Fused Multiple Graphical Lasso

Sen Yang[1], Zhisong Pan[2], Xiaotong Shen[3], Peter Wonka[1], Jieping Ye[1]
[1]Computer Science and Engineering, Arizona State University, USA
[2]School of Computer Science and Engineering,
Nanjing University of Aeronautics & Astronautics, P.R. China
[3]School of Statistics, University of Minnesota, USA

September 12, 2012

Abstract

In this paper, we consider the problem of estimating multiple graphical models simultaneously using the fused lasso penalty, which encourages adjacent graphs to share similar structures. A motivating example is the analysis of brain networks of Alzheimer's disease using neuroimaging data. Specifically, we may wish to estimate a brain network for the normal controls (NC), a brain network for the patients with mild cognitive impairment (MCI), and a brain network for Alzheimer's patients (AD). We expect the two brain networks for NC and MCI to share common structures but not to be identical to each other; similarly for the two brain networks for MCI and AD. The proposed formulation can be solved using a blockwise coordinate descent method. Our key technical contribution is to establish the necessary and sufficient condition for the graphs to be decomposable. Based on this key property, a simple screening rule is presented, which decomposes the large graphs into small subgraphs and allows an efficient estimation of multiple independent (small) subgraphs, dramatically reducing the computational cost. We perform experiments on both synthetic and real data; our results demonstrate the effectiveness and efficiency of the proposed approach.

1 Introduction

Undirected graphical models explore the relationships among a set of random variables through their joint distribution. The estimation of undirected graphical models has applications in many domains, such as computer vision, biology, and medicine. An instance is the analysis of gene expression data. As shown in many biological studies, genes tend to work in groups based on their biological functions, and there exist some regulatory relationships between genes [1]. Such biological knowledge can be represented as a graph, where nodes are the genes, and edges describe the regulatory relationships. Graphical models provide a useful tool for modeling these relationships, and can be used to explore gene activities. One of the most widely used graphical models is the Gaussian graphical model (GGM), which assumes the variables to be Gaussian distributed [2]. In the framework of GGM, the problem of learning a graph is equivalent to estimating the inverse of the covariance matrix (precision matrix), since the nonzero off-diagonal elements of the precision matrix represent edges in the graph [2].

In recent years many research efforts have focused on estimating the precision matrix and the corresponding graphical model. Meinshausen and Bühlmann [3] estimated edges for each node in the graph by fitting a lasso problem [4] using the remaining variables as predictors. Yuan and Lin [5] and Banerjee et al. [2] proposed a penalized maximum likelihood approach using ℓ_1 regularization, and used interior point optimization to estimate the sparse precision matrix. Friedman et al. [6] introduced a blockwise coordinate descent method to solve the same problem, referred to as Graphical lasso (GLasso). Huang et al. [7] derived the monotone property of GLasso. Liu et al. [8] introduced a stability-based method for choosing the regularization parameters for GLasso. Although GLasso is faster than previous approaches, it usually fails to converge with warm-starts. To resolve this issue, Mazumder and Hastie [9] proposed a new algorithm called DP-GLasso, each step of which is a box-constrained QP problem. The main challenge of estimating a sparse precision matrix is its intensive computation. Witten et al. [10] and Mazumder and Hastie [11] independently derived a simple screening rule, achieving great computational gain when the regularization parameter is large. However, these formulations assume that observations are independently drawn from a single Gaussian distribution. In many applications the observations may be drawn from multiple Gaussian distributions; in this case, multiple graphical models need to be estimated.

There is some recent work on the estimation of multiple precision matrices. Guo et al. [12] proposed a method to jointly estimate multiple graphical models using a hierarchical penalty. However, this method is not convex. Danaher et al. [13] estimated multiple precision matrices simultaneously using a pairwise fused penalty and grouping penalty. The generalized gradient method was used to solve the problem, but it required computing the inverse of precision matrices and checking the positive definiteness of precision matrices at each iteration. A screening rule for the two graph case was also proposed in [13]. However, it is not clear whether the screening rule can be extended to the more general case with more than two graphs, which is the case in brain network modeling. Time-varying graphical models were also studied by Zhu et al. [14], and Kolar et al. [15, 16].

In this paper, we consider the problem of estimating multiple graphical models by maximizing a penalized log likelihood with ℓ_1 and fused regularization as in [13]. The ℓ_1 regularization yields a sparse solution, and the fused regularization encourages adjacent graphs to be similar. A motivating example is the modeling of brain networks for Alzheimer's disease using neuroimaging data such as Positron emission tomography (PET). In this case, we want to estimate graphical models for three groups: normal controls (NC), patients of mild cognitive impairment (MCI), and Alzheimer's patients (AD). These networks are expected to share some common connections, but they are not identical. Furthermore, the networks are expected to evolve over time, in the order of disease progression from NC to MCI to AD. Estimating the graphical models separately fails to exploit the common structures among them. It is thus desirable to jointly estimate the three networks (graphs). We employ the blockwise coordinate descent method to solve the fused multiple graphical lasso (FMGL), where each step is solved by the accelerated gradient method [17]. Our key technical contribution is to establish the necessary and sufficient condition for the FMGL solution to be block diagonal. Based on this key property of FMGL, we develop a screening rule which enables the efficient estimation of large multiple precision matrices. Specifically, we derive a set of necessary conditions for the solution of FMGL to be block diagonal. We prove that these conditions are sufficient when $K \leq 3$ (K is the number of graphs to be estimated). Our simulation studies strongly indicate that these conditions are also sufficient for any $K > 3$ as well. Our results significantly extend the recent work presented in [13]. We conduct experiments on both synthetic and real data; our results demonstrate the effectiveness and efficiency of the proposed approach.

The rest of the paper is organized as follows. We introduce the fused multiple graphical lasso formulation in Section 2. The screening rule is presented in Section 3. The experimental results are shown in Section 4. We conclude the paper in Section 5.

2 Fused multiple graphical lasso

Assume we are given K data sets, $\mathbf{X}^{(k)} \in \mathcal{R}^{n_k \times p},\ k = 1, \ldots, K$ with $K \geq 2$, where n_k is the number of samples, and p is the number of features. The p features are common for all K data sets, and all $\sum_{k=1}^{K} n_k$ samples are independent. Furthermore, the samples within each data set $\mathbf{X}^{(k)}$ are identically distributed with a p-variate Gaussian distribution with zero mean and positive definite covariance matrix $\mathbf{\Sigma}^{(k)}$, and there are many conditionally independent pairs of features, i.e. the precision matrix $\mathbf{\Theta}^{(k)} = (\mathbf{\Sigma}^{(k)})^{-1}$ should be sparse. For notational simplicity, we assume that $n_1 = \cdots = n_K = n$. Denote the sample covariance matrix for each data set $\mathbf{X}^{(k)}$ as $\mathbf{S}^{(k)}$ such that $\mathbf{S}^{(k)} = \frac{1}{n}(\mathbf{X}^{(k)})^T\mathbf{X}^{(k)}$, and $\mathbf{\Theta} = \{\mathbf{\Theta}^{(1)}, \ldots, \mathbf{\Theta}^{(K)}\}$. Then the negative log likelihood for the data takes the form of

$$\mathcal{L}(\mathbf{\Theta}) = \sum_{k=1}^{K} \Big(-\log det(\mathbf{\Theta}^{(k)}) + tr(\mathbf{S}^{(k)}\mathbf{\Theta}^{(k)}) \Big). \tag{1}$$

Minimizing Eq.(1) leads to the maximum likelihood estimate (MLE) $\hat{\mathbf{\Theta}}^{(k)} = (\mathbf{S}^{(k)})^{-1}$. However, MLE fails in the high-dimensional setting. In this setting, the sample size n is less than the number of features p, thus $\mathbf{S}^{(k)}$ is singular. Furthermore, the MLE is unlikely to be sparse. The ℓ_1 regularization has been employed to induce sparsity, resulting in the sparse inverse covariance estimation [2, 5, 6]. In this paper, we employ both the ℓ_1 regularization and the fused regularization for simultaneously estimating multiple graphs. The ℓ_1 regularization leads to a sparse solution, and the fused penalty encourages $\mathbf{\Theta}^{(k)}$ to be similar to its neighbors. Mathematically, we solve the following formulation:

$$\min_{\mathbf{\Theta}^{(k)} \succ 0, k=1 \ldots K} \sum_{k=1}^{K} \Big(-\log det(\mathbf{\Theta}^{(k)}) + tr(\mathbf{S}^{(k)}\mathbf{\Theta}^{(k)}) \Big) + P(\mathbf{\Theta}), \tag{2}$$

where

$$P(\mathbf{\Theta}) = \lambda_1 \sum_{k=1}^{K} \sum_{i \neq j} |\theta_{ij}^{(k)}| + \lambda_2 \sum_{k=1}^{K-1} \sum_{i \neq j} |\theta_{ij}^{(k)} - \theta_{ij}^{(k+1)}|,$$

λ_1 and λ_2 are nonnegative regularization parameters. The algorithm for solving Eq. (2) and the corresponding complexity analysis are given in Appendix A.

