Journal of Machine Learning Research 22 (2021) 1-48 Submitted 12/19; Revised 1/21; Published 2/21

GemBag: Group Estimation of Multiple Bayesian Graphical Models

Xinming Yang XYANG104@ILLINOIS.EDU
Department of Statistics
University of Illinois at Urbana-Champaign
Champaign, IL, USA
Lingrui Gan LGAN6@ILLINOIS.EDU
Facebook
Menlo Park, CA, USA
Naveen N. Narisetty NAVEEN@ILLINOIS.EDU
Department of Statistics
University of Illinois at Urbana-Champaign
Champaign, IL, USA
Feng Liang LIANGF@ILLINOIS.EDU
Department of Statistics
University of Illinois at Urbana-Champaign
Champaign, IL, USA

Editor: Garvesh Raskutti

Abstract

In this paper, we propose a novel hierarchical Bayesian model and an efficient estimation method for the problem of joint estimation of multiple graphical models, which have similar but different sparsity structures and signal strength. Our proposed hierarchical Bayesian model is well suited for sharing of sparsity structures, and our procedure, called as GemBag, is shown to enjoy optimal theoretical properties in terms of ℓ_∞ norm estimation accuracy and correct recovery of the graphical structure even when some of the signals are weak. Although optimization of the posterior distribution required for obtaining our proposed estimator is a non-convex optimization problem, we show that it turns out to be convex in a large constrained space facilitating the use of computationally efficient algorithms. Through extensive simulation studies and an application to a bike sharing data set, we demonstrate that the proposed GemBag procedure has strong empirical performance in comparison with alternative methods.

Keywords: graphical models, Bayesian regularization, spike-and-slab priors, selection consistency, non-convex optimization, EM algorithm

1. Introduction

Graphical models provide a natural framework to study the dependence relationship among random variables, in which a node represents a variable and an edge between two nodes means that the two variables are conditionally dependent given the others. When all the variables jointly follow a Gaussian distribution, the corresponding graphical model is known as the Gaussian graphical model (GGM). For GGMs, it is well known that edges in the

graph correspond to non-zero entries in the precision matrix of the underlying Gaussian distribution (Dempster, 1972). Due to this connection, estimating the precision matrix and its support for a high-dimensional Gaussian distribution has drawn a lot of research interest both from the frequentist framework (Yuan and Lin, 2007; Friedman et al., 2008, 2010; Cai et al., 2011; Mazumder and Hastie, 2012) and from the Bayesian framework (Carvalho and Scott, 2009; Dobra et al., 2011; Wang and Li, 2012; Banerjee and Ghosal, 2015; Mohammadi and Wit, 2015; Gan et al., 2019a).

Estimating a single graphical model is not sufficient in applications where observations are naturally grouped into different classes. For example, in social network data, users are grouped by users' characteristics; in biological experiments, subjects are classified into categories based on their experimental conditions; and in gene expression analysis, samples are classified into different tissues or disease states. In the capital bike sharing dataset that we shall analyze in Section 6, user data can be divided into two different classes: "registered" and "casual" or over different years. In such situations, useful insights on the differences between different classes of observations will be missed if we only estimate a single Gaussian graphical model. On the other hand, it is natural to expect that the precision matrices from different classes would share some similarities, especially in terms of their sparsity patterns (i.e., their graph structures). Therefore, it is advantageous to utilize the cross-class similarities and estimate the multiple graphs jointly.

Several methods have been proposed for jointly estimating multiple GGMs based on various penalized likelihood approaches. To encourage shared sparsity, Guo et al. (2011); Danaher et al. (2014); Ma and Michailidis (2016) adopted the group lasso formulation; Zhu and Barber (2015) introduced a non-convex log-shift penalty at the group-level; and Lee and Liu (2015) extended the CLIME method from Cai et al. (2011) for estimating a single graph to the setting of multiple graphs. When the graphs are ordered, Yang et al. (2015) proposed to use a fused lasso penalty to encourage nearby graphs to have similar structures. For theoretical properties, Guo et al. (2011); Ma and Michailidis (2016) established the rate of convergence of their estimators in terms of the averaged Frobenius norm, while Lee and Liu (2015) established the rate of convergence in terms of an averaged version of ℓ_∞ norm where the average is taken over classes.

The Bayesian framework can naturally facilitate sharing of information across different precision matrices through a hierarchical prior. For example, Peterson et al. (2015) used a Markov random field prior on multiple graphs to encourage the selection of common edges in related graphs, and Tan et al. (2017) used a Chung-Lu random graph model (Chung and Lu, 2002) as the prior for hierarchical modeling of multiple graphs. However, theoretical properties of the Bayesian methods for multiple graphs are not studied in the literature. Moreover, implementations of existing Bayesian methods for multiple graphical models have severe computational limitations. Neither the block Gibbs sampler used in Peterson et al. (2015) nor the sequential Monte Carlo algorithm in Tan et al. (2017) scales well with the number of nodes/variables in the graph. The maximum number of nodes considered by these papers is 22, while in practice, many applications have graphs with hundreds of nodes. Moreover, theoretical properties of the existing Bayesian approaches for this problem are not studied. A major motivation of our paper is to develop a Bayesian hierarchical model for multiple GGMs with a scalable computational implementation that can accommodate

a large number of nodes and to study the theoretical properties in terms of estimation consistency and graph structure recovery.

In this paper, we develop a Bayesian approach for jointly estimating multiple GGMs. In the proposed hierarchical model, we allow the GGMs to have similar but different sparsity structures and heterogeneous signal magnitudes. Our method extends the joint estimation method proposed by Gan et al. (2019b) and includes it as a special case where all the graphical models are assumed to have a common sparsity structure. We devise a computationally scalable EM algorithm to obtain the MAP (maximum a posteriori) estimators of the precision matrices. Our theoretical results show that the MAP estimators have an optimal rate of convergence in ℓ_∞ norm under a general setting where the graphical models may have different sparsity structures and signal strength. In terms of sparsity structure recovery, we show that our proposed estimators can consistently recover the sparsity structure of each individual graphical model under the usual minimal signal strength condition that requires a $(\sqrt{(\log p)/n})$ rate for the signals. When the sparsity patterns of the graphical models are similar, we show that our Bayesian regularization method can achieve a smaller misclassification error compared to individual estimation method. In the special case where the multiple GGMs do share a common sparsity structure, the selection consistency of our method can be retained under a weaker minimal signal strength condition that allows signals presented within groups to be even smaller than the $(\sqrt{(\log p)/n})$ rate.

The remaining part of the paper is organized as follows. The proposed Bayesian hierarchical formulation and the parameter estimation procedure are described in Section 2. Theoretical guarantees of our approach are presented in Section 3 and computational techniques are discussed in Section 4. Finally, empirical studies are presented in Sections 5 and 6.

Notations

We use the following notations throughout the paper. For a square matrix $A_{p\times p} = (a_{ij})$, we denote its spectral norm by $\|A\|_2$; its element-wise ℓ_∞ norm by $\|A\|_\infty = \max_{1\leqslant i,j\leqslant p}|a_{ij}|$; and its Frobenius norm by $\|A\|_F$. We denote the largest eigenvalue and the smallest eigenvalue of A by $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$, respectively. When A is a square symmetric matrix, we note $\|A\|_2 = \lambda_{\max}(A)$, and use $A \succ 0$ to indicate that A is positive definite. For a collection of K square matrices of the same dimension $\boldsymbol{A} = (A_1, \ldots, A_K)$, write $\|\boldsymbol{A}\|_\infty = \max_{1\leqslant k\leqslant K}\|A_k\|_\infty$. Let $\boldsymbol{\Theta}^0 = (\Theta_1^0, \ldots, \Theta_K^0)$ denote the collection of true precision matrices and $\boldsymbol{\mathcal{S}}^0 = (\mathcal{S}_1^0, \ldots, \mathcal{S}_K^0)$ where $\mathcal{S}_k^0 = \{(i,j) : \theta_{k,ij}^0 \neq 0\}$ is the index set of nonzero entries in the true precision matrix Θ_k^0. Define column sparsity of Θ_k^0 as $d_k = \max_{1\leqslant i\leqslant p} card(\{j : \theta_{k,ij}^0 \neq 0\})$ where $card(\cdot)$ denotes the cardinality of a set and let $d = \max_{1\leqslant k\leqslant K} d_k$.

2. Group Estimation of Multiple Bayesian Graphical Models

Suppose we observe multivariate datasets $Y_1, \ldots, Y_K$ from K distinct and related classes, where the k-th dataset consists of n_k observations $Y_k = (Y_{k,1}, \ldots, Y_{k,n_k})$ of a p-dimensional random vector Y that is common to the K classes. Assume that $Y_1, \ldots, Y_K$ follow indepen-

dent Gaussian graphical models:

$$Y_{k,1}, \ldots, Y_{k,n_k} \sim N_p(\mathbf{0}, \Theta_k^{-1}), \quad k = 1, \ldots, K.$$

Our target is to estimate the K precision matrices $\Theta_k = (\theta_{k,ij})$ and identify their sparsity structures. Because of the equivalence of $\theta_{k,ij} = 0$ and the conditional independence between the i-th and j-th components of Y (Dempster, 1972), the sparsity structure of Θ_k reveals the dependence structure of Y in the k-th class.

A naive approach for this problem is to estimate $\Theta_k = (\theta_{k,ij})$, $k = 1, \ldots, K$, individually. However, since the p-dimensional random vector Y is common to the K related classes, it is reasonable to assume that the K matrices share some common structure. The naive approach, which does not utilize the cross-class similarity information, may not be effective. To borrow strength across classes and improve the efficiency of estimation and structure recovery, we treat entries of the same indices across K precision matrices $\boldsymbol{\theta}_{ij} = (\theta_{1,ij}, \ldots, \theta_{K,ij})$ as one group and encourage them to have similar sparsity patterns.

2.1 Bayesian Model Formulation

We now specify the Bayesian formulation of our model by first introducing the group level binary indicators γ_{ij}'s. Each indicator γ_{ij} is used to gather the group-level sparsity information for the corresponding group $\boldsymbol{\theta}_{ij}$ and indicate whether there is an edge between nodes i and j in at least one of the K graphs ($\gamma_{ij} = 1$) or not ($\gamma_{ij} = 0$). As γ_{ij}'s are binary, we place a Bernoulli prior on the γ_{ij}'s:

$$\gamma_{ij} \sim \text{Bern}(p_1).$$

Conditioning on γ_{ij}, we then introduce individual binary indicators $r_{k,ij}$'s to indicate the sparsity of the corresponding entries $\theta_{k,ij}$'s. Priors on $\{r_{k,ij}\}_{k=1}^K \mid \gamma_{ij}$ are placed to encourage similarities among them:

$$r_{1,ij}, \ldots, r_{K,ij} \mid \gamma_{ij} \sim f(\cdot \mid \gamma_{ij}),$$

where $f(\cdot \mid \gamma_{ij})$ is a distribution that encodes the sparsity sharing pattern of our model. We shall discuss the choices for $f(\cdot \mid \gamma_{ij})$ in Section 2.2.

We now discuss prior specification on $\theta_{k,ij}$ given the binary indicator $r_{k,ij}$. When $r_{k,ij} = 1$, we assume $\theta_{k,ij}$ is a "signal" entry that is likely to take a large value; when $r_{k,ij} = 0$, $\theta_{k,ij}$ is a "noise" entry that is tightly centered around zero. To achieve this mechanism, we place the following spike-and-slab Lasso prior (Ročková, 2018; Gan et al., 2019a) on the upper triangular entries $\theta_{k,ij}$ $(i < j)$:

$$\pi(\theta_{k,ij}|r_{k,ij}) = \begin{cases} \text{LP}(\theta_{k,ij}; v_1) \text{ when } r_{k,ij} = 1, \\ \text{LP}(\theta_{k,ij}; v_0) \text{ when } r_{k,ij} = 0, \end{cases}$$

where $v_1 > v_0 > 0$ and $\text{LP}(\cdot; v)$ denotes a Laplace distribution with density function $\exp(-|\theta|/v)/(2v)$. Here, $\text{LP}(\theta_{k,ij}; v_0)$ represents the spike component with a small variance that would induce values close to zero and $\text{LP}(\theta_{k,ij}; v_1)$ represents the slab component with a large variance that would produce large signals. Additionally, we enforce $\theta_{k,ij} = \theta_{k,ji}$ for $i < j$ to keep Θ_k's symmetric.

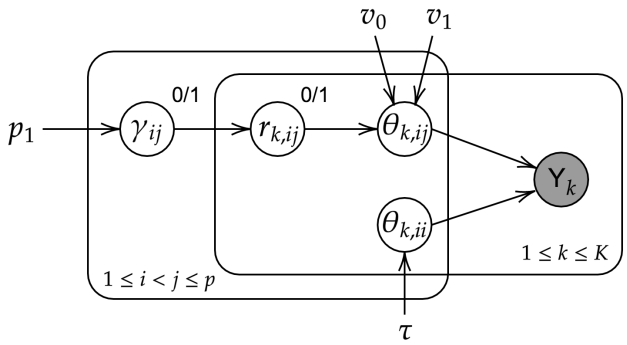


Figure 1: A graphical representation of our model. The boxes are "plates" representing replicates: the left plate represents indices of precision matrices and the right plate represents different classes.

We also place an exponential prior on the positive diagonal entries of the K precision matrices to introduce proper shrinkage:

$$\theta_{k,ii} \sim \text{Exp}(\tau).$$

The full Bayesian prior formulation of our proposed method is thus given by:

$$\begin{aligned} \theta_{k,ii} &\sim \text{Exp}(\tau), \\ \theta_{k,ij} \mid r_{k,ij} &\sim r_{k,ij}\text{LP}(\theta_{k,ij}; v_1) + (1 - r_{k,ij})\text{LP}(\theta_{k,ij}; v_0), \\ r_{k,ij} \mid \gamma_{ij} &\sim f(\cdot \mid \gamma_{ij}), \\ \gamma_{ij} &\sim \text{Bern}(p_1), \end{aligned} \tag{1}$$

for $k = 1, \ldots, K$; $1 \leqslant i < j \leqslant p$. Figure 1 gives a graphical representation of our Bayesian model.

2.2 Priors for Group-level and Bi-level Sparsity

In our prior formulation (1), a key element is the distribution $f(\cdot \mid \gamma_{ij})$, which encodes the sparsity sharing pattern of our model. In this paper, we consider two specific forms of $f(\cdot \mid \gamma_{ij})$: one is used to encourage a common *group-level sparsity*; the other one is used to encourage *bi-level sparsity*, where some components within a non-sparse group are allowed to be sparse.

Group-Level Sparsity. If the sparsity structure at the group-level is the main target in a real application, or if it is reasonable to assume that entries within $\boldsymbol{\theta}_{ij}$ are either signals or noises simultaneously, we set $f(\cdot \mid \gamma_{ij}) = \gamma_{ij}$, which enforces $r_{k,ij}$'s to be the same as γ_{ij}, i.e.,

$$r_{k,ij} = \gamma_{ij}. \tag{2}$$

Under this formulation, the individual binary indicators $r_{k,ij}$'s in (1) become redundant and can all be replaced by their corresponding group level indicator γ_{ij}.

Bi-Level Sparsity. In some applications, we may be interested in not only the group-level sparsity patterns, but also the within-group individual sparsity patterns. This is usually referred to as the bi-level selection problem in literature and a variety of methods have been proposed to perform bi-level selection in the regression setting (Huang et al., 2009; Simon et al., 2013; Xu and Ghosh, 2015). In this paper, we consider $f(\cdot \mid \gamma_{ij})$ to have the following hierarchical structure for the bi-level sparsity:

$$r_{k,ij} \mid \gamma_{ij} \sim \gamma_{ij}\text{Bern}(p_2) + (1-\gamma_{ij})\delta_0(r_{k,ij}), \tag{3}$$

where $\delta_0(\cdot)$ denotes a point mass at zero. Under this setup, when the group level indicator $\gamma_{ij} = 0$, all the $r_{k,ij}$'s within the group indexed by (i,j) have to be 0 altogether; when the group sparsity indicator $\gamma_{ij} = 1$, each of $r_{k,ij}$'s can still take 0 with probability $1-p_2$ independently.

The bi-level prior provides a flexible approach to encourage information sharing at the group level and meanwhile allow within-group heterogeneity. In (1), when $p_1 = 1$ so that γ_{ij}'s have to be all 1's, the bi-level formulation (3) becomes a special case that every class of the data is modeled independently and Θ_k's are estimated separately. On the other hand, when $p_2 = 1$, the bi-level formulation (3) degenerates to the one for the group-level sparsity (2). Thus, the joint estimation method in Gan et al. (2019b) is a special case of our general framework of bi-level prior specification (3). The bi-level prior with $p_1 \in (0,1)$ and $p_2 \in (0,1)$ represents a middle ground where different classes share commonality of structures while not subjected to have the exact same structure.

In the remaining part of the paper, we only focus on the bi-level prior specification for convenience of illustration. Our prior on the precision matrices $\mathbf{\Theta} = (\Theta_1, \ldots, \Theta_k)$ after integrating out γ_{ij}'s and $r_{k,ij}$'s is specified as follows:

$$\begin{aligned} \pi(\theta_{k,ii}) &= \tau \exp(-\tau\theta_{k,ii}), \quad \theta_{k,ii} > 0, \\ \pi(\boldsymbol{\theta}_{ij}) &= p_1 \prod_{k=1}^{K} [p_2 \text{LP}(\theta_{k,ij}; v_1) + (1-p_2)\text{LP}(\theta_{k,ij}; v_0)] + (1-p_1) \prod_{k=1}^{K} \text{LP}(\theta_{k,ij}; v_0), \end{aligned} \tag{4}$$

where $k = 1, \ldots, K$ and $1 \leqslant i < j \leqslant p$. The Bayesian joint model (1) with this prior will be called **GemBag**, short for **G**roup **e**stimation of **m**ultiple **Ba**yesian **g**raphical models.

For the problem of multi-task GGM, a key element is the modeling of the similarity of the multiple underlying graphs. It is quite natural in the Bayesian framework to consider a hierarchical prior to induce similar sparsity patterns of the multiple graphs (Peterson et al., 2015; Tan et al., 2017). While the general idea of using a hierarchical prior is natural, it is not trivial to design the hierarchies and choose appropriate prior distributions to achieve desirable theoretical properties and computational efficiency. In comparison with the existing approaches, our prior specification allows more efficient computations while allowing efficient estimation by sharing information from across multiple graphs.

Peterson et al. (2015) use a Markov random field (MRF) model for learning the similarity between the structure of the multiple graphs which involves a $K \times K$ matrix valued parameter with a hyper-prior placed on it. Due to this, their approach is computationally

not feasible unless the number of graphs K being considered is very small due to the computational intractability of the normalizing constant of the MRF model. In contrast, our proposal of the simple bi-level Bernoulli prior allows efficient computation even when the number of graphs K is large and yet allows sharing of information across multiple graphs. For modeling the precision matrices given the structure of the underlying graphs, Peterson et al. (2015) use a G-Wishart prior which has normalizing constants in both the prior and the posterior that pose computational difficulties. In contrast, we place the spike-and-slab Lasso prior on the off-diagonal precision matrices entries. This formulation has the nice interpretation of differentiating "signal" and "noise" entries and facilitate the computation due to the conditional independence of the precision matrices entries given the structure of the underlying graphs. Tan et al. (2017) also use G-Wishart priors with intractable normalizing constants making it not computationally scalable for large graphs.

Besides the computational challenge, theoretical properties are not established for the procedures of Peterson et al. (2015); Tan et al. (2017) and it is unclear how they would perform theoretically in terms of estimation accuracy and recovering the underlying graphs. In contrast, the theoretical properties we establish for our prior specification help us show that GemBag procedure enjoys optimal theoretical properties in terms of ℓ_∞ norm estimation accuracy and correct recovery of the graphical structure (see Section 3).

We close this subsection with a remark on Covariate Dependent Sparsity, which allows covariates to impact sparsity structures of graphs. For example, in our model, $p_1,\ p_2$ can be treated as functions of covariates, if additional covariates are available. Let x_k denote class specific covariates of the k-th sample class, we can extend the prior placed on $r_{k,ij}$ in (3) in the following manner:

$$r_{k,ij} \mid \gamma_{ij}, \pi_{k,i}, \pi_{k,j} \sim \gamma_{ij}\text{Bern}(\pi_{k,i}\pi_{k,j}) + (1-\gamma_{ij})\delta_0(r_{k,ij}),\ \text{where logit}(\pi_{k,i}) = \beta_i^T x_k.$$

While it is possible to generalize our methodology for this prior, the theoretical analysis and computational techniques deserve an independent study and we leave it for future exploration. In the remaining part of the paper, we focus on the settings when $p_1,\ p_2$ do not depend on covariates.

2.3 Parameter Inference

2.3.1 Point Estimation

For efficient computations, we estimate $\mathbf{\Theta}$ via point estimation. Motivated by Gan et al. (2019a), we consider the following MAP estimator:

$$\begin{aligned}\hat{\mathbf{\Theta}} &= \underset{\Theta_k \in \mathcal{S}_+^p,\ k=1,\ldots,K}{\arg\min} \left(-\log \prod_k p(Y_k \mid \Theta_k) - \log \prod_{i=1}^p \prod_{k=1}^K \pi(\theta_{k,ii}) - \log \prod_{i<j} \pi(\boldsymbol{\theta}_{ij}) \right) \\ &= \underset{\Theta_k \in \mathcal{S}_+^p,\ k=1,\ldots,K}{\arg\min} \left(-\sum_k \log p(Y_k \mid \Theta_k) + \sum_{i=1}^p \sum_{k=1}^K \text{Pen}_{L_1}(\theta_{k,ii}) + \sum_{i<j} \text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij}) \right), \quad (5)\end{aligned}$$

where $\mathcal{S}_+^p$ denotes the cone of positive definite matrices with dimension $p \times p$,

$$\log p(Y_k \mid \Theta_k) = \frac{n_k}{2}\Big(\log\det(\Theta_k) - \text{tr}(S_k\Theta_k) \Big)$$

is the log likelihood with $S_k = \frac{1}{n_k}\sum_{i=1}^{n_k} Y_{k,i}Y_{k,i}^T$ being the sample covariance matrix of the k-th class, and $\text{Pen}_{L_1}(\theta_{k,ii})$ and $\text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij})$ are $(-\log)$ of the prior on the diagonal entry $\theta_{k,ii}$ and the prior on the off-diagonal entries $\boldsymbol{\theta}_{ij}$, respectively:

$$\begin{cases} \text{Pen}_{L_1}(\theta_{k,ii}) & = \tau\theta_{k,ii}, \\ \text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij}) & = -\log\left(p_1 \prod_{k=1}^K \left(\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}} \right) \right. \\ & \qquad \left. + (1-p_1) \prod_{k=1}^K \left(\frac{1}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}} \right) \right), \end{cases}$$

The MAP estimator can be interpreted as a penalized likelihood estimator with penalty functions, $\text{Pen}_{L_1}(\theta_{k,ii})$ and $\text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij})$, induced from our prior specification (4). The exponential prior on the diagonal entry $\theta_{k,ii}$ leads to the usual ℓ_1 penalization. The induced penalty from the bi-level prior on a group of parameters $\boldsymbol{\theta}_{ij}$, however, does not correspond to any standard penalty function, which we shall discuss in Section 2.3.3.

2.3.2 Structure Recovery

Given the point estimation $\hat{\boldsymbol{\Theta}}$, we further quantify the uncertainty of the sparsity patterns through the marginal inclusion probability $\mathbb{P}(r_{k,ij} = 1 \mid \hat{\boldsymbol{\Theta}})$.

