Gaussian Mixture Graphical Lasso with Application to Edge Detection in Brain Networks

***Abstract*—Sparse inverse covariance estimation (*i.e.*, edge detection) is an important research problem in recent years, where the goal is to discover the direct connections between a set of nodes in a networked system based upon the observed node activities. Existing works mainly focus on unimodal distributions, where it is usually assumed that the observed activities are generated from a *single* Gaussian distribution (*i.e.*, one graph). However, this assumption is too strong for many real-world applications. In many real-world applications (*e.g.*, brain networks), the node activities usually exhibit much more complex patterns that are difficult to be captured by one single Gaussian distribution. In this work, we are inspired by Latent Dirichlet Allocation (LDA) [6] and consider modeling the edge detection problem as estimating a mixture of *multiple* Gaussian distributions, where each corresponds to a separate sub-network. To address this problem, we propose a novel model called Gaussian Mixture Graphical Lasso (MGL). It learns the proportions of signals generated by each mixture component and their parameters iteratively via an EM framework. To obtain more interpretable networks, MGL imposes a special regularization, called Mutual Exclusivity Regularization (MER), to minimize the overlap between different sub-networks. MER also addresses the common issues in read-world data sets, *i.e.*, noisy observations and small sample size. Through the extensive experiments on synthetic and real brain data sets, the results demonstrate that MGL can effectively discover multiple connectivity structures from the observed node activities.**

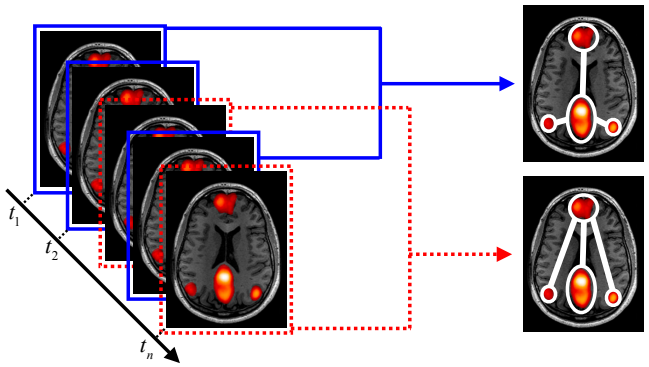
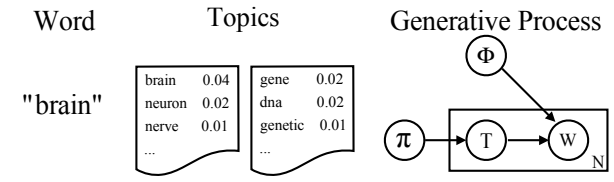


Fig. 1: The problem of Gaussian mixture sparse inverse covariance estimation. The brain activities over time may originate from the mixture of multiple latent cognitive brain modes (*i.e.* different connectivity structures among nodes). Without knowing the mode proportions and assignments in the observed brain images, our goal is to discover these underlying sub-networks for different modes.

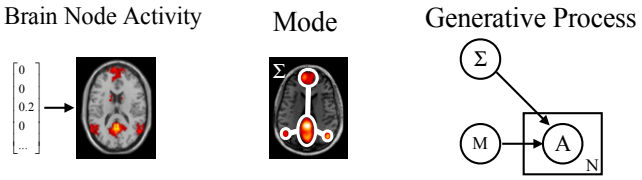
I. Introduction

Edge detection of brain network [9], [13], [11], [19] aims at identifying the edges between nodes (*i.e.*, functionally coherent brain regions) of a brain mapping [5], [8] from a temporal sequence of observed activities (*e.g.*, fMRI scans). Since a well-constructed connectivity network servers as the prerequisite for many graph mining algorithms on brain disorder diagnosis and brain functionality analysis [1], it is significant to design a more effective and accurate edge detection method. Existing edge detection methods usually rely on the assumption that all nodes' activities obey a multivariate Gaussian distribution, and the connections between nodes could be depicted by their inverse covariance matrix (*a.k.a.* precision matrix) [26]. A widely used variation of this line of works is known as Graphical Lasso (GLasso) [14], which additionally imposes sparseness on the precision matrix. However, in many neurology studies such as [20], [7], [3], [12], human brains usually exhibit dramatically different activity modes when they perform different tasks. Based on these studies, we believe that the cognitive structure of the human mind can be paralleled into several sub-graphs based on different cognitive control processes and behavior. Cognitive control means a set of dynamic processes that engage and disengage different nodes of brain to modulate attention and switch between tasks. Applying GLasso without considering different latent cognitive modes is equivalent to deriving an "average" network representation. Since the behavior of different brain modes varies significantly, the derived "average" network may lose crucial information. Under such context, as illustrated in Figure 1, it is natural to investigate whether and how one could extend the edge detection methods applied in brain network to capture the connectivity structures of multiple underlying cognitive brain modes.

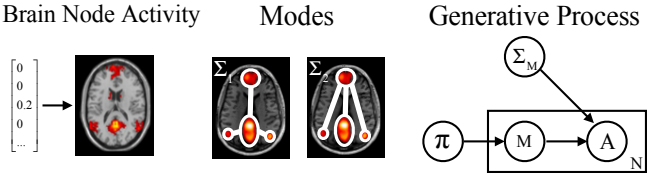
To incorporate the concept of multiple connectivity structure into edge detection, we follow the idea of latent Dirichlet allocation (LDA) [6] to adopt Gaussian mixture model on this problem. LDA views a document as a mixture