3 The screening rule for fused multiple graphical lasso

Witten et al. [10] and Mazumder and Hastie [11] independently derived a necessary and sufficient condition for the solution of a single graphical lasso to be block diagonal. A simple screening test can be used to identify the blocks, thus the original graphical lasso problem can be decomposed into several smaller problems. When the number of blocks is large, it can achieve massive computational gain. Danaher et al. [13] developed a similar necessary and sufficient condition for fused graphical lasso with two graphs. However, it remains a challenge to derive the necessary and sufficient condition for the solution of fused multiple graphical lasso to be block diagonal for $K > 2$ graphs.

In this section, we first present a theorem demonstrating that FMGL can be decomposable once its solution is block diagonal. Then we derive a set of necessary conditions for the solution of FMGL to be block diagonal. We also prove that these conditions are sufficient when $K = 3$. We conjecture that these conditions are sufficient for any $K > 3$ as well (see the discussion in Section 4.2.2).

Let $C_1, \ldots, C_L$ be a partition of the p features into L nonoverlapping sets, with $C_l \cap C_{l'} = \emptyset,\ \forall l \neq l'$ and $\bigcup_{l=1}^{L} C_l = \{1, \ldots, p\}$. Then we have the following result [13]:

Theorem 1. *Suppose that the FMGL solution $\hat{\mathbf{\Theta}}$ is block diagonal with L known blocks only consisting of features in the set $C_l,\ l = 1, \ldots, L$, i.e. each estimation precision matrix takes the form*

$$\hat{\mathbf{\Theta}}^{(k)} = \begin{pmatrix} \hat{\mathbf{\Theta}}_1^{(k)} & & \\ & \ddots & \\ & & \hat{\mathbf{\Theta}}_L^{(k)} \end{pmatrix}, k = 1, \ldots, K,$$

then Eq. (2) can be solved by applying FMGL on just the corresponding set of features:

$$\hat{\mathbf{\Theta}}_l = \arg\min_{\mathbf{\Theta}_l \succ 0} \sum_{k=1}^{K} \left(-\log det(\mathbf{\Theta}_l^{(k)}) + tr(\mathbf{S}_l^{(k)} \mathbf{\Theta}_l^{(k)}) \right) + P(\mathbf{\Theta}_l),\ l = 1, \ldots, L,$$

where $\hat{\mathbf{\Theta}}_l^{(k)}$ and $\mathbf{S}_l^{(k)}$ are the corresponding $|C_l| \times |C_l|$ symmetric submatrices of $\hat{\mathbf{\Theta}}^{(k)}$ and $\hat{\mathbf{S}}^{(k)}$.

Theorem 1 can be directly derived from Eq.(2), since $det(\hat{\mathbf{\Theta}}^{(k)}) = \prod_{l=1}^{L} det(\hat{\mathbf{\Theta}}_l^{(k)})$, $tr(\mathbf{S}^{(k)} \hat{\mathbf{\Theta}}^{(k)}) = \sum_{l=1}^{L} tr(\mathbf{S}_l^{(k)} \hat{\mathbf{\Theta}}_l^{(k)})$, and $P(\hat{\mathbf{\Theta}}) = \sum_{l=1}^{L} P(\hat{\mathbf{\Theta}}_l)$. The key is how to efficiently identify the block structures. We address this problem in the remaining part of this section.

Theorem 2. *The following set of conditions are necessary for the FMGL solution $\hat{\mathbf{\Theta}}^{(k)}, k = 1, \ldots, K$ to be block diagonal with L known blocks $C_l, l = 1, \ldots, L$:*

$$\begin{aligned}
&|\sum_{k=1}^{t} s_{ij}^{(k)}| \leq t\lambda_1 + \lambda_2,\ 1 \leq t \leq K-1, \\
&|\sum_{k=t_1}^{t_2} s_{ij}^{(k)}| \leq (t_2 - t_1 + 1)\lambda_1 + 2\lambda_2,\ 2 \leq t_1 \leq t_2 \leq K-1, \\
&|\sum_{k=t}^{K} s_{ij}^{(k)}| \leq (K - t + 1)\lambda_1 + \lambda_2,\ 2 \leq t \leq K, \\
&|\sum_{k=1}^{K} s_{ij}^{(k)}| \leq K\lambda_1,
\end{aligned} \tag{3}$$

for $i \in C_l, j \in C_{l'}, l \neq l'$.

Proof. Denote the inverse of $\mathbf{\Theta}^{(k)}$ as $\mathbf{W}^{(k)}$, for $k = 1, \ldots, K$. For the diagonal elements of $\mathbf{\Theta}^{(k)}$, the Karush-Kuhn-Tucker (KKT) optimality conditions [18] for Eq.(2) are

$$-w_{ii}^{(k)} + s_{ii}^{(k)} = 0,\ 1 \leq k \leq K.$$

Thus, the optimal $\hat{w}_{ii}^{(k)}$ can be directly computed as $s_{ii}^{(k)}$. For the off-diagonal elements of $\mathbf{\Theta}^{(k)}$, the KKT conditions for Eq.(2) can be written as

$$\begin{aligned}
&-w_{ij}^{(1)} + s_{ij}^{(1)} + \lambda_1\gamma_{ij}^{(1)} + \lambda_2\upsilon_{ij}^{(1,2)} = 0 \\
&-w_{ij}^{(k)} + s_{ij}^{(k)} + \lambda_1\gamma_{ij}^{(k)} + \lambda_2(-\upsilon_{ij}^{(k-1,k)} + \upsilon_{ij}^{(k,k+1)}) = 0, \text{for } 2 \leq k \leq K-1 \\
&-w_{ij}^{(K)} + s_{ij}^{(K)} + \lambda_1\gamma_{ij}^{(K)} - \lambda_2\upsilon_{ij}^{(K-1,K)} = 0,
\end{aligned} \tag{4}$$

where $\gamma_{ij}^{(k)}$ is the subgradient of $|\theta_{ij}^{(k)}|$: $\gamma_{ij}^{(k)} = 1$ if $\theta_{ij}^{(k)} > 0$, $\gamma_{ij}^{(k)} = -1$ if $\theta_{ij}^{(k)} < 0$, and $\gamma_{ij}^{(k)} \in [-1, 1]$ if $\theta_{ij}^{(k)} = 0$; $\upsilon_{ij}^{(k,k+1)}$ is the subgradient of $|\theta_{ij}^{(k)} - \theta_{ij}^{(k+1)}|$ with respect to $\theta_{ij}^{(k)}$: $\upsilon_{ij}^{(k,k+1)} = 1$ if $\theta_{ij}^{(k)} > \theta_{ij}^{(k+1)}$, $\upsilon_{ij}^{(k,k+1)} = -1$ if $\theta_{ij}^{(k)} < \theta_{ij}^{(k+1)}$, and $\upsilon_{ij}^{(k,k+1)} \in [-1, 1]$ if $\theta_{ij}^{(k)} = \theta_{ij}^{(k+1)}$.

Note that $\hat{\mathbf{W}}^{(k)}$ has the same block diagonal structure as $\hat{\mathbf{\Theta}}^{(k)}$, thus $\hat{\theta}_{ij}^{(k)} = \hat{w}_{ij}^{(k)} = 0$ for $i \in C_l, j \in C_{l'}, l \neq l'$. Then Eq.(4) can be rewritten as

$$\begin{aligned}
&s_{ij}^{(1)} + \lambda_1\gamma_{ij}^{(1)} + \lambda_2\upsilon_{ij}^{(1,2)} = 0 \\
&s_{ij}^{(k)} + \lambda_1\gamma_{ij}^{(k)} + \lambda_2(-\upsilon_{ij}^{(k-1,k)} + \upsilon_{ij}^{(k,k+1)}) = 0, \text{for } 2 \leq k \leq K-1 \\
&s_{ij}^{(K)} + \lambda_1\gamma_{ij}^{(K)} - \lambda_2\upsilon_{ij}^{(K-1,K)} = 0.
\end{aligned} \tag{5}$$

As a result, we have $\sum_{k=1}^{t} s_{ij}^{(k)} = -\lambda_1 \sum_{k=1}^{t} \gamma_{ij}^{(t)} - \lambda_2\upsilon_{ij}^{(t,t+1)}$, for $1 \leq t \leq K-1$, implying $|\sum_{k=1}^{t} s_{ij}^{(k)}| \leq t\lambda_1 + \lambda_2$, for $1 \leq t \leq K-1$ since $\gamma_{ij}^{(1)}, \ldots, \gamma_{ij}^{(K)}, \upsilon_{ij}^{(1,2)}, \ldots, \upsilon_{ij}^{(K-1,K)} \in [-1, 1]$. Similarly, we can prove the other three conditions. □

Next, we show that the conditions (3) in Theorem 2 are also sufficient for $K = 2, 3$. Danaher et al. [13] have proved the sufficiency when $K = 2$. We here give a more concise and simpler proof for $K = 2$. More importantly, our proof can be easily extended to the case when $K = 3$. Before proving the sufficiency, we first prove the following lemmas:

Lemma 1. *Suppose $|\alpha + \beta| \leq (t_1 + 1)\lambda_1$, $|\alpha| \leq \lambda_1 + t_2\lambda_2$, and $|\beta| \leq t_1\lambda_1 + t_2\lambda_2$ with $t_1, t_2 > 0$, the following three intervals intersect: (1) $|a| \leq t_2\lambda_2$; (2) $-\lambda_1 + \alpha \leq a \leq \lambda_1 + \alpha$; (3) $-t_1\lambda_1 - \beta \leq a \leq t_1\lambda_1 - \beta$.*