For any $\boldsymbol{\Theta}$ value, the marginal inclusion probability can be computed as follows (see the detailed derivation in Appendix C).

$$\begin{aligned} \omega_{k,ij}(\boldsymbol{\theta}_{ij}) \triangleq \mathbb{P}\left(r_{k,ij} = 1 \mid \boldsymbol{\Theta}\right) &= \mathbb{P}\left(r_{k,ij} = 1 \mid \boldsymbol{\theta}_{ij}\right) \\ &= \mathbb{P}\left(\gamma_{ij} = 1 \mid \boldsymbol{\theta}_{ij}\right) \cdot \mathbb{P}\left(r_{k,ij} = 1 \mid \gamma_{ij} = 1,\ \theta_{k,ij}\right) \\ &= \eta_1(\boldsymbol{\theta}_{ij}) \cdot \eta_2(\theta_{k,ij}), \end{aligned} \tag{6}$$

where

$$\begin{cases} \eta_1(\boldsymbol{\theta}_{ij}) \triangleq \mathbb{P}\left(\gamma_{ij} = 1 \mid \boldsymbol{\theta}_{ij}\right) & = \frac{p_1 S_1(\boldsymbol{\theta}_{ij})}{p_1 S_1(\boldsymbol{\theta}_{ij}) + (1-p_1) S_2(\boldsymbol{\theta}_{ij})}, \\ \eta_2(\theta_{k,ij}) \triangleq \mathbb{P}\left(r_{k,ij} = 1 \mid \gamma_{ij} = 1,\ \theta_{k,ij}\right) & = \frac{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}}}, \end{cases} \tag{7}$$

with

$$S_1(\boldsymbol{\theta}_{ij}) = \prod_{k=1}^K \left(\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}} \right) \text{ and } S_2(\boldsymbol{\theta}_{ij}) = \prod_{k=1}^K \left(\frac{1}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}} \right). \tag{8}$$

Based on the MAP estimator $\hat{\boldsymbol{\Theta}}$ from (5), we obtain an estimate of the sparsity patterns $\boldsymbol{\mathcal{S}}$ by thresholding the posterior inclusion probabilities $\omega_{k,ij}(\hat{\boldsymbol{\theta}}_{ij})$:

$$\hat{r}_{k,ij} = 1 \Longleftrightarrow \omega_{k,ij}(\hat{\boldsymbol{\theta}}_{ij}) > t, \quad \text{for } t \in (0,1). \tag{9}$$

The expressions (6)-(8) provide some insight about how information across classes is shared for structure recovery. The first quantity $\eta_1(\boldsymbol{\theta}_{ij})$ represents the group-level inclusion probability for $\boldsymbol{\theta}_{ij}$, the group of entries indexed by (i,j) across Θ_k's, and the second quantity $\eta_2(\theta_{k,ij})$ is the within-group inclusion probability for entry $\theta_{k,ij}$ from the k-th class. The product of them, which takes both the group level and individual level information into consideration, is in effect when performing the sparsity structure recovery.

2.3.3 The Bi-level Bayesian Penalty

To understand the Bayesian penalty induced from our bi-level prior specification, we examine the gradient of $\text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij}) = -\log \pi(\boldsymbol{\theta}_{ij})$. Calculation reveals that

$$\frac{\partial}{\partial|\theta_{k,ij}|}\text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij}) = \omega_{k,ij}(\boldsymbol{\theta}_{ij})\frac{1}{v_1} + (1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}))\frac{1}{v_0}, \tag{10}$$

where $\omega_{k,ij}(\boldsymbol{\theta}_{ij})$ is defined in (6).

Equations (6) and (10) together provide intuition on how our proposed prior formulation works. The penalty induced penalty by our prior is a weighted sum of two ℓ_1 penalties: a large penalty $1/v_0$ that is used to eliminate noise entries and a smaller penalty $1/v_1$ that is used to reduce bias due to over-shrinkage on relevant entries. The weight, determined by the conditional probability of $r_{k,ij} = 1$, can be decoupled as the product of group-level posterior inclusion probability and within-group level posterior inclusion probability as shown by (6) and measures the group information and the second term measures the individual information, respectively.

The surprisingly simple representation (10) of the gradient unveils how adaptive shrinkage is induced based on posterior inclusion probabilities, and also plays an important role in our theoretical analysis. In contrast to the alternative prior choices proposed in the literature such as the Markov random field prior (Peterson et al., 2015) and the Chung-Lu random graph prior (Tan et al., 2017), our prior allows us to investigate the theoretical properties of the proposed Bayesian framework.

Note that the gradient (10) is a decreasing function with respect to the magnitude of $\theta_{k,ij}$, if we fix the other elements of $\boldsymbol{\theta}_{ij}$. That is, the bi-level Bayesian penalty is a concave function of $|\theta_{k,ij}|$. In the regression setting, several concave penalties have been proposed and shown to have the oracle property for bi-level selection (Huang et al., 2012; Breheny and Huang, 2009).

3. Theoretical Guarantees

In this section, we provide theoretical guarantees for MAP estimators that take the following general form including (5) as a special case:

$$\hat{\boldsymbol{\Theta}} = \underset{\Theta_k \in \mathcal{S}_+^p,\ k=1,\ldots,K}{\arg\min} \left(-\sum_k \log p(Y_k \mid \Theta_k) + \alpha \sum_{i=1}^{p}\sum_{k=1}^{K} \text{Pen}_{L_1}(\theta_{k,ii}) + \alpha \sum_{i<j} \text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij}) \right), \tag{11}$$

where $\alpha \geqslant 1$. From the Bayesian perspective, we can view the estimator above as a MAP estimator of the posterior distribution $\pi(\boldsymbol{\Theta} \mid \boldsymbol{Y}) \propto p(\boldsymbol{Y} \mid \boldsymbol{\Theta})\,\pi(\boldsymbol{\Theta})^{\alpha}$ or of $\pi(\boldsymbol{\Theta} \mid \boldsymbol{Y}) \propto p(\boldsymbol{Y} \mid \boldsymbol{\Theta})^{1/\alpha}\,\pi(\boldsymbol{\Theta})$. The latter, known as the fractional Bayesian posterior (Jiang and Tanner, 2008; Yang et al., 2017), has often been considered for Bayesian theoretical analysis.

The objective function (11) is not convex due to the non-convexity of the penalty function $\text{Pen}_{\text{GB}}(\cdot)$ and, consequently, has multiple local solutions. Despite this, we show that there exists a local solution $\tilde{\boldsymbol{\Theta}}$ of (11) enjoys optimal rate of convergence in ℓ_∞ norm and has a stronger performance for structure recovery performance when compared to other

alternatives. If an additional constraint $\|\Theta_k\|_2 \leqslant B, k = 1, \ldots, K$, is imposed on the parameter space Ω with a reasonably large B, we show the objective function (11) is strictly convex, and thus, the local solution $\tilde{\boldsymbol{\Theta}}$ is the unique solution.

3.1 Conditions

In our theoretical analysis, we allow the true data generating distribution to be non-Gaussian and consider distributions with exponential tails (sub-Gaussian distributions) or with polynomial tails (such as the t distributions). The same conditions are also considered in Cai et al. (2011) and Gan et al. (2019a) when only one class is present in the data, that is, $K = 1$. For all the p-dimensional random vectors $Y_{k,i} = (Y_{k,i}^{(1)}, \ldots, Y_{k,i}^{(p)}), i = 1, \ldots, n_k$ and $k = 1, \ldots, K$, we define

(A.1) Exponential tail condition: there exist some constants $0 < \zeta < 1/4$ and $U > 0$ such that $(\log p)/n < \zeta$ and

$$\mathbb{E}(e^{tY_{k,i}^{(j)}}) \leqslant U \quad \text{for any } |t| \leqslant \zeta \text{ and } j = 1, \ldots, p;$$

(A.2) Polynomial tail condition: there exist some constants $\kappa_1, \kappa_2, \kappa_3, U > 0$ such that $p \leqslant \kappa_1 n^{\kappa_2}$ and

$$\mathbb{E}|Y_{k,i}^{(j)}|^{4+4\kappa_2+\kappa_3} \leqslant U \quad \text{for } j = 1, \ldots, p.$$

(A.3) Eigenvalue condition: $1/\xi_0 \leqslant \lambda_{\min}(\Theta_k^0) \leqslant \lambda_{\max}(\Theta_k^0) \leqslant 1/\xi_1 \quad$ for $k = 1, \ldots, K$.

When presenting the theoretical results, we also assume the sample sizes of the K classes to be the same with $n_1 = \cdots = n_K = n$ for simplicity.

3.2 Estimation Accuracy

With the aforementioned assumptions, we are now ready to present our result on the rate of convergence in ℓ_∞ norm.

Theorem 1 *Suppose that one of the tail conditions, (A.1) or (A.2), and the eigenvalue condition (A.3) hold. Let* $C_1 = \zeta^{-1}(2 + \kappa_0 + \zeta^{-1}U^2)$ *when the exponential tail condition (A.1) holds and* $C_1 = \sqrt{(\|\boldsymbol{\Theta}^0\|_\infty + 1)(4 + \kappa_0)}$ *when the polynomial tail condition (A.2) holds for some* $\kappa_0 > 0$. *In addition, assume that:*

(i) the hyperparameters $(v_1,\ v_0,\ p_1,\ p_2,\ \tau)$ *satisfy the following when* $p_2 = 1$*:*

$$\begin{cases} \max(\frac{5}{nv_1}, \frac{2\tau}{n}) < \frac{C_3}{\alpha}\sqrt{\frac{\log p}{n}}, \quad \frac{1}{nv_0} > \frac{C_4}{\alpha}\sqrt{\frac{\log p}{n}}, \\ \epsilon_2 < \frac{v_1^K(1-p_1)}{v_0^K p_1} \leqslant \frac{v_1^{K+2}(1-p_1)}{v_0^{K+2}p_1} \leqslant 2p^{\epsilon_0/\alpha}, \end{cases}$$

or when $p_2 < 1$*:*

$$\begin{cases} \max(\frac{5}{nv_1}, \frac{2\tau}{n}) < \frac{C_3}{\alpha}\sqrt{\frac{\log p}{n}}, \quad \frac{1}{nv_0} > \frac{C_4}{\alpha}\sqrt{\frac{\log p}{n}}, \\ \frac{v_1^{K+2}(1-p_1)}{v_0^{K+2}p_1} \leqslant 2p^{\epsilon_0/\alpha}p_2^K, \\ \epsilon_2 < \frac{v_1(1-p_2)}{v_0 p_2} < \frac{v_1^3(1-p_2)}{v_0^3 p_2} \leqslant 2p^{\epsilon_0/\alpha}; \end{cases}$$

(ii) the sample size n satisfies: $\sqrt{n} \geqslant M_0 \max(d, \sqrt{K})\sqrt{\log p}$;
(iii) the parameter α satisfies: $\alpha p^{\epsilon_0/\alpha} > KC_3^2 \log p/(2\xi_1^2)$.
Then, there exists a local solution of (11) *that satisfies*

$$\|\tilde{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^0\|_\infty < C_5\sqrt{\frac{\log p}{n}}$$

with probability greater than $1 - K\delta$, *where* $\delta = 2p^{-\kappa_0}$ *when condition (A.1) holds, and* $\delta = O(n^{-\kappa_3/8} + p^{-\kappa_0/2})$ *when condition (A.2) holds. Moreover,* $\tilde{\theta}_{k,ij} = 0$ *for* $(i,j) \in \left(\mathcal{S}_k^0\right)^c$. *Here,* C_3, ϵ_2 *are sufficiently small positive constants;* $M_0, C_4, C_5, \epsilon_0$ *are positive constants that only depend on the ground truth* $\boldsymbol{\Theta}^0$.

Our proof of Theorem 1 is motivated by the constructive proof technique used in Ravikumar et al. (2011) and Gan et al. (2019a). The exact definitions of $M_0, C_4, C_5, \epsilon_0$ and the proof of Theorem 1 are provided in Appendix A.1. We note that the penalty from our bi-level prior specification (10) is a weighted combination of two ℓ_1 penalties. Through the interplay of (v_0, v_1, p_1, p_2), the consequence of condition (i) is to ensure that $\text{Pen}_{\text{GB}}(\boldsymbol{\theta}_{ij})$ achieves adaptive shrinkage: small entries in $\boldsymbol{\theta}_{ij}$ lead to penalty that is bigger than the order of $\sqrt{(\log p)/n}$ to suppress noise, whereas large entries lead to penalty that is smaller in order than $\sqrt{(\log p)/n}$ to reduce bias. To make condition (i) hold, we require the rate of $1/v_1$ to be at most $\sqrt{n \log p}/\alpha$ and the rate of $1/v_0$ to be at least $\sqrt{n \log p}/\alpha$. While condition (i) does not impose any requirement on the rate of α, a large α may imply extreme rates of other parameters like p_1 and p_2 to satisfy condition (i). For example, when $\alpha = n$, $p^{\epsilon_0/\alpha}$ would go to 1 when $n \to \infty$ and $\log p/n \to 0$. Therefore, to make the condition $\frac{v_1^{K+2}(1-p_1)}{v_0^{K+2}p_1} \leqslant 2p^{\epsilon_0/\alpha}$ hold, we require p_1 to be close enough to 1. While an α of moderate rate makes condition (i) easier to satisfy in theory, we observe in empirical studies that our method with α of different rates such as $\alpha = 1, \sqrt{n}, n$, has a comparable performance. Under the conditions stated, Theorem 1 guarantees that there exists a local minimizer $\tilde{\boldsymbol{\Theta}}$ that has an optimal rate of convergence for estimation in ℓ_∞ norm.

3.3 Sparsity Structure Recovery

The local minimizer $\tilde{\boldsymbol{\Theta}}$ from Theorem 1 also has desirable properties in sparsity structure recovery. Let us first consider the simple case when sparsity patterns of Θ_k^0's are the same, that is, $\mathcal{S}_1^0 = \cdots = \mathcal{S}_K^0$. The following theorem establishes the selection consistency of $\tilde{\boldsymbol{\Theta}}$ with a mild condition on the minimal signal strength.

Theorem 2 *Under the same conditions (i) – (iii) in Theorem 1 and the following conditions:*
(iv) minimal signal strength satisfies the following when $\mathcal{S}_1^0 = \cdots = \mathcal{S}_K^0$:

$$\min_{i \neq j} \left(\max_{k:(i,j) \in \mathcal{S}_k^0} (|\theta_{k,ij}^0|) \right) \geqslant L_0\sqrt{\log p/n},$$

where $L_0 > C_5$ *is a sufficiently large constant;*

(v) the hyperparameters $(v_1,\ v_0,\ p_1,\ p_2)$ *satisfy:*

$$\eta_1(\mathbf{0}) \cdot \eta_2(0) < t < \frac{1}{1 + \frac{2p^{\epsilon_0/\alpha}}{p^{(C_4-C_3)(L_0-C_5)/\alpha}}} \cdot \eta_2(0),$$

where $t \in (0,1)$ *is an arbitrary thresholding value,*

$$\eta_1(\mathbf{0}) = \frac{1}{1 + \frac{(1-p_1)v_1^K}{p_1[p_2v_0+(1-p_2)v_1]^K}} \text{ and } \eta_2(0) = \frac{1}{1 + \frac{(1-p_2)v_1}{p_2v_0}},$$

we have

$$\mathbb{P}(\tilde{\boldsymbol{\mathcal{S}}} = \boldsymbol{\mathcal{S}}^0) \to 1.$$

A proof of Theorem 2 is provided in Appendix A.2. In the proof, we show that for any threshold t between 0 and 1, we can differentiate the signal entries and the noise entries based on their posterior inclusion probabilities with a proper choice of the hyperparameters $(v_1,\ v_0,\ p_1,\ p_2)$. It is worth noting that in condition (iv), we only impose a condition on the maximum magnitude within each group. This illustrates the advantage of GemBag for identification of group sparsity when many signals in a group may be weak. When the sparsity structures of different graphs are not exactly the same, GemBag still performs well. The following theorem shows that GemBag consistently recovers the sparsity structure of each individual graphical model.

Theorem 3 *Under the same conditions (i) – (iii) in Theorem 1 and the following conditions:*

(vi) minimal signal strength satisfies the following when sparsity patterns of Θ_k^0*'s are not the same:*

$$\min_{i\neq j}\left(\min_{k:(i,j)\in\mathcal{S}_k^0}(|\theta_{k,ij}^0|)\right) \geqslant L_1\sqrt{\log p/n},$$

where $L_1 > C_5$ *is a sufficiently large constant;*

(vii) the hyperparameters $(v_1,\ v_0,\ p_1,\ p_2)$ *satisfy:*

$$\frac{(1-p_2)v_1}{p_2v_0} > \frac{1-t}{t} > \left(1 + \frac{2p^{\epsilon_0/\alpha}}{p^{(C_4-C_3)(L_1-C_5)/\alpha}}\right)^2 - 1,$$

we have

$$\mathbb{P}(\tilde{\boldsymbol{\mathcal{S}}} = \boldsymbol{\mathcal{S}}^0) \to 1.$$

A proof of Theorem 3 is provided in Appendix A.3. In condition (vi), we require the minimal signal strength on each individual signal entry, which is the same signal strength requirement from individual estimation method (Gan et al., 2019a). Compared to individual estimation, GemBag would have better performance and smaller misclassification error when the sparsity patterns are similar across different classes. Recall that our posterior inclusion probability (6) takes both group-level inclusion probability $\eta_1(\boldsymbol{\theta}_{ij})$ and within-group inclusion probability $\eta_2(\theta_{k,ij})$ into consideration. As a result, for entries with weak

signal strength and small within-group inclusion probabilities, they could still be correctly identified as long as signal strength at the group level is strong and group-level inclusion probabilities are large. In contrast, by individual estimation method, those entries with weak signals would be misclassified as noises without the group-level information. A more rigorous comparison is provided in Appendix D.

3.4 Local Strict Convexity

A potential challenge is caused by the non-convexity of the objective function.

It is easy to check that the whole objective function (11) is non-convex with multiple local solutions. Therefore finding $\hat{\mathbf{\Theta}}$, the minimizer of (11), is not straightforward.

Motivated by Loh and Wainwright (2017); Gan et al. (2019a), we consider estimating the MAP estimator in the parameter space with a side constraint $\Omega = \{\|\Theta_k\|_2 \leqslant B, \Theta_k \succ 0, k = 1, \ldots, K\}$:

$$\hat{\mathbf{\Theta}} = \underset{\Theta_k \in \mathcal{S}_+^p,\ k=1,\ldots,K}{\arg\min} \left(-\sum_k \log p(Y_k \mid \Theta_k) + \alpha \sum_{i=1}^{p} \sum_{k=1}^{K} \mathrm{Pen}_{L_1}(\theta_{k,ii}) + \alpha \sum_{i<j} \mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij}) \right) \quad \text{subject to } \|\Theta_k\|_2 \leqslant B, \Theta_k \succ 0, k = 1, \ldots, K. \tag{12}$$

As shown in the following theorem, even when the bound B diverges to $+\infty$, the optimization problem is strictly convex. That is to say, within a reasonably large parameter space, the solution we obtained from any algorithm that solves (11) will be unique and globally optimal. A proof of Theorem 4 is provided in Appendix A.4.

Theorem 4 *If* $1/\xi_1 + dC_5\sqrt{\frac{\log p}{n}} < B < \left(\frac{nv_0^2}{\alpha K}\right)^{1/2}$, *then the local minimizer* $\tilde{\mathbf{\Theta}}$ *of* (11) *is also the unique solution of* (12).

Theorem 4 suggests that the objective function in (12) is strictly convex even when B is quite large and is growing to infinity. For a given prior parameter v_0, the upper bound on B decreases as α increases which is expected because the weight of the non-convex regularization function becomes larger. However, for establishing the optimal theoretical results of ℓ_∞ norm convergence in Theorem 1, the hyperparameter v_0 depends on α and is required to be $O(\alpha/\sqrt{n \log p})$. To empirically corroborate the rate of v_0 for achieving optimal performance, we provide a simulation study in Appendix E.2 which supports our theoretical results that v_0 scales linearly with $\alpha/\sqrt{n \log p}$. Therefore, the upper bound on B to achieve both strict local convexity and optimal estimation accuracy is $O(\sqrt{\alpha/(K \log p)})$, which grows to infinity as long as α grows faster than the rate of $K \log p$. To gain more insight into the convexity result, recall that the rate of v_0 from Theorem 1, $O(\alpha/\sqrt{n \log p})$, can be larger with a larger α and the density of the spike component would be flatter and thus, the non-convexity of the regularization function becomes less severe. While this does not guarantee that the MAP estimator with $\alpha = 1$ is unique for the specific choice of v_0 required for attaining the consistency results, we observe in empirical studies that the performance of a MAP estimator we find is as good as the unique estimator with a larger α.

3.5 Comparisons with Existing Works

We consider existing joint estimation methods proposed by Guo et al. (2011) and Lee and Liu (2015) for comparison in terms of estimation accuracy and selection consistency. In the following discussion, we use $\hat{\mathbf{\Theta}}'$ as a generic notation to denote estimators proposed by these authors.

Guo et al. (2011) established the estimation consistency of their estimator $\hat{\mathbf{\Theta}}'$ in Frobenius norm for a fixed K value:

$$\sum_{k=1}^{K} \|\hat{\Theta}'_k - \Theta^0_k\|_F = O_p\left(\sqrt{\frac{(p+q_1)\log p}{n}}\right),$$

where $q_1 = card(\cup_k \mathcal{S}^0_k) - p$. Our Theorem 1 implies that the estimation error upper bound of our estimator $\tilde{\mathbf{\Theta}}$ under this Frobenius norm is $O\left(\sqrt{\frac{(p+q_2)\log p}{n}}\right)$, where $q_2 = \max_k\{card(\mathcal{S}^0_k)\} - p$. This rate is the same as that of $\hat{\mathbf{\Theta}}'$ when K is fixed. For graph structure recovery, Guo et al. (2011) obtained sparsistency, a notion from Lam and Fan (2009), in the sense that the zero entries in the true precision matrices are estimated as zeroes with probability tending to one (For our method, sparsistency is stated in Theorem 1). However, there is no guarantee that the nonzero entries could be detected. This is weaker than our selection consistency result as we can recover the entire graph structure. Moreover, to achieve sparsistency, Guo et al. (2011) required the minimum signal $\min_k \min_{i\neq j,(i,j)\in\mathcal{S}^0_k}(|\theta^0_{k,ij}|)$ to be lower bounded by some constant while we allow it to go to 0.

Lee and Liu (2015) established the estimation consistency of their joint estimator of multiple precision matrices (JEMP), $\hat{\mathbf{\Theta}}'$, in the averaged version of the ℓ_∞-ℓ_1 norm:

$$\max_{i,j}\left(\frac{1}{K}\sum_{k=1}^{K}\left|\hat{\theta}'_{k,ij} - \theta^0_{k,ij}\right|\right) = O_p\left(\sqrt{\frac{\log p}{n}}\right).$$

Our estimation error rate under ℓ_∞ norm in Theorem 1 implies this result. Besides, our error rate in Theorem 1 is on the maximum over all entries of all precision matrices without averaging, and therefore is stronger. For selection consistency, JEMP has the same result as ours but requires a stronger condition on the signal strength. Specifically, JEMP implicitly requires $\min_k \min_{i\neq j,(i,j)\in\mathcal{S}^0_k}(|\theta^0_{k,ij}|)$ to be lower bounded at the rate of $K(\log p/n)^{1/2}$. In comparison, we only require a smaller rate of $(\log p/n)^{1/2}$. Moreover, when the sparsity patterns of Θ^0_k's are the same, our condition becomes even weaker as our minimum signal strength rate would be imposed on $\min_k \max_{i\neq j,(i,j)\in\mathcal{S}^0_k}(|\theta^0_{k,ij}|)$ rather than $\min_k \min_{i\neq j,(i,j)\in\mathcal{S}^0_k}(|\theta^0_{k,ij}|)$.

