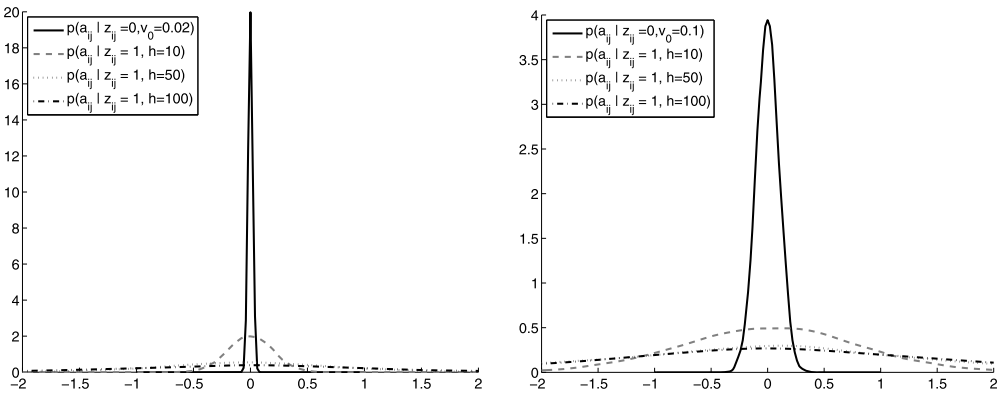

Figure 2: The univariate density of $p(a_{ij} \mid z_{ij})$ for different values of h and v_0.

Nevertheless, the downward bias seems to be gentle, as $\Pr(z_{ij} = 1)$ is never extremely small; consequently the prior (8)–(9) is able to let the data reflect the $\mathbf{Z}$ if the likelihood is strong.

Another concern is that the lack of analytical relation between $\Pr(z_{ij} = 1)$ and π might raise challenges against the incorporation of prior information about $\Pr(z_{ij} = 1)$ into π. This problem can be side-stepped by prior simulation and interpolation. Take a $p = 150$ node problem as an example. If the popular choice $\Pr(z_{ij} = 1) = 2/(p-1) = 0.013$ is desirable, interpolating the points $(\pi, \Pr(z_{ij} = 1))$ in Panel (b) of Figure 1 suggests that π should be set approximately at 0.018, 0.027, and 0.048 for $v_0 = 0.02$, 0.05, and 0.1, respectively. Our view is that obtaining these points under prior (8)–(9) is much faster than evaluating $C(\mathbf{Z}, v_0, v_1, \lambda)$ at each configuration of $\mathbf{Z}$ under the traditional prior (2).

3.2 Choice of v_0 and v_1

From (8), the choice of v_0 should be such that if the data supports $z_{ij} = 0$ over $z_{ij} = 1$, then a_{ij} is small enough to be replaced by zero. The choice of v_1 should be such that, if the data favor $z_{ij} = 1$ against $z_{ij} = 0$, then a_{ij} can be accurately estimated to be substantially different from zero. One general strategy for choosing v_0 and v_1, as recommended by George and McCulloch (1993), is based on the concept of practical significance. Specifically, suppose a small δ can be chosen such that $|a_{ij}| < \delta$ might be regarded as practically insignificantly different from zero. Incorporating such a prior belief is then achieved by choosing v_0 and v_1, such that the density $p(a_{ij} \mid z_{ij} = 0)$ is larger than the density $p(a_{ij} \mid z_{ij} = 1)$, precisely within the interval $(-\delta, \delta)$. An explicit expression of v_0 as a function of δ and h can be derived when $p(a_{ij} \mid z_{ij})$ are normal. However, the implied distribution $p(a_{ij} \mid z_{ij})$ from (8)–(9) is neither normal nor even analytically tractable. A numerical study will illustrate some aspects of $p(a_{ij} \mid z_{ij})$.

Figure 2 draws the Monte Carlo estimated density of $p(a_{ij} \mid z_{ij} = 0)$ and $p(a_{ij} \mid z_{ij} = 1)$ for several settings of v_0 and h. In all cases, there is a clear separation between

$p(a_{ij} \mid z_{ij} = 0)$ and $p(a_{ij} \mid z_{ij} = 1)$, with a larger h resulting in a sharper separation. This property of separation between a small and a large variance component is the essence of the prior for structural learning that aims to separate small and large a_{ij}s. Clearly, the marginal densities are no longer normal. For example, the density of $p(a_{ij} \mid z_{ij} = 0)$ is more spiky than that of $\mathrm{N}(0, v_0)$; the difference between $p(a_{ij} \mid z_{ij} = 1)$ when $h = 50$ and $h = 100$ is less clear than the difference between $\mathrm{N}(0, 2500v_0^2)$ and $\mathrm{N}(0, 10000v_0^2)$. The lack of an explicit form of $p(a_{ij} \mid z_{ij})$ makes the strategies of analytically calculating v_0 from the threshold δ infeasible. Numerical methods that estimate $p(a_{ij} \mid z_{ij})$ from Markov chain Monte Carlo (MCMC) samples might be used to choose v_0 according to δ from a range of possible values.

Another perspective is that, when v_0 is chosen to be very small and h is chosen to be very large, the prior for a_{ij} is close to a point-mass mixture that selects any $a_{ij} \neq 0$ as an edge. Because the point-mass mixture prior provides a noninformative method of structure learning when the threshold δ cannot be meaningfully specified, it makes sense to choose v_0 to be as small as possible, but not so small that it could cause MCMC convergence issues, and to choose v_1 to allow for reasonable values of a_{ij}. In our experiments with standardized data, MCMC converges quickly and mixes quite well, as long as $v_0 \geq 0.01$ and $h \leq 1000$.

3.3 Choice of λ

The value of λ controls the distribution of the diagonal elements a_{ii}. Because the data are usually standardized, a choice of $\lambda = 1$ assigns substantial probability to the entire region of plausible values of a_{ii}, without overconcentration or overdispersion. From our experience, the data generally contain sufficient information about the diagonal elements, and the structure learning results are insensitive to a range of λ values, such as $\lambda = 5$ and 10.

4 Fast block Gibbs samplers

The primary advantage of the SSSL prior (8)–(9) over traditional approaches is its scalability to larger p problems. The reduction in computing time comes from two improvements. One is that (8)–(9) enable block updates of all $p(p-1)/2$ edge-inclusion indicators in $\mathbf{Z}$ simultaneously, while other methods only update one edge-inclusion indicator z_{ij} at a time. The other is that there is no need of a Monte Carlo approximation of the intractable constants, while all of the other methods require some sort of Monte Carlo integration to evaluate any graph. The general sampling scheme for generating posterior samples of graphs is to sample from the joint distribution $p(\mathbf{A}, \mathbf{Z} \mid \mathbf{Y})$ by iteratively generating from $p(\mathbf{A} \mid \mathbf{Z}, \mathbf{Y})$ and $p(\mathbf{Z} \mid \mathbf{A}, \mathbf{Y})$. The first conditional $p(\mathbf{A} \mid \mathbf{Z}, \mathbf{Y})$ is sampled in a column-wise manner and the second conditional $p(\mathbf{Z} \mid \mathbf{A}, \mathbf{Y})$ is generated all at once. The details depend on whether $\mathbf{A} = \boldsymbol{\Omega}$ for concentration graph models or $\mathbf{A} = \boldsymbol{\Sigma}$ for covariance graph models, and they are described below.

4.1 Block Gibbs samplers for concentration graph models

Consider concentration graph models with $\mathbf{A} = \boldsymbol{\Omega}$ in the hierarchical prior (8)–(9). To sample from $p(\boldsymbol{\Omega} \mid \mathbf{Z}, \mathbf{Y})$, the following proposition provides necessary conditional distributions. The proof is in the Appendix.