Proof. We first prove by contradiction that the first and second intervals intersect. If they do not intersect, we must have $|\alpha| > \lambda_1 + t_2\lambda_2$, which contradicts with the condition $|\alpha| \leq \lambda_1 + t_2\lambda_2$. Similarly, the first and third intervals intersect. Since $|\alpha + \beta| \leq (t_1 + 1)\lambda_1$, we have $\lambda_1 + \alpha \geq -t_1\lambda_1 - \beta$ and $t_1\lambda_1 - \beta \geq -\lambda_1 + \alpha$, indicating that the second and third intervals intersect. Thus, the three intervals intersect. □

Lemma 2. *Suppose $|y_1 + y_2 + y_3| \leq 3\lambda_1$. Then the following region*

$$\begin{aligned}
&\max\{-\lambda_1 - y_1, -2\lambda_1 + y_2 + y_3\} \leq a_1 \leq \min\{\lambda_1 - y_1, 2\lambda_1 + y_2 + y_3\}, \\
&\max\{-\lambda_1 + y_3, -2\lambda_1 - y_1 - y_2\} \leq a_2 \leq \min\{\lambda_1 + y_3, 2\lambda_1 - y_1 - y_2\}
\end{aligned} \tag{6}$$

is a square (a single point is considered as a special case of a square). Let B and C be the lower-left and upper-right vertices of this square, then the diagonal BC belongs to the region

$$\max\{-\lambda_1 + y_2, -2\lambda_1 - y_1 - y_3\} \leq a_1 - a_2 \leq \min\{\lambda_1 + y_2, 2\lambda_1 - y_1 - y_3\}. \tag{7}$$

The proof is given in Appendix B.

Lemma 3. *Under the conditions (3) and $K = 2, 3$, the linear system in Eq.(5) has a solution such that $\gamma_{ij}^{(1)}, \ldots, \gamma_{ij}^{(K)}, \upsilon_{ij}^{(1,2)}, \ldots, \upsilon_{ij}^{(K-1,K)} \in [-1, 1]$.*

Proof. Eq.(5) can be rewritten as a matrix form $\mathbf{A}\mathbf{x}+\mathbf{y}=\mathbf{0}$, where $\mathbf{A}=[\lambda_2\mathbf{H}_K,\ \lambda_1\mathbf{I}_{K\times K}]$, $\mathbf{x}=[\upsilon_{ij}^{(1,2)},\ldots,\upsilon_{ij}^{(K-1,K)},\gamma_{ij}^{(1)},\ldots,\gamma_{ij}^{(K)}]^T$, $\mathbf{y}=[s_{ij}^{(1)},\ldots,s_{ij}^{(K)}]^T$, and

$$\mathbf{H}_K=\begin{pmatrix}1&&&&\\-1&1&&&\\&-1&&&\\&&\ddots&&\\&&&1\\&&&-1\end{pmatrix}_{K\times(K-1)}$$

Note that one solution of $\mathbf{A}\mathbf{x}+\mathbf{y}=\mathbf{0}$ is $\mathbf{x}^*=[\mathbf{0}_{1\times(K-1)},-\mathbf{y}^T/\lambda_1]^T$, and the null space of $\mathbf{A}$ is

$$\mathbf{B}=Null(\mathbf{A})=\begin{pmatrix}\mathbf{I}_{(K-1)\times(K-1)}\\-\frac{\lambda_2}{\lambda_1}\mathbf{H}_K\end{pmatrix},$$

thus the general solution of $\mathbf{A}\mathbf{x}+\mathbf{y}=\mathbf{0}$ takes the form of $\tilde{\mathbf{x}}=\mathbf{B}\mathbf{a}+\mathbf{x}^*$ with an arbitrary $\mathbf{a}\in\mathcal{R}^{K-1}$. Next we need to show that there exists a solution $\hat{\mathbf{x}}$ such that $\|\hat{\mathbf{x}}\|_\infty\le 1$ under the conditions (3) when $K=2,3$. For notational simplicity, we use y_k to represent $s_{ij}^{(k)}$ in the proof.

***K*=2**: The solution $\tilde{\mathbf{x}}$ takes the form of

$$\tilde{\mathbf{x}}=\begin{pmatrix}a\\\frac{-\lambda_2 a-y_1}{\lambda_1}\\\frac{\lambda_2 a-y_2}{\lambda_1}\end{pmatrix}.$$

$\|\tilde{\mathbf{x}}\|_\infty\le 1$ can be expressed as $|a|\le 1$, $\frac{-\lambda_1-y_1}{\lambda_2}\le a\le\frac{\lambda_1-y_1}{\lambda_2}$, and $\frac{-\lambda_1+y_2}{\lambda_2}\le a\le\frac{\lambda_1+y_2}{\lambda_2}$. If these three intervals intersect, there exists $\hat{\mathbf{x}}$ such that $\|\hat{\mathbf{x}}\|_\infty\le 1$. The problem of finding the desired $\hat{\mathbf{x}}$ is therefore transformed to identifying the intersection of the above set of conditions. According to Lemma 1, these three intervals intersect.

***K*=3**: Following the same idea, we can obtain the conditions for $K=3$:

$$|a_1|\le 1,\ |a_2|\le 1, \tag{8}$$

and

$$-\frac{\lambda_1+y_1}{\lambda_2}\le a_1\le\frac{\lambda_1-y_1}{\lambda_2} \tag{9}$$

$$\frac{-\lambda_1+y_2}{\lambda_2}\le a_1-a_2\le\frac{\lambda_1+y_2}{\lambda_2} \tag{10}$$

$$\frac{-\lambda_1+y_3}{\lambda_2}\le a_2\le\frac{\lambda_1+y_3}{\lambda_2}. \tag{11}$$

Consider the summation of Eqs.(10) and (11), we can obtain

$$\frac{-2\lambda_1+y_2+y_3}{\lambda_2}\le a_1\le\frac{2\lambda_1+y_2+y_3}{\lambda_2}. \tag{12}$$

The refined feasible region of a_1 by combining Eqs.(9) with (12) is given by

$$\frac{\max\{-\lambda_1-y_1,-2\lambda_1+y_2+y_3\}}{\lambda_2}\le a_1\le\frac{\min\{\lambda_1-y_1,2\lambda_1+y_2+y_3\}}{\lambda_2}. \tag{13}$$

Based on $|y_1|\le\lambda_1+\lambda_2$, $|y_2+y_3|\le 2\lambda_1+\lambda_2$, $|y_1+y_2+y_3|\le 3\lambda_1$ as well as Lemma 1, we can show that Eqs.(9), (12), and $|a_1|\le 1$ intersect. Thus, Eq. (13) also intersects with $|a_1|\le 1$.

Similarly, we can also obtain the refined feasible region of a_2 that intersects with $|a_2|\le 1$

$$\frac{\max\{-\lambda_1+y_3,-2\lambda_1-y_1-y_2\}}{\lambda_2}\le a_2\le\frac{\min\{\lambda_1+y_3,2\lambda_1-y_1-y_2\}}{\lambda_2}, \tag{14}$$

and the refined feasible region of a_1-a_2,

$$\frac{\max\{-\lambda_1+y_2,-2\lambda_1-y_1-y_3\}}{\lambda_2}\le a_1-a_2\le\frac{\min\{\lambda_1+y_2,2\lambda_1-y_1-y_3\}}{\lambda_2}, \tag{15}$$

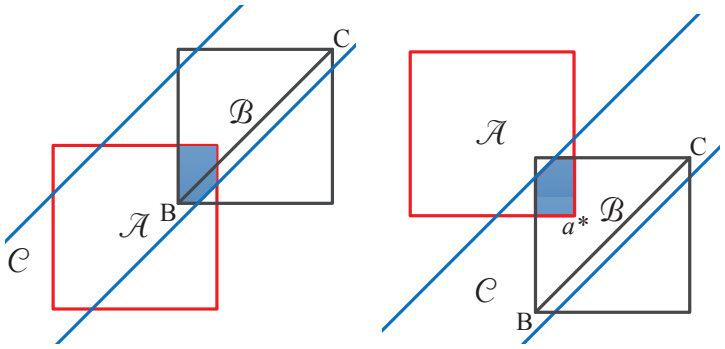


Figure 1: Illustration of the regions $\mathcal{A}, \mathcal{B}$, and $\mathcal{C}$: Red square ($\mathcal{A}$), black square ($\mathcal{B}$), the band between blues lines ($\mathcal{C}$), and the blue region ($\mathcal{A} \cap \mathcal{B} \cap \mathcal{C}$).

which intersects with the region of $|a_1| \leq 1, |a_2| \leq 1$.

Let $\mathcal{A}$ represent the feasible region of Eq. (8), $\mathcal{B}$ be the feasible region of Eqs.(13) and (14), and $\mathcal{C}$ be the feasible region of Eq.(15) (see Figure 1 for illustration). We have shown that $\mathcal{A} \cap \mathcal{B} \neq \emptyset$ and $\mathcal{A} \cap \mathcal{C} \neq \emptyset$. Next, we will show $\mathcal{A} \cap \mathcal{B} \cap \mathcal{C} \neq \emptyset$.