4. EM Algorithm

In this section, we present an EM algorithm to compute the MAP estimator in (12). We treat the indicator variable $\Gamma = (\gamma_{ij})$ and $\boldsymbol{R} = (R_1, \ldots, R_K)$ with $R_k = (r_{k,ij})$ as latent variables and obtain the MAP estimator of $\mathbf{\Theta}$ by iterating the following two steps:

- Expectation step (E-step): compute the Q function defined as the expectation of the full log likelihood with respect to the joint posterior distribution of the latent

variables, Γ and $\boldsymbol{R}$, given $\boldsymbol{Y}$ and $\boldsymbol{\Theta}^{(t)}$, the current estimate of $\boldsymbol{\Theta}$:

$$\begin{aligned} Q(\boldsymbol{\Theta} \mid \boldsymbol{\Theta}^{(t)}) &= \mathbb{E}_{\Gamma,\boldsymbol{R}|\boldsymbol{\Theta}^{(t)},\boldsymbol{Y}} \log\Big(p(\boldsymbol{Y} \mid \boldsymbol{\Theta}, \Gamma, \boldsymbol{R})^{\frac{1}{\alpha}} \pi(\boldsymbol{\Theta}, \Gamma, \boldsymbol{R})\Big) \\ &= \mathbb{E}_{\Gamma,\boldsymbol{R}|\boldsymbol{\Theta}^{(t)},\boldsymbol{Y}} \left(-\sum_{i<j}\sum_{k=1}^{K}\left[r_{k,ij}\frac{|\theta_{k,ij}|}{v_1} + (1-r_{k,ij})\frac{|\theta_{k,ij}|}{v_0}\right]\right) \\ &\quad + \frac{1}{\alpha}\log p(\boldsymbol{Y} \mid \boldsymbol{\Theta}) - \sum_{i=1}^{p}\sum_{k=1}^{K}\tau\theta_{k,ii} + C, \end{aligned} \tag{13}$$

where C is some constant not depending on $\boldsymbol{\Theta}$.

- Maximization step (M-step): update the estimate of $\boldsymbol{\Theta}$ by maximizing the Q function (13) under the constraints that $\Theta_k \succ 0$ and $\|\Theta_k\|_2 \leqslant B$ for $k = 1, \ldots, K$.

4.1 E-step

To evaluate the Q function, we need to compute the conditional expectation $\mathbb{E}_{\Gamma,\boldsymbol{R}|\boldsymbol{\Theta}^{(t)},\boldsymbol{Y}}(r_{k,ij})$, which is the posterior inclusion probability of $r_{k,ij}$'s given $\boldsymbol{\Theta}^{(t)}$ and $\boldsymbol{Y}$ defined in (6):

$$\begin{aligned} \mathbb{E}_{\Gamma,\boldsymbol{R}|\boldsymbol{\Theta}^{(t)},\boldsymbol{Y}}(r_{k,ij}) &= \mathbb{P}\left(r_{k,ij} = 1 \mid \boldsymbol{\Theta}^{(t)}, \boldsymbol{Y}\right) \\ &= \mathbb{P}\left(\gamma_{ij} = 1 \mid \boldsymbol{\theta}_{ij}^{(t)}\right)\mathbb{P}\left(r_{k,ij} = 1 \mid \gamma_{ij} = 1,\ \theta_{k,ij}^{(t)}\right) \\ &= \eta_1(\boldsymbol{\theta}_{ij}^{(t)}) \cdot \eta_2(\theta_{k,ij}^{(t)}). \end{aligned} \tag{14}$$

We write the Q function as

$$\begin{aligned} Q(\boldsymbol{\Theta} \mid \boldsymbol{\Theta}^{(t)}) = \sum_{k=1}^{K}\Bigg\{ &\frac{n_k}{2\alpha}\log\det(\Theta_k) - \frac{n_k}{2\alpha}\mathrm{tr}(S_k\Theta_k) - \sum_{i=1}^{p}\tau\theta_{k,ii} \\ &- \sum_{i<j}\left[\omega_{k,ij}(\boldsymbol{\theta}_{ij}^{(t)})\frac{|\theta_{k,ij}|}{v_1} + \left(1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}^{(t)})\right)\frac{|\theta_{k,ij}|}{v_0}\right]\Bigg\} + C. \end{aligned}$$

4.2 M-step

The function $Q(\boldsymbol{\Theta} \mid \boldsymbol{\Theta}^{(t)})$ is a summation of K terms where the k-th term is a function of Θ_k alone. Therefore, maximizing $Q(\boldsymbol{\Theta} \mid \boldsymbol{\Theta}^{(t)})$ is equivalent to optimizing K sub-problems, where the objective function of the k-th sub-problem is only related to Θ_k and takes the following form:

$$\frac{n_k}{2\alpha}\log\det(\Theta_k) - \frac{n_k}{2\alpha}\mathrm{tr}(S_k\Theta_k) - \sum_{i=1}^{p}\tau\theta_{k,ii} - \sum_{i<j}\left[\omega_{k,ij}(\boldsymbol{\theta}_{ij}^{(t)})\frac{|\theta_{k,ij}|}{v_1} + \left(1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}^{(t)})\right)\frac{|\theta_{k,ij}|}{v_0}\right], \tag{15}$$

subject to the constraints that $\Theta_k \succ 0$ and $\|\Theta_k\|_2 \leqslant B$.

Optimizing (15) is a graphical lasso problem with an adaptive ℓ_1 penalty. The magnitude of shrinkage on $\theta_{k,ij}$ is controlled by $\omega_{k,ij}(\boldsymbol{\theta}_{ij}^{(t)})$. Compared to BAGUS (Gan et al., 2019a),

the adaptive weights $\omega_{k,ij}(\boldsymbol{\theta}_{ij}^{(t)})$ take the group information into consideration through the sharing of $\eta_1(\boldsymbol{\theta}_{ij}^{(t)})$. We adopt the BAGUS algorithm in Gan et al. (2019a) to solve (15). The time complexity of our algorithm is $O(p^3)$, the same as the state-of-the-art algorithms for graphical Lasso (Friedman et al., 2008; Hsieh et al., 2011; Gan et al., 2019a). Our EM algorithm is summarized in Algorithm 1, which outputs estimates of K precision matrices and K posterior inclusion probability matrices.

Algorithm 1 EM algorithm for computing the MAP estimator in (12)

repeat
  Calculate $P_k = (\omega_{k,ij}), k = 1, \ldots, K$ using (14).
  for $k = 1$ to K **do**
    Update Θ_k by maximizing (15).
  end for
until Convergence
return P_k, Θ_k, $k = 1, \ldots, K$.

Implementation of existing Bayesian methods for multiple graphical models such as the proposal in Peterson et al. (2015); Tan et al. (2017) have severe computational limitations due to the use of MCMC-based samplers to sample from the full posterior. While it might be possible to develop an EM algorithm for obtaining the MAP estimators corresponding to the models in Peterson et al. (2015); Tan et al. (2017), it is unclear whether these approaches will be scalable to large graphs due to the intractable normalizing constants of their priors.

5. Simulation Studies

Following the simulation setups in Guo et al. (2011), Danaher et al. (2014); Peterson et al. (2015), Lee and Liu (2015), and Gan et al. (2019a), we assess the performance of GemBag under four different designs: an AR(2) model, a circle model, a nearest-neighbor network, and a scale-free network. In the first two designs, we set $K = 10$ and $p = 50$ and in the last two designs, we set $K = 3$ and $p = 100, 150, 200$. For each $k = 1, \ldots, K$, we generate n_k independently and identically distributed observations from a multivariate normal distribution with mean $\mathbf{0}$ and precision matrix Θ_k. To assess the performance of GemBag when data is not normally distributed, we consider multivariate t-distribution and the details are provided in Appendix E.1. To evaluate the performance of GemBag for a higher dimension, we also consider $p = 400$ and the results for this setting are provided in Appendix E.3. In the first two designs, we consider $n_1 = \cdots = n_K = n = 50, 100, 200$; in the last two designs, we set $n_1 = \cdots = n_K = n = 100$. The details of the four designs are described as follows.

1. AR(2) model: we construct a baseline precision matrix Θ^0 with $\theta_{ii}^0 = 1, \theta_{i,i-1}^0 = \theta_{i-1,i}^0 = 0.5$, and $\theta_{i,i-2}^0 = \theta_{i-2,i}^0 = 0.25$. Then $K = 10$ individual precision matrices are generated by randomly setting 30% of the nonzero off-diagonal elements of Θ^0 to zero.

2. Circle model: we construct a baseline precision matrix Θ^0 with $\theta^0_{ii} = 1, \theta^0_{i,i-1} = \theta^0_{i-1,i} = 0.5$, and $\theta^0_{1p} = \theta^0_{p1} = 0.4$. Then K individual precision matrices and n_k random samples are generated in the same way as in the first design.

3. Nearest-neighbor network: we randomly generate p points on a unit square and construct a baseline nearest-neighbor network by linking any two points that are the o nearest neighbors (based on the Euclidean distance) of each other. We set $o = 3$ in our simulation. Then, we generate $K = 3$ individual networks by adding ρM individual edges to the baseline graph where M is the number of edges in the baseline graph and $\rho = 0, 0.25, 1$.

   Given an individual network, we generate a precision matrix Θ_k that has the same sparsity structure as the corresponding network. The diagonal entries of Θ_k are set to be one, and non-zero off diagonal entries are independently generated from a uniform distribution with support on $[-1, -0.5] \cup [0.5, 1]$. W then divide each off-diagonal element $\theta_{k,ij}$ by $1.01\sqrt{\sum_{i:i\neq j} |\theta_{k,ij}|}\sqrt{\sum_{j:j\neq i} |\theta_{k,ij}|}$ to ensure Θ_k to be positive definite.

4. Scale-free network: many real-world large networks, such as the world wide web, social networks, and collaboration networks, are thought to be scale-free. We construct a baseline scale-free network using the Barabási-Albert model in Barabási and Albert (1999). Then, K individual networks and precision matrices, as well as the n_k random samples, are generated in the same way as in the third design.

We compare GemBag with three alternative methods: (i) fitting each class of observations individually using the BAGUS method (denoted as BAGUS) in Gan et al. (2019a); (ii) fitting a single model on data from all classes using the BAGUS method (denoted as Pooled); (iii) using the group graphical lasso (denoted as GGL) from Danaher et al. (2014). Bayesian approaches based on full posterior sampling (Peterson et al., 2015; Tan et al., 2017) are not considered for comparison as their Markov chain Monte Carlo samplers are not scalable with large p. For all methods, we use a grid search to select the set of hyperparamters that minimizes BIC:

$$\text{BIC} = \sum_{k=1}^{K} \Bigg[n_k \left(-\log\det(\hat{\Theta}_k) + \text{tr}(S_k \hat{\Theta}_k) \right) + \log(n_k) \times card(\{(i,j) : \hat{\theta}_{k,ij} \neq 0, 1 \leqslant i < j \leqslant p\}) \Bigg].$$

For GemBag, we consider $\alpha = 1, \sqrt{n}$, and n and estimate the sparsity patterns using a threshold $t = 0.5$ on the posterior inclusion probabilities. We set $p_1 = p_2 = \sqrt{0.5}$ so that the prior inclusion probability $\mathbb{P}(r_{k,ij} = 1) = p_1 p_2 = 0.5$ and tune (v_0, v_1) with $v_0 = \tau = (0.25, 0.5, 0.75, 1) \times \sqrt{1/(n \log p)}$ and $v_1 = (2.5, 5, 7.5, 10) \times \sqrt{1/(n \log p)}$ when $\alpha = 1$, with $v_0 = \tau = (1, 1.5, 2, 2.5, 3) \times 10^{-2} \times \sqrt{1/\log p}$ and $v_1 = (2, 4, 6, 8) \times \sqrt{1/\log p}$ when $\alpha = \sqrt{n}$, or with $v_0 = \tau = (1, 1.5, 2, 2.5, 3) \times 10^{-3} \times \sqrt{n/\log p}$ and $v_1 = (2, 4, 6, 8) \times \sqrt{n/\log p}$ when $\alpha = n$. We recommend the same range of constants as default choices for tuning v_0 and v_1 when applying our method to new settings. Analysis on how sensitive the performance of our method in graph estimation is to different choices of hyperparameters is provided in

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
			$n = 50,\ p = 50$			
GemBag ($\alpha = 1$)	**0.997(0.001)**	0.581(0.056)	0.721(0.034)	0.973(0.013)	3.095(0.139)	0.656(0.135)
GemBag ($\alpha = \sqrt{n}$)	0.995(0.001)	0.700(0.028)	**0.783(0.021)**	0.974(0.010)	2.923(0.149)	0.707(0.151)
GemBag ($\alpha = n$)	0.995(0.001)	0.660(0.034)	0.751(0.019)	0.967(0.013)	3.158(0.115)	0.665(0.137)
BAGUS	0.983(0.002)	0.660(0.024)	0.657(0.014)	0.859(0.019)	3.227(0.073)	0.938(0.193)
Pooled	0.966(0.004)	**0.983(0.010)**	0.771(0.020)	**0.985(0.004)**	**2.900(0.067)**	**0.492(0.027)**
GGL	0.965(0.009)	0.622(0.039)	0.545(0.021)	0.802(0.018)	4.647(0.133)	0.644(0.072)
			$n = 100,\ p = 50$			
GemBag ($\alpha = 1$)	**0.998(0.000)**	0.899(0.029)	**0.928(0.016)**	**0.998(0.001)**	**1.813(0.111)**	0.502(0.095)
GemBag ($\alpha = \sqrt{n}$)	0.997(0.001)	0.900(0.017)	0.923(0.011)	**0.998(0.001)**	1.829(0.070)	0.506(0.075)
GemBag ($\alpha = n$)	0.996(0.001)	0.891(0.017)	0.909(0.010)	**0.998(0.001)**	1.914(0.083)	0.525(0.071)
BAGUS	0.988(0.002)	0.873(0.018)	0.830(0.019)	0.953(0.012)	2.084(0.064)	0.575(0.098)
Pooled	0.964(0.006)	**0.996(0.005)**	0.771(0.026)	0.989(0.002)	2.851(0.064)	**0.485(0.023)**
GGL	0.963(0.008)	0.763(0.022)	0.629(0.025)	0.876(0.011)	4.158(0.104)	0.551(0.018)
			$n = 200,\ p = 50$			
GemBag ($\alpha = 1$)	0.999(0.000)	0.986(0.008)	**0.986(0.004)**	**1.000(0.000)**	**1.106(0.038)**	0.342(0.052)
GemBag ($\alpha = \sqrt{n}$)	0.999(0.001)	0.987(0.006)	0.981(0.006)	**1.000(0.000)**	1.126(0.041)	**0.315(0.047)**
GemBag ($\alpha = n$)	**1.000(0.000)**	0.973(0.007)	0.981(0.004)	**1.000(0.000)**	1.176(0.050)	0.346(0.078)
BAGUS	0.991(0.002)	0.979(0.011)	0.918(0.016)	0.992(0.005)	1.288(0.048)	0.360(0.055)
Pooled	0.959(0.009)	**0.999(0.002)**	0.751(0.034)	0.989(0.001)	2.825(0.054)	0.480(0.019)
GGL	0.925(0.010)	0.929(0.014)	0.600(0.027)	0.959(0.007)	3.267(0.083)	0.455(0.016)

Table 1: Simulation results for AR(2) model.

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
			$n = 50,\ p = 50$			
GemBag ($\alpha = 1$)	**1.000(0.000)**	0.972(0.010)	**0.983(0.005)**	**1.000(0.000)**	1.912(0.087)	0.898(0.210)
GemBag ($\alpha = \sqrt{n}$)	0.999(0.000)	0.987(0.006)	0.979(0.008)	**1.000(0.000)**	**1.877(0.055)**	0.651(0.164)
GemBag ($\alpha = n$)	0.999(0.000)	0.984(0.007)	0.973(0.009)	**1.000(0.000)**	1.888(0.059)	0.695(0.191)
BAGUS	0.990(0.002)	0.960(0.015)	0.840(0.020)	0.980(0.008)	2.264(0.142)	0.938(0.227)
Pooled	0.981(0.004)	**1.000(0.000)**	0.776(0.035)	0.996(0.000)	2.473(0.040)	**0.503(0.024)**
GGL	0.952(0.009)	0.955(0.020)	0.583(0.033)	0.975(0.011)	3.768(0.140)	0.745(0.194)
			$n = 100,\ p = 50$			
GemBag ($\alpha = 1$)	**1.000(0.000)**	0.999(0.002)	**0.998(0.002)**	**1.000(0.000)**	1.258(0.037)	0.559(0.087)
GemBag ($\alpha = \sqrt{n}$)	**1.000(0.000)**	**1.000(0.001)**	0.996(0.003)	**1.000(0.000)**	1.199(0.038)	**0.462(0.097)**
GemBag ($\alpha = n$)	**1.000(0.000)**	0.999(0.002)	**0.998(0.002)**	**1.000(0.000)**	**1.193(0.035)**	0.485(0.078)
BAGUS	0.991(0.001)	0.998(0.002)	0.875(0.015)	0.999(0.001)	1.415(0.039)	0.551(0.081)
Pooled	0.979(0.004)	**1.000(0.000)**	0.759(0.030)	0.996(0.000)	2.448(0.029)	0.489(0.020)
GGL	0.953(0.004)	0.998(0.003)	0.609(0.018)	0.999(0.002)	3.115(0.069)	0.528(0.056)
			$n = 200,\ p = 50$			
GemBag ($\alpha = 1$)	**1.000(0.000)**	**1.000(0.000)**	0.999(0.001)	**1.000(0.000)**	0.865(0.031)	0.346(0.057)
GemBag ($\alpha = \sqrt{n}$)	**1.000(0.000)**	**1.000(0.000)**	0.999(0.002)	**1.000(0.000)**	0.865(0.029)	0.302(0.040)
GemBag ($\alpha = n$)	**1.000(0.000)**	**1.000(0.000)**	**1.000(0.001)**	**1.000(0.000)**	**0.854(0.029)**	**0.301(0.045)**
BAGUS	0.993(0.002)	**1.000(0.000)**	0.892(0.023)	1.000(0.000)	0.973(0.029)	0.341(0.054)
Pooled	0.974(0.005)	**1.000(0.000)**	0.722(0.035)	0.996(0.000)	2.432(0.027)	0.477(0.014)
GGL	0.957(0.005)	**1.000(0.000)**	0.627(0.021)	1.000(0.000)	2.550(0.071)	0.424(0.019)

Table 2: Simulation results for circle model.

Appendix E.4. For BAGUS and Pooled, we follow the same tuning procedure in Gan et al. (2019a) and tune (v_0, v_1) with the same values as in GemBag when $\alpha = 1$. For GGL, we tune the penalty parameters (λ_1, λ_2) as in Danaher et al. (2014) with $\lambda_1 = (0.1, 0.2, \ldots, 1)$ and $\lambda_2 = (0.1, 0.3, 0.5)$.

Tables 1-4 summarize the results of selection accuracy and estimation accuracy for all methods. The metrics we use for selection accuracy are specificity (Spec), sensitivity (Sens),

Matthews correlation coefficient (MCC), and area under the receiver operating characteristic (ROC) curve (AUC); the metrics we use for estimation accuracy are Frobenius norm (F-norm) and element-wise ℓ_∞ norm (ℓ_∞ norm). In Tables 1-4, we report the maximum of ℓ_∞ norm and the average for the other metrics over the K classes and the results are aggregated over 100 replications.

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
	$n = 100,\ p = 100,\ \rho = 0$					
GemBag ($\alpha = 1$)	**0.999(0.000)**	0.850(0.037)	**0.906(0.021)**	0.979(0.011)	**2.853(0.163)**	**0.517(0.083)**
GemBag ($\alpha = \sqrt{n}$)	0.990(0.005)	**0.895(0.049)**	0.794(0.059)	**0.993(0.007)**	3.503(0.415)	0.759(0.298)
GemBag ($\alpha = n$)	0.997(0.002)	0.852(0.046)	0.857(0.043)	0.991(0.008)	3.337(0.371)	0.789(0.257)
BAGUS	0.994(0.002)	0.816(0.039)	0.794(0.033)	0.903(0.022)	3.184(0.190)	0.551(0.093)
Pooled	0.989(0.003)	0.664(0.056)	0.616(0.048)	0.840(0.029)	7.115(0.380)	0.983(0.035)
GGL	0.948(0.008)	0.707(0.074)	0.401(0.044)	0.845(0.038)	6.338(0.382)	0.604(0.037)
	$n = 100,\ p = 100,\ \rho = 0.25$					
GemBag ($\alpha = 1$)	**0.998(0.001)**	0.794(0.044)	**0.857(0.025)**	0.964(0.015)	**2.940(0.142)**	0.495(0.081)
GemBag ($\alpha = \sqrt{n}$)	**0.998(0.001)**	0.798(0.038)	0.853(0.022)	0.959(0.010)	2.960(0.177)	**0.480(0.110)**
GemBag ($\alpha = n$)	0.997(0.001)	0.795(0.030)	0.845(0.026)	**0.964(0.009)**	3.058(0.178)	0.495(0.138)
BAGUS	0.988(0.003)	**0.813(0.030)**	0.732(0.025)	0.917(0.017)	3.372(0.148)	0.591(0.102)
Pooled	0.976(0.004)	0.571(0.045)	0.472(0.029)	0.783(0.024)	6.179(0.256)	0.871(0.104)
GGL	0.966(0.010)	0.769(0.043)	0.552(0.054)	0.879(0.022)	5.274(0.122)	0.529(0.029)
	$n = 100,\ p = 150,\ \rho = 0.25$					
GemBag ($\alpha = 1$)	**0.998(0.001)**	0.793(0.036)	0.838(0.023)	**0.960(0.012)**	**3.672(0.162)**	**0.485(0.076)**
GemBag ($\alpha = \sqrt{n}$)	**0.998(0.000)**	0.789(0.030)	**0.845(0.022)**	0.959(0.007)	3.757(0.260)	0.494(0.100)
GemBag ($\alpha = n$)	**0.998(0.001)**	0.759(0.055)	0.826(0.026)	0.954(0.014)	3.925(0.290)	0.514(0.108)
BAGUS	0.987(0.002)	**0.816(0.028)**	0.663(0.026)	0.915(0.014)	4.453(0.196)	0.622(0.103)
Pooled	0.983(0.003)	0.561(0.038)	0.459(0.024)	0.778(0.020)	7.608(0.268)	0.907(0.089)
GGL	0.976(0.005)	0.719(0.051)	0.513(0.044)	0.855(0.026)	6.782(0.258)	0.551(0.027)
	$n = 100,\ p = 200,\ \rho = 0.25$					
GemBag ($\alpha = 1$)	0.998(0.000)	0.793(0.027)	0.823(0.019)	**0.959(0.011)**	**4.320(0.125)**	**0.483(0.064)**
GemBag ($\alpha = \sqrt{n}$)	**0.999(0.000)**	0.780(0.034)	**0.836(0.016)**	0.958(0.010)	4.516(0.242)	0.495(0.122)
GemBag ($\alpha = n$)	**0.999(0.000)**	0.723(0.024)	0.822(0.016)	0.946(0.009)	4.740(0.163)	0.515(0.132)
BAGUS	0.986(0.002)	**0.819(0.019)**	0.613(0.019)	0.915(0.009)	5.551(0.137)	0.693(0.107)
Pooled	0.986(0.002)	0.558(0.031)	0.442(0.020)	0.777(0.016)	8.801(0.252)	0.924(0.082)
GGL	0.981(0.005)	0.684(0.045)	0.486(0.057)	0.838(0.023)	8.062(0.209)	0.569(0.020)
	$n = 100,\ p = 100,\ \rho = 1$					
GemBag ($\alpha = 1$)	**0.996(0.001)**	0.430(0.036)	**0.591(0.028)**	**0.847(0.016)**	**3.672(0.098)**	0.490(0.105)
GemBag ($\alpha = \sqrt{n}$)	0.993(0.003)	0.444(0.060)	0.564(0.030)	0.793(0.030)	3.793(0.122)	**0.407(0.040)**
GemBag ($\alpha = n$)	0.993(0.002)	0.431(0.054)	0.559(0.032)	0.791(0.026)	3.806(0.116)	0.427(0.062)
BAGUS	0.985(0.001)	**0.484(0.032)**	0.525(0.024)	0.773(0.017)	4.212(0.103)	0.595(0.108)
Pooled	0.977(0.005)	0.300(0.037)	0.314(0.025)	0.697(0.015)	4.810(0.099)	0.551(0.087)
GGL	0.994(0.003)	0.372(0.103)	0.516(0.059)	0.684(0.051)	4.767(0.167)	0.481(0.026)

Table 3: Simulation results for nearest-neighbor network.