Proposition 1. *Suppose $\mathbf{A} = \boldsymbol{\Omega}$ in the hierarchical prior (8)–(9). Focus on the last column and row. Let $\mathbf{V} = (v^2_{z_{ij}})$ be a $p \times p$ symmetric matrix with zeros in the diagonal entries and $(v^2_{ij})_{i<j}$ in the upper diagonal entries. Partition $\boldsymbol{\Omega}, \mathbf{S} = \mathbf{Y}'\mathbf{Y}$ and $\mathbf{V}$ as follows:*

$$\boldsymbol{\Omega} = \begin{pmatrix} \boldsymbol{\Omega}_{11}, \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}'_{12}, \omega_{22} \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} \mathbf{S}_{11}, \mathbf{s}_{12} \\ \mathbf{s}'_{12}, s_{22} \end{pmatrix}, \quad \mathbf{V} = \begin{pmatrix} \mathbf{V}_{11}, \mathbf{v}_{12} \\ \mathbf{v}'_{12}, 0 \end{pmatrix}. \tag{10}$$

Consider a change of variables: $(\boldsymbol{\omega}_{12}, \omega_{22}) \to (\mathbf{u} = \boldsymbol{\omega}_{12}, v = \omega_{22} - \boldsymbol{\omega}'_{12}\boldsymbol{\Omega}^{-1}_{11}\boldsymbol{\omega}_{12})$. We then have the full conditionals:

$$(\mathbf{u} \mid -) \sim \mathrm{N}(-\mathbf{C}\mathbf{s}_{12}, \mathbf{C}), \quad \textit{and } (v \mid -) \sim \mathrm{GA}(\frac{n}{2} + 1, \frac{s_{22} + \lambda}{2}), \tag{11}$$

where $\mathbf{C} = \{(s_{22} + \lambda)\boldsymbol{\Omega}^{-1}_{11} + \mathrm{diag}(\mathbf{v}^{-1}_{12})\}^{-1}$.

Permuting any column to be updated to the last one and using (11) will lead to a simple block Gibbs step for generating $(\boldsymbol{\Omega} \mid \mathbf{Z}, \mathbf{Y})$. For $p(\mathbf{Z} \mid \boldsymbol{\Omega}, \mathbf{Y})$, prior (8)–(9) implies all z_{ij} are independent Bernoulli with probability

$$\Pr(z_{ij} = 1 \mid \boldsymbol{\Omega}, \mathbf{Y}) = \frac{\mathrm{N}(\omega_{ij} \mid 0, v_1^2)\pi}{\mathrm{N}(\omega_{ij} \mid 0, v_1^2)\pi + \mathrm{N}(\omega_{ij} \mid 0, v_0^2)(1-\pi)}. \tag{12}$$

A closer look at the conditional distributions of the last column $\mathbf{u} = \boldsymbol{\omega}_{12}$ in (11) and the corresponding edge-inclusion indicator vector $\boldsymbol{\gamma} \equiv (\gamma_1, \ldots, \gamma_{p-1})' = (z_{1p}, \ldots, z_{p-1,p})'$ in (12) reveals something interesting. These distributions look like the Gibbs samplers used in the stochastic search variable selection (SSVS) algorithm (George and McCulloch, 1993). Indeed, consider $\boldsymbol{\beta} \equiv (\beta_1, \ldots, \beta_{p-1})' = -\mathbf{u}$ and note that $s_{22} = n$ for standardized data. If $\boldsymbol{\Omega}^{-1}_{11} = \frac{1}{n}\mathbf{S}_{11}$ and $\lambda = 0$, then (11) implies that

$$(\boldsymbol{\beta} \mid \mathbf{z}_{12}, \mathbf{Y}) \sim \mathrm{N}\left[\left\{\mathbf{S}_{11} + \mathrm{diag}(\mathbf{v}^{-1}_{12})\right\}^{-1}\mathbf{s}_{12}, \left\{\mathbf{S}_{11} + \mathrm{diag}(\mathbf{v}^{-1}_{12})\right\}^{-1}\right],$$

and (12) implies that

$$\Pr(\gamma_j = 1 \mid \boldsymbol{\beta}) = \frac{\mathrm{N}(\beta_j \mid 0, v_1^2)\pi}{\mathrm{N}(\beta_j \mid 0, v_1^2)\pi + \mathrm{N}(\beta_j \mid 0, v_0^2)(1-\pi)}, \quad j = 1, \ldots, p-1,$$

which are exactly the Gibbs sampler of SSVS for the p-th variable. Thus, the problem of SSSL for concentration graph models can be viewed as a p-coupled SSVS regression problem, as the use of $\boldsymbol{\Omega}^{-1}_{11}$ in the conditional distribution of $\boldsymbol{\omega}_{12}$ in place of $\mathbf{S}_{11}$ shares information across p regressions in a coherent fashion. This interesting connection has not been noted elsewhere, to the best of our knowledge.

4.2 Block Gibbs samplers for covariance graph models

Now, consider covariance graph models with $\mathbf{A} = \mathbf{\Sigma}$ in the hierarchical prior (8)–(9). To sample from $p(\mathbf{\Sigma} \mid \mathbf{Z}, \mathbf{Y})$, the following proposition provides necessary conditional distributions; its proof is in the Appendix.

Proposition 2. *Suppose $\mathbf{A} = \mathbf{\Sigma}$ in the hierarchical prior (8)–(9). Focus on the last column and row. Let $\mathbf{V} = (v^2_{z_{ij}})$ be a $p \times p$ symmetric matrix with zeros in the diagonal entries and $(v^2_{z_{ij}})_{i<j}$ in the upper diagonal entries. Partition $\mathbf{\Sigma}, \mathbf{S}$ and $\mathbf{V}$ as follows:*

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_{11}, \boldsymbol{\sigma}_{12} \\ \boldsymbol{\sigma}'_{12}, \sigma_{22} \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} \mathbf{S}_{11}, \mathbf{s}_{12} \\ \mathbf{s}'_{12}, s_{22} \end{pmatrix}, \quad \mathbf{V} = \begin{pmatrix} \mathbf{V}_{11}, \mathbf{v}_{12} \\ \mathbf{v}'_{12}, 0 \end{pmatrix}. \tag{13}$$

Consider a change of variables: $(\boldsymbol{\sigma}_{12}, \sigma_{22}) \to (\mathbf{u} = \boldsymbol{\sigma}_{12}, v = \sigma_{22} - \boldsymbol{\sigma}'_{12}\mathbf{\Sigma}_{11}^{-1}\boldsymbol{\sigma}_{12})$. We then have the full conditionals:

$$\begin{aligned} (\mathbf{u} \mid \mathbf{Y}, \mathbf{Z}, \mathbf{\Sigma}_{11}, v) &\sim \mathrm{N}\left[\left\{\mathbf{B} + \mathrm{diag}(\mathbf{v}_{12}^{-1})\right\}^{-1}\mathbf{w}, \left\{\mathbf{B} + \mathrm{diag}(\mathbf{v}_{12}^{-1})\right\}^{-1}\right], \\ (v \mid \mathbf{Y}, \mathbf{Z}, \mathbf{\Sigma}_{11}, \mathbf{u}) &\sim \mathrm{GIG}(1 - n/2, \lambda, \mathbf{u}'\mathbf{\Sigma}_{11}^{-1}\mathbf{S}_{11}\mathbf{\Sigma}_{11}^{-1}\mathbf{u} - 2\mathbf{s}'_{12}\mathbf{\Sigma}_{11}^{-1}\mathbf{u} + s_{22}), \end{aligned} \tag{14}$$

where $\mathbf{B} = \mathbf{\Sigma}_{11}^{-1}\mathbf{S}_{11}\mathbf{\Sigma}_{11}^{-1}v^{-1} + \lambda\mathbf{\Sigma}_{11}^{-1}$, $\mathbf{w} = \mathbf{\Sigma}_{11}^{-1}\mathbf{s}_{12}v^{-1}$, and $\mathrm{GIG}(q, a, b)$ denotes the generalized inverse Gaussian distribution with a probability density function:

$$p(x) = \frac{(a/b)^{q/2}}{2K_q(\sqrt{ab})} x^{(p-1)} e^{-(ax+b/x)/2},$$

with K_q as a modified Bessel function of the second kind.