According to Lemma 2, $\mathcal{B}$ is a square. Let B and C be the lower-left and upper-right vertices of $\mathcal{B}$, then the diagonal BC of $\mathcal{B}$ belongs to $\mathcal{C}$. Denote $\mathbf{a}^*$ as the closest point in $\mathcal{A}$ to the diagonal BC. $\mathbf{a}^*$ belongs to $\mathcal{B}$, since $\mathcal{A} \cap \mathcal{B} \neq \emptyset$. We can easily prove that $\mathbf{a}^* \in \mathcal{A} \cap \mathcal{B} \cap \mathcal{C}$. If $\mathbf{a}^*$ is in the diagonal BC, then $\mathbf{a}^* \in \mathcal{C}$ (see Figure 1(a)). If not, we prove it by contradiction. Suppose $\mathbf{a}^* \notin \mathcal{C}$, there exists one closest point $\hat{\mathbf{a}} \in \mathcal{A} \cap \mathcal{C}$ to the diagonal BC. The distance from $\hat{\mathbf{a}}$ to the diagonal must be less than that of $\mathbf{a}^*$, which contradicts with the condition that $\mathbf{a}^*$ is the closest point to the diagonal BC (see Figure 1(b)).

Thus, we have proved that $\mathcal{A} \cap \mathcal{B} \cap \mathcal{C} \neq \emptyset$, and there exists a solution $\hat{\mathbf{x}}$ such that $\|\hat{\mathbf{x}}\|_\infty \leq 1$. □

Now we are ready to prove the sufficiency, stated in the following theorem:

Theorem 3. *The conditions* (3) *are sufficient for the FMGL solution* $\hat{\mathbf{\Theta}}^{(k)}, k = 1, \ldots, K$ *to be block diagonal with* L *known blocks* $C_l, l = 1, \ldots, L$ *when* $K = 2, 3$.

Proof. We construct matrices $\hat{\mathbf{\Theta}}^{(k)}$ with the block diagonal structure $C_l, l = 1, \ldots, L$ based on the conditions (3), and show that they are a solution to FMGL. The l-th block diagonal elements $\hat{\mathbf{\Theta}}_l^{(k)}, k = 1, \ldots, K$ are obtained by performing FMGL on $\mathbf{S}_l^{(k)}$. According to Theorem 1, the solution $\hat{\mathbf{\Theta}}_l^{(k)}, k = 1, \ldots, K$ satisfies the KKT condition. According to Lemma 3, the linear system in Eq. (5) has a solution $\hat{\mathbf{x}}$ such that $\|\hat{\mathbf{x}}\|_\infty \leq 1$ for $i \in C_l,\ j \in C_{l'}, l \neq l'$ under the conditions (3). Obviously, the choice $\hat{\mathbf{w}}_{ij}^{(k)} = \hat{\theta}_{ij}^{(k)} = 0$ for $i \in C_l,\ j \in C_{l'}, l \neq l'$ satisfies the KKT condition. Hence, $\hat{\mathbf{\Theta}}^{(k)}, k = 1, \ldots, K$ form a solution to FMGL. □

According to Theorem 2 and Theorem 3, the conditions (3) can be used as screening rule to identify the block structure of the FMGL solution. The steps about how to use the conditions (3) are standard [10, 11]:

1. Construct an adjacency matrix $\mathbf{E} = \mathbf{I}_{p \times p}$. Set $E_{ij} = E_{ji} = 0$ if $s_{ij}^{(k)}, k = 1, \ldots, K$ satisfy the conditions (3). Otherwise, set $E_{ij} = E_{ji} = 1$.

2. Identify the connected components of $\mathbf{E}$. Note that the connected components are the partition of the p features into nonoverlapping sets.

An obvious consequence of Theorem 2 and Theorem 3 is that the off-diagonal elements in the i-th row and column are zeros, if $s_{ii}^{(k)}, k = 1, 2, 3$ satisfy the conditions (3). In addition, $\theta_{ii}^{(k)}$ and $w_{ii}^{(k)}$ can be directly computed as $1/s_{ii}^{(k)}$ and $s_{ii}^{(k)}$.

4 Experimental results

In this section, we evaluate the proposed algorithm and screening rule on synthetic datasets and two real datasets: ADHD-200 [19] and FDG-PET images [20]. The experiments are performed on a PC with dual-core Intel 3.0GHz CPU and 4GB memory. The code is written in C.

4.1 Convergence

To examine the convergence rate of the proposed method, we randomly generate 3 graphs with 10% sparsity and $p = 500$. The matrices $\tilde{\mathbf{S}}^{(k)}, k = 1, 2, 3$ are constructed in the following way: the diagonal elements of $\tilde{\mathbf{S}}^{(k)}$ are set to 0.65, the off-diagonal elements are selectively set to 0.65 based on the corresponding graphs, and the remaining ones are set to 0. A noise term $0.35(Q^{(k)})^T Q^{(k)}$ is added to $\tilde{\mathbf{S}}^{(k)}$, where $Q^{(k)}$ is a $p \times p$ matrix with standard Gaussian distribution. The sample covariance matrix $\mathbf{S}^{(k)} = \tilde{\mathbf{S}}^{(k)} + 0.35(Q^{(k)})^T Q^{(k)}$. $Q^{(k)}$ is standardized so that the diagonal elements of $\mathbf{S}^{(k)}$ are 1. λ_1 varies from 0.25 to 0.45 with a step size of 0.1, and λ_2 is set to 0.1. From Figure 2, we can observe that FMGL usually converges within 20 rounds (update all p columns/rows). When the regularization parameter value is small, FMGL needs more rounds to converge.

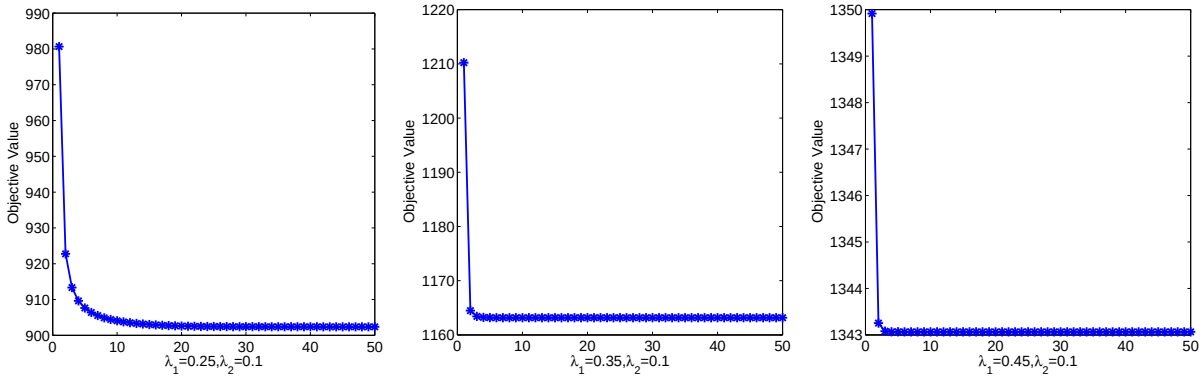


Figure 2: Convergence curves of FMGL. λ_1 is set to 0.25,0.35,0.45, and λ_2 is set to 0.1.

4.2 Simulation

4.2.1 Screening rule

The synthetic covariance matrices are generated as follows. A block diagonal matrix $\tilde{\mathbf{S}}$ with L blocks is created, and each block is of size $(p/L)\times(p/L)$ with all ones. The sample covariance matrices are generated as $\mathbf{S}^{(k)} = 0.5\tilde{\mathbf{S}} + 0.5(Q^{(k)})^T Q^{(k)}$, where $Q^{(k)}$ is a $p \times p$ matrix with standard Gaussian distribution. To make sure that the solution has L blocks, we vary λ_1 from 0.2 to 0.5 with a step size of 0.05, and fix λ_2 to 0.2. The convergence criterion is set to 1e-5, and the maximal iteration number is set to 1000. We use the speedup rate t_o/t_s and the error $|f_o - f_s|$ to measure the performance of the screening rule, where t_o, t_s are the computational times without and with the screening rule, and f_o, f_s are the objective values without and with the screening rule. The results with varying p are shown in Figure 3. As shown in Figure 3, the screening rule can achieve great computational gain. Since the complexity of FMGL is $O(Kp^3)$, the speedup rate for a FMGL solution with L same size blocks is $O(L^2)$. It can be observed that the speedup rate varies from 10 to 55 for $L = 5$, and from 25 to 320 for $L = 10$. We can also find that the speedup rate decreases when sparsity increases, and increases when p increases.

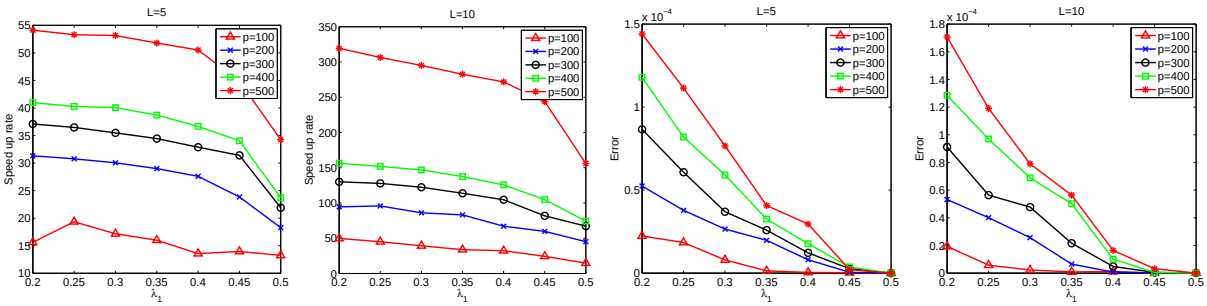


Figure 3: The speedup rate and the error without and with the screening rule.