As shown in the tables, GemBag performs the best in almost all the designs in terms of both selection accuracy (MCC and AUC) and estimation accuracy (F-norm and ℓ_∞ norm). In the first two designs, signals within the same group have the same magnitude. Therefore, Pooled method has some advantage by aggregating all the samples from different classes and fitting a single model when n is small. As n grows larger, GemBag outperforms all the other competitors. The last two designs are more challenging, since both the sparsity structures and signal strength can be different within a group. We observe that GemBag

still has the best performance under different values of ρ, which controls the discrepancy between the sparsity patterns across classes.

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
	$n = 100,\ p = 100,\ \rho = 0$					
GemBag ($\alpha = 1$)	**1.000(0.000)**	0.999(0.002)	**0.991(0.005)**	**1.000(0.000)**	**1.677(0.095)**	0.515(0.114)
GemBag ($\alpha = \sqrt{n}$)	0.996(0.002)	0.946(0.015)	0.892(0.038)	0.990(0.007)	2.212(0.102)	**0.455(0.096)**
GemBag ($\alpha = n$)	0.994(0.002)	0.943(0.015)	0.854(0.036)	0.990(0.007)	2.263(0.098)	0.458(0.096)
BAGUS	0.997(0.001)	0.995(0.004)	0.936(0.019)	0.998(0.002)	1.747(0.096)	0.492(0.107)
Pooled	0.958(0.003)	0.746(0.043)	0.429(0.027)	0.903(0.018)	7.148(0.300)	0.869(0.024)
GGL	0.938(0.007)	**1.000(0.001)**	0.483(0.022)	1.000(0.001)	5.043(0.282)	0.545(0.019)
	$n = 100,\ p = 100,\ \rho = 0.25$					
GemBag ($\alpha = 1$)	**0.999(0.001)**	0.936(0.019)	**0.939(0.014)**	**0.992(0.004)**	**2.017(0.082)**	0.485(0.091)
GemBag ($\alpha = \sqrt{n}$)	0.998(0.001)	0.879(0.032)	0.892(0.024)	0.970(0.012)	2.413(0.102)	**0.427(0.073)**
GemBag ($\alpha = n$)	0.997(0.001)	0.867(0.029)	0.880(0.022)	0.969(0.012)	2.422(0.080)	0.437(0.071)
BAGUS	0.990(0.001)	0.919(0.021)	0.801(0.019)	0.967(0.009)	2.407(0.100)	0.518(0.088)
Pooled	0.959(0.004)	0.654(0.040)	0.415(0.027)	0.833(0.021)	6.331(0.229)	0.799(0.040)
GGL	0.959(0.006)	**0.964(0.013)**	0.591(0.029)	0.980(0.007)	4.705(0.137)	0.540(0.024)
	$n = 100,\ p = 150,\ \rho = 0.25$					
GemBag ($\alpha = 1$)	**0.999(0.001)**	0.929(0.021)	**0.925(0.017)**	**0.990(0.004)**	**2.522(0.090)**	0.483(0.082)
GemBag ($\alpha = \sqrt{n}$)	**0.999(0.000)**	0.861(0.023)	0.890(0.017)	0.962(0.011)	3.019(0.080)	**0.437(0.059)**
GemBag ($\alpha = n$)	0.998(0.001)	0.846(0.041)	0.864(0.019)	0.960(0.018)	3.070(0.093)	0.462(0.079)
BAGUS	0.991(0.001)	0.911(0.019)	0.757(0.022)	0.963(0.008)	3.121(0.150)	0.542(0.080)
Pooled	0.971(0.003)	0.639(0.031)	0.403(0.022)	0.826(0.015)	7.740(0.240)	0.823(0.040)
GGL	0.977(0.005)	**0.940(0.022)**	0.619(0.032)	0.969(0.011)	6.222(0.325)	0.582(0.025)
	$n = 100,\ p = 200,\ \rho = 0.25$					
GemBag ($\alpha = 1$)	**0.999(0.001)**	**0.932(0.017)**	**0.909(0.023)**	**0.989(0.004)**	**2.986(0.111)**	0.503(0.107)
GemBaG ($\alpha = \sqrt{n}$)	**0.999(0.000)**	0.857(0.026)	0.884(0.015)	0.961(0.012)	3.515(0.098)	**0.464(0.075)**
GemBag ($\alpha = n$)	**0.999(0.000)**	0.815(0.024)	0.875(0.016)	0.949(0.011)	3.602(0.091)	0.501(0.082)
BAGUS	0.991(0.002)	0.910(0.016)	0.713(0.039)	0.962(0.007)	3.890(0.376)	0.598(0.110)
Pooled	0.977(0.002)	0.639(0.026)	0.396(0.018)	0.826(0.014)	8.916(0.209)	0.835(0.027)
GGL	0.985(0.003)	0.931(0.019)	0.636(0.034)	0.964(0.010)	7.467(0.143)	0.610(0.025)
	$n = 100,\ p = 100,\ \rho = 1$					
GemBag ($\alpha = 1$)	**0.996(0.001)**	0.571(0.023)	**0.686(0.017)**	**0.902(0.011)**	**3.210(0.083)**	0.506(0.109)
GemBag ($\alpha = \sqrt{n}$)	0.993(0.003)	0.569(0.046)	0.658(0.019)	0.838(0.023)	3.390(0.083)	**0.422(0.051)**
GemBag ($\alpha = n$)	0.994(0.001)	0.553(0.032)	0.647(0.020)	0.835(0.017)	3.411(0.088)	0.428(0.057)
BAGUS	0.985(0.001)	**0.616(0.022)**	0.606(0.018)	0.845(0.011)	3.785(0.107)	0.613(0.115)
Pooled	0.969(0.003)	0.362(0.031)	0.315(0.026)	0.739(0.016)	5.083(0.110)	0.606(0.072)
GGL	0.986(0.002)	0.609(0.033)	0.616(0.019)	0.801(0.016)	4.616(0.069)	0.487(0.022)

Table 4: Simulation results for scale-free network.

To gain more insights into the performance of each method in recovering the sparsity structure, we plot the ROC curves of all methods for each design in Figure 2. Each point on the ROC curve corresponds to a possible threshold of the posterior probabilities in case of GemBag, BAGUS, and Pooled, and the magnitude of the estimated precision matrix entries for GGL. Therefore, the ROC curve provides an overall measurement on the performance in recovering the sparsity structure at various thresholds, which can be used to assess the performance of the methods irrespective of the choice of the threshold. We observe that the ROC curves of GemBag are above the other curves in almost all the designs. We also plot the estimated sparsity structure from one replication for each design in Figure 3. We note that the estimated graphs from GemBag are closest to the truth while the others tend to include more false edges.

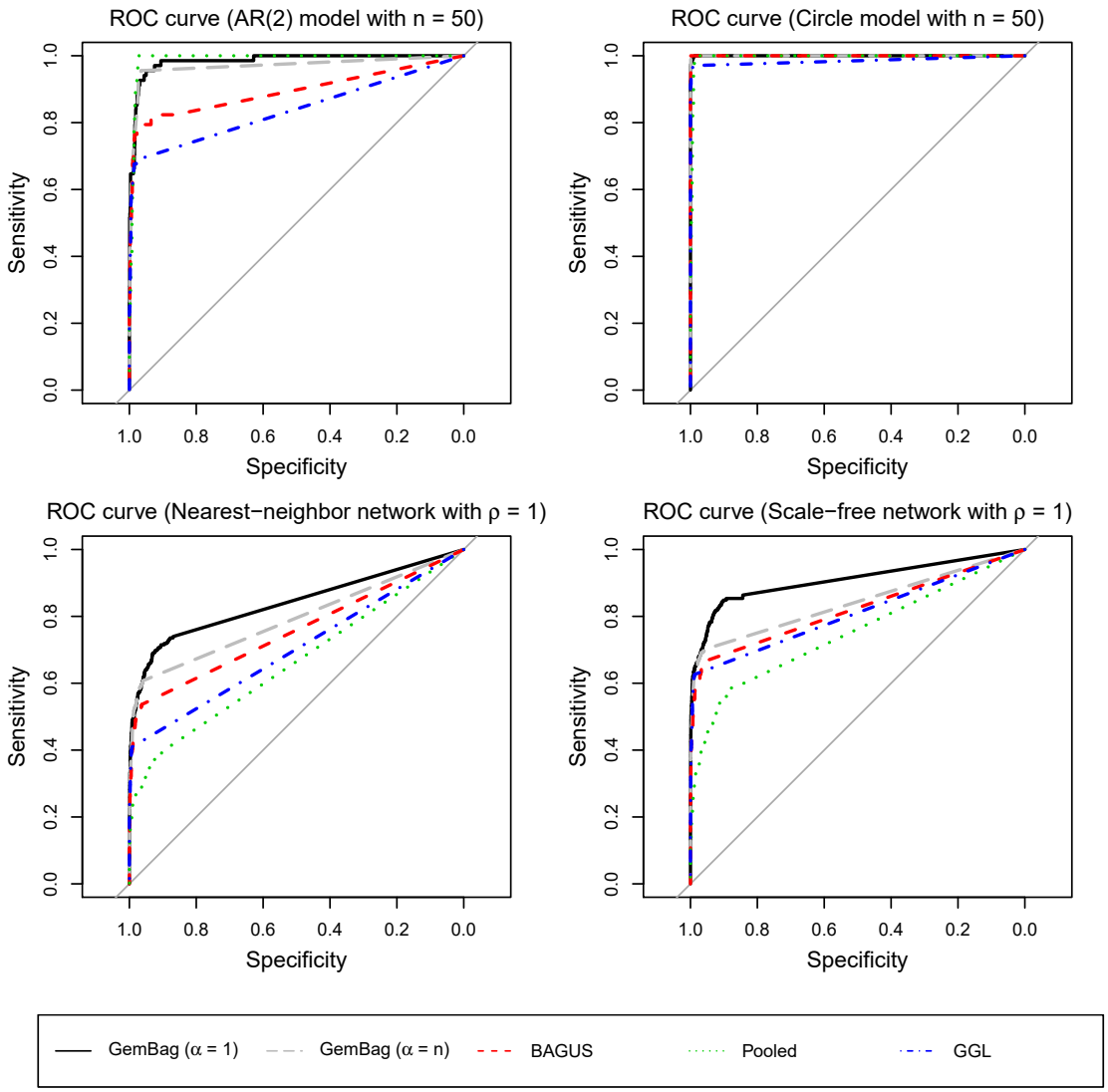


Figure 2: ROC curves under different designs.

We report the computational time of GemBag along with the competing methods using a MacBook Pro with 2.9 GHz Intel Core i5 processor and 8.00 GB memory in Table 5. The computational time of our method is comparable to the competitors except the Pooled method, which restrictively assumes the same precision matrix for all classes and has much worse performance compared to our method. GemBag not only has better performance but is also computationally more efficient even compared to group graphical lasso.

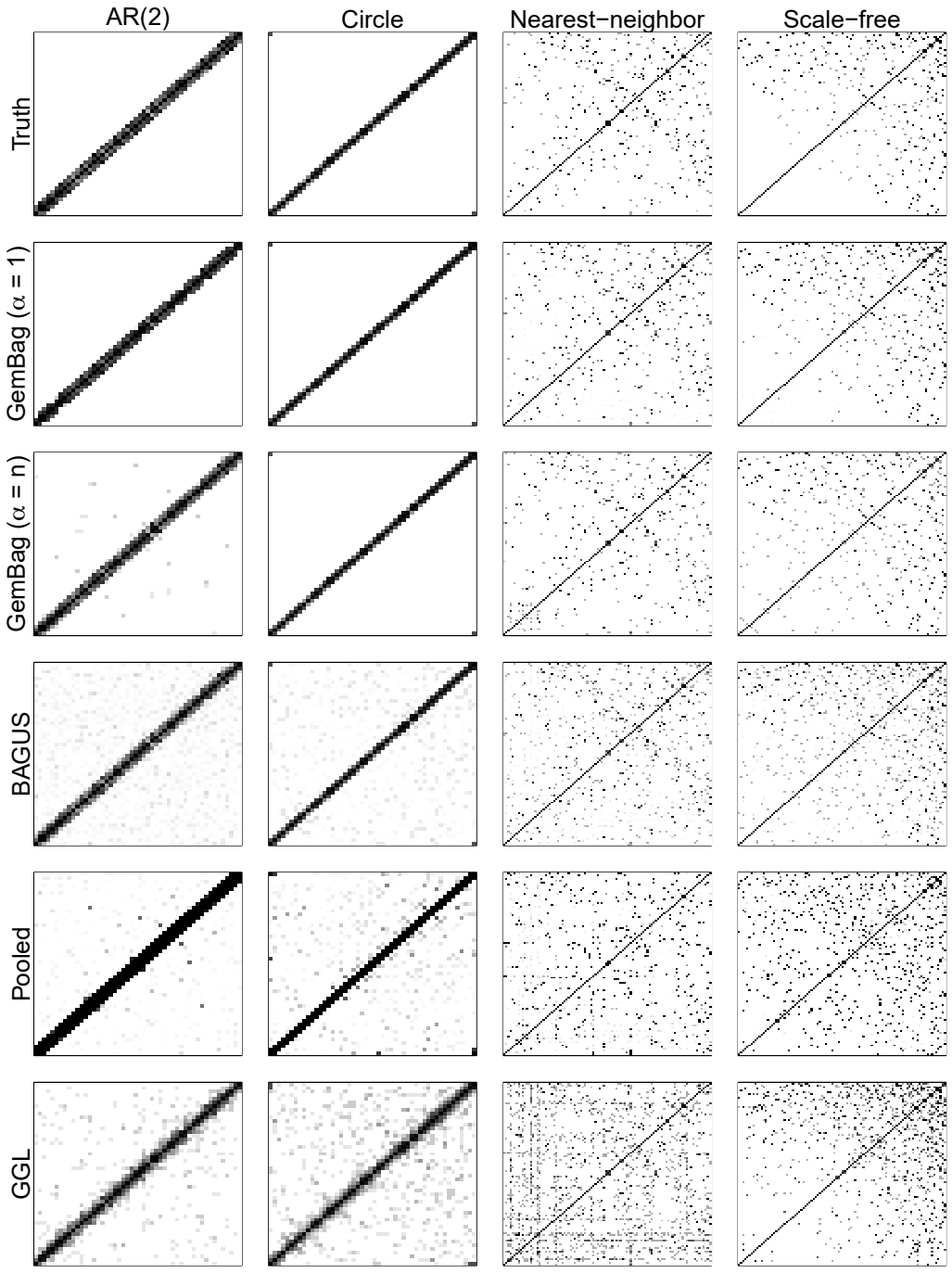


Figure 3: True sparsity pattern and estimated sparsity patterns by all the methods under AR(2) model with $n = 100$, circle model with $n = 100$, nearest-neighbor network with $\rho = 0.25$, and scale-free network with $\rho = 0.25$ (from leftmost column to rightmost column).

	Nearest-neighbor network ($n = 100,\ \rho = 0.25$)			Scale-free network ($n = 100,\ \rho = 0.25$)		
	$p = 100$	$p = 150$	$p = 200$	$p = 100$	$p = 150$	$p = 200$
GemBag ($\alpha = 1$)	3.637(0.122)	12.272(1.018)	27.867(1.054)	3.590(0.078)	11.864(0.226)	28.454(0.215)
GemBag ($\alpha = n$)	5.877(0.871)	17.324(2.680)	42.479(5.734)	3.906(0.185)	16.504(1.455)	36.946(3.248)
BAGUS	3.601(0.046)	12.120(0.443)	28.065(1.222)	3.557(0.047)	12.142(1.060)	28.665(0.401)
Pooled	1.182(0.013)	3.968(0.155)	9.192(0.381)	1.174(0.005)	3.937(0.089)	9.495(0.101)
GGL	10.555(0.768)	27.445(2.717)	60.421(9.453)	8.188(0.912)	26.068(2.727)	55.364(4.471)

Table 5: Average computational time (in seconds) based on 10 replications.

6. Application to Capital Bikeshare Data

We use the Capital Bikeshare trip data[1] to evaluate the performance of GemBag. The data contains records of bike rentals by either a "registered" rider or a "casual" rider in a bicycle sharing system with more than 500 stations. We consider $p = 237$ stations located in Washington, D.C. and use the registered and casual rental data from these stations for every day in 2016, 2017 and 2018. Following the same processing procedure in Zhu and Barber (2015), we removed the seasonal trend and marginally transformed each station's data to a normal distribution, then divided the observations of daily rental counts by year and rider membership into $K = 3 \times 2 = 6$ classes.

For each class, we use the first 80 percent observations as training data and the rest 20 percent as test data. We apply GemBag using $\alpha = 1$, as well as BAGUS and GGL, on the training data to estimate μ_k and $\Theta_k,\ k = 1, \ldots, 6$. To evaluate the prediction performance, we follow the same procedure as described in Fan et al. (2009): we divide each observation from the test data of the k-th class, $Y_{k,i} = (y_{k,i}^{(1)}, \ldots, y_{k,i}^{(237)})$, into two parts, $Y_{k,i_1} = (y_{k,i}^{(1)}, \ldots, y_{k,i}^{(118)})$ and $Y_{k,i_2} = (y_{k,i}^{(119)}, \ldots, y_{k,i}^{(237)})$, and predict the second half Y_{k,i_2} by the best linear predictor based on the first half Y_{k,i_1},

$$\hat{Y}_{k,i_2} = \mathbb{E}(Y_{k,i_2} \mid Y_{k,i_1}) = \hat{\mu}_{k2} + \hat{\Theta}_{k_{21}} \hat{\Theta}_{k_{11}}^{-1} (Y_{k,i_1} - \hat{\mu}_{k1}),\ \text{for } k = 1, \ldots, 6,\ \text{and } i \in \mathcal{T}_k,$$

where $\mathcal{T}_k$ is the index set of the test data of the k-th class, $\mu_k = (\mu_{k1}, \mu_{k2})$, and $\Theta_k = \begin{pmatrix} \Theta_{k_{11}} & \Theta_{k_{12}} \\ \Theta_{k_{21}} & \Theta_{k_{22}} \end{pmatrix}$. Finally we report the average absolute forecast error (AAFE) of each class:

$$\text{AAFE}_k = \frac{1}{119} \sum_{j=119}^{237} \frac{1}{card(\mathcal{T}_k)} \sum_{i \in \mathcal{T}_k} |\hat{y}_{k,i}^{(j)} - y_{k,i}^{(j)}|,\ k = 1, \ldots, 6.$$

In Figure 4, we plot the averaged AAFE versus the number of nonzero off-diagonal entries in the estimated precision matrices. We observe that although GemBag and GGL have roughly the same lowest averaged AAFE, the precision matrices estimated by GemBag are sparser than the ones estimated by GGL. In other wrods, to achieve the same prediction performance, GemBag is based on a sparser dependence structure than GGL, indicating a stronger performance of the GemBag estimator in terms of parsimony. The parsimonious model from GemBag is preferred since it is more interpretable with the same level of prediction accuracy. In the case of Gaussian graphical model, it is known that zero entries

1. Data available at `https://www.capitalbikeshare.com/system-data`.

in the precision matrix have a correspondence to the conditional independence of the two variables given the others. Therefore, with a sparser estimate of precision matrix, we can get a sparser underlying graph to extract useful information.

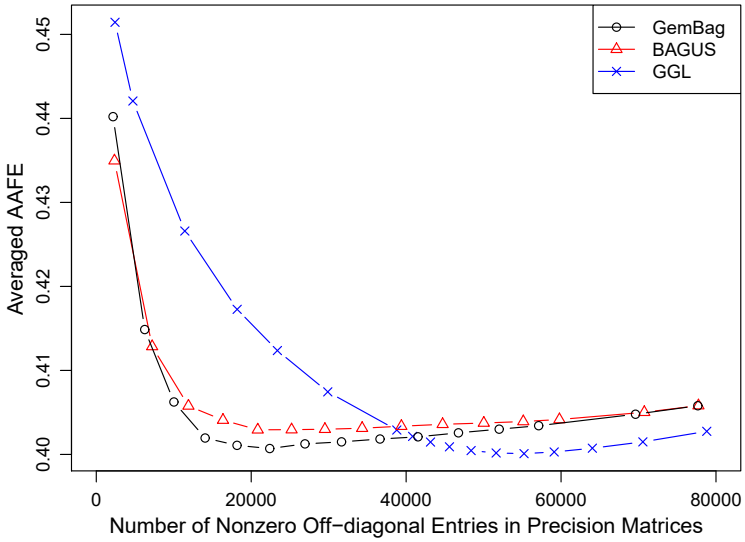


Figure 4: Averaged AAFE versus the total number of nonzero off-diagonal entries in 6 estimated precision matrices. For GemBag and BAGUS, we plot the curves by fixing $v_1 = 0.2$ and varying v_0. For GGL, we fix the ratio between its two tuning parameters and varying them together. Different ratios would output similar curves and only one of them is plotted.

To get estimates for the networks of the stations, we select the tuning parameters for GemBag and BAGUS by BIC and summarize the number of edges in the estimated networks in Table 6. We note that networks estimated by GemBag are sparser compared to those estimated by BAGUS. Besides, GemBag outputs denser registered user networks compared to casual user networks in all years, which is more interpretable as we expect more dependence in registered user networks.

	Registered User			Casual User		
	2016	2017	2018	2016	2017	2018
GemBag	562	643	620	402	488	493
BAGUS	1042	1121	996	1209	1101	1280

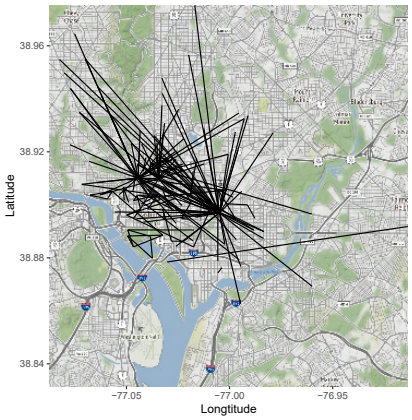
Table 6: Number of edges in estimated networks.

To further understand the estimated registered user networks and casual user networks, we plot the intersection of the networks across three years in Figure 5[2]. The intersection of registered user networks is denser than the intersection of casual user networks, which

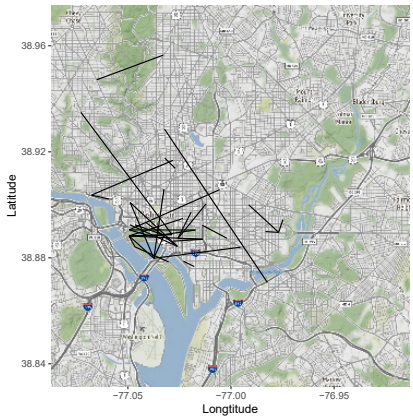
2. Map built using the R package ggmap (Kahle and Wickham, 2013).

implies that registered users have more regular activities. When comparing GemBag and BAGUS, we observe that although networks estimated by BAGUS are denser individually, the networks estimated by GemBag share more common edges. More spatial visualization is provided in Appendix F.