Surprisingly, Proposition (2) shows that the conditional distribution of any column (row) in $\mathbf{\Sigma}$ is also multivariate normal. This suggests direct column-wise block Gibbs updates of $\mathbf{\Sigma}$. Sampling from $p(\mathbf{Z} \mid \mathbf{\Sigma}, \mathbf{Y})$ is similar to that in (12) for $p(\mathbf{Z} \mid \mathbf{\Omega}, \mathbf{Y})$ with only the modification of replacing ω_{ij} with σ_{ij}.

5 Effectiveness of the new methods

5.1 Computational efficiency

The computational speed and the scalability of SSSL block Gibbs samplers are evaluated empirically. The data of dimension $p \in \{50, 100, 150, 200, 250\}$ and sample size $n = 2p$ are first generated from $\mathrm{N}(0, \mathbf{I}_p)$ and then standardized. The samplers are implemented under the hyperparameters $v_0 = 0.05, h = 50, \pi = 2/(p-1)$ and $\lambda = 1$. All chains are initialized at the sample covariance matrix. All computations are implemented on a six-core CPU 3.33GHz desktop using MATLAB. For each run, we measure the time it takes the block Gibbs sampler to sweep across all columns (rows) once, which is called one iteration. One iteration actually updates each element a_{ij} twice: once when updating column i and again when updating column j. This property improves its efficiency. The

solid and dashed curves in Figure 3 display the minutes taken for 1000 iterations versus p for covariance graph models and concentration graph models respectively.

Overall, the SSSL algorithms run fast. Covariance graph models take approximately 2 and 9 minutes to generate 1000 iterations for $p = 100$ and 200; concentration graph models take even less time, approximately 1.2 and 5 minutes. The relatively slower speed of covariance graph models is due to a few more matrix inversion steps in updating the columns in $\mathbf{\Sigma}$. We also measure the mixing of the MCMC output by calculating the inefficiency factor $1+2\sum_{k=1}^{\infty}\rho(k)$ where $\rho(k)$ is the sample autocorrelation at lag k. We use 5000 samples after 2000 burn-ins and K lags in the estimation of the inefficiency factors, where $K = \text{argmin}_k\{\rho(k) < 2/\sqrt{M}, k \geq 1\}$ with $M = 5000$ being the total number of saved iterations. The median inefficiency factor among all of the elements of $\mathbf{\Omega}$ was 1 when $p = 100$, further suggesting the efficiency of SSSL. In our experience, a MCMC sample of 5000 iterations after 2000 burnins usually generates reliable results in terms of Monte Carlo errors for $p = 100$ node problems, meaning that the computing time is usually less than 10 minutes, far less than the few days of computing time required by the existing methods.

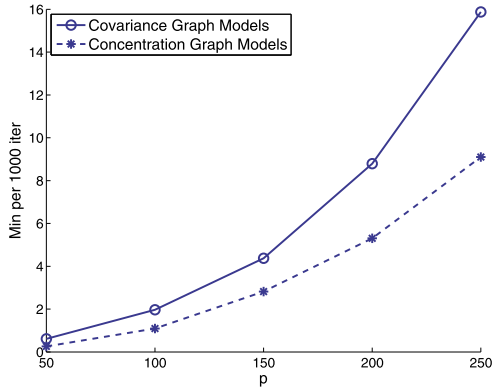

Figure 3: Time in minutes for 1000 iterations of SSSL versus dimension p for covariance graph models (solid) and concentration graph models (dashed).

5.2 Structure learning accuracy

The preceding section shows tremendous computational gains of SSSL over existing methods. This section evaluates these methods on their structure learning performance using two synthetic scenarios, both of which are motivated by real-world applications.

Scenario 1. The first scenario mimics the dependence pattern of daily currency exchange rate returns, which has been previously analyzed via concentration graph models (Carvalho and West, 2007; Wang et al., 2011). We use two different synthetic datasets for the two types of graphical models. For concentration graph models, Jones et al. (2005) generated a simulated dataset of $p = 12$ and $n = 250$ that mimics the exchange rate return pattern. We use their original dataset downloaded from their

website. The true underlying concentration graph has 13 edges and is given by Figure 2 of Jones et al. (2005). For covariance graph models, we estimate a sparse covariance matrix with 13 edges based on the data of Jones et al. (2005) and then use this sparse covariance matrix to generate $n = 250$ samples. The true sparse covariance matrix is as follows:

$$\begin{pmatrix}
0.239 & 0.117 & & & & & & 0.031 & & & & \\
0.117 & 1.554 & & & & & & & & & & \\
 & & 0.362 & 0.002 & & & & & & & & \\
 & & 0.002 & 0.199 & 0.094 & & & & & & & \\
 & & & 0.094 & 0.349 & & & & & & & -0.036 \\
 & & & & & 0.295 & -0.229 & 0.002 & & & & \\
 & & & & & -0.229 & 0.715 & & & & & \\
0.031 & & & & & 0.002 & & 0.164 & 0.112 & -0.028 & -0.008 & \\
 & & & & & & & 0.112 & 0.518 & -0.193 & -0.090 & \\
 & & & & & & & -0.028 & -0.193 & 0.379 & 0.167 & \\
 & & & & & & & -0.008 & -0.090 & 0.167 & 0.159 & \\
 & & & & -0.036 & & & & & & & 0.207
\end{pmatrix}.$$

To assess the performance of structure learning, we compute the number of true positives (TP) and the number of false positives (FP). We begin by evaluating some benchmark models against which to compare SSSL. For concentration graph models, we consider the classical adaptive graphical lasso (Fan et al., 2009) and the Bayesian G-Wishart prior $\mathrm{W}_G(3, \mathbf{I}_p)$ in (1). For covariance graph models, we consider the classical adaptive thresholding (Cai and Liu, 2011) and the Bayesian G-inverse Wishart prior $\mathrm{IW}_G(3, \mathbf{I}_p)$ in (4). The two classical methods require some tuning parameters, which are chosen by 10-fold cross-validation. The adaptive graphical lasso has TP = 8 and FP = 7 for concentration graph models and the adaptive thresholding has TP = 13 and FP = 45 for covariance graph models. They seem to favor less sparse models, most likely because the sample size is large relative to the dimensions. The two Bayesian methods are implemented under the priors of graphs (2) with $\pi = 2/(p-1)$. They perform better than the classical methods. The G-Wishart has TP = 9 and FP = 1 for concentration graph models, and the G-inverse Wishart has TP = 7 and FP = 0 for covariance graph models.

We investigate the performance of SSSL by considering a range of hyperparameter values that represent different prior beliefs about graphs: $\pi \in \{2/(p-1), 4/(p-1), 0.5\}$, $h \in \{10, 50, 100\}$, and $v_0 \in \{0.02, 0.05, 0.1\}$. Under each hyperparameter setting, a sample of 10000 iterations is used to estimate the posterior median graph, which is then compared against the true underlying graph. Table 1 and 2 summarize TP and FP for concentration and covariance graphs respectively. Within each table, patterns can be observed by comparing results across different values of one hyperparameter while fixing the others. For v_0, a larger value lowers both TP and FP because v_0 is positively related to the threshold of practical significance that a_{ij} can be treated as zero. For π, a larger value increases both TP and FP, and the case of $\pi = 0.5$ seems to produce graphs that are too dense and have high FP, especially for the concentration graph models in Table 1. For h, increasing h reduces both TP and FP, partly because h is positively related to the threshold that a_{ij} can be practically treated as zero and in part because h is negatively related to the implied prior edge-inclusion probability $\Pr(z_{ij})$ as discussed in Panel (a) of Figure 1. Next, comparing the results in Tables 1–2 with the two Bayesian benchmarks reported above, we can see that SSSL is competitive. For

example, Table 2 shows that, except for the extreme cases of $\pi = 0.5$ that favor graphs that are too dense or cases of $v_0 = 0.1$ that favor graphs that are too sparse, SSSL has approximately the same TP = 7 and FP = 0 as the G-inverse Wishart method for covariance graph models.