4.2.2 Sufficiency when $K \geq 3$

We conduct simulations to examine whether the conditions (3) can guarantee that there exists a solution to the constrained linear system $\mathbf{Ax} + \mathbf{y} = \mathbf{0}$, $s.t.\ \|\mathbf{x}\|_\infty \leq 1$ when $K \geq 3$. The first and last elements of $\mathbf{y}$ are uniformly drawn from $[-\lambda_1 - \lambda_2, \lambda_1 + \lambda_2]$, and the remaining ones are uniformly drawn from $[-\lambda_1 - 2\lambda_2, \lambda_1 + 2\lambda_2]$. If the error $\|\mathbf{Ax} + \mathbf{y}\|_2 >$1e-8, a counterexample is found. We perform 2 million replications for $K = 3, 4, \ldots, 10, 15, 20$ respectively. About half of the replications give a $\mathbf{y}$ satisfying the conditions (3). For these $\mathbf{y}$, we minimize $\|\mathbf{Ax} + \mathbf{y}\|_2$ subject to the constraint using the gradient method. No counterexample is found, implying that the conditions (3) are likely sufficient for any $K > 3$.

4.2.3 Stability

We conduct experiments to demonstrate the effectiveness of FMGL. The synthetic sparse precision matrices are generated in the following way: we set the first precision matrix $\mathbf{\Theta}^{(1)}$ as $0.25I_{p\times p}$, where $p = 100$. When adding an edge (i, j) in the graph, we add σ to $\theta^{(1)}_{ii}$ and $\theta^{(1)}_{jj}$, and subtract σ from $\theta^{(1)}_{ij}$ and $\theta^{(1)}_{ji}$ to keep the positive definiteness of $\mathbf{\Theta}^{(1)}$, where σ is uniformly drawn from $[0.1, 0.3]$. When deleting an edge (i, j) from the graph, we reverse the above steps with $\sigma = \theta^{(1)}_{ij}$. We randomly assign 200 edges for $\mathbf{\Theta}^{(1)}$. $\mathbf{\Theta}^{(2)}$ is obtained by adding 25 edges and deleting 25 different edges from $\mathbf{\Theta}^{(1)}$. $\mathbf{\Theta}^{(3)}$ is obtained from $\mathbf{\Theta}^{(2)}$ in the same way. For each precision matrix, we randomly draw n samples from the Gaussian distribution with the corresponding precision matrix, where n varies from 40 to 200 with a step of 20. We perform 500 replications for each n. For each n, λ_2 is fixed to 0.08, and λ_1 is adjusted to make sure that the edge number is about 200. The accuracy n_d/n_g is used to measure the performance of FMGL and GLasso, where n_d is the number of true edges detected by FGML and GLasso, and n_g is the number of true edges. The results are shown in Figure 4. We can see from the figure that FMGL achieves higher accuracies, demonstrating the effectiveness of FMGL for learning multiple graphical models.

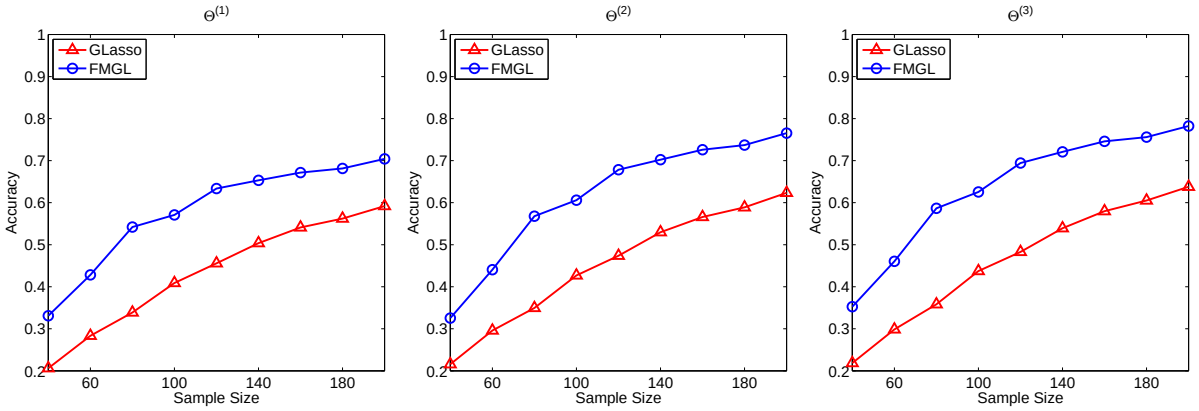


Figure 4: Comparison of FMGL and GLasso in detecting true edges.

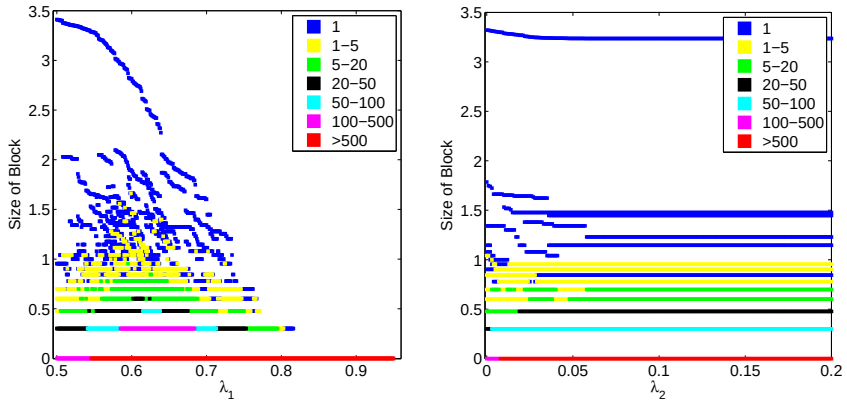
4.3 Real data

4.3.1 ADHD-200

The Attention Deficit Hyperactivity Disorder (ADHD) affects at least 5-10% of school-age children with annual costs exceeding 36 billion/year in the United States. The ADHD-200 project has released resting-state functional magnetic resonance images (fMRI) of 491 typically developing children and 285 ADHD children, aiming to encourage the research on ADHD. The data used in this experiment is the preprocessed data using the NIAK pipeline downloaded from neurobureau [21]. More details about the preprocessing strategy can be found in the same website. The dataset we choose includes 116 typically developing children (TDC), 29 ADHD-Combined (ADHD-C), and 49 ADHD-Inattentive (ADHD-I). There are 231 time series and 2834 brain regions for each subject. We want to estimate the graphs of the three groups simultaneously. The sample covariance matrix is computed using all data from the same group. Since the number of brain regions p is 2834, obtaining the precision matrices is computationally intensive. We use this data to test the effectiveness of the proposed screening rule. λ_1 and λ_2 are set to 0.6 and 0.015. The convergence criterion is 1e-5. The computational time is about 4.13 hours without screening, and

172 seconds with screening, demonstrating the superiority of the screening rule. The obtained solution has 1443 blocks. The largest one including 634 features is given in Appendix C.

The block structures of the FMGL solution are the same as those identified by the screening rule. The screening rule can be used to analyze the rough structures of the graphs. The cost of identifying blocks using the screening rule is negligible compared to that of estimating the graphs. For high-dimensional data such as ADHD-200, it is practical to use the screening rule to identify the block structure before estimating the large graphs. We use the screening rule to identify block structures on ADHD-200 data with varying λ_1 and λ_2. The size distribution is shown in Figure 5. We can observe that the number of blocks increases, and the size of blocks deceases when the regularization parameter value increases.



(a) (b)

Figure 5: The size distribution of blocks (in logarithmic scale) identified by the proposed screening rule. The color represents the number of blocks of a specified size. (a): λ_1 varies from 0.5 to 0.95 with λ_2 fixed to 0.015. (b): λ_2 varies from 0 to 0.2 with λ_1 fixed to 0.55.

4.3.2 FDG-PET

In this experiment, we use FDG-PET images from 74 Alzheimer's disease (AD), 172 mild cognitive impairment (MCI), and 81 normal control (NC) subjects downloaded from the Alzheimer's disease neuroimaging initiative (ADNI) database. The different regions of the whole brain volume can be represented by 116 anatomical volumes of interest (AVOI), defined by Automated Anatomical Labeling (AAL) [22]. Then we extracted data from each of the 116 AVOIs, and derived the average of each AVOI for each subject. The 116 AVOIs can be categorized into 10 groups: prefrontal lobe, other parts of the frontal lobe, parietal lobe, occipital lobe, thalamus, insula, temporal lobe, corpus striatum, cerebellum, and vermis. More details about the categories can be found in [23, 22]. We remove two small groups (thalamus and insula) containing only 4 AVOIs in our experiments.