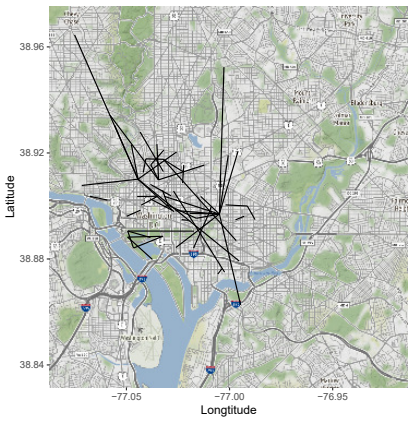
To get a better understanding of the common structure of registered user networks over three years, we provide another plot of the intersection of registered user networks estimated by GemBag and BAGUS in Figure 6. In Figure 6a, we notice three hubs, stations that have higher connectivity. It turns out that the three stations are close to Union Station, Dupont Circle, and Logan Circle, respectively. As Figure 6 suggests, GemBag provides a much more convincing case for the three hubs it detects as compared to BAGUS. Although one may still see the three hubs in BAGUS to have slightly more edges than the other nodes, the degree of difference is not as pronounced as in the case of GemBag. In conclusion, compared to BAGUS, GemBag not only accomplishes better prediction accuracy as shown in Figure 4, but also provides more interpretable insights about the underlying graph structure. This advantage is due to the Bayesian hierarchical modeling of Gembag which utilizes the cross-class similarity information as desired.



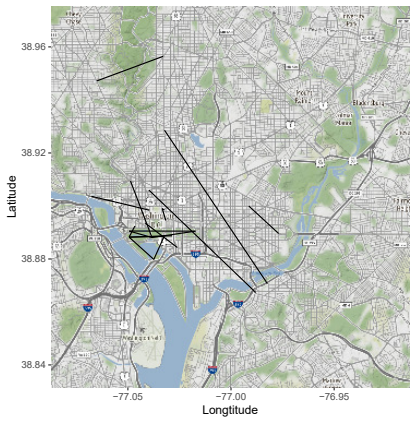
(a) Registered user network by GemBag



(b) Casual user network by GemBag

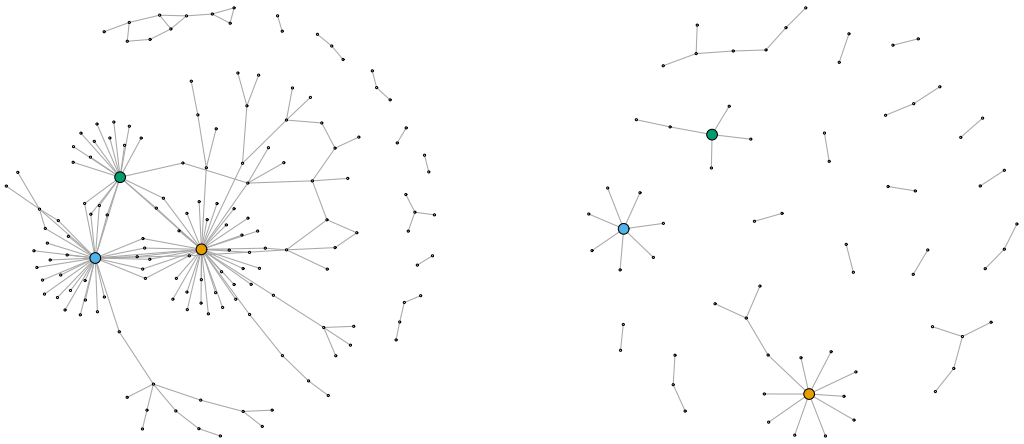


(c) Registered user network by BAGUS



(d) Casual user network by BAGUS

Figure 5: Intersection of the estimated networks across three years.



(a) GemBag (b) BAGUS

Figure 6: Intersection of registered user networks across three years estimated by GemBag and BAGUS.

Acknowledgments

We sincerely thank the editor and three reviewers for their insightful comments that improved the manuscript. Naveen Narisetty gratefully acknowledges partial funding support from NSF grants DMS-1811768 and CAREER-1943500. Feng Liang gratefully acknowledges funding support from NSF-DMS 1916472.

Appendix A. Proofs of the Main Results

In this section, we provide the proofs of the main results, Theorem 1-4. For simplicity, we assume sample sizes of the K classes to be the same: $n_1 = \cdots = n_K = n$.

Recall our objective function is

$$L(\mathbf{\Theta}) = \sum_{k=1}^{K} -l(\Theta_k) + \alpha \sum_{i=1}^{p}\sum_{k=1}^{K} \mathrm{Pen}_{L_1}(\theta_{k,ii}) + \frac{\alpha}{2}\sum_{i\neq j} \mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij}),$$

where

$$l(\Theta_k) = -\frac{n}{2}\left(\mathrm{tr}(S_k\Theta_k) - \log\det(\Theta_k)\right),$$

$$\mathrm{Pen}_{L_1}(\theta_{k,ii}) = \tau\theta_{k,ii},$$

$$\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij}) = -\log\left(p_1\prod_{k=1}^{K}\left(\frac{p_2}{2v_1}e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0}e^{-\frac{|\theta_{k,ij}|}{v_0}}\right) + (1-p_1)\prod_{k=1}^{K}\frac{1}{2v_0}e^{-\frac{|\theta_{k,ij}|}{v_0}}\right) + C,$$

where C is some constant that lets $\mathrm{Pen}_{\mathrm{GB}}(\mathbf{0}) = 0$.

A.1 Proof of Theorem 1

In this subsection, we provide the proof of Theorem 1 when $p_2 < 1$. For the case of $p_2 = 1$, a proof is provided in Gan et al. (2019b). We first introduce some notations that we use in the proof. Let Σ_k^0 denote the k-th true covariance matrix and $W_k = S_k - \Sigma_k^0$ denote the difference between the k-th sample covariance matrix and Σ_k^0. Use $\boldsymbol{\Sigma}^0 = (\Sigma_1^0, \ldots, \Sigma_K^0)$ and $\boldsymbol{W} = (W_1, \ldots, W_K)$ to denote the collection of Σ_k^0's and W_k's, respectively. For any subset $\mathcal{M}_k$ of $\{(i,j) : 1 \leqslant i, j \leqslant p\}$ and a $p \times p$ matrix Θ_k, let $(\Theta_k)_{\mathcal{M}_k}$ denote the submatrix of Θ_k with entries indexed by $\mathcal{M}_k$; for $\boldsymbol{\mathcal{M}} = (\mathcal{M}_1, \ldots, \mathcal{M}_K)$, a collection of $\mathcal{M}_k$'s, and $\boldsymbol{\Theta} = (\Theta_1, \ldots, \Theta_K)$, let $\boldsymbol{\Theta}_{\boldsymbol{\mathcal{M}}} = ((\Theta_1)_{\mathcal{M}_1}, \ldots, (\Theta_K)_{\mathcal{M}_K})$; for a vector of dimension K, let $(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{M}}} = (\theta_{k,ij})_{k:(i,j)\in\mathcal{M}_k}$. We use $\Gamma^0_{\boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{M}}}$ to denote the Hessian matrix $\nabla^2_{\boldsymbol{\Theta}_{\boldsymbol{\mathcal{M}}},\boldsymbol{\Theta}_{\boldsymbol{\mathcal{M}}}}(-\sum_k \log\det\Theta_k)$ evaluated at $\boldsymbol{\Theta}^0$, and $(\Gamma_k^0)_{\mathcal{M}_k\mathcal{M}_k}$ to denote the Hessian matrix $\nabla^2_{(\Theta_k)_{\mathcal{M}_k},(\Theta_k)_{\mathcal{M}_k}}(-\log\det\Theta_k)$ evaluated at Θ_k^0.

Our proof is motivated by the proof techniques in Ravikumar et al. (2011); Gan et al. (2019a). We prove the rate of convergence in ℓ_∞ norm of some local minimizer $\tilde{\boldsymbol{\Theta}}$ in Theorem 1 following three steps:

- Step 1. Construct a solution set $\mathcal{A}$ for the following constrained minimization problem:

$$\min_{\boldsymbol{\Theta}\in\Omega_1} L(\boldsymbol{\Theta}),$$

  where $\Omega_1 := \{\boldsymbol{\Theta} : \boldsymbol{\Theta}_{\boldsymbol{\mathcal{B}}^c} = 0 \text{ and } \Theta_k \in \mathcal{S}_+^p,\ k = 1, \ldots, K\}$ and $\boldsymbol{\mathcal{B}}^c = (\mathcal{B}_1^c, \ldots, \mathcal{B}_K^c)$ with $\mathcal{B}_k = \{(i,j) : |\theta_{k,ij}^0| \geqslant C_5\sqrt{\log p/n} \text{ or } i = j\}$.

- Step 2. Show that there exists some $\tilde{\boldsymbol{\Theta}}' \in \mathcal{A}$ such that $\|\tilde{\boldsymbol{\Theta}}' - \boldsymbol{\Theta}^0\|_\infty < C_5\sqrt{\log p/n}$.

- Step 3. Show that the constructed $\tilde{\boldsymbol{\Theta}}'$ in Step 2 is a local minimizer of $L(\boldsymbol{\Theta})$.

Let $M_{\Gamma^0} = |||(\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})^{-1}|||_\infty = \sup_k |||(\Gamma_k^0)_{\mathcal{B}_k\mathcal{B}_k}{}^{-1}|||_\infty$ and $M_{\boldsymbol{\Sigma}^0} = \sup_k |||\Sigma_k^0|||_\infty$ where $|||\cdot|||_\infty$ denotes the matrix maximum absolute row sum norm. We first describe a more general result of Theorem 5. Theorem 1 directly follows from Theorem 5 by checking its condition (iv) with standard concentration inequalities (cf. Theorem 1 in Cai et al. (2011)).

Theorem 5 *Under the following conditions:*

(i) rates of the hyperparameters $v_1,\ v_0,\ p_1,\ p_2$, and τ:

$$\begin{aligned}
\frac{\alpha}{nv_1} &< C_3/(1+2\epsilon_1)\sqrt{\log p/n} \\
\frac{\alpha}{nv_0} &> C_4\sqrt{\log p/n} \\
\frac{v_1^{K+2}(1-p_1)}{v_0^{K+2}p_1p_2^K} &\leqslant \epsilon_1 p^{(C_2-C_3)M_{\Gamma^0}(C_4-C_3)/\alpha} \\
\epsilon_2 < \frac{v_1(1-p_2)}{v_0p_2} < \frac{v_1^3(1-p_2)}{v_0^3p_2} &\leqslant \epsilon_1 p^{(C_2-C_3)M_{\Gamma^0}(C_4-C_3)/\alpha} \\
\frac{\alpha\tau}{n} &\leqslant \frac{C_3}{2}\sqrt{\log p/n},
\end{aligned}$$

where $C_2 > C_3 > 0$, $C_4 > (1+\epsilon_2)/\epsilon_2[C_1 + 2(C_1+C_2)M_{\Gamma^0}M^2_{\mathbf{\Sigma}^0} + 6(C_1+C_2)^2 d M^2_{\Gamma^0}M^3_{\mathbf{\Sigma}^0}/M]$, *and* $\epsilon_1 > \epsilon_2 > 0$,

(ii) the eigenvalues of the true precision matrices:

$$1/\xi_0 < \lambda_{\min}(\Theta^0_k) < \lambda_{\max}(\Theta^0_k) \leqslant 1/\xi_1 < \sqrt{\frac{\alpha p^{\epsilon_0/\alpha}}{KC_3^2 \log p}} \quad \text{for } k = 1, \ldots, K,$$

where $\epsilon_0 := (C_2 - C_3)M_{\Gamma^0}(C_4 - C_3)$,

(iii) the sample size n: $\sqrt{n} \geqslant M\sqrt{\log p}$ *with*

$$M = 2d(C_1+C_2)M_{\Gamma^0}\max\{2M_{\Gamma^0}[2M_{\mathbf{\Sigma}^0}(M^2_{\mathbf{\Sigma}^0} + \frac{3}{2}M^3_{\mathbf{\Sigma}^0}) + (M^2_{\mathbf{\Sigma}^0} + \frac{3}{2}M^3_{\mathbf{\Sigma}^0})^2], 3M_{\mathbf{\Sigma}^0}, 1, \xi_0\},$$

(iv) difference between the sample covariance matrices and the true covariance matrices: $\|\boldsymbol{W}\|_\infty \leqslant C_1\sqrt{\log p/n}$,

there exists a local solution $\tilde{\mathbf{\Theta}}$ *of* (11) *such that*

$$\|\tilde{\mathbf{\Theta}} - \mathbf{\Theta}^0\|_\infty < 2(C_1+C_2)M_{\Gamma^0}\sqrt{\log p/n},$$

where C_2 *is chosen such that* $2(C_1+C_2)M_{\Gamma^0} = C_5$.

Proof of Theorem 5

- Step 1. Construct a solution set for minimizing the objective function (11) as follows:

$$\mathcal{A} = \{\mathbf{\Theta} : G(\mathbf{\Theta})_{\boldsymbol{\mathcal{B}}} = \mathbf{0}, \mathbf{\Theta}_{\boldsymbol{\mathcal{B}}^c} = \mathbf{0}, \text{ and } \Theta_k \in \mathcal{S}^p_+,\ k = 1, \ldots, K\},$$

where $\boldsymbol{\mathcal{B}} = (\mathcal{B}_1, \ldots, \mathcal{B}_K)$ with $\mathcal{B}_k = \{(i,j) : |\theta^0_{k,ij}| \geqslant r_e \text{ or } i = j\}$ for $r_e = 2(C_1 + C_2)M_{\Gamma^0}\sqrt{\log p/n}$ and

$$G(\mathbf{\Theta}) = \frac{n}{2}(\boldsymbol{S} - \mathbf{\Theta}^{-1} + \frac{2\alpha}{n}\boldsymbol{Z}(\mathbf{\Theta})) \tag{16}$$

is the subgradient of the objective function $L(\mathbf{\Theta})$ with $\mathbf{\Theta}^{-1} = (\Theta_1^{-1}, \ldots, \Theta_K^{-1})$ and

$$Z_{k,ij}(\mathbf{\Theta}) = Z_{k,ij}(\boldsymbol{\theta}_{ij}) = \begin{cases} \tau & \text{if } i = j, \\ \frac{1}{2}\frac{\partial}{\partial|\theta_{k,ij}|}\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})\mathrm{sign}(\theta_{k,ij}) & \text{if } i \neq j, \end{cases}$$

where

$$\frac{\partial}{\partial|\theta_{k,ij}|}\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij}) = \omega_{k,ij}(\boldsymbol{\theta}_{ij})\frac{1}{v_1} + (1 - \omega_{k,ij}(\boldsymbol{\theta}_{ij}))\frac{1}{v_0}$$

and

$$\mathrm{sign}(\theta_{k,ij}) = \begin{cases} 1 & \text{when } \theta_{k,ij} > 0, \\ -1 & \text{when } \theta_{k,ij} < 0, \\ [-1,1] & \text{when } \theta_{k,ij} = 0. \end{cases}$$

- Step 2. Show that there exists some $\tilde{\boldsymbol{\Theta}}' \in \mathcal{A}$ such that $\|\tilde{\boldsymbol{\Theta}}' - \boldsymbol{\Theta}^0\|_\infty < r_e$.

  We only need to consider the entries of $\boldsymbol{\Theta}$ indexed by $\boldsymbol{\mathcal{B}}$, because $\|(\tilde{\boldsymbol{\Theta}}' - \boldsymbol{\Theta}^0)_{\boldsymbol{\mathcal{B}}^c}\|_\infty < r_e$ due to the way that $\mathcal{A}$ is constructed.

  Define the following mapping from $\mathbb{R}^{|\boldsymbol{\mathcal{B}}|}$ to $\mathbb{R}^{|\boldsymbol{\mathcal{B}}|}$ with $|\boldsymbol{\mathcal{B}}| = \sum_k card(\mathcal{B}_k)$:

$$F(\mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}})) = -\frac{2}{n}(\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})^{-1}\mathrm{vec}\left(G(\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})_{\boldsymbol{\mathcal{B}}}\right) + \mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}}),$$

  where $G(\boldsymbol{\Theta})$ is the subgradient of the objective function $L(\boldsymbol{\Theta})$ defined in (16), and $\boldsymbol{\Delta}$ satisfies $(\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})_{\boldsymbol{\mathcal{B}}^c} = \mathbf{0}$, that is, $\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}^c} = -\boldsymbol{\Theta}^0_{\boldsymbol{\mathcal{B}}^c}$. Note that the factor $2/n$ is introduced to cancel the factor $n/2$ from G.

  Let $\mathbb{B}(r)$ denote the ℓ_∞ ball in $\mathbb{R}^{|\boldsymbol{\mathcal{B}}|}$ where $r = 2(C_1 + C_3)M_{\Gamma^0}\sqrt{\log p/n} < r_e$. For any $\mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}}) \in \mathbb{B}(r)$, we have

$$\begin{aligned}
F(\mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}})) = & -(\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})^{-1}\mathrm{vec}\Big(\big(\boldsymbol{S} - (\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})^{-1} + \frac{2\alpha}{n}\boldsymbol{Z}(\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})\big)_{\boldsymbol{\mathcal{B}}}\Big) + \mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}}) \\
= & -(\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})^{-1}\Big(\mathrm{vec}\big(\boldsymbol{\Sigma}^0 - (\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})^{-1}\big)_{\boldsymbol{\mathcal{B}}}\big) - \Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}}\mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}})\Big) \qquad (17) \\
& -(\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})^{-1}\mathrm{vec}\Big(\boldsymbol{W}_{\boldsymbol{\mathcal{B}}} + \frac{2\alpha}{n}\boldsymbol{Z}(\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})_{\boldsymbol{\mathcal{B}}}\Big) \qquad (18)
\end{aligned}$$

  where $(\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})^{-1} = ((\Theta^0_1 + \Delta_1)^{-1}, \ldots, (\Theta^0_K + \Delta_K)^{-1})$ and $\boldsymbol{W} = \boldsymbol{S} - \boldsymbol{\Sigma}^0$. Next we bound $F(\mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}}))$.

  For (17), we have

$$\begin{aligned}
& \left\|-(\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})^{-1}\left(\mathrm{vec}\left(\left(\boldsymbol{\Sigma}^0 - (\boldsymbol{\Theta}^0 + \boldsymbol{\Delta})^{-1}\right)_{\boldsymbol{\mathcal{B}}}\right) - (\Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}})\mathrm{vec}(\boldsymbol{\Delta}_{\boldsymbol{\mathcal{B}}})\right)\right\|_\infty \\
\leqslant & \sup_k \left\|-(\Gamma^0_k)_{\mathcal{B}_k\mathcal{B}_k}{}^{-1}\left(\mathrm{vec}\left(\left(\Sigma^0_k - (\Theta^0_k + \Delta_k)^{-1}\right)_{\mathcal{B}_k}\right) - (\Gamma^0_k)_{\mathcal{B}_k\mathcal{B}_k}\mathrm{vec}((\Delta_k)_{\mathcal{B}_k})\right)\right\|_\infty \\
\leqslant & \sup_k |||(\Gamma^0_k)_{\mathcal{B}_k\mathcal{B}_k}{}^{-1}|||_\infty \left\|\mathrm{vec}\left(\left(\Sigma^0_k - (\Theta^0_k + \Delta_k)^{-1}\right)_{\mathcal{B}_k}\right) - (\Gamma^0_k)_{\mathcal{B}_k\mathcal{B}_k}\mathrm{vec}((\Delta_k)_{\mathcal{B}_k})\right\|_\infty \\
\leqslant & \sup_k |||(\Gamma^0_k)_{\mathcal{B}_k\mathcal{B}_k}{}^{-1}|||_\infty \Big(\left\|\mathrm{vec}\left(\left(\Sigma^0_k - (\Theta^0_k + \Delta_k)^{-1}\right)_{\mathcal{B}_k}\right) - \left(\Gamma^0_k\mathrm{vec}(\Delta_k)\right)_{\mathcal{B}_k}\right\|_\infty \\
& \qquad + \left\|\left(\Gamma^0_k\mathrm{vec}(\Delta_k)\right)_{\mathcal{B}_k} - (\Gamma^0_k)_{\mathcal{B}_k\mathcal{B}_k}\mathrm{vec}((\Delta_k)_{\mathcal{B}_k})\right\|_\infty \Big) \\
\leqslant & M_{\Gamma^0}\sup_k \Bigg(\left|\left|\left|\int_0^1 \left\{(\Theta^0_k + t\Delta_k)^{-1} \otimes (\Theta^0_k + t\Delta_k)^{-1} - \Gamma^0_k\right\} dt\right|\right|\right|_\infty \|\mathrm{vec}((\Delta_k)_{\mathcal{B}_k})\|_\infty \\
& \qquad + M^2_{\boldsymbol{\Sigma}^0}\left\|(\Delta_k)_{\mathcal{B}^c_k}\right\|_\infty \Bigg) \\
\leqslant & M_{\Gamma^0} r \sup_k \int_0^1 \left|\left|\left|(\Theta^0_k + t\Delta_k)^{-1} \otimes (\Theta^0_k + t\Delta_k)^{-1} - (\Theta^0_k)^{-1} \otimes (\Theta^0_k)^{-1}\right|\right|\right|_\infty dt \\
\leqslant & M_{\Gamma^0} r \sup_k \max_{0\leqslant t\leqslant 1} \left|\left|\left|(\Theta^0_k + t\Delta_k)^{-1} \otimes (\Theta^0_k + t\Delta_k)^{-1} - (\Theta^0_k)^{-1} \otimes (\Theta^0_k)^{-1}\right|\right|\right|_\infty \\
\leqslant & M_{\Gamma^0} r \sup_k \max_{0\leqslant t\leqslant 1} \Big(2\left|\left|\left|(\Theta^0_k)^{-1}\right|\right|\right|_\infty \left|\left|\left|(\Theta^0_k + t\Delta_k)^{-1} - (\Theta^0_k)^{-1}\right|\right|\right|_\infty \\
& \qquad + \left|\left|\left|(\Theta^0_k + t\Delta_k)^{-1} - (\Theta^0_k)^{-1}\right|\right|\right|^2_\infty \Big)
\end{aligned}$$

$$\begin{aligned}
&\leqslant M_{\Gamma^0} r[2M_{\mathbf{\Sigma}^0}(M^2_{\mathbf{\Sigma}^0} + \frac{3}{2}M^3_{\mathbf{\Sigma}^0})dr_e + (M^2_{\mathbf{\Sigma}^0} + \frac{3}{2}M^3_{\mathbf{\Sigma}^0})^2 d^2 r_e^2] \\
&\leqslant \frac{r}{2}, \qquad (19)
\end{aligned}$$

where the second inequality is because $\|AB\|_\infty \leqslant |||A|||_\infty \|B\|_\infty$, the fourth inequality is because $\Gamma^0_k = (\Theta^0_k)^{-1} \otimes (\Theta^0_k)^{-1}$ and entries of Δ_k indexed by $\mathcal{B}^c_k$ have rates lower than $\sqrt{\log p/n}$, the seventh inequality is because $|||A\otimes A - B\otimes B|||_\infty \leqslant 2|||B|||_\infty |||A-B|||_\infty + |||A-B|||^2_\infty$ from Lemma 13 of Loh and Wainwright (2017), the eighth inequality is because

$$\begin{aligned}
|||(\Theta^0_k + t\Delta_k)^{-1} - (\Theta^0_k)^{-1}|||_\infty &\leqslant M^2_{\Sigma^0_k} |||t\Delta_k|||_\infty + \frac{3}{2}M^3_{\Sigma^0_k}|||t\Delta_k|||^2_\infty \\
&\leqslant M^2_{\Sigma^0_k} dtr_e + \frac{3}{2}M^3_{\Sigma^0_k} d^2t^2 r_e^2 \\
&\leqslant (M^2_{\Sigma^0_k} t + \frac{3}{2}M^3_{\Sigma^0_k} t^2) dr_e
\end{aligned}$$

and $r_e < \frac{1}{d}$ from condition (iii), and last inequality is by $M_{\Gamma^0}[2M_{\mathbf{\Sigma}^0}(M^2_{\mathbf{\Sigma}^0} + \frac{3}{2}M^3_{\mathbf{\Sigma}^0}) + (M^2_{\mathbf{\Sigma}^0} + \frac{3}{2}M^3_{\mathbf{\Sigma}^0})^2]dr_e < \frac{1}{2}$ and $r_e < \frac{1}{d}$ from condition (iii).