Table 1: Summary of performance measures under different hyperparameters for a $p = 12$ concentration graph example. As for benchmarks, the classical adaptive graphical lasso has TP=8 and FP=7; the Bayesian G-Wishart has TP=9 and FP=1.

	$\pi = 2/(p-1)$			$\pi = 4/(p-1)$			$\pi = 0.5$		
v_0	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$
				Number of true positives (TP)					
0.02	9	9	9	10	10	10	10	10	10
0.05	10	9	9	10	10	9	10	10	10
0.1	9	8	8	10	8	8	10	9	8
				Number of false positives (FP)					
0.02	3	2	0	7	3	1	14	4	3
0.05	2	0	0	4	1	0	8	2	0
0.1	1	0	0	1	0	0	4	0	0

Table 2: Summary of performance measures under different hyperparameters for a $p = 12$ covariance graph example. As for benchmarks, the classical adaptive thresholding has TP=13 and FP=45; the Bayesian G-inverse Wishart has TP=7 and FP=0.

	$\pi = 2/(p-1)$			$\pi = 4/(p-1)$			$\pi = 0.5$		
v_0	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$
				Number of true positives (TP)					
0.02	7	7	7	8	7	7	9	7	7
0.05	7	7	6	7	7	7	7	7	7
0.1	5	3	3	6	5	4	7	5	5
				Number of false positives (FP)					
0.02	0	0	0	1	0	0	5	0	0
0.05	0	0	0	0	0	0	0	0	0
0.1	0	0	0	0	0	0	0	0	0

Scenario 2. The second scenario mimics the dependence pattern of gene expression data, for which graphical models are used extensively to understand the underlying biological relationships. The real data are the breast cancer data (Jones et al., 2005; Castelo and Roverato, 2006), which contain $p = 150$ genes related to the estrogen receptor pathway. Similar to the first scenario, we generate two synthetic datasets for the two graphical models. For concentration graph models, we first estimate a sparse concentration matrix with 179 edges based on the real data, and then generate a sample of 200 observations from this estimated sparse concentration matrix. For covariance graph models, we estimate a sparse covariance matrix with 101 edges based on the real data and then use this sparse covariance matrix to generate a synthetic data of $n = 200$ samples. Panels (a) and (b) of Figure 4 display the frequencies of the non-zero partial correlations and correlations implied by these two underlying sparse matrices, respectively. Among the nonzero elements, 13% of the partial correlations and 60% of the correlations are within 0.1.

We repeat the same procedures of fitting the benchmark and proposed models as in Scenario 1. As for benchmarks, the adaptive graphical lasso has TP = 145 and

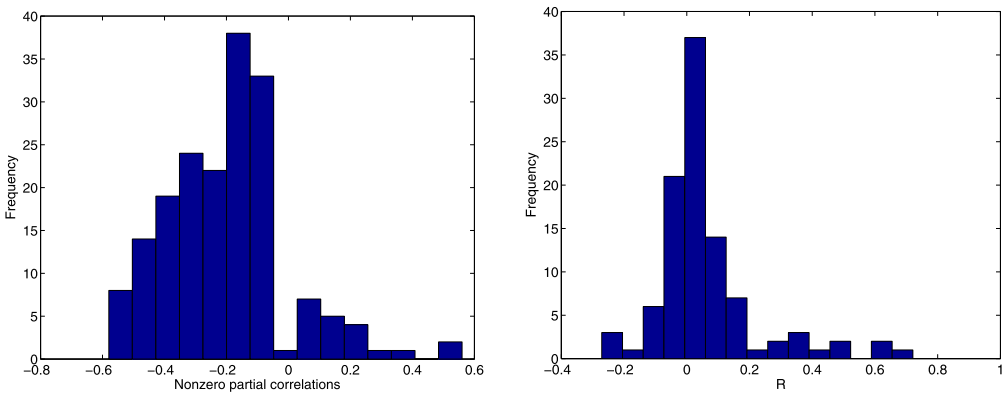

Figure 4: Histograms showing the empirical frequency of the non-zero elements of the partial correlation matrix for the first dataset (left) and of the nonzero elements of the correlation matrix for the second dataset (right) in Scenario 2 of $p = 150$.

FP = 929 for concentration graph models, and the adaptive thresholding has TP = 14 and FP = 12 for covariance graph models. The G-Wishart prior has TP = 105 and FP = 2 and takes about four days to run. The evaluation of the G-inverse Wishart is worth elaborating since existing experiments are conducted in smaller p settings and little is known about its performance in high-dimensional problems.

The original model fitting algorithm for the G-inverse Wishart relies on the computationally expensive importance sampling for approximating the normalizing constant and thus is slow and numerically instable for this p=150 dataset. Silva (2013) develops a new approximation that requires no Monte Carlo integration, which greatly speeds up the computation. The MATLAB routine implementing his algorithm is publicly available on that paper's website. We adopt these functions with one modification that sets the edge-inclusion probability to be $2/(p-1)$. The algorithm takes about 2 hours to complete 1000 sweeps, which is substantially slower than SSSL that costs about 5 minutes – see Figure 3. Since both SSSL and Silva (2013) require no Monte Carlo approximation, the difference in run time is a result of the fact that the SSSL's block update of $\mathbf{Z}$ is faster than Silva (2013)'s one-edge-at-a-time update.

Using the posterior median graph as an estimate of G, the G-inverse Wishart prior $\text{IW}_G(3, \mathbf{I})$ produces TP = 3 and FP = 3. These two numbers are surprisingly small. An exploration of the G-inverse Wishart distribution provides some explanations. The main reason is that when G is sparse and p is large, the G-inverse Wishart prior inadvertently enforces the free elements in $\mathbf{\Sigma}$ towards zero and hence exerts a strong prior influence on the posterior distribution. To see this, suppose G is empty, then (4) implies that the diagonal elements $\{\sigma_{ii}\}$ follow independent inverse Gamma distributions

$$p(\sigma_{ii} \mid b) \propto \sigma_{ii}^{-\frac{b+2p}{2}} \exp\{-\frac{d_{ii}}{2\sigma_{ii}}\}, \quad i = 1, \ldots, p,$$

which clearly depend on the dimension p and converge to zero rapidly as p increases for a fixed b. Now suppose G is arbitrarily sparse. Although the theoretical marginal distributions of the free elements in $\mathbf{\Sigma}$ are unknown, the distribution of $\{\sigma_{ii}\}$ under an empty graph leads us to conjecture that the free elements in $\mathbf{\Sigma}$ could be extremely concentrated around zero for large p as well. In our $p = 150$ example, a simulation from $\text{IW}_G(3, \mathbf{I}_p)$ under the ground truth G that contains 101 edges supports this conjecture. The estimated prior mean of these 101 off-diagonal free elements $\{\sigma_{ij}\}$ is in the range of -6×10^{-6} and 6×10^{-6}; the estimated prior standard deviation is between 1.9×10^{-4} and 2.3×10^{-4}. Such a tightly concentrated prior provides little probability support for the true graph. The implication on structure learning is that the Bayes factor might not truly respond to the data, but largely reflect prior prejudices that σ_{ij} are extremely small. In other words, the overly concentrated prior does not allow the data to speak about G and consequently the posterior distribution of G is dominated by its prior. In fact, the posterior sample mean and standard deviation of the number of edges in $\mathbf{Z}$ computed from the MCMC output of Silva (2013) are 170 and 12.9, which are close to the prior expected number of edges, computed as $\binom{p}{2} \times \frac{2}{p-1} = 150$ and its standard deviation, computed as $\sqrt{\binom{p}{2} \times \frac{2}{p-1} \times (1 - \frac{2}{p-1})} = 12.16$.