To examine whether FMGL can effectively utilize the information of common structures, we randomly select g percent samples from each group, where g varies from 20 to 100 with a step size of 10. For each g, λ_2 is fixed to 0.1, and λ_1 is adjusted to make sure the number of edges in each group is about the same. We perform 500 replications for each g. The edges with probability larger than 0.85 are considered as stable edges. The results showing the numbers of stable edges are summarized in Figure 6. We can observe that FMGL is more stable than GLasso. When the sample size is too small (say 20%), there are only 20 stable edges in the graph of NC obtained by GLasso. But the graph of NC obtained by FMGL still has about 140 edges, illustrating the superiority of FMGL in stability.

The brain connectivity models obtained by FMGL are shown in Figure 7. We can see that the number of connections within the prefrontal lobe significantly increases, and the number of connections within the temporal lobe significantly decreases from NC to AD, which are supported by previous literatures [24, 25]. The connections between the prefrontal and occipital lobes increase from NC to AD, and connections within cerebellum decrease. We can also find that the adjacent graphs are similar, indicating that FMGL can identify the common structures, but also keep the meaningful differences.

5 Conclusion

In this paper, we consider simultaneously estimating multiple graphical models by maximizing a fused penalized log likelihood. The blockwise coordinate descent method is employed to solve the fused multiple

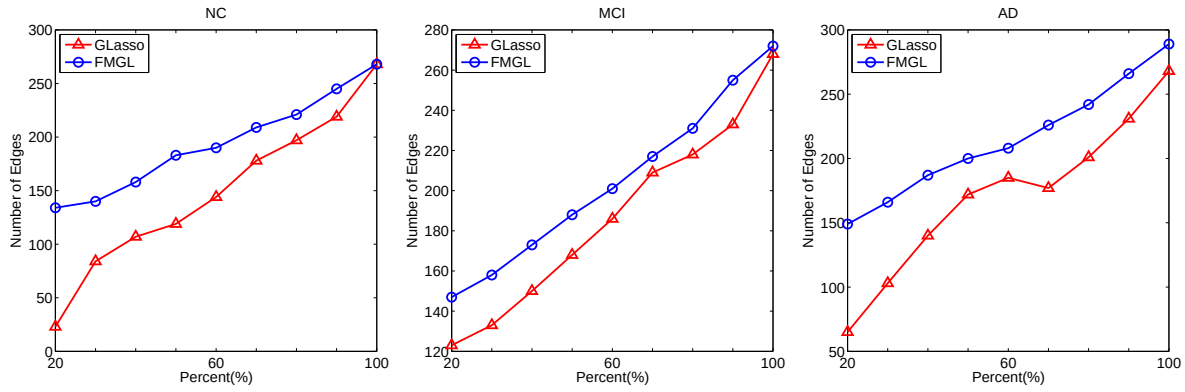


Figure 6: The number of stable edges in NC, MCI, and AD.

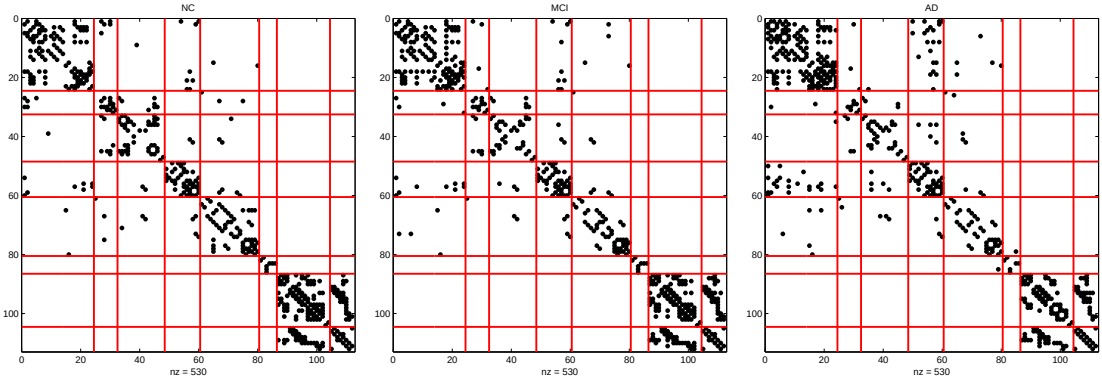


Figure 7: Brain connection models with 265 edges: NC, MCI, and AD. In each figure, the diagonal blocks are prefrontal lobe, other parts of frontal lobe, parietal lobe, occipital lobe, temporal lobe, corpus striatum, cerebellum, and vermis respectively.

graphical lasso. We have derived a set of necessary conditions for the FMGL solution to be block diagonal, and prove that they are also sufficient in the case of three graphs, extending the recent work in [13]. A screening rule has been developed to enable the efficient estimation of large multiple graphs. Numerical experiments on synthetic and real data demonstrate the effectiveness of the proposed method and screening rule. Based on our extensive simulation studies, we conjecture that the proposed necessary conditions are also sufficient for the general case with more than 3. A future direction is to prove the necessary and sufficient conditions in the general case.

References

[1] H.Y. Chuang, E. Lee, Y.T. Liu, D. Lee, and T. Ideker. Network-based classification of breast cancer metastasis. *Molecular systems biology*, 3(1), 2007.

[2] O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516, 2008.

[3] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006.

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

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

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

[7] S. Huang, J. Li, L. Sun, J. Liu, T. Wu, K. Chen, A. Fleisher, E. Reiman, and J. Ye. Learning brain connectivity of alzheimers disease from neuroimaging data. In *NIPS*, pages 808–816, 2009.

[8] H. Liu, K. Roeder, and L. Wasserman. Stability approach to regularization selection (StARS) for high dimensional graphical models. In *NIPS*, 2011.

[9] R. Mazumder and T. Hastie. The graphical lasso: New insights and alternatives. *Arxiv preprint arXiv:1111.5479*, 2011.

[10] D.M. Witten, J.H. Friedman, and N. Simon. New insights and faster computations for the graphical lasso. *Journal of Computational and Graphical Statistics*, 20(4):892–900, 2011.

[11] R. Mazumder and T. Hastie. Exact covariance thresholding into connected components for large-scale graphical lasso. *Arxiv preprint arXiv:1108.3829*, 2011.

[12] J. Guo, E. Levina, G. Michailidis, and J. Zhu. Joint estimation of multiple graphical models. *Biometrika*, 98(1):1–15, 2011.

[13] P. Danaher, P. Wang, and D.M. Daniela. The joint graphical lasso for inverse covariance estimation across multiple classes. *Arxiv preprint arXiv:1111.0324v3*, 2012.

[14] S. Zhou, J. Lafferty, and L. Wasserman. Time varying undirected graphs. *COLT*, 2008.

[15] M. Kolar, L. Song, A. Ahmed, and E.P. Xing. Estimating time-varying networks. *The Annals of Applied Statistics*, 4(1):94–123, 2010.

[16] M. Kolar and E.P. Xing. On time varying undirected graphs. In *AISTAT*, 2011.

[17] Y. Nesterov. *Gradient methods for minimizing composite objective function*. CORE, 2007.

[18] S.P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge Univ Pr, 2004.

[19] `http://fcon_1000.projects.nitrc.org/indi/adhd200/`.

[20] `http://adni.loni.ucla.edu/`.

[21] `http://www.nitrc.org/plugins/mwiki/index.php?title=neurobureau:NIAKPipeline/`.

[22] N. Tzourio-Mazoyer, B. Landeau, D. Papathanassiou, F. Crivello, O. Etard, N. Delcroix, B. Mazoyer, and M. Joliot. Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *Neuroimage*, 15(1):273–289, 2002.

[23] K. Wang, M. Liang, L. Wang, L. Tian, X. Zhang, K. Li, and T. Jiang. Altered functional connectivity in early alzheimer's disease: A resting-state fMRI study. *Human brain mapping*, 28(10):967–978, 2007.

[24] NP Azari, SI Rapoport, CL Grady, MB Schapiro, JA Salerno, A. Gonzales-Aviles, and B. Horwitz. Patterns of interregional correlations of cerebral glucose metabolic rates in patients with dementia of the alzheimer type. *Neurodegeneration*, 1:101–111, 1992.

[25] B. Horwitz, C.L. Grady, NL Schlageter, R. Duara, and SI Rapoport. Intercorrelations of regional cerebral glucose metabolic rates in alzheimer's disease. *Brain research*, 407(2):294–306, 1987.

[26] J. Liu, L. Yuan, and J. Ye. An efficient algorithm for a class of fused lasso problems. In *KDD*, pages 323–332, 2010.

[27] L. Condat. A direct algorithm for 1d total variation denoising. 2012.