For (18), we have

$$\begin{aligned}
\left\| \Gamma^0_{\boldsymbol{\mathcal{B}}\boldsymbol{\mathcal{B}}} \mathrm{vec}\left(\boldsymbol{W}_{\boldsymbol{\mathcal{B}}} + \frac{2\alpha}{n} \boldsymbol{Z}(\mathbf{\Theta}^0 + \mathbf{\Delta})_{\boldsymbol{\mathcal{B}}} \right) \right\|_\infty &\leqslant \sup_k \left\| \Gamma^0_{\mathcal{B}_k \mathcal{B}_k} \mathrm{vec}\left((W_k)_{\mathcal{B}_k} + \frac{2\alpha}{n} Z_k(\mathbf{\Theta}^0 + \mathbf{\Delta})_{\mathcal{B}_k} \right) \right\|_\infty \\
&\leqslant \sup_k M_{\Gamma^0} \left(\|W_k\|_\infty + \left\| \frac{2\alpha}{n} Z_k(\mathbf{\Theta}^0 + \mathbf{\Delta})_{\mathcal{B}_k} \right\|_\infty \right) \\
&\leqslant M_{\Gamma^0} \left(C_1\sqrt{\log p/n} + C_3\sqrt{\log p/n} \right) = \frac{r}{2} \qquad (20)
\end{aligned}$$

where the third inequality is because of $\|W_k\|_\infty \leqslant \|\boldsymbol{W}\|_\infty \leqslant C_1\sqrt{\log p/n}$ from condition (iv) and the upper bound on the magnitude of the first derivatives from Lemma 6 since $\|(\mathbf{\Theta}^0 + \mathbf{\Delta})_{\mathcal{B}_k}\|_\infty \geqslant r_e - r = 2(C_2 - C_3)M_{\Gamma^0}\sqrt{\log p/n}$.

Thus, combining (17), (18), (19), and (20), we have $\|F(\mathrm{vec}(\mathbf{\Delta}_{\boldsymbol{\mathcal{B}}}))\|_\infty \leqslant r$, that is, $F(\mathrm{vec}(\mathbf{\Delta}_{\boldsymbol{\mathcal{B}}})) \subseteq \mathbb{B}(r)$ for any $\mathrm{vec}(\mathbf{\Delta}_{\boldsymbol{\mathcal{B}}}) \in \mathbb{B}(r)$. Therefore, we have $F(\mathbb{B}(r)) \subseteq \mathbb{B}(r)$. By Brouwer's fixed point theorem (Ortega and Rheinboldt, 1970), there exists a fixed point $\mathrm{vec}(\tilde{\mathbf{\Delta}}'_{\boldsymbol{\mathcal{B}}}) \in \mathbb{B}(r)$ such that $F(\mathrm{vec}(\tilde{\mathbf{\Delta}}'_{\boldsymbol{\mathcal{B}}})) = \mathrm{vec}(\tilde{\mathbf{\Delta}}'_{\boldsymbol{\mathcal{B}}})$, which is equivalent to $G(\mathbf{\Theta}^0 + \tilde{\mathbf{\Delta}}')_{\boldsymbol{\mathcal{B}}} = \mathbf{0}$. Let $\tilde{\mathbf{\Theta}}' = \mathbf{\Theta}^0 + \tilde{\mathbf{\Delta}}'$ and we have

$$\begin{cases} \tilde{\mathbf{\Theta}}'_{\boldsymbol{\mathcal{B}}} = \mathbf{\Theta}^0_{\boldsymbol{\mathcal{B}}} + \tilde{\mathbf{\Delta}}'_{\boldsymbol{\mathcal{B}}} \\ \tilde{\mathbf{\Theta}}'_{\boldsymbol{\mathcal{B}}^{\mathbf{c}}} = \mathbf{\Theta}^0_{\boldsymbol{\mathcal{B}}^{\mathbf{c}}} + \tilde{\mathbf{\Delta}}'_{\boldsymbol{\mathcal{B}}^{\mathbf{c}}} = \mathbf{0} \end{cases}$$

Then, this estimate $\tilde{\mathbf{\Theta}}'$ satisfies $G(\tilde{\mathbf{\Theta}}')_{\boldsymbol{\mathcal{B}}} = \mathbf{0}$, $\tilde{\mathbf{\Theta}}'_{\boldsymbol{\mathcal{B}}^{\mathbf{c}}} = \mathbf{0}$, and $\|(\tilde{\mathbf{\Theta}}' - \mathbf{\Theta}^0)_{\boldsymbol{\mathcal{B}}}\|_\infty \leqslant r < r_e$. As long as $\tilde{\mathbf{\Theta}}'$ is in Ω, that is, $\tilde{\Theta}'_k \succ 0$, we establish the statement for Step 2.

Indeed, for any k, and by conditions (ii), we have

$$\lambda_{\min}(\tilde{\Theta}'_k) \geqslant \lambda_{\min}(\Theta^0_k) - \|\tilde{\Delta}'_k\|_2 \geqslant \lambda_{\min}(\Theta^0_k) - dr_e > 0.$$

Therefore, we have $\tilde{\mathbf{\Theta}}' \in \mathcal{A}$ and $\|\tilde{\mathbf{\Theta}}' - \mathbf{\Theta}^0\|_\infty < r_e$.

- Step 3. Show that $\tilde{\mathbf{\Theta}}'$ constructed in Step 2 is a local minimizer of $L(\mathbf{\Theta})$.

  It suffices to show that there exists some $\epsilon > 0$ such that $H(\mathbf{\Delta}) = L(\tilde{\mathbf{\Theta}}' + \mathbf{\Delta}) - L(\tilde{\mathbf{\Theta}}') \geqslant 0$ for any $\mathbf{\Delta}$ with $\|\mathbf{\Delta}\|_\infty < \epsilon$. We have

$$H(\mathbf{\Delta}) = \sum_{k=1}^{K} \frac{n}{2} \left\{ \mathrm{tr}(\Delta_k (S_k - \tilde{\Theta}_k'^{-1})) + \mathrm{tr}(\Delta_k \tilde{\Theta}_k'^{-1}) - \left[\log\det(\tilde{\Theta}_k' + \Delta_k) - \log\det(\tilde{\Theta}_k') \right] \right\} + \alpha \sum_{k=1}^{K} \sum_{i=1}^{p} \tau \Delta_{k,ii} + \frac{\alpha}{2} \sum_{i \neq j} \left[\mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}) - \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}') \right].$$

  Under the condition $\sqrt{n} \geqslant 2d(C_1 + C_2) M_{\Gamma^0} \xi_0 \sqrt{\log p}$ and with the same proof for Theorem 1 in Rothman et al. (2008), we have the following upper bound on $\log\det(\tilde{\Theta}_k' + \Delta_k) - \log\det(\tilde{\Theta}_k')$:

$$\log\det(\tilde{\Theta}_k' + \Delta_k) - \log\det(\tilde{\Theta}_k') \leqslant \mathrm{tr}(\Delta_k \tilde{\Theta}_k'^{-1}) - \frac{1}{4} \xi_1^2 \|\Delta_k\|_F^2, \quad k = 1, \ldots, K. \tag{21}$$

  For any (i, j) pair with $i \neq j$, we have

$$\begin{aligned} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}) - \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}') = & \left(\mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}) - \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}') \right) \\ & + \left(\mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}') - \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}') \right), \end{aligned} \tag{22}$$

  where $\mathbf{\Delta}_{ij}'$ is defined as follows:

$$\begin{cases} (\mathbf{\Delta}_{ij}')_{\boldsymbol{\mathcal{B}}} = (\mathbf{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}} \\ (\mathbf{\Delta}_{ij}')_{\boldsymbol{\mathcal{B}}^c} = \mathbf{0} \end{cases}$$

  For the first term in (22), we have

$$\mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}) - \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}') = \nabla_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}^c}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}' + u_1(\mathbf{\Delta}_{ij} - \mathbf{\Delta}_{ij}'))(\mathbf{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}^c} \tag{23}$$

  for some u_1 in $(0, 1)$.

  For the second term in (22), we have

$$\begin{aligned} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + \mathbf{\Delta}_{ij}') - \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}') &= \nabla_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}')(\mathbf{\Delta}_{ij}')_{\boldsymbol{\mathcal{B}}} \\ &\quad + \frac{1}{2} (\mathbf{\Delta}_{ij}')_{\boldsymbol{\mathcal{B}}}^T \nabla^2_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}, (\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + u_2 \mathbf{\Delta}_{ij}')(\mathbf{\Delta}_{ij}')_{\boldsymbol{\mathcal{B}}} \\ &= \nabla_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}')(\mathbf{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}} \\ &\quad + \frac{1}{2} (\mathbf{\Delta}_{ij}')_{\boldsymbol{\mathcal{B}}}^T \nabla^2_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}, (\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}_{ij}' + u_2 \mathbf{\Delta}_{ij}')(\mathbf{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}} \end{aligned} \tag{24}$$

  for some u_2 in $(0, 1)$.

Combining (21), (23), and (24), we have the following lower bound for $H(\boldsymbol{\Delta})$:

$$\begin{aligned}
H(\boldsymbol{\Delta}) \geqslant & \sum_{k=1}^{K} \frac{n}{2} \left(\sum_{i,j} \Delta_{k,ij} (s_{k,ij} - \tilde{\Theta}'^{-1}_{k,ij}) + \frac{1}{4} \xi_1^2 \|\Delta_k\|_F^2 \right) + \alpha \sum_{k=1}^{K} \sum_{i=1}^{p} \tau \Delta_{k,ii} \\
& + \frac{\alpha}{2} \sum_{i \neq j} \left(\sum_{k:(i,j) \in \mathcal{B}_k^c} \frac{\partial}{\partial |\theta_{k,ij}|} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij} + \boldsymbol{\Delta}'_{ij} + u_1(\boldsymbol{\Delta}_{ij} - \boldsymbol{\Delta}'_{ij})) \Delta_{k,ij} \right) \\
& + \frac{\alpha}{2} \sum_{i \neq j} \Bigg[\sum_{k:(i,j) \in \mathcal{B}_k} \frac{\partial}{\partial \theta_{k,ij}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij}) \Delta_{k,ij} \\
& \qquad + \frac{1}{2} (\boldsymbol{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}}^T \nabla^2_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}, (\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij} + u_2 \boldsymbol{\Delta}'_{ij}) (\boldsymbol{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}} \Bigg] \\
= & \text{ (I) + (II) + (III)}
\end{aligned}$$

where

$$\text{(I)} = \frac{n}{2} \sum_{k=1}^{K} \sum_{(i,j) \in \mathcal{B}_k} \Delta_{k,ij} \left(s_{k,ij} - \tilde{\Theta}'^{-1}_{k,ij} + \frac{2\alpha}{n} Z_{k,ij}(\tilde{\boldsymbol{\theta}}'_{ij}) \right),$$

$$\text{(II)} = \sum_{k=1}^{K} \frac{n}{8} \xi_1^2 \|\Delta_k\|_F^2 + \sum_{i \neq j} \frac{\alpha}{4} (\boldsymbol{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}}^T \nabla^2_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}, (\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij} + u_2 \boldsymbol{\Delta}'_{ij}) (\boldsymbol{\Delta}_{ij})_{\boldsymbol{\mathcal{B}}},$$

$$\begin{aligned}
\text{(III)} = \frac{n}{2} \sum_{i \neq j} \Bigg\{ & \sum_{k:(i,j) \in \mathcal{B}_k^c} \Bigg[\Delta_{k,ij} (s_{k,ij} - \tilde{\Theta}'^{-1}_{k,ij}) \\
& + |\Delta_{k,ij}| \frac{\alpha}{n} \frac{\partial}{\partial |\theta_{k,ij}|} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij} + \boldsymbol{\Delta}'_{ij} + u_1(\boldsymbol{\Delta}_{ij} - \boldsymbol{\Delta}'_{ij})) \Bigg] \Bigg\}.
\end{aligned}$$

Due to the construction of $\tilde{\boldsymbol{\Theta}}'$, we have (I) = 0.

For (II), by condition (ii) and the upper bound on the magnitude of the second derivatives of $\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})$ in Lemma 6, we have

$$\begin{aligned}
\text{(II)} \geqslant & \sum_{i,j} \sum_{k=1}^{K} \frac{n}{8} \xi_1^2 \Delta_{k,ij}^2 - \sum_{(i,j):i \neq j} \frac{\alpha \|\nabla^2_{(\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}, (\boldsymbol{\theta}_{ij})_{\boldsymbol{\mathcal{B}}}} \mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij} + u_2 \boldsymbol{\Delta}'_{ij})\|_\infty}{4} \left(\sum_{k:(i,j) \in \mathcal{B}_k} \Delta_{k,ij} \right)^2 \\
\geqslant & \sum_{i,j} \sum_{k=1}^{K} \frac{n}{8} \xi_1^2 \Delta_{k,ij}^2 - \sum_{(i,j):i \neq j} \frac{\frac{nC_3^2 \log p}{2\alpha p^{\epsilon_0/\alpha}}}{4} \left(\sum_{k:(i,j) \in \mathcal{B}_k} \Delta_{k,ij} \right)^2 \\
\geqslant & \sum_{(i,j):i \neq j} \left(\frac{n}{8K} \xi_1^2 - \frac{nC_3^2 \log p}{8\alpha p^{\epsilon_0/\alpha}} \right) \left(\sum_{k:(i,j) \in \mathcal{B}_k} \Delta_{k,ij} \right)^2 \geqslant 0.
\end{aligned}$$

In (III), we have the following upper bound on the magnitude of $s_{k,ij} - \tilde{\Theta}'^{-1}_{k,ij}$:

$$\left| s_{k,ij} - \tilde{\Theta}^{-1}_{k,ij} \right| \leqslant |s_{k,ij} - \sigma^0_{k,ij}| + |\tilde{\Theta}'^{-1}_{k,ij} - \sigma^0_{k,ij}|$$

$$\leqslant C_1\sqrt{\frac{\log p}{n}} + M^2_{\Sigma^0_k} r_e + \frac{3}{2} d_k M^3_{\Sigma^0_k} r_e^2,$$

where $r_e = 2(C_1 + C_2) M_{\Gamma^0}\sqrt{\log p/n}$, defined in the first line of the proof of Theorem 5, and $|\tilde{\Theta}'^{-1}_{k,ij} - \sigma^0_{k,ij}| < M^2_{\Sigma^0_k} r_e + \frac{3}{2} d_k M^3_{\Sigma^0_k} r_e^2$ can be shown with the same proof for Corollary 4 in Ravikumar et al. (2008).

For the rest of the first derivative inside the inner summation in (III), we have

$$\begin{aligned}
\lim_{\boldsymbol{\Delta}_{ij}\to\mathbf{0}} \frac{\alpha}{n}\frac{\partial}{\partial|\theta_{k,ij}|}\mathrm{Pen}_{\mathrm{GB}}(\tilde{\boldsymbol{\theta}}'_{ij} + \boldsymbol{\Delta}'_{ij} + u_1(\boldsymbol{\Delta}_{ij} - \boldsymbol{\Delta}'_{ij})) &= \omega_{k,ij}(\tilde{\boldsymbol{\theta}}'_{ij})\frac{\alpha}{nv_1} + (1 - \omega_{k,ij}(\tilde{\boldsymbol{\theta}}'_{ij}))\frac{\alpha}{nv_0} \\
&> (1 - \omega_{k,ij}(\tilde{\boldsymbol{\theta}}'_{ij}))\frac{\alpha}{nv_0} \\
&= (1 - \eta_1(\tilde{\boldsymbol{\theta}}'_{ij})\eta_2(0))\frac{\alpha}{nv_0} \qquad (25) \\
&> (1 - \eta_2(0))\frac{\alpha}{nv_0} \\
&> \frac{\epsilon_2}{1+\epsilon_2}\frac{\alpha}{nv_0} \qquad (26) \\
&> C_1\sqrt{\frac{\log p}{n}} + M^2_{\Sigma^0_k} r_e + \frac{3}{2} d_k M^3_{\Sigma^0_k} r_e^2 \\
&\geqslant \left| s_{k,ij} - \tilde{\Theta}'^{-1}_{k,ij} \right|
\end{aligned}$$

where (25) is because $\tilde{\theta}'_{k,ij} = 0$ for $(i,j) \in \mathcal{B}^c_k$ and (26) is due to $\epsilon_2 < v_1(1-p_2)/(v_0 p_2)$. Therefore, there exists some small enough $\epsilon > 0$ such that (III) $\geqslant 0$.

Hence, there exists some $\epsilon > 0$ such that $H(\boldsymbol{\Delta}) \geqslant$ (I) + (II) + (III) $\geqslant 0$ and we can conclude that the constructed $\tilde{\boldsymbol{\Theta}}'$ is a local minimizer of $L(\boldsymbol{\Theta})$.

■

A.2 Proof of Theorem 2

In this subsection, we provide the proof of Theorem 2 of selection consistency when the sparsity patterns of Θ^0_k's are the same, that is, $\mathcal{S}^0_1 = \cdots = \mathcal{S}^0_K$:

- If $\boldsymbol{\theta}^0_{ij} = \mathbf{0}$, then we have $\tilde{\boldsymbol{\theta}}_{ij} = \mathbf{0}$ by Theorem 1. Therefore,

  $$\omega_{k,ij}(\tilde{\boldsymbol{\theta}}_{ij}) = \eta_1(\mathbf{0})\eta_2(0) < t.$$

  Hence, we have $(i,j) \notin \tilde{\mathcal{S}}_k$.

- If $\boldsymbol{\theta}^0_{ij} \neq \mathbf{0}$, then we have $\|\tilde{\boldsymbol{\theta}}_{ij}\|_1 > (L_0 - C_5)\sqrt{\log p/n}$ with probability going to one by Theorem 1 and the minimal signal strength condition (iv) in Theorem 2. Thus,

  $$\omega_{k,ij}(\tilde{\boldsymbol{\theta}}_{ij}) = \eta_1(\tilde{\boldsymbol{\theta}}_{ij})\eta_2(\tilde{\theta}_{k,ij})$$

$$
\begin{aligned}
&= \frac{p_1 S_1(\tilde{\boldsymbol{\theta}}_{ij})}{p_1 S_1(\tilde{\boldsymbol{\theta}}_{ij}) + (1-p_1) S_2(\tilde{\boldsymbol{\theta}}_{ij})} \frac{\frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_0}}} \\
&\geqslant \frac{p_1 \prod_{k=1}^K \frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}}}{p_1 \prod_{k=1}^K \frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}} + (1-p_1) \prod_{k=1}^K \frac{1}{2v_0} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_0}}} \frac{1}{1 + \frac{(1-p_2)v_1}{p_2 v_0}} \\
&= \frac{1}{1 + \frac{(1-p_1)v_1^K}{p_1 p_2^K v_0^K} e^{-\|\tilde{\boldsymbol{\theta}}_{ij}\|_1 (1/v_0 - 1/v_1)}} \eta_2(0) \\
&\geqslant \frac{1}{1 + \frac{2p^{\epsilon_0/\alpha}}{p^{(C_4 - C_3)(L_0 - C_5)/\alpha}}} \eta_2(0) \\
&> t.
\end{aligned}
$$

So, we have $(i,j) \in \tilde{\mathcal{S}}_k$.

Therefore, we have

$$\mathbb{P}(\tilde{\boldsymbol{\mathcal{S}}} = \boldsymbol{\mathcal{S}}^0) \to 1.$$

A.3 Proof of Theorem 3

In this subsection, we provide the proof of Theorem 3 of selection consistency when the sparsity patterns of Θ_k^0's are not the same.

- If $\theta_{k,ij}^0 = 0$, then we have $\tilde{\theta}_{k,ij} = 0$ by Theorem 1. Therefore,

  $$\omega_{k,ij}(\tilde{\boldsymbol{\theta}}_{ij}) = \eta_1(\tilde{\boldsymbol{\theta}}_{ij}) \eta_2(0) \leqslant \eta_2(0) < t.$$

  Hence, we have $(i,j) \notin \tilde{\mathcal{S}}_k$.

- If $\theta_{k,ij}^0 \neq 0$, then we have $|\tilde{\theta}_{k,ij}| > (L_1 - C_5)\sqrt{\log p / n}$ with probability going to one by Theorem 1 and the minimal signal strength condition (vi) in Theorem 3. Thus,

  $$
  \begin{aligned}
  \omega_{k,ij}(\tilde{\boldsymbol{\theta}}_{ij}) &= \eta_1(\tilde{\boldsymbol{\theta}}_{ij}) \eta_2(\tilde{\theta}_{k,ij}) \\
  &= \frac{p_1 S_1(\tilde{\boldsymbol{\theta}}_{ij})}{p_1 S_1(\tilde{\boldsymbol{\theta}}_{ij}) + (1-p_1) S_2(\tilde{\boldsymbol{\theta}}_{ij})} \frac{\frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_0}}} \\
  &\geqslant \frac{p_1 \prod_{k=1}^K \frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}}}{p_1 \prod_{k=1}^K \frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}} + (1-p_1) \prod_{k=1}^K \frac{1}{2v_0} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_0}}} \frac{\frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\tilde{\theta}_{k,ij}|}{v_0}}} \\
  &\geqslant \frac{1}{1 + \frac{(1-p_1)v_1^K}{p_1 p_2^K v_0^K} e^{-|\tilde{\theta}_{k,ij}|(1/v_0 - 1/v_1)}} \frac{1}{1 + \frac{(1-p_2)v_1}{p_2 v_0} e^{-|\tilde{\theta}_{k,ij}|(1/v_0 - 1/v_1)}} \\
  &\geqslant \frac{1}{\left(1 + \frac{2p^{\epsilon_0/\alpha}}{p^{(C_4 - C_3)(L_1 - C_5)/\alpha}}\right)^2}
  \end{aligned}
  $$

$$> t.$$

So, we have $(i,j) \in \tilde{\mathcal{S}}_k$.

Therefore, we have

$$\mathbb{P}(\tilde{\boldsymbol{\mathcal{S}}} = \boldsymbol{\mathcal{S}}^0) \to 1.$$

A.4 Proof of Theorem 4

To establish the uniqueness result of $\hat{\boldsymbol{\Theta}}$, we first show that $L(\boldsymbol{\Theta})$ is strictly convex under the constraint $\|\Theta_k\|_2 \leqslant B, k = 1, \ldots, K$. Then, we show the local minimizer $\tilde{\boldsymbol{\Theta}}$ we constructed is inside the constraint set, i.e., $\|\tilde{\Theta}_k\|_2 \leqslant B$.

To establish the strict convexity, we decompose $L(\boldsymbol{\Theta})$ into the following two parts and show that each part is strictly convex:

(I) $L_1(\boldsymbol{\Theta}) = \sum_k \left[-l(\Theta_k) - \frac{\alpha K}{4v_0^2} \|\Theta_k\|_F^2 \right]$

(II) $L_2(\boldsymbol{\Theta}) = \alpha \sum_i \sum_k \mathrm{Pen}_{L_1}(\theta_{k,ii}) + \frac{\alpha}{2} \sum_{i \neq j} \mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij}) + \frac{\alpha K}{4v_0^2} \sum_k \|\Theta_k\|_F^2$

- For (I):

  The Hessian matrix of $L_1(\Theta_k)$ is

  $$\nabla^2 L_1(\Theta_k) = \frac{n}{2}(\Theta_k \otimes \Theta_k)^{-1} - \frac{\alpha K}{2v_0^2} I_{p^2 \times p^2}, \quad k = 1, \ldots . K.$$

  Consider the minimum eigenvalue of the Hessian matrix of $L_1(\Theta_k)$,

  $$\begin{aligned} \lambda_{\min}(\nabla^2 L_1(\Theta_k)) =& \frac{n}{2} \lambda_{\max}^{-1}(\Theta_k \otimes \Theta_k) - \frac{\alpha K}{2v_0^2} \\ =& \frac{n}{2} \lambda_{\max}^{-2}(\Theta_k) - \frac{\alpha K}{2v_0^2} \\ \geqslant& \frac{n}{2} \|\Theta_k\|_2^{-2} - \frac{\alpha K}{2v_0^2} \\ \geqslant& \frac{n}{2} B^{-2} - \frac{\alpha K}{2v_0^2} > 0. \end{aligned}$$

  Therefore, $L_1(\Theta_k), k = 1, \ldots, K$ and $L_1(\boldsymbol{\Theta}) = \sum_k L_1(\Theta_k)$ are all strictly convex.