The fundamental cause of this strong prior influence is perhaps that the parameter space $\{b : b > 0\}$ assumed by Silva and Ghahramani (2009) is too restrictive. It might be reasonable to let the parameter space depend on G. For example, the standard inverse-Wishart theory implies that the parameter space should be $\{b : b > 2 - 2p\}$ when G is empty and so b could be even negative, and $\{b : b > 0\}$ when G is full. Hierarchical models that allow the value of b to be G-dependent might be helpful. A thorough examination along these lines is beyond the scope of the current paper. However, it is probably safe to conclude that further investigation should be called upon to follow the innovative framework of Silva and Ghahramani (2009).

As for SSSL, Tables 3 and 4 summarize its performance. When the results are compared across different levels of one hyperparameter, the general relations between TP or FP and a hyperparameter are similar to those in Scenario 1. In fact, the patterns appear to be more significant in Scenario 2 because priors have greater influences in this relatively small sample size problem. When compared with benchmarks, SSSL is competitive. For concentration graph models, Table 3 suggests that, except for a few extreme priors that favor overly dense graphs, SSSL produces much sparser graphs than the classical adaptive graphical lasso, for which FP = 929 is too high. When $v_0 = 0.02$, SSSL is also comparable to the Bayesian G-Wishart prior. When v_0 increases, TP drops quickly because many signals are weak (Figure 4) and are thus treated as practically insignificant by SSSL. For covariance graph models, Table 4 suggests that SSSL generally performs better than the adaptive thresholding. The only exceptions are cases in which the hyperparameters favor overly dense graphs (e.g., $\pi = 0.5$) or overly sparse graphs (e.g., $v_0 = 0.1$).

In summary, under sensible choices of hyperparameters, such as $v_0 = 0.02, h = 50$, and $\pi = 2/(p-1)$, SSSL is comparable to existing Bayesian methods in terms of

Table 3: Summary of performance measures under different hyperparameters for a $p = 150$ concentration graph example. As for benchmarks, the classical adaptive lasso has TP=145 and FP=929; the Bayesian G-Wishart has TP=105 and FP=2.

	$\pi = 2/(p-1)$			$\pi = 4/(p-1)$			$\pi = 0.5$		
v_0	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$
				Number of true positives (TP)					
0.02	106	101	100	110	105	102	162	136	125
0.05	90	82	78	97	89	83	146	117	109
0.1	72	59	56	78	67	58	122	101	93
				Number of false positives (FP)					
0.02	3	0	0	9	1	0	1533	238	91
0.05	0	0	0	0	0	0	667	39	9
0.1	0	0	0	0	0	0	169	4	0

Table 4: Summary of performance measures under different hyperparameters for a $p = 150$ covariance graph example. As for benchmarks, the classical adaptive thresholding has TP=14 and FP=12; the Bayesian G-inverse Wishart cannot run.

	$\pi = 2/(p-1)$			$\pi = 4/(p-1)$			$\pi = 0.5$		
v_0	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$	$h = 10$	$h = 50$	$h = 100$
				Number of true positives (TP)					
0.02	20	19	15	20	20	17	38	27	24
0.05	12	10	10	14	11	10	27	15	15
0.1	7	6	5	7	7	6	15	11	10
				Number of false positives (FP)					
0.02	4	1	1	6	3	1	1084	163	69
0.05	0	0	0	0	0	0	229	17	3
0.1	0	0	0	0	0	0	21	0	0

structure learning accuracy. However, SSSL's computational advantage of sheer speed and simplicity makes it very attractive for routine uses.

6 Graphs for credit default swap data

This section illustrates the practical utility of graphical models by applying them to credit default swap (CDS) data. CDS is a credit protection contract in which the buyer of the protection periodically pays a small amount of money, known as "spread", to the seller of protection in exchange for the seller's payoff to the buyer if the reference entity defaults on its obligation. The spread depends on the creditworthiness of the reference entity and thus can be used to monitor how the market views the credit risk of the reference entity. The aim of this statistical analysis is to understand the cross-sectional dependence structure of CDS series and thus the joint credit risks of the entities. The interconnectedness of credit risks is important, as it is an example of the systemic risk – the risk that many financial institutions fail together.

The data are provided by Markit via Wharton Research Data Services (WRDS) and comprise daily closing CDS spread quotes for 79 widely traded North American reference entities from January 2, 2001 through April 23, 2012. The quotes are for five-year maturity, as five-year maturity is the most widely traded and studied term. The spread quote is then transformed into log returns for graphical model analysis.

To assess the variation in the graphs over time, we estimate graphs using a one-year moving window. In particular, at the end of each month t, we use the daily CDS returns over the period of month $t-11$ to month t to estimate the graph G_t. The choice of a one-year window is intended to balance the number of observations, as well as to accommodate the time-varying nature of the graphs. The first estimation period begins in January 2001 and continues through December 2001, and the last is from May 2011 to April 2012. In total, there are 89 estimation periods, corresponding to 89 time-varying graphs for each type of graph. We set the prior hyperparameters at $v_0 = 0.02, h = 50, \pi = 2/(p-1)$ and $\lambda = 1$. The MCMC are run for 10000 iterations and the first 3000 iterations are discarded. The graphs are estimated by the posterior median graph.

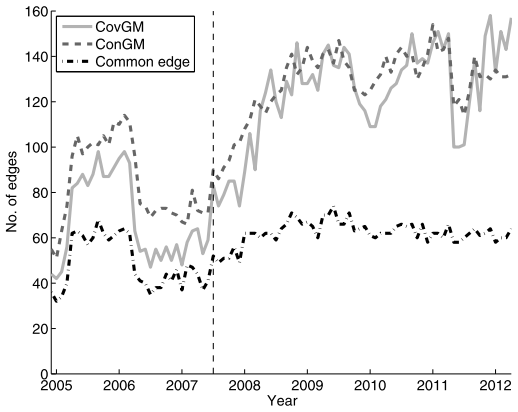

Figure 5: Time series plots of the estimated numbers of edges for the covariance graph (solid line), the concentration graph (dashed), and the common graph (dash dotted).

Figure 5 shows changes over time of the estimated number of edges for the two types of graphs and the number of common edges. The numbers of edges are in the range of 40 and 160 out of a total of 3081 possible edges, indicating a very high level of sparsity. At each time point, both types of graph reflect about the same level of sparsity. Over time, they show similar patterns of temporal variations. There is a steady upward trend in the number of edges starting from mid 2007 for both types of graph. The timing of the rise of the number of edges is suggestive. Mid-2007 saw the start of the subprime-mortgage crisis, which was signified by a number of credit events, including bankruptcy and the significant loss of several banks and mortgage firms, such as Bear Stearns, GM finance, UBS, HSBC and Countrywide. If the number of edges can be viewed as the "degree of connectedness" among CDS series, then this observed increase implies that the market tends to be more integrated during periods of credit crisis and consequently tends to have a higher systemic risk.