Appendix

A. Optimization

We solve the problem in Eq.(2) by the blockwise coordinate descent method. Consider the partition of $\mathbf{\Theta}^{(k)}$

$$\mathbf{\Theta}^{(k)} = \begin{pmatrix} \mathbf{\Theta}_{11}^{(k)} & \boldsymbol{\theta}_{12}^{(k)} \\ \boldsymbol{\theta}_{21}^{(k)} & \theta_{22}^{(k)} \end{pmatrix}, \quad k = 1, \ldots, K \tag{16}$$

where $\mathbf{\Theta}_{11}^{(k)}$ is $(p-1)\times(p-1)$, $\boldsymbol{\theta}_{12}^{(k)}$ is $(p-1)\times 1$, and $\theta_{22}^{(k)}$ is scalar. We solve $\boldsymbol{\theta}_{12}^{(k)}$ and $\theta_{22}^{(k)}$, $k = 1, \ldots, K$ at each iteration while fixing the rest. Then Eq.(2) is equivalent to the following problem:

$$\min_{\boldsymbol{\theta}_{12}, \theta_{22}} \sum_{i=1}^{K} \Big(-\log(\theta_{22}^{(k)} - (\boldsymbol{\theta}_{12}^{(k)})^T (\mathbf{\Theta}_{11}^{(k)})^{-1} \boldsymbol{\theta}_{12}^{(k)}) + s_{22}^{(k)} \theta_{22}^{(k)} + 2(\mathbf{s}_{12}^{(k)})^T \boldsymbol{\theta}_{12}^{(k)} \Big) + 2P(\boldsymbol{\theta}_{12}). \tag{17}$$

where

$$P(\boldsymbol{\theta}_{12}) = \lambda_1 \sum_{k=1}^{K} \|\boldsymbol{\theta}_{12}^{(k)}\|_1 + \lambda_2 \sum_{k=1}^{K-1} \|\boldsymbol{\theta}_{12}^{(k)} - \boldsymbol{\theta}_{12}^{(k+1)}\|_1,$$

$\boldsymbol{\theta}_{12} = \{\boldsymbol{\theta}_{12}^{(1)}, \ldots, \boldsymbol{\theta}_{12}^{(K)}\}$, and $\theta_{22} = \{\theta_{22}^{(1)}, \ldots, \theta_{22}^{(K)}\}$.

Optimizing Eq.(17) with respect to $\theta_{22}^{(k)}$ yields

$$\hat{\theta}_{22}^{(k)} = \frac{1}{s_{22}^{(k)}} + (\boldsymbol{\theta}_{12}^{(k)})^T(\boldsymbol{\Theta}_{11}^{(k)})^{-1}\boldsymbol{\theta}_{12}^{(k)},\ k = 1, \ldots, K. \tag{18}$$

Plugging $\hat{\theta}_{22}^{(k)}$ into Eq.(17), we can get the following optimization problem

$$\hat{\boldsymbol{\theta}}_{12} = \arg\min_{\boldsymbol{\theta}_{12}} \sum_{i=1}^{K} \left(s_{22}^{(k)}(\boldsymbol{\theta}_{12}^{(k)})^T(\boldsymbol{\Theta}_{11}^{(k)})^{-1}\boldsymbol{\theta}_{12}^{(k)}) + 2(\mathbf{s}_{12}^{(k)})^T\boldsymbol{\theta}_{12}^{(k)} \right) + 2P(\boldsymbol{\theta}_{12}). \tag{19}$$

The objective in Eq. (19) consists of a smooth convex term and a nonsmooth convex penalty term. Many algorithms can be applied to solve Eq. (19) . In this paper, we employ the accelerated gradient method [17] and the fused lasso signal approximator [26]. From $\hat{\boldsymbol{\theta}}_{12}^{(k)}$, it is easy to obtain $\hat{\theta}_{22}^{(k)}$ based on Eq.(18).

A.1 Accelerated gradient method

Denote $\mathbf{Y} = [\boldsymbol{\theta}_{12}^{(1)}, \ldots, \boldsymbol{\theta}_{12}^{(K)}] \in \mathcal{R}^{(p-1)\times K}$, and $\mathbf{U} = [\mathbf{s}_{12}^{(1)}, \ldots, \mathbf{s}_{12}^{(K)}] \in \mathcal{R}^{(p-1)\times K}$, then Eq. (19) can be equivalently written as

$$\min_{\mathbf{Y}} h(\mathbf{Y}) := f(\mathbf{Y}) + P(\mathbf{Y}) \tag{20}$$

where $f(\mathbf{Y}) = \sum_{i=1}^{K} \left(\frac{1}{2} s_{22}^{(k)}(\boldsymbol{\theta}_{12}^{(k)})^T(\boldsymbol{\Theta}_{11}^{(k)})^{-1}\boldsymbol{\theta}_{12}^{(k)}) + (\mathbf{s}_{12}^{(k)})^T\boldsymbol{\theta}_{12}^{(k)} \right)$. Note that $f(\mathbf{Y})$ is quadratic, whose gradient is Lipschitz continuous with constant L, i.e.

$$\|\nabla f(\mathbf{Y}_1) - \nabla f(\mathbf{Y}_2)\|_F \leq L\|\mathbf{Y}_1 - \mathbf{Y}_2\|_F, \forall \mathbf{Y}_1, \mathbf{Y}_2 \in \mathcal{R}^{(p-1)\times K}$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

We use the Nesterov's method [17] to solve Eq. (20). The Nesterov's method is based on two sequences $\{\mathbf{Y}_i\}$ and $\{\mathbf{V}_i\}$ in which $\{\mathbf{Y}_i\}$ is the sequence of approximate solutions, $\{\mathbf{V}_i\}$ is the sequence of search points, and i represents the i-th iteration. The search point $\mathbf{V}_i$ is the affine combination of $\mathbf{Y}_{i-1}$ and $\mathbf{Y}_i$ as

$$\mathbf{V}_i = \mathbf{Y}_i + \beta_i(\mathbf{Y}_i - \mathbf{Y}_{i-1}),$$

where β_i is a properly chosen coefficient. The approximate solution $\mathbf{Y}_{i+1}$ is computed as the minimizer of the linearized function of $h(\mathbf{Y})$ at $\mathbf{V}_i$:

$$\mathbf{Y}_{i+1} = \arg\min_{\mathbf{Z}} h_{L_i,\mathbf{V}_i}(\mathbf{Z}) := f(\mathbf{V}_i) + \langle \mathbf{Z} - \mathbf{V}_i, \nabla f(\mathbf{V}_i)\rangle + P(\mathbf{Z}) + \frac{L_i}{2}\|\mathbf{Z} - \mathbf{V}_i\|_F^2$$

where L_i is determined by line search so that L_i should be appropriate for the search point $\mathbf{V}_i$. By ignoring the terms that do not depend on $\mathbf{Z}$, the above equation can be expressed equivalently as

$$\mathbf{Y}_{i+1} = \arg\min_{\mathbf{Z}} \frac{L_i}{2}\|\mathbf{Z} - (\mathbf{V}_i - \frac{1}{L_i}\nabla f(\mathbf{V}_i))\|_F^2 + P(\mathbf{Z}). \tag{21}$$

Eq. (21) is well decoupled, and each row of $\mathbf{Y}_{i+1}$ can be separately computed by the fused lasso signal approximator [26, 27], i.e.

$$\mathbf{y}^j = \arg\min_{z} \frac{1}{2}\|\mathbf{z} - \mathbf{t}^i\|^2 + \frac{\lambda_1}{L_i}\|\mathbf{z}\|_1 + \frac{\lambda_2}{L_i}\sum_{k=1}^{K-1}|z_k - z_{k+1}|,$$

where $\mathbf{y}^j$ is the j-th row of $\mathbf{Y}_{i+1}$, and $\mathbf{t}^j$ is the j-th row of $\mathbf{V}_i - 1/L_i\nabla f(\mathbf{V}_i)$.

The key steps of the accelerated gradient method to solve Eq. (19) are summarized in Algorithm 1.

Algorithm 1: Accelerated Gradient Method

Input: $\mathbf{U}, \lambda_1, \lambda_2, \mathbf{Y}_0, L_0$
Output: $\mathbf{Y}, L$
Initialization: $\mathbf{Y}_1 = \mathbf{Y}_0,\ \alpha_{-1} = 0,\ \alpha_0 = 1$, and $L = L_0$;
while *Not Converged* **do**
  Set $\beta_i = \frac{\alpha_{i-2}-1}{\alpha_{i-1}}$, $\mathbf{V}_i = \mathbf{Y}_i + \beta_i(\mathbf{Y}_i - \mathbf{Y}_{i-1})$.
  Find the smallest $L = L_{i-1}, 2L_{i-1}, \ldots$ such that $h(\mathbf{Y}_{i+1}) \leq h_{L,\mathbf{V}_i}(\mathbf{Y}_{i+1})$ where $\mathbf{Y}_{i+1}$ is computed using Eq. (21).
  Set $L_i = L$ and $\alpha_{i+1} = \frac{1+\sqrt{1+4\alpha^2}}{2}$.
end
return $\mathbf{Y}_{i+1}, L_i$;

A.2 Computing $(\boldsymbol{\Theta}_{11}^{(k)})^{-1}$

Eq. (19) involves the inverse of $\boldsymbol{\Theta}_{11}^{(k)}$ that can be efficiently computed. Let the covariance matrix $\mathbf{W}^{(k)} = (\boldsymbol{\Theta}^{(k)})^{-1}, k = 1, \ldots, K$. Consider the same partition of $\mathbf{W}^{(k)}$

$$\mathbf{W}^{(k)} = \begin{pmatrix} \mathbf{W}_{11}^{(k)} & \mathbf{w}_{12}^{(k)} \\ \mathbf{w}_{21}^{(k)} & w_{22}^{(k)} \end{pmatrix}, \quad k = 1, \ldots, K,$$

we have

$$\begin{aligned} \mathbf{W}_{11}^{(k)} &= (\boldsymbol{\Theta}_{11}^{(k)})^{-1} + s_{22}^{(k)} (\boldsymbol{\Theta}_{11}^{(k)})^{-1} \boldsymbol{\theta}_{12}^{(k)} (\boldsymbol{\theta}_{12}^{(k)})^T (\boldsymbol{\Theta}_{11}^{(k)})^{-1}, \\ \mathbf{w}_{12}^{(k)} &= -s_{22}^{(k)} (\boldsymbol{\Theta}_{11}^{(k)})^{-1} \boldsymbol{\theta}_{12}^{(k)}, \; w_{22}^{(k)} = s_{22}^{(k)} \end{aligned} \tag{22}$$

since $s_{22}^{(k)} = 1/(\theta_{22}^{(k)} - (\boldsymbol{\theta}_{12}^{(k)})^T (\boldsymbol{\Theta}_{11}^{(k)})^{-1} \boldsymbol{\theta}_{12}^{(k)})$. Thus, we can obtain $(\boldsymbol{\Theta}_{11}^{(k)})^{-1}$ by $\mathbf{W}^{(k)}$

$$(\boldsymbol{\Theta}_{11}^{(k)})^{-1} = \mathbf{W}_{11}^{(k)} - \mathbf{w}_{12}^{(k)} (\mathbf{w}_{12}^{(k)})^T / w_{22}^{(k)}, \tag{23}$$

which only needs $O(p^2)$ operations.