- For (II):

  Denote $\mathrm{Pen}(\boldsymbol{\Theta}) = \sum_i \sum_k \mathrm{Pen}_{L_1}(\theta_{k,ii}) + \frac{1}{2} \sum_{i \neq j} \mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})$ and consider the magnitude of its second-order subgradients. For any $k \neq k' \in \{1, \ldots, K\}$ and $i \neq j \in \{1, \ldots, p\}$,

we have

$$\begin{cases} \left|\nabla^2_{\theta_{k,ij},\theta_{k,ij}}\mathrm{Pen}(\boldsymbol{\Theta})\right| = \left(\frac{1}{v_0}-\frac{1}{v_1}\right)^2 \omega_{k,ij}(\boldsymbol{\theta}_{ij})(1-\omega_{k,ij}(\boldsymbol{\theta}_{ij})) \\ \qquad\qquad\qquad\qquad \leqslant \frac{1}{4}\left(\frac{1}{v_0}-\frac{1}{v_1}\right)^2 \\ \qquad\qquad\qquad\qquad \leqslant \frac{1}{4v_0^2} \\ \left|\nabla^2_{\theta_{k,ij},\theta_{k',ij}}\mathrm{Pen}(\boldsymbol{\Theta})\right| = \left(\frac{1}{v_0}-\frac{1}{v_1}\right)^2 \eta_2(\theta_{k,ij})\eta_2(\theta_{k',ij})\eta_1(\boldsymbol{\theta}_{ij})(1-\eta_1(\boldsymbol{\theta}_{ij})) \\ \qquad\qquad\qquad\qquad \leqslant \left(\frac{1}{v_0}-\frac{1}{v_1}\right)^2 \eta_1(\boldsymbol{\theta}_{ij})(1-\eta_1(\boldsymbol{\theta}_{ij})) \\ \qquad\qquad\qquad\qquad \leqslant \frac{1}{4v_0^2} \\ \nabla^2_{\theta_{k,ii},\theta_{k,ii}}\mathrm{Pen}(\boldsymbol{\Theta}) = \nabla^2_{\theta_{k,ii},\theta_{k',ii}}\mathrm{Pen}(\boldsymbol{\Theta}) = 0 \end{cases}$$

Therefore, for any (i,j) pair, we have

$$\lambda_{\min}(\nabla^2_{\boldsymbol{\theta}_{ij},\boldsymbol{\theta}_{ij}}\mathrm{Pen}(\boldsymbol{\Theta})) > -\frac{K}{4v_0^2}.$$

Hence,

$$\begin{aligned} \lambda_{\min}(\nabla^2_{\boldsymbol{\theta}_{ij},\boldsymbol{\theta}_{ij}}L_2(\boldsymbol{\Theta})) &= \alpha\lambda_{\min}(\nabla^2_{\boldsymbol{\theta}_{ij},\boldsymbol{\theta}_{ij}}\mathrm{Pen}(\boldsymbol{\Theta})) + \frac{\alpha K}{2v_0^2} \\ &> -\frac{\alpha K}{4v_0^2} + \frac{\alpha K}{2v_0^2} \\ &= \frac{\alpha K}{4v_0^2} > 0, \end{aligned}$$

that is, $\nabla^2_{\boldsymbol{\theta}_{ij},\boldsymbol{\theta}_{ij}}L_2(\boldsymbol{\Theta})$ is positive definite for any (i,j) pair. Since $\nabla^2_{\boldsymbol{\theta}_{ij},\boldsymbol{\theta}_{i'j'}}L_2(\boldsymbol{\Theta}) = \mathbf{0}$ for any two different (i,j), (i',j') pairs, we have $\nabla^2 L_2(\boldsymbol{\Theta})$ is positive definite. Therefore, $L_2(\boldsymbol{\Theta})$ is strictly convex.

Thus, $L(\boldsymbol{\Theta}) = L_1(\boldsymbol{\Theta}) + L_2(\boldsymbol{\Theta})$ is strictly convex.

The only thing left to show is $\tilde{\boldsymbol{\Theta}}$ is inside the constraint set, i.e., $\|\tilde{\Theta}_k\|_2 \leqslant B, k = 1,\ldots,K$. Indeed, we have

$$\begin{aligned} \|\tilde{\Theta}_k\|_2 &\leqslant \|\tilde{\Theta}_k - \Theta_k^0\|_2 + \|\Theta_k^0\|_2 \\ &\leqslant |||\tilde{\Theta}_k - \Theta_k^0|||_\infty + \frac{1}{\xi_1} \\ &\leqslant dr_e + \frac{1}{\xi_1} < B. \end{aligned}$$

Therefore, the theorem is proved.

Appendix B. Other Proof

The following lemma gives upper bounds on the magnitudes of the first derivative and second derivative of $\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})$ when $\theta_{k,ij}$ is $O(\sqrt{\log p/n})$ away from zero.

Lemma 6 *Under condition (i) in Theorem 5, if* $|\theta_{k,ij}| \geqslant 2(C_2 - C_3)M_{\Gamma^0}\sqrt{\log p/n}$, *then we have*

$$\frac{\alpha}{n}\left|\frac{\partial}{\partial\theta_{k,ij}} Pen_{GB}(\boldsymbol{\theta}_{ij})\right| < C_3\sqrt{\log p/n},$$

and

$$\frac{\alpha}{2n}\left|\frac{\partial^2}{\partial^2\theta_{k,ij}} Pen_{GB}(\boldsymbol{\theta}_{ij})\right| < \frac{C_3^2\log p}{4\alpha p^{\epsilon_0/\alpha}}$$

Furthermore, for any $k' \neq k$, *we have*

$$\frac{\alpha}{2n}\left|\frac{\partial^2}{\partial\theta_{k,ij}\partial\theta_{k',ij}} Pen_{GB}(\boldsymbol{\theta}_{ij})\right| < \frac{C_3^2\log p}{8\alpha p^{\epsilon_0/\alpha}}$$

Proof We first show that $\eta_1(\boldsymbol{\theta}_{ij})$ and $\eta_2(\theta_{k,ij})$ are close to 1 when $|\theta_{k,ij}| \geqslant 2(C_2 - C_3)M_{\Gamma^0}\sqrt{\log p/n}$.

$$\begin{aligned}
\eta_1(\boldsymbol{\theta}_{ij}) &= \frac{p_1 S_1(\boldsymbol{\theta}_{ij})}{p_1 S_1(\boldsymbol{\theta}_{ij}) + (1-p_1)S_2(\boldsymbol{\theta}_{ij})} \\
&\geqslant \frac{p_1\prod_{l=1}^K \frac{p_2}{2v_1}e^{-\frac{|\theta_{l,ij}|}{v_1}}}{p_1\prod_{l=1}^K \frac{p_2}{2v_1}e^{-\frac{|\theta_{l,ij}|}{v_1}} + (1-p_1)\prod_{l=1}^K \frac{1}{2v_0}e^{-\frac{|\theta_{l,ij}|}{v_0}}} \\
&= \frac{1}{1 + \frac{(1-p_1)v_1^K}{p_1 p_2^K v_0^K}e^{-|\theta_{k,ij}|(1/v_0 - 1/v_1)}} \geqslant \frac{1}{1+\frac{v_0^2}{v_1^2}\epsilon_1 p^{-\epsilon_0/\alpha}} \geqslant 1 - \frac{v_0^2}{v_1^2}\epsilon_1 p^{-\epsilon_0/\alpha}, \\
\eta_2(\theta_{k,ij}) &= \frac{\frac{p_2}{2v_1}e^{-\frac{|\theta_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1}e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0}e^{-\frac{|\theta_{k,ij}|}{v_0}}} \\
&= \frac{1}{1 + \frac{(1-p_2)v_1}{p_2 v_0}e^{-|\theta_{k,ij}|(1/v_0 - 1/v_1)}} \geqslant \frac{1}{1+\frac{v_0^2}{v_1^2}\epsilon_1 p^{-\epsilon_0/\alpha}} \geqslant 1 - \frac{v_0^2}{v_1^2}\epsilon_1 p^{-\epsilon_0/\alpha},
\end{aligned}$$

where $\epsilon_0 := (C_2 - C_3)M_{\Gamma^0}(C_4 - C_3)$, due to the following inequalities:

$$\begin{cases} \frac{(1-p_2)v_1^3}{p_2 v_0^3},\ \frac{v_1^{K+2}(1-p_1)}{v_0^{K+2}p_1 p_2^K} < \epsilon_1 p^{(C_2-C_3)M_{\Gamma^0}(C_4-C_3)/\alpha} \\ e^{|\theta_{k,ij}|(1/v_0 - 1/v_1)} > p^{2(C_2-C_3)M_{\Gamma^0}(C_4-C_3)/\alpha}. \end{cases}$$

Therefore

$$\omega_{k,ij}(\boldsymbol{\theta}_{ij}) = \eta_1(\boldsymbol{\theta}_{ij})\eta_2(\theta_{k,ij}) \geqslant 1 - \frac{2\epsilon_1}{p^{\epsilon_0/\alpha}}\cdot\frac{v_0^2}{v_1^2}.$$

- For the first derivative, we have

$$\begin{aligned}\left|\frac{\partial}{\partial\theta_{k,ij}}\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})\right| &= \omega_{k,ij}(\boldsymbol{\theta}_{ij})\frac{1}{v_1} + (1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}))\frac{1}{v_0}\\ &\leqslant \frac{1}{v_1} + (1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}))\frac{1}{v_0}\\ &\leqslant \frac{1}{v_1} + \frac{2\epsilon_1}{p^{\epsilon_0/\alpha}}\cdot\frac{v_0^2}{v_1^2}\cdot\frac{1}{v_0}\\ &\leqslant \frac{1}{v_1}(1+2\epsilon_1)\end{aligned}$$

  when p is large enough so that $p^{\epsilon_0/\alpha} > 1$. Therefore,

$$\frac{\alpha}{n}\left|\frac{\partial}{\partial\theta_{k,ij}}\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})\right| \leqslant \frac{\alpha}{nv_1}(1+2\epsilon_1) \leqslant C_3\sqrt{\log p/n}.$$

- For the second derivative, we have

$$\begin{aligned}\frac{\alpha}{2n}\left|\frac{\partial^2}{\partial^2\theta_{k,ij}}\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})\right| &= \frac{\alpha}{2n}\left(\frac{1}{v_0}-\frac{1}{v_1}\right)^2\omega_{k,ij}(\boldsymbol{\theta}_{ij})(1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}))\\ &\leqslant \frac{\alpha}{2n}\cdot\frac{1}{v_1^2}\cdot\frac{v_1^2}{v_0^2}(1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}))\\ &\leqslant \frac{C_3^2\log p}{4\alpha p^{\epsilon_0/\alpha}}\end{aligned}$$

  and for $k \neq k'$, we have

$$\begin{aligned}\frac{\alpha}{2n}\left|\frac{\partial^2}{\partial\theta_{k,ij}\partial\theta_{k',ij}}\mathrm{Pen}_{\mathrm{GB}}(\boldsymbol{\theta}_{ij})\right| &= \frac{\alpha}{2n}\left(\frac{1}{v_0}-\frac{1}{v_1}\right)^2\eta_2(\theta_{k,ij})\eta_2(\theta_{k',ij})\eta_1(\boldsymbol{\theta}_{ij})(1-\eta_1(\boldsymbol{\theta}_{ij}))\\ &\leqslant \frac{\alpha}{2n}\cdot\frac{1}{v_1^2}\cdot\frac{v_1^2}{v_0^2}(1-\eta_1(\boldsymbol{\theta}_{ij}))\\ &\leqslant \frac{C_3^2\log p}{8\alpha p^{\epsilon_0/\alpha}}\end{aligned}$$

  due to

$$\begin{cases} 1-\omega_{k,ij}(\boldsymbol{\theta}_{ij}) \leqslant 2\frac{v_0^2}{v_1^2}\cdot\frac{\epsilon_1}{p^{\epsilon_0/\alpha}}, \quad 1-\eta_1(\boldsymbol{\theta}_{ij}) \leqslant \frac{v_0^2}{v_1^2}\cdot\frac{\epsilon_1}{p^{\epsilon_0/\alpha}},\\ \frac{1}{v_1^2} < \frac{C_3^2}{(1+2\epsilon_1)^2}\cdot\frac{n\log p}{\alpha^2} < \frac{C_3^2}{4\epsilon_1}\cdot\frac{n\log p}{\alpha^2}.\end{cases}$$

■

Appendix C. Marginal Posterior Inclusion Probability

In this section, we derive the marginal inclusion probability (6). Given $\boldsymbol{\Theta}$ and $\boldsymbol{Y}$, the indicators $\Gamma = (\gamma_{ij})$ and $\boldsymbol{R} = (R_1, \ldots, R_K)$ with $R_k = (r_{k,ij})$ in different locations on the

precision matrices would be independent:

$$\pi(\Gamma, \boldsymbol{R} \mid \boldsymbol{\Theta}, \boldsymbol{Y}) \propto \prod_{i<j} \pi(\gamma_{ij}, \boldsymbol{r}_{ij} \mid \boldsymbol{\Theta}, \boldsymbol{Y}),$$

where $\boldsymbol{r}_{ij} = (r_{1,ij}, \ldots, r_{K,ij})$. Therefore, it suffices to look at one location and the posterior for indicators in other locations can be similarly derived:

$$\begin{aligned}\pi(\gamma_{ij}, \boldsymbol{r}_{ij} \mid \boldsymbol{\Theta}, \boldsymbol{Y}) \propto & \left[p_1 \prod_{k=1}^{K} p_2^{r_{k,ij}} (1-p_2)^{1-r_{k,ij}} \right]^{\gamma_{ij}} \left[(1-p_1) \prod_{k=1}^{K} \delta_0(r_{k,ij}) \right]^{1-\gamma_{ij}} \\ & \times \prod_{k=1}^{K} \mathrm{LP}(\theta_{k,ij}; v_1)^{r_{k,ij}} \mathrm{LP}(\theta_{k,ij}; v_0)^{1-r_{k,ij}}\end{aligned}$$

After summing out γ_{ij}, we have

$$\begin{aligned}\pi(\boldsymbol{r}_{ij} \mid \boldsymbol{\Theta}, \boldsymbol{Y}) \propto & p_1 \prod_{k=1}^{K} \left[p_2 \mathrm{LP}(\theta_{k,ij}; v_1) \right]^{r_{k,ij}} \left[(1-p_2) \mathrm{LP}(\theta_{k,ij}; v_0) \right]^{1-r_{k,ij}} \\ & + (1-p_1) \prod_{k=1}^{K} \mathrm{LP}(\theta_{k,ij}; v_0) \prod_{k=1}^{K} \delta_0(r_{k,ij}) \\ \propto & p_1 S_1(\boldsymbol{\theta}_{ij}) \prod_{k=1}^{K} \left[\frac{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}}} \right]^{r_{k,ij}} \left[\frac{\frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}}}{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}}} \right]^{1-r_{k,ij}} \\ & + (1-p_1) S_2(\boldsymbol{\theta}_{ij}) \prod_{k=1}^{K} \delta_0(r_{k,ij}),\end{aligned}$$

where

$$S_1(\boldsymbol{\theta}_{ij}) = \prod_{k=1}^{K} \left(\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}} \right) \text{ and } S_2(\boldsymbol{\theta}_{ij}) = \prod_{k=1}^{K} \frac{1}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}}.$$

As can be seen, it is a mixture of a point mass at $\mathbf{0}$ and a product of independent Bernoulli distributions, so we have

$$\mathbb{E}_{\Gamma, \boldsymbol{R} \mid \boldsymbol{\Theta}, \boldsymbol{Y}}(r_{k,ij}) = \frac{p_1 S_1(\boldsymbol{\theta}_{ij})}{p_1 S_1(\boldsymbol{\theta}_{ij}) + (1-p_1) S_2(\boldsymbol{\theta}_{ij})} \frac{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}}}{\frac{p_2}{2v_1} e^{-\frac{|\theta_{k,ij}|}{v_1}} + \frac{1-p_2}{2v_0} e^{-\frac{|\theta_{k,ij}|}{v_0}}}.$$

Appendix D. Comparison of Sparsity Structure Recovery Performance

In this section, we provide a more rigorous comparison of sparsity structure recovery performance between our proposed GemBag method and individual estimation method. Let us consider the entries that might have different classification results by GemBag method and individual estimation method and divide them into different groups: (1) entries with

weak signals that are identified as signals by GemBag and noises by individual estimation (2) noise entries that are identified as signals by GemBag and noises by individual estimation (3) entries with strong signals that are identified as noises by GemBag and signals by individual estimation.

We introduce the following notations to represent the count of entries in each group. For any threshold t, let

$$\begin{cases} N_1 = \sum_k \sum_{i\neq j} \mathbb{1}(0 < |\theta^0_{k,ij}| < C_5\sqrt{(\log p)/n} \text{ and } \omega_{k,ij}((|\boldsymbol{\theta}^0_{ij}| - C_5\sqrt{(\log p)/n}\mathbf{1_K})^+) > t), \\ N_2 = \sum_k \sum_{i\neq j} \mathbb{1}(|\theta^0_{k,ij}| = 0 \text{ and } \omega_{k,ij}(|\boldsymbol{\theta}'_{ij}|) > t), \\ N_3 = \sum_k \sum_{i\neq j} \mathbb{1}(|\theta^0_{k,ij}| \geqslant C_5\sqrt{(\log p)/n} \text{ and } \omega_{k,ij}((|\boldsymbol{\theta}^0_{ij}| - C_5\sqrt{(\log p)/n}\mathbf{1_K})^+) \leqslant t), \end{cases}$$

where $C_5\sqrt{(\log p)/n}$ is the estimation error bound from Theorem 1, the $(\cdot)^+$ operator elementwisely takes the positive part of the input vector, and entries of $\boldsymbol{\theta}'_{ij} = (\theta'_{1,ij}, \ldots, \theta'_{K,ij})$ have the following form:

$$\theta'_{k,ij} = \begin{cases} 0 & \text{if } |\theta^0_{k,ij}| < C_5\sqrt{(\log p)/n}, \\ |\theta^0_{k,ij}| + C_5\sqrt{(\log p)/n} & \text{otherwise,} \end{cases} \quad k = 1, \ldots, K.$$

Here, N_1 measures the number of entries that are weak individually but strong groupwisely in signal strength; N_2 measures the number of entries that are noises but have strong signal strength jointly; and N_3 measures the number of entries that are strong individually but weak groupwisely in signal strength. Thus, large N_1 indicates that sparsity patterns are more likely to be the same within the group while large N_2 and N_3 imply that there are large discrepancies in the sparsity pattern of entries within the group. We introduce the following criterion to measure the similarity of the sparsity patterns of Θ_k's:

(A.4) Similarity condition:

$$N_1 > N_2 + N_3.$$

The following proposition illustrates the advantage of GemBag over individual estimation when sparsity patterns are similar across classes.

Proposition 7 *Under the same conditions in Theorem 1 and the similarity condition (A.4), the misclassification error by GemBag is smaller than the one by estimating each graph individually.*

Proof We first look at the misclassification error when we estimate each graph individually, that is, set $p_1 = 1$. From the constructive proof of Theorem 1, when $|\theta^0_{k,ij}| < C_5\sqrt{(\log p)/n}$, we have $\tilde{\theta}_{k,ij} = 0$. Besides, from (7) and $p_1 = 1$, we have the posterior inclusion probability $\omega_{k,ij}(\tilde{\boldsymbol{\theta}}_{ij}) = \eta_2(\tilde{\theta}_{k,ij})$ which only depends on the individual estimate $\tilde{\theta}_{k,ij}$. So there is no way that we can differentiate the small signals with magnitude less than $C_5\sqrt{(\log p)/n}$ from noises with magnitude 0. Therefore, the number of misclassified elements is at least $N_4 = \sum_k \sum_{i\neq j} \mathbb{1}(0 < |\theta^0_{k,ij}| < C_5\sqrt{(\log p)/n})$.

In the case of using our proposed estimator to perform sparsity structure recovery, the misclassification error could come from two sources:

1. Noises but classified as signals $\sum_k \sum_{i\neq j} \mathbb{1}(|\theta^0_{k,ij}| = 0 \text{ and } \omega_{k,ij}(|\tilde{\boldsymbol{\theta}}_{ij}|) > t) < N_2$,

2. Signals but classified as noises $\sum_k \sum_{i \neq j} \mathbb{1}(|\theta^0_{k,ij}| > 0 \text{ and } \omega_{k,ij}(|\tilde{\boldsymbol{\theta}}_{ij}|) \leqslant t) < N_3 + (N_4 - N_1)$.

By the similarity condition, we have $N_2 + N_3 + (N_4 - N_1) = N_4 + (N_2 + N_3 - N_1) < N_4$. Hence, we conclude that the misclassification error from thresholding the posterior inclusion probability of our proposed estimator is lower than the one from individual estimation method.

■

Appendix E. Additional Simulation Results

In this section, we provide some additional simulation results.

E.1 Non-normally Distributed Data

We consider the multivariate t-distribution with degrees of freedom = 5 or 10 to generate data and use the four network designs as described in Section 5 to construct the true precision matrices. As can be seen in Table 7 and 8, GemBag still has the best performance in terms of both selection accuracy (MCC and AUC) and estimation accuracy (F-norm). As the degrees of freedom parameter of the multivariate t-distribution grows larger and the tails become lighter, GemBag outperforms all the other competitors also in ℓ_∞ norm in the nearest-neighbor network and scale-free network. In the AR(2) model and circle model, signals within the same group have the same magnitude and thus, Pooled method has some advantage by aggregating all the samples from different classes to get small estimation error in ℓ_∞ norm, which is similar to what we observe in the multivariate normal case.

E.2 Empirical Study on the Rate of v_0

We consider the nearest-neighbor network with $\rho = 1$ as described in Section 5 and try different combinations of (n, p, α) values with $n = \alpha = 100, 150, 200, 250$ and $p = 50, 75, 100, 125, 150$. For each combination, we select v_0 that produces an unique estimate under different initial values with the smallest error in F-norm:

$$\arg\min_{v_0} \sum_k \|\hat{\Theta}_k(v_0) - \Theta^0_k\|_F.$$

This process is repeated 30 times for each combination of (n, p, α) values. In Figure 7, we plot the average of selected v_0 over 30 repetitions versus the value of $\alpha/\sqrt{n \log p}$. As can be seen, the points approximately form a straight line with a positive slope, which supports our theory that v_0 scales linearly with $\alpha/\sqrt{n \log p}$.