To further illustrate the interpretability of graphs, Figure 6 provides four snapshots on graph details at two time points, in December 2004 and in December 2008. The covariance graph for December 2004 (Panel a) has 44 edges and 30 completely isolated

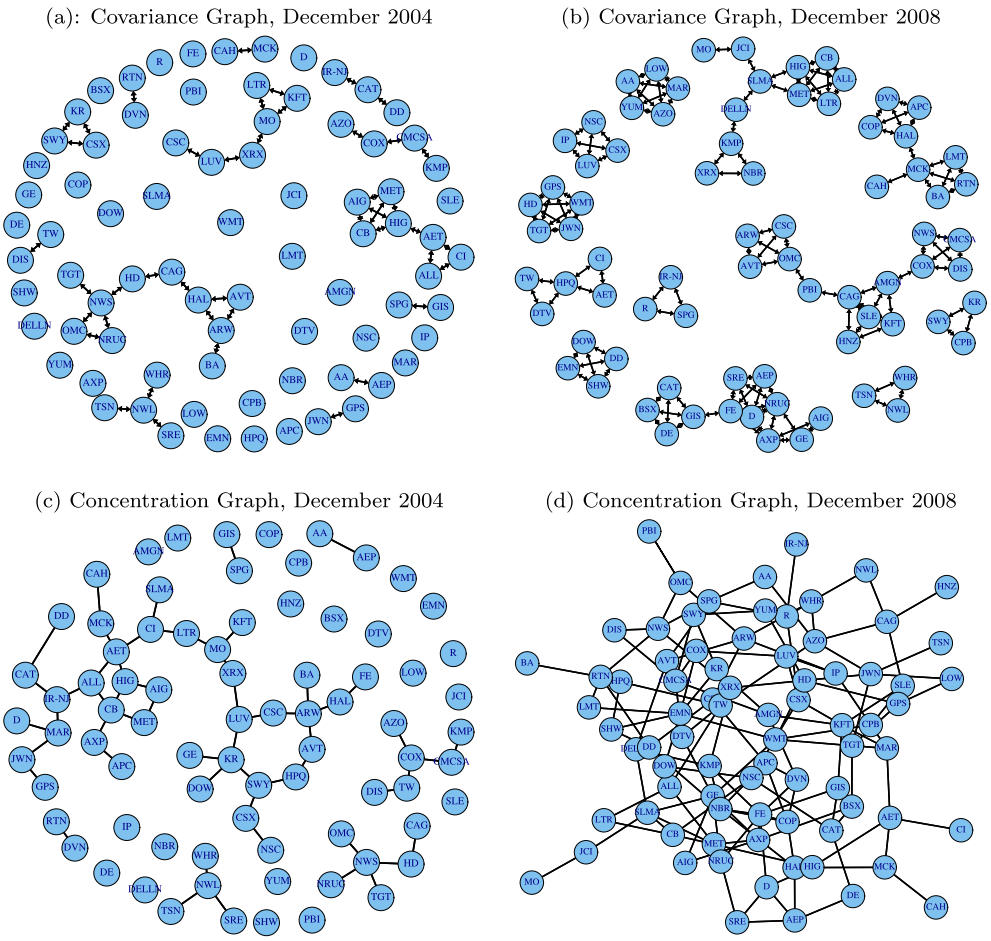

Figure 6: Estimated graphs of 79 CDS returns at two different time points. The four panels are: Panel (a), covariance graph in December 2004; Panel (b), covariance graph in December 2008; Panel (c), concentration graph in December 2004; and Panel (d) concentration graph in December 2008.

nodes, as opposed to the covariance graph for December 2008 (Panel b), which has 128 edges and no isolated nodes. These differences suggest that the connectedness of the network rises as the credit risks of many reference entities become linked with each other. The same message is further confirmed by concentration graphs. The concentration graph for December 2004 (Panel c) has 55 edges and 20 completely isolated nodes, while the concentration graph for December 2008 (Panel d) has 135 edges and no isolated nodes. The increase of connectedness is also manifested by the fact that every pair of nodes in the concentration graph on December 2008 is connected by a path.

Finally, we zoom into subgraphs involving the American International Group (AIG)

to study whether the graphs make economic sense at the firm level. AIG provides a good example for study because it suffered a significant credit deterioration during the 2007–2008 crisis when its credit ratings were downgraded below "AA" levels in September 2008. Panels (a) and (b) of Figure 7 show the covariance subgraph in December 2004 and in December 2008. The subgraphs show that AIG experienced some interesting dependence structure shifts between these two periods. In December 2004, AIG formed a clique with three other major insurance companies: the Metropolitan Life Insurance Company (MET), the Chubb Corporation (CB) and the Hartford Financial Services Group (HIG). The credit risks of these four insurance companies are all linked to each other. By December 2008, the linkages between AIG and the other three insurance companies disappear, while the linkages among the other three firms remain. On the other hand, AIG is now connected with two non-insurance financial companies, the GE Capital (GE) and the American Express Company (AXP). The concentration subgraphs of AIG displayed in Panels (c) and (d) of Figure 7 show similar structural shift patterns as in the covariance graphs, although here dependence functions differently as conditional dependence. AIG initially formed a prime component with the other three insurance firms in December 2004. The connections between AIG and the other insurance firms were severed in December 2008, and new connections between AIG and GE and between AIG and AXP arose.

Given that AIG suffered a more severe credit crisis than other insurance companies, the uncovered network shift might indicate that the CDS market was able to adjust its view regarding the credit risk of AIG and treat it as unrelated to its peer insurance firms during the credit crisis. Such timely changes in dependence structures may shed light on the question of the information efficiency of the CDS market, from a point of view that is different from the usual analysis of changes in levels.

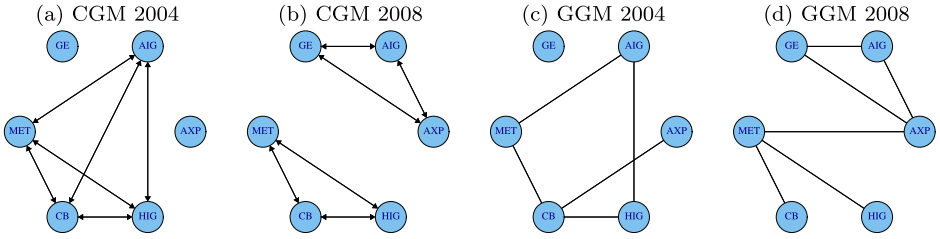

Figure 7: Subgraphs involving AIG estimated for two different time periods. There are four snapshots: Panel (a), covariance graph in December 2004; Panel (b), covariance graph in December 2008; Panel (c), concentration graph in December 2004; and Panel (d) concentration graph in December 2008. The six subgraph nodes are: American International Group (AIG); GE Capital (GE); Metropolitan Life Insurance Company (MET); The Chubb Corporation (CB); Hartford Financial Services Group (HIG); and American Express Company (AXP).

7 Conclusion

This paper proposes a new framework for the structure learning of concentration graph models and covariance graph models. The main goal is to overcome the scalability limitation of existing methods without sacrificing structure learning accuracy. The key idea is to use absolutely continuous spike and slab priors instead of the popular point-mass mixture priors to enable accurate, fast, and simple structure learning. Our analysis suggests that the accuracy of these new methods is comparable to that of existing Bayesian methods, but the model-fitting is vastly faster to run and simpler to implement. Problems with 100–250 nodes can now be fitted in a few minutes, as opposed to a few days or even numeric infeasibility under existing methods. This remarkable efficiency will facilitate the application of Bayesian graphical models to large problems and will provide scalable and effective modular structures for more complicated models.

The focus of the paper is on the structure learning of graphs. A related yet different question is the parameter estimation of the covariance matrix. Since our priors place zero probability mass on any sparse matrix containing exact zeros, as opposed to the point-mass mixture priors, one concern is then about where and how the posterior of $\mathbf{\Omega}$ or $\mathbf{\Sigma}$ will concentrate when the true covariance/concentration matrix is sparse. Our limited experiments suggest that the posterior distribution from the proposed two-component normals is indeed more dispersed around zero than those from the point-mass mixture priors. Take the concentration graph in Scenario 1 as an example. Under SSSL, the average magnitude of the posterior mean estimates of these $\{\omega_{ij}\}$ corresponding to the true zeros in $\mathbf{\Omega}$ is in the range of 0.0085 and 0.045, depending on the hyperparameters of (v_0, h, π). In contrast, under the G-Wishart prior, the estimates of these $\{\omega_{ij}\}$ have a smaller average magnitude of 0.0040. The small-variance normal shrinks parameters less aggressively than the point-mass mixture. Although we do not find it problematic for structure learning, such weaker shrinkage may cause the SSSL's performance of parameter estimation to be suboptimal. A refinement is to replace the normal distribution with distributions having higher densities near zero. The Bayesian shrinkage regression literature shows that some heavy-tailed distributions offer comparable parameter estimation performance to the point-mass mixture priors (e.g., Armagan et al. 2013; Griffin and Brown 2010; Carvalho et al. 2010). Computational tractability of SSSL is maintained by applying the data-augmentation to the mixture normal representation of these alternative distributions.

Appendix

Proof of Proposition 1

Clearly, the conditional distribution of $\mathbf{\Omega}$ given the edge-inclusion indicators $\mathbf{Z}$ is

$$p(\mathbf{\Omega} \mid \mathbf{Y}, \mathbf{Z}) \propto |\mathbf{\Omega}|^{\frac{n}{2}} \exp\{-\mathrm{tr}(\frac{1}{2}\mathbf{S\Omega})\} \prod_{i<j} \left\{ \exp(-\frac{\omega_{ij}^2}{2v_{z_{ij}}^2}) \right\} \prod_{i=1}^{p} \left\{ \exp(-\frac{\lambda}{2}\omega_{ii}) \right\}. \quad (15)$$

Under the partitions (10), the conditional distribution of the last column in $\mathbf{\Omega}$ is

$$p(\boldsymbol{\omega}_{12}, \omega_{22} \mid \mathbf{Y}, \mathbf{Z}, \boldsymbol{\Omega}_{11}) \propto (\omega_{22} - \boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12})^{\frac{n}{2}} \exp\big[-\frac{1}{2}\{\boldsymbol{\omega}_{12}'\mathbf{D}^{-1}\boldsymbol{\omega}_{12} + 2\mathbf{s}_{12}'\boldsymbol{\omega}_{12} + (s_{22}+\lambda)\omega_{22}\}\big],$$

where $\mathbf{D} = \text{diag}(\mathbf{v}_{12})$. Consider a change of variables $(\mathbf{u}, \omega_{22}) \to (\mathbf{u} = \boldsymbol{\omega}_{12}, v = \omega_{22} - \boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12})$, whose Jacobian is a constant not involving $(\boldsymbol{\omega}_{12}, v)$. So

$$p(\mathbf{u}, v \mid \mathbf{Y}, \mathbf{Z}, \boldsymbol{\Omega}_{11}) \propto v^{\frac{n}{2}} \exp(-\frac{s_{22}+\lambda}{2}v) \exp\left(-\frac{1}{2}\big[\mathbf{u}'\{\mathbf{D}^{-1} + (s_{22}+\lambda)\boldsymbol{\Omega}_{11}^{-1}\}\mathbf{u} + 2\mathbf{s}_{12}'\mathbf{u}\big]\right).$$

This implies that:

$$(\mathbf{u}, v) \mid (\boldsymbol{\Omega}_{11}, \mathbf{Z}, \mathbf{Y}) \sim \text{N}(-\mathbf{C}\mathbf{s}_{12}, \mathbf{C})\text{GA}(\frac{n}{2}+1, \frac{s_{22}+\lambda}{2}),$$

where $\mathbf{C} = \{(s_{22}+\lambda)\boldsymbol{\Omega}_{11}^{-1} + \mathbf{D}^{-1}\}^{-1}$.

Proof of Proposition 2

Given edge-inclusion indicators $\mathbf{Z}$, the conditional posterior distribution of $\boldsymbol{\Sigma}$ is

$$p(\boldsymbol{\Sigma} \mid \mathbf{Y}, \mathbf{Z}) \propto |\boldsymbol{\Sigma}|^{-\frac{n}{2}} \exp\{-\frac{1}{2}\text{tr}(\mathbf{S}\boldsymbol{\Sigma}^{-1})\} \prod_{i<j} \left\{ \exp(-\frac{\sigma_{ij}^2}{2v_{z_{ij}}^2}) \right\} \prod_{i=1}^{p} \left\{ \exp(-\frac{\lambda}{2}\sigma_{ii}) \right\}. \quad (16)$$

Under partitions (13), consider a transformation $(\boldsymbol{\sigma}_{12}, \sigma_{22}) \to (\mathbf{u} = \boldsymbol{\sigma}_{12}, v = \sigma_{22} - \boldsymbol{\sigma}_{12}'\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\sigma}_{12})$, whose Jacobian is a constant not involving $(\mathbf{u}, v)$ and apply the block matrix inversion to $\boldsymbol{\Sigma}$ using blocks $(\boldsymbol{\Sigma}_{11}, \mathbf{u}, v)$:

$$\boldsymbol{\Sigma}^{-1} = \begin{pmatrix} \boldsymbol{\Sigma}_{11}^{-1} + \boldsymbol{\Sigma}_{11}^{-1}\mathbf{u}\mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}v^{-1} & -\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u}v^{-1} \\ -\mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}v^{-1} & v^{-1} \end{pmatrix}. \quad (17)$$

After removing some constants not involving $(\mathbf{u}, v)$, the terms in (16) can be expressed as a function of $(\mathbf{u}, v)$:

$$\begin{aligned} |\boldsymbol{\Sigma}| &\propto v, \\ \text{tr}(\mathbf{S}\boldsymbol{\Sigma}^{-1}) &\propto \mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{S}_{11}\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u}v^{-1} - 2\mathbf{s}_{12}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u}v^{-1} + s_{22}v^{-1}, \\ \prod_{i<j}\left\{ \exp(-\frac{\sigma_{ij}^2}{2v_{z_{ij}}^2}) \right\} &\propto \exp(-\frac{1}{2}\mathbf{u}'\mathbf{D}^{-1}\mathbf{u}), \\ \prod_{i}\left\{ \exp(-\frac{\lambda}{2}\sigma_{ii}) \right\} &\propto \exp(-\frac{1}{2}\lambda(\mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u} + v)), \end{aligned}$$

where $\mathbf{D} = \text{diag}(\mathbf{v}_{12})$. Holding all but $(\mathbf{u}, v)$ fixed, we can then rewrite the logarithm of (16) as

$$\log p(\mathbf{u}, v \mid -) = -\frac{1}{2}\Big\{ n\log(v) + \mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{S}_{11}\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u}v^{-1} - 2\mathbf{s}_{12}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u}v^{-1} + s_{22}v^{-1} + \mathbf{u}'\mathbf{D}^{-1}\mathbf{u} + \lambda\mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u} + \lambda v \Big\} + constant.$$

This gives the conditionals of $\mathbf{u}$ and v as

$$(\mathbf{u} \mid \mathbf{Y}, \mathbf{Z}, \boldsymbol{\Sigma}_{11}, v) \sim \mathrm{N}\big\{(\mathbf{B} + \mathbf{D}^{-1})^{-1}\mathbf{w}, (\mathbf{B} + \mathbf{D}^{-1})^{-1}\big\},$$
$$(v \mid \mathbf{Y}, \mathbf{Z}, \boldsymbol{\Sigma}_{11}, \mathbf{u}) \sim \mathrm{GIG}(1 - n/2, \lambda, \mathbf{u}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{S}_{11}\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u} - 2\mathbf{s}_{12}'\boldsymbol{\Sigma}_{11}^{-1}\mathbf{u} + s_{22}),$$

where $\mathbf{B} = \boldsymbol{\Sigma}_{11}^{-1}\mathbf{S}_{11}\boldsymbol{\Sigma}_{11}^{-1}v^{-1} + \lambda\boldsymbol{\Sigma}_{11}^{-1}$, $\mathbf{w} = \boldsymbol{\Sigma}_{11}^{-1}\mathbf{s}_{12}v^{-1}$ and $\mathrm{GIG}(q, a, b)$ denotes the generalized inverse Gaussian distribution with probability density function:

$$p(x) = \frac{(a/b)^{q/2}}{2K_q(\sqrt{ab})}x^{(q-1)}e^{-(ax+b/x)/2},$$

with K_q a modified Bessel function of the second kind.

Supplementary materials

The MATLAB routines implementing all frequentist and Bayesian procedures used in the paper are available from the author's web site of the paper at: https://www.msu.edu/ haowang/RESEARCH/SSSL/sssl.html.

References

Armagan, A., Dunson, D. B., and Lee, J. (2013). "Generalized double Pareto shrinkage." *Statistica Sinica*, 23(1): 119. 352, 372

Atay-Kayis, A. and Massam, H. (2005). "The marginal likelihood for decomposable and non-decomposable graphical Gaussian models." *Biometrika*, 92: 317–335. 354

Banerjee, O., El Ghaoui, L., and d'Aspremont, A. (2008). "Model Selection Through Sparse Maximum Likelihood Estimation for Multivariate Gaussian or Binary Data." *The Journal of Machine Learning Research*, 9: 485–516. 351

Bhattacharya, A. and Dunson, D. B. (2011). "Sparse Bayesian infinite factor models." *Biometrika*, 98(2): 291–306. 352

Bickel, P. J. and Levina, E. (2008). "Covariance Regularization by Thresholding." *The Annals of Statistics*, 36(6): 2577–2604. 351, 355

Bien, J. and Tibshirani, R. J. (2011). "Sparse estimation of a covariance matrix." *Biometrika*, 98(4): 807–820. 351, 355

Cai, T. and Liu, W. (2011). "Adaptive Thresholding for Sparse Covariance Matrix Estimation." *Journal of the American Statistical Association*, 106(494): 672–684. URL http://amstat.tandfonline.com/doi/abs/10.1198/jasa.2011.tm10560 355, 364

Carvalho, C. M., Chang, J., Lucas, J. E., Nevins, J. R., Wang, Q., and West, M. (2008). "High-Dimensional Sparse Factor Modeling: Applications in Gene Expression Genomics." *Journal of the American Statistical Association*, 103(484): 1438–1456. 352

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2010). "The horseshoe estimator for sparse signals." *Biometrika*, 97(2): 465–480. 372

Carvalho, C. M. and West, M. (2007). "Dynamic matrix-variate graphical models." *Bayesian Analysis*, 2: 69–98. 363

Castelo, R. and Roverato, A. (2006). "A Robust Procedure For Gaussian Graphical Model Search From Microarray Data With p Larger Than n." *Journal of Machine Learning Research*, 7: 2621–2650. 365

Chaudhuri, S., Drton, M., and Richardson, T. S. (2007). "Estimation of a covariance matrix with zeros." *Biometrika*, 94(1): 199–216. 355

Cheng, Y. and Lenkoski, A. (2012). "Hierarchical Gaussian graphical models: Beyond reversible jump." *Electronic Journal of Statistics*, 6: 2309–2331. 352, 354

Cox, D. R. and Wermuth, N. (1993). "Linear Dependencies Represented by Chain Graphs." *Statistical Science*, 8(3): 204–218. 355

Dawid, A. P. and Lauritzen, S. L. (1993). "Hyper-Markov laws in the statistical analysis of decomposable graphical models." *Annals of Statistics*, 21: 1272–1317. 353

Dempster, A. (1972). "Covariance selection." *Biometrics*, 28: 157–175. 353

Dobra, A., Lenkoski, A., and Rodriguez, A. (2011). "Bayesian inference for general Gaussian graphical models with application to multivariate lattice data." *Journal of the American Statistical Association*, 106(496): 1418–1433. 352, 354

Fan, J., Feng, Y., and Wu, Y. (2009). "Network exploration via the adaptive LASSO and SCAD penalties." *Annals of Applied Statistics*, 3(2): 521–541. 364

Friedman, J., Hastie, T., and Tibshirani, R. (2008). "Sparse inverse covariance estimation with the graphical lasso." *Biostatistics*, 9(3): 432–441. 354

George, E. I. and McCulloch, R. E. (1993). "Variable selection via Gibbs sampling." *Journal of the American Statistical Association*, 88: 881–889. 352, 359, 361

Giudici, P. and Green, P. J. (1999). "Decomposable graphical Gaussian model determination." *Biometrika*, 86: 785–801. 354

Griffin, J. E. and Brown, P. J. (2010). "Inference with normal-gamma prior distributions in regression problems." *Bayesian Analysis*, 5(1): 171–188. 352, 372

Jones, B., Carvalho, C., Dobra, A., Hans, C., Carter, C., and West, M. (2005). "Experiments in stochastic computation for high-dimensional graphical models." *Statistical Science*, 20: 388–400. 354, 363, 364, 365

Kauermann, G. (1996). "On a Dualization of Graphical Gaussian Models." *Scandinavian Journal of Statistics*, 23(1): pp. 105–116. 355

Khare, K. and Rajaratnam, B. (2011). "Wishart distributions for decomposable covariance graph models." *Annals of Statistics*, 39(1): 514–555. 355

Khondker, Z. S., Zhu, H., Chu, H., Lin, W., and Ibrahim, J. G. (2013). "The Bayesian covariance lasso." *Statistics and its interface*, 6(2): 243. 352

Lenkoski, A. and Dobra, A. (2011). "Computational Aspects Related to Inference in Gaussian Graphical Models With the G-Wishart Prior." *Journal of Computational and Graphical Statistics*, 20(1): 140–157. 354

Murray, I. (2007). *Advances in Markov chain Monte Carlo methods*. University College London: PhD. Thesis. 358

Park, T. and Casella, G. (2008). "The Bayesian Lasso." *Journal of the American Statistical Association*, 103(482): 681–686. 352

Rothman, A. J., Bickel, P. J., Levina, E., and Zhu, J. (2008). "Sparse permutation invariant covariance estimation." *Electronic Journal of Statistics*, 2: 494–515. 354

Rothman, A. J., Levina, E., and Zhu, J. (2009). "Generalized Thresholding of Large Covariance Matrices." *Journal of the American Statistical Association*, 104(485): 177–186. 351, 355

Roverato, A. (2002). "Hyper-Inverse Wishart Distribution for Non-decomposable Graphs and its Application to Bayesian Inference for Gaussian Graphical Models." *Scandinavian Journal of Statistics*, 29: 391–411. 353

Scott, J. G. and Carvalho, C. M. (2008). "Feature-Inclusion Stochastic Search for Gaussian Graphical Models." *Journal of Computational and Graphical Statistics*, 17(4): 790–808. 354

Silva, R. (2013). "A MCMC Approach for Learning the Structure of Gaussian Acyclic Directed Mixed Graphs." In *Statistical Models for Data Analysis*, 343–351. Springer. 366, 367

Silva, R. and Ghahramani, Z. (2009). "The Hidden Life of Latent Variables: Bayesian Learning with Mixed Graph Models." *Journal of Machine Learning Research*, 10: 1187–1238. 352, 355, 367

Wang, H. (2010). "Sparse seemingly unrelated regression modelling: Applications in finance and econometrics." *Computational Statistics & Data Analysis*, 54(11): 2866–2877. 357

— (2012). "The Bayesian Graphical Lasso and Efficient Posterior Computation." *Bayesian Analysis*, 7(2): 771–790. 352, 354, 356

— (2014). "Coordinate descent algorithm for covariance graphical lasso." *Statistics and Computing*, 24(4): 521–529. 351

Wang, H. and Li, S. Z. (2012). "Efficient Gaussian graphical model determination under G-Wishart prior distributions." *Electronic Journal of Statistics*, 6: 168–198.

352, 354

Wang, H., Reeson, C., and Carvalho, C. M. (2011). "Dynamic Financial Index Models: Modeling conditional dependencies via graphs." *Bayesian Analysis*, 6(4): 639–664. 363

Wermuth, N., Cox, D. R., and Marchetti, G. M. (2006). "Covariance Chains." *Bernoulli*, 12: 841–862. 355

Yuan, M. and Lin, Y. (2007). "Model selection and estimation in the Gaussian graphical model." *Biometrika*, 94(1): 19–35. 351, 354

Acknowledgments

The author would like to thank two anonymous referees for helpful comments that improved the quality of the manuscript.