The outline of FMGL is shown in Algorithm 2.

Algorithm 2: Fused Multiple Graphical Lasso (FMGL)

Input: $\mathbf{S}^{(k)}, k = 1, \ldots, K, \lambda_1, \lambda_2$
Output: $\boldsymbol{\Theta}^{(k)}, \mathbf{W}^{(k)}, k = 1, \ldots, K$
Initialization: $\mathbf{W}^{(k)} = \text{diag}(\mathbf{S}^{(k)})$ and $\boldsymbol{\Theta}^{(k)} = (\mathbf{W}^{(k)})^{-1}$;
for $j = 1, 2, \ldots, p, 1, 2, \ldots, p, \ldots$
  Compute $(\boldsymbol{\Theta}_{11}^{(k)})^{-1}, k = 1, \ldots, K$ according to Eq.(23).
  Solve Eq.(19) using results from the previous round as warm-start. Update $\boldsymbol{\theta}_{12}^{(k)}$ and $\theta_{22}^{(k)}$ using Eq.(18).
  Update $\boldsymbol{\Theta}^{(k)}$ and $\mathbf{W}^{(k)}$ using Eq.(22) so that $\boldsymbol{\Theta}^{(k)} \mathbf{W}^{(k)} = \mathbf{I}_{p \times p}$.
Until *Convergence*;
return $\boldsymbol{\Theta}^{(k)}, \mathbf{W}^{(k)}, k = 1, \ldots, K$;

A.3 Computational complexity

For each iteration, we compute $(\boldsymbol{\Theta}_{11}^{(k)})^{-1}$ and update $\mathbf{W}^{(k)}$, which involves rank-one operations with total cost of $O(Kp^2)$ for K graphs. Each iteration of the accelerated gradient method involves computation of the gradient and the computation of $p - 1$ fused lasso signal approximators. The complexity of computing the gradient is $O(Kp^2)$. The fused lasso signal approximator usually takes less than 10 iterations to converge [26]. The number of iterations in the accelerated gradient method to obtain an ϵ solution is $O(1/\sqrt{\epsilon})$. Thus, the total complexity of each iteration is $O(Kp^2/\sqrt{\epsilon})$, and the complexity of each round (update all p columns/rows) of Algorithm 2 is $O(Kp^3/\sqrt{\epsilon})$.

B. Proof of Lemma 2

We split the problem into three cases.

- Case 1: $-3\lambda_1 \leq y_1 + y_2 + y_3 \leq -\lambda_1$
  Eq.(6) can be rewritten as

$$-\lambda_1 - y_1 \leq a_1 \leq 2\lambda_1 + y_2 + y_3, -2\lambda_1 - y_1 - y_2 \leq a_2 \leq \lambda_1 + y_3. \tag{24}$$

  Eq.(7) can be written as

$$-2\lambda_1 - y_1 - y_3 \leq a_1 - a_2 \leq \lambda_1 + y_2. \tag{25}$$

  Based on Eq.(24), we have

$$-2\lambda_1 - y_1 - y_3 \leq a_1 - a_2 \leq 4\lambda_1 + y_1 + 2y_2 + y_3. \tag{26}$$

  From Eq.(24), we can see that the region of Eq. (24) is a square since $2\lambda_1 + y_2 + y_3 + (\lambda_1 + y_1) = \lambda_1 + y_3 + (2\lambda_1 + y_1 + y_2)$. Since $4\lambda_1 + y_1 + 2y_2 + y_3 \geq \lambda_1 + y_2$ and $-2\lambda_1 - y_1 - y_3 = -2\lambda_1 - y_1 - y_3$, Eqs. (25) and (26) intersect. Moreover, the region of Eq.(25) belongs to that of Eq.(26), and the width of the region of Eq.(25) is half of that of Eq.(26), which means that the diagonal BC belongs to the region of Eq.(26).

- Case 2: $-\lambda_1 \leq y_1 + y_2 + y_3 \leq \lambda_1$
  Eq.(6) can be rewritten as

$$-\lambda_1 - y_1 \leq a_1 \leq \lambda_1 - y_1, -\lambda_1 + y_3 \leq a_2 \leq \lambda_1 + y_3. \tag{27}$$

  Eq.(7) can be written as

$$-\lambda_1 + y_2 \leq a_1 - a_2 \leq \lambda_1 + y_2. \tag{28}$$

  Based on Eq.(27), we have

$$-2\lambda_1 - y_1 - y_3 \leq a_1 - a_2 \leq 2\lambda_1 - y_1 - y_3. \tag{29}$$

  Similar to Case 1, we can obtain the same conclusion as in Case 1.

- Case 3: $\lambda_1 \leq y_1 + y_2 + y_3 \leq 3\lambda_1$
  Eq.(6) can be rewritten as

$$-2\lambda_1 + y_2 + y_3 \leq a_1 \leq \lambda_1 - y_1, -\lambda_1 + y_3 \leq a_2 \leq 2\lambda_1 - y_1 - y_2. \tag{30}$$

  Eq.(7) can be written as

$$-\lambda_1 + y_2 \leq a_1 - a_2 \leq 2\lambda_1 - y_1 - y_3. \tag{31}$$

  Based on Eq.(30), we have

$$-4\lambda_1 + y_1 + 2y_2 + y_3 \leq a_1 - a_2 \leq 2\lambda_1 - y_1 - y_3. \tag{32}$$

  Similar to Case 1, we can obtain the same conclusion as in Case 1.

C. Graph of ADHD-200

Figure 8 shows a subgraph of ADHD-200 identified by FMGL with the screening rule.

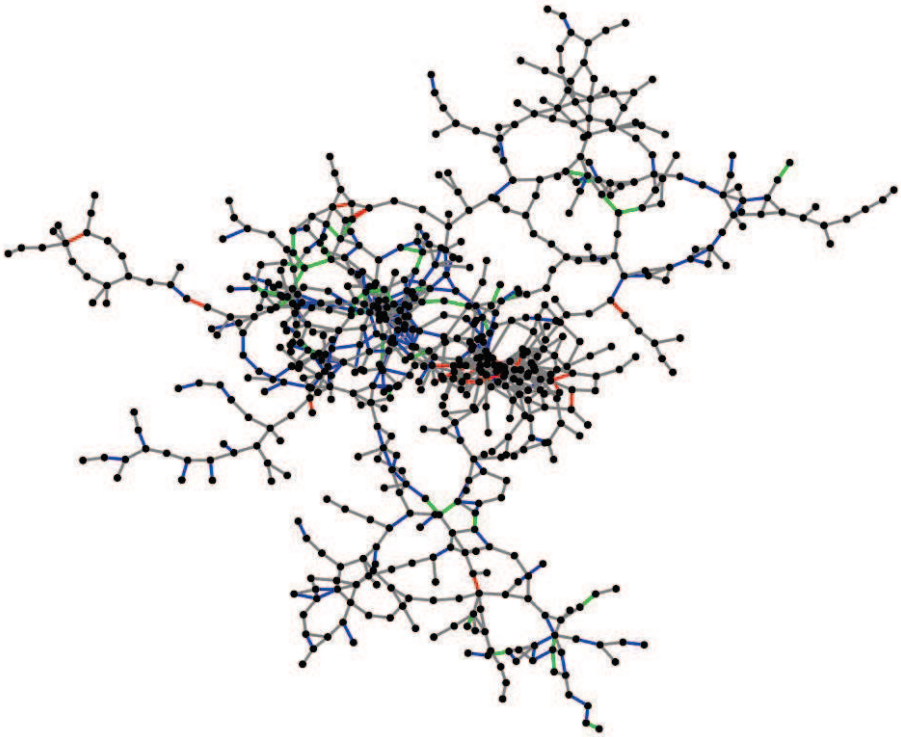


Figure 8: A subgraph of ADHD-200 identified by FMGL with the proposed screening rule. The grey edges are common edges among the three graphs; the red, green, and blue edges are the specific edges for TDC, ADHD-I, and ADHD-C respectively.