E.3 Empirical Performance with Large p

To evaluate the performance of our method with higher dimensions, we consider the nearest-neighbor network with $p = 400$ and $\rho = 0.25$ as described in Section 5. We implement our method with $\alpha = 1$ and $\alpha = n$ and the competitors and report the results aggregated over 50 replication in Table 9. As shown in the table, GemBag still has the best performance in

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
	AR(2) model ($n = 100,\ p = 50$)					
GemBag ($\alpha = 1$)	**0.996(0.001)**	0.817(0.040)	**0.862(0.023)**	**0.990(0.006)**	**2.486(0.093)**	0.724(0.143)
GemBag ($\alpha = n$)	**0.996(0.002)**	0.748(0.045)	0.818(0.026)	0.987(0.009)	2.722(0.167)	0.727(0.138)
BAGUS	0.952(0.006)	0.805(0.034)	0.608(0.025)	0.902(0.021)	3.322(0.184)	0.938(0.154)
Pooled	0.900(0.023)	**0.993(0.008)**	0.579(0.051)	0.986(0.004)	2.920(0.117)	**0.508(0.037)**
GGL	0.918(0.014)	0.731(0.021)	0.474(0.030)	0.850(0.010)	4.167(0.094)	0.621(0.071)
	Circle model ($n = 100,\ p = 50$)					
GemBag ($\alpha = 1$)	0.999(0.000)	0.981(0.027)	0.977(0.018)	**1.000(0.000)**	2.024(0.138)	0.822(0.151)
GemBag ($\alpha = n$)	**1.000(0.000)**	0.982(0.010)	**0.985(0.007)**	**1.000(0.000)**	**1.790(0.121)**	0.677(0.123)
BAGUS	0.961(0.005)	0.991(0.006)	0.646(0.025)	0.995(0.003)	2.694(0.153)	0.899(0.139)
Pooled	0.914(0.025)	**1.000(0.000)**	0.492(0.058)	0.996(0.000)	2.553(0.091)	**0.515(0.028)**
GGL	0.905(0.012)	0.988(0.006)	0.468(0.023)	0.993(0.003)	3.169(0.083)	0.669(0.147)
	Nearest-neighbor network ($n = 100,\ p = 100,\ \rho = 0.25$)					
GemBag ($\alpha = 1$)	0.985(0.004)	0.753(0.044)	0.668(0.039)	**0.939(0.017)**	4.002(0.295)	0.667(0.151)
GemBag ($\alpha = n$)	**0.997(0.001)**	0.634(0.073)	**0.738(0.038)**	0.918(0.026)	**3.963(0.322)**	0.620(0.125)
BAGUS	0.963(0.006)	**0.765(0.037)**	0.528(0.028)	0.878(0.020)	5.660(0.517)	0.967(0.177)
Pooled	0.945(0.012)	0.561(0.037)	0.339(0.034)	0.764(0.021)	6.225(0.249)	0.887(0.098)
GGL	0.938(0.016)	0.720(0.056)	0.418(0.042)	0.848(0.028)	5.288(0.305)	**0.585(0.082)**
	Scale-free network ($n = 100,\ p = 100,\ \rho = 0.25$)					
GemBag ($\alpha = 1$)	0.984(0.003)	0.903(0.028)	0.725(0.040)	**0.977(0.009)**	3.360(0.238)	0.699(0.126)
GemBag ($\alpha = n$)	**0.997(0.001)**	0.765(0.048)	**0.811(0.029)**	0.937(0.020)	**3.208(0.266)**	0.632(0.116)
BAGUS	0.964(0.006)	0.874(0.025)	0.571(0.035)	0.940(0.013)	4.935(0.603)	0.964(0.180)
Pooled	0.927(0.011)	0.631(0.039)	0.310(0.030)	0.803(0.021)	6.320(0.234)	0.828(0.046)
GGL	0.932(0.012)	**0.931(0.022)**	0.479(0.034)	0.960(0.012)	4.772(0.350)	**0.559(0.036)**

Table 7: Simulation results for data following multivariate t-distribution (degrees of freedom = 5).

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
	AR(2) model ($n = 100,\ p = 50$)					
GemBag ($\alpha = 1$)	**0.997(0.001)**	0.878(0.031)	**0.910(0.017)**	**0.996(0.003)**	**2.057(0.103)**	0.574(0.090)
GemBag ($\alpha = n$)	**0.997(0.002)**	0.827(0.023)	0.879(0.012)	0.994(0.003)	2.226(0.084)	0.591(0.098)
BAGUS	0.974(0.004)	0.840(0.029)	0.727(0.025)	0.929(0.019)	2.525(0.069)	0.682(0.119)
Pooled	0.951(0.010)	**0.995(0.005)**	0.721(0.037)	0.988(0.002)	2.868(0.078)	**0.489(0.028)**
GGL	0.950(0.007)	0.750(0.016)	0.570(0.019)	0.866(0.008)	4.185(0.050)	0.577(0.028)
	Circle model ($n = 100,\ p = 50$)					
GemBag ($\alpha = 1$)	**1.000(0.000)**	0.997(0.003)	0.994(0.002)	**1.000(0.000)**	1.528(0.068)	0.650(0.108)
GemBag ($\alpha = n$)	**1.000(0.000)**	0.996(0.004)	**0.995(0.003)**	**1.000(0.000)**	**1.408(0.060)**	0.557(0.078)
BAGUS	0.981(0.002)	0.996(0.003)	0.777(0.014)	0.998(0.002)	1.824(0.067)	0.667(0.107)
Pooled	0.968(0.008)	**1.000(0.000)**	0.684(0.048)	0.996(0.000)	2.469(0.047)	**0.491(0.022)**
GGL	0.936(0.006)	0.994(0.004)	0.546(0.019)	0.997(0.002)	3.160(0.051)	0.575(0.063)
	Nearest-neighbor network ($n = 100,\ p = 100,\ \rho = 0.25$)					
GemBag ($\alpha = 1$)	0.994(0.002)	0.792(0.036)	0.787(0.030)	**0.955(0.013)**	**3.263(0.151)**	**0.500(0.078)**
GemBag ($\alpha = n$)	**0.998(0.002)**	0.713(0.043)	**0.800(0.025)**	0.942(0.015)	3.482(0.168)	0.503(0.077)
BAGUS	0.977(0.005)	**0.797(0.030)**	0.628(0.030)	0.902(0.017)	4.128(0.276)	0.708(0.111)
Pooled	0.966(0.005)	0.569(0.043)	0.415(0.027)	0.776(0.021)	6.235(0.275)	0.897(0.099)
GGL	0.957(0.011)	0.747(0.052)	0.494(0.045)	0.866(0.026)	5.343(0.256)	0.545(0.030)
	Scale-free network ($n = 100,\ p = 100,\ \rho = 0.25$)					
GemBag ($\alpha = 1$)	0.994(0.002)	0.931(0.020)	**0.854(0.028)**	**0.987(0.005)**	**2.497(0.118)**	0.530(0.110)
GemBag ($\alpha = n$)	**0.998(0.002)**	0.811(0.040)	0.852(0.027)	0.950(0.018)	2.729(0.137)	**0.516(0.086)**
BAGUS	0.981(0.003)	0.899(0.021)	0.695(0.026)	0.955(0.010)	3.187(0.218)	0.657(0.122)
Pooled	0.948(0.005)	0.649(0.035)	0.373(0.025)	0.824(0.017)	6.358(0.222)	0.818(0.043)
GGL	0.951(0.007)	**0.950(0.019)**	0.549(0.027)	0.972(0.010)	4.812(0.282)	0.557(0.030)

Table 8: Simulation results for data following multivariate t-distribution (degrees of freedom = 10).

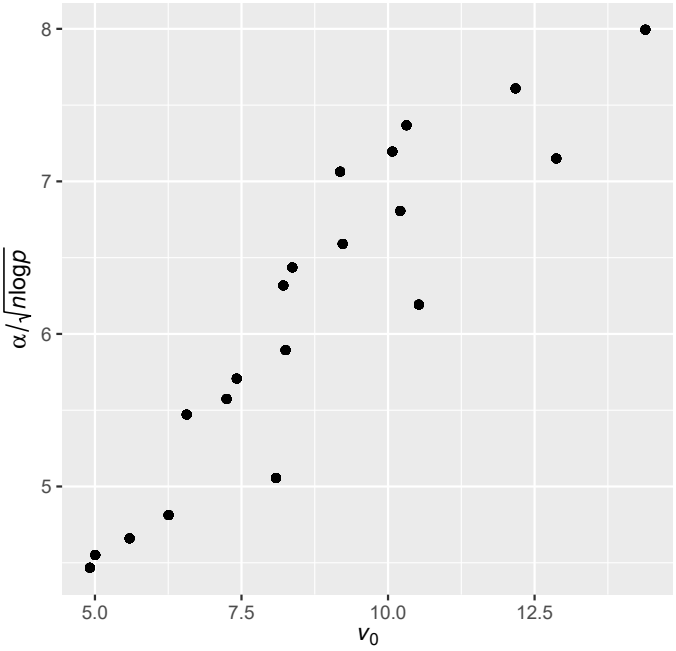


Figure 7: Selected v_0 versus $\alpha/\sqrt{n \log p}$ with different combinations of (n, p, α) values.

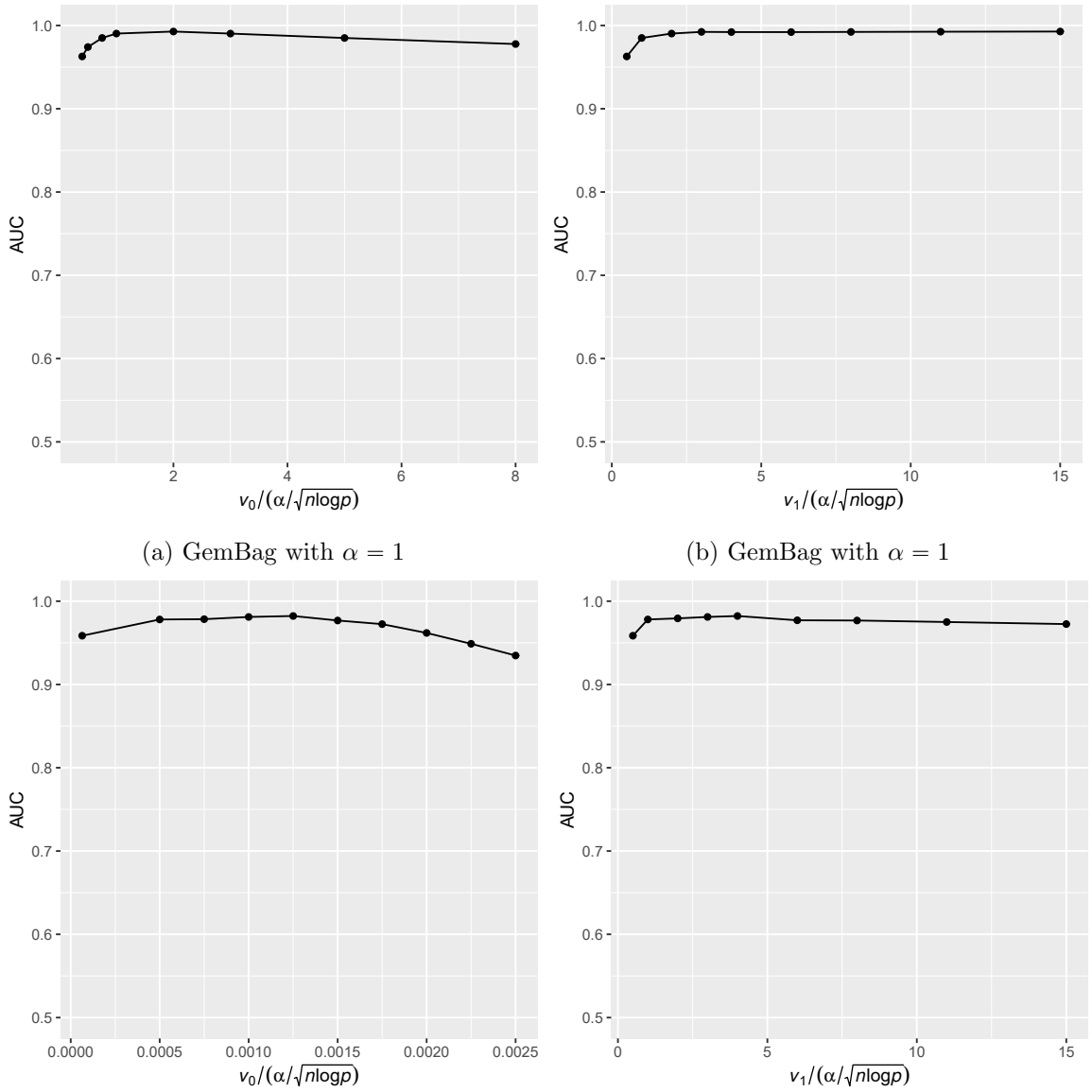
terms of both selection accuracy (MCC and AUC) and estimation accuracy (F-norm and ℓ_∞ norm) when p is large.

	Spec	Sens	MCC	AUC	F-norm	ℓ_∞ norm
GemBaG ($\alpha = 1$)	0.998(0.000)	0.754(0.023)	0.765(0.012)	**0.944(0.009)**	**6.512(0.174)**	**0.509(0.051)**
GemBaG ($\alpha = n$)	**1.000(0.000)**	0.632(0.020)	**0.780(0.014)**	0.916(0.008)	7.435(0.170)	0.529(0.061)
BAGUS	0.986(0.000)	**0.802(0.015)**	0.478(0.008)	0.902(0.007)	10.205(0.160)	0.901(0.089)
Pooled	0.991(0.001)	0.532(0.022)	0.398(0.015)	0.765(0.012)	12.480(0.254)	0.980(0.043)
GGL	0.989(0.002)	0.566(0.064)	0.400(0.054)	0.780(0.032)	12.340(0.326)	0.618(0.021)

Table 9: Simulation results for nearest-neighbor network when $p = 400$.

E.4 Sensitivity Analysis of v_0 and v_1 on Graph Estimation

We implement our method with $\alpha = 1$ and $\alpha = n$ on the nearest-neighbor network with $\rho = 0.25$ as described in Section 5 to analyze how sensitive the performance of our method in estimating the graphs is to different choices of hyperparameters. In Figure 8, we plot the best AUC value versus v_0 while varying v_1 or versus v_1 while varying v_0 in each setting of α. As can be seen, the performance of graph estimation is more sensitive to v_0 compared to v_1 in both settings of α. Besides, compared to the situation where α is small, the performance of our method with a large α in graph estimation is more sensitive to different choices of hyperparameters v_0 and v_1. This suggests that more careful tuning in v_0 and in the setting of large α is desired to get better performance by our method in graph estimation.



(a) GemBag with $\alpha = 1$

(b) GemBag with $\alpha = 1$

(c) GemBag with $\alpha = n$

(d) GemBag with $\alpha = n$

Figure 8: AUC versus different choices of hyperparameters v_0 and v_1.

Appendix F. Spatial Visualization of the Bike Sharing Network

In this section, we provide spatial visualization of both the intersection and the union of the estimated bike sharing networks from Section 6 in Figure 9. As can be seen, though networks estimated by GemBag are much sparser compared to BAGUS, they turn out to share more common edges, which illustrates that GemBag is more capable of capturing common structures.

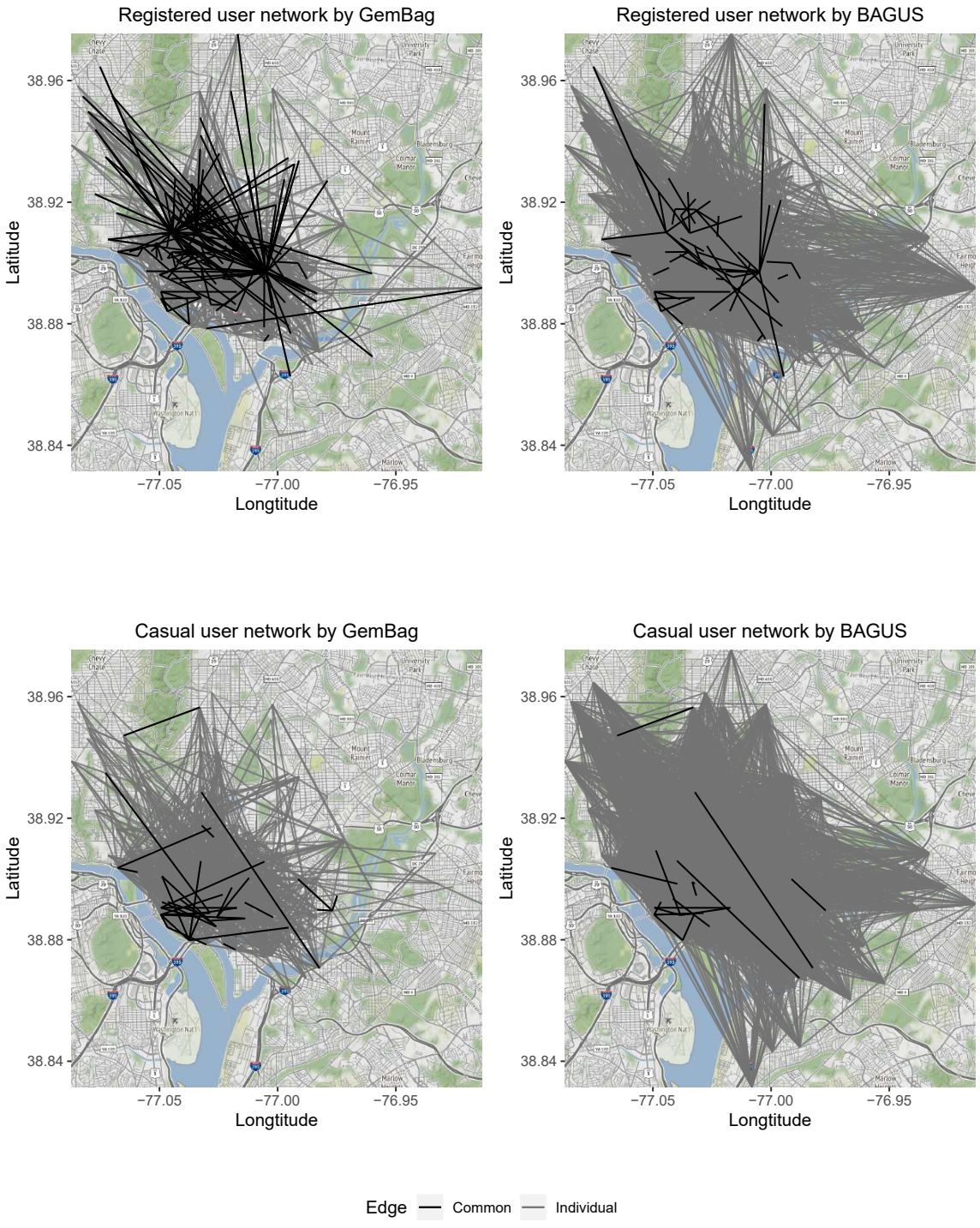


Figure 9: Intersection and union of the estimated networks across three years.

References

Sayantan Banerjee and Subhashis Ghosal. Bayesian structure learning in graphical models. *Journal of Multivariate Analysis*, 136:147–162, 2015.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

Patrick Breheny and Jian Huang. Penalized methods for bi-level variable selection. *Statistics and its interface*, 2(3):369, 2009.

Tony Cai, Weidong Liu, and Xi Luo. A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494): 594–607, 2011.

Carlos M. Carvalho and James G. Scott. Objective Bayesian model selection in Gaussian graphical models. *Biometrika*, 96(3):497–512, 2009.

Fan Chung and Linyuan Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences*, 99(25):15879–15882, 2002.

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.

Arthur P. Dempster. Covariance selection. *Biometrics*, 28(1):157–175, 1972.

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

Jianqing Fan, Yang Feng, and Yichao Wu. Network exploration via the adaptive lasso and SCAD penalties. *The Annals of Applied Statistics*, 3(2):521–541, 2009.

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

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Applications of the lasso and grouped lasso to the estimation of sparse graphical models. Technical report, Stanford University, 2010.

Lingrui Gan, Naveen N. Narisetty, and Feng Liang. Bayesian regularization for graphical models with unequal shrinkage. *Journal of the American Statistical Association*, 114 (527):1218–1231, 2019a. doi: 10.1080/01621459.2018.1482755.

Lingrui Gan, Xinming Yang, Naveen N. Narisetty, and Feng Liang. Bayesian joint estimation of multiple graphical models. In *Advances in Neural Information Processing Systems 32*, pages 9802–9812, 2019b.

Jian Guo, Elizaveta Levina, George Michailidis, and Ji Zhu. Joint estimation of multiple graphical models. *Biometrika*, 98(1):1–15, 2011.

Cho-Jui Hsieh, Inderjit S. Dhillon, Pradeep K. Ravikumar, and Mátyás A. Sustik. Sparse inverse covariance matrix estimation using quadratic approximation. In *Advances in Neural Information Processing Systems*, pages 2330–2338, 2011.

Jian Huang, Shuange Ma, Huiliang Xie, and Cun-Hui Zhang. A group bridge approach for variable selection. *Biometrika*, 96(2):339–355, 2009.

Jian Huang, Patrick Breheny, and Shuangge Ma. A selective review of group selection in high-dimensional models. *Statistical Science*, 27(4):481–499, 2012.

Wenxin Jiang and Martin A. Tanner. Gibbs posterior for variable selection in high-dimensional classification and data mining. *The Annals of Statistics*, 36(5):2207–2231, 2008.

David Kahle and Hadley Wickham. ggmap: Spatial visualization with ggplot2. *The R Journal*, 5(1):144–161, 2013.

Clifford Lam and Jianqing Fan. Sparsistency and rates of convergence in large covariance matrix estimation. *The Annals of Statistics*, 37(6B):4254–4278, 2009.

Wonyul Lee and Yufeng Liu. Joint estimation of multiple precision matrices with common structures. *Journal of Machine Learning Research*, 16(1):1035–1062, 2015.

Po-Ling Loh and Martin J. Wainwright. Support recovery without incoherence: A case for nonconvex regularization. *The Annals of Statistics*, 45(6):2455–2482, 2017.

Jing Ma and George Michailidis. Joint structural estimation of multiple graphical models. *Journal of Machine Learning Research*, 17(1):5777–5824, 2016.

Rahul Mazumder and Trevor Hastie. The graphical lasso: New insights and alternatives. *Electronic Journal of Statistics*, 6:2125–2149, 2012.

Abdolreza Mohammadi and Ernst C. Wit. Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Analysis*, 10(1):109–138, 2015.

James M. Ortega and Werner C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*, volume 30. Siam, 1970.

Christine Peterson, Francesco C. Stingo, and Marina Vannucci. Bayesian inference of multiple Gaussian graphical models. *Journal of the American Statistical Association*, 110 (509):159–174, 2015.

Pradeep Ravikumar, Martin J. Wainwright, Garvesh Raskutti, and Bin Yu. High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *arXiv preprint arXiv:0811.3628*, 2008.

Pradeep Ravikumar, Martin J. Wainwright, Garvesh Raskutti, and Bin Yu. High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980, 2011.

Veronika Ročková. Bayesian estimation of sparse signals with a continuous spike-and-slab prior. *The Annals of Statistics*, 46(1):401–437, 2018.

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

Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245, 2013.

Linda S. L. Tan, Ajay Jasra, Maria De Iorio, and Timothy M. D. Ebbels. Bayesian inference for multiple Gaussian graphical models with application to metabolic association networks. *The Annals of Applied Statistics*, 11(4):2222–2251, 2017.

Hao Wang and Sophia Li. Efficient Gaussian graphical model determination under G-Wishart prior distributions. *Electronic Journal of Statistics*, 6:168–198, 2012.

Xiaofan Xu and Malay Ghosh. Bayesian variable selection and estimation for group lasso. *Bayesian Analysis*, 10(4):909–936, 2015.

Sen Yang, Zhaosong Lu, Xiaotong Shen, Peter Wonka, and Jieping Ye. Fused multiple graphical lasso. *SIAM Journal on Optimization*, 25(2):916–943, 2015.

Yun Yang, Debdeep Pati, and Anirban Bhattacharya. α-variational inference with statistical guarantees. *arXiv preprint arXiv:1710.03266*, 2017.

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

Yuancheng Zhu and Rina Foygel Barber. The log-shift penalty for adaptive estimation of multiple Gaussian graphical models. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, pages 1153–1161, 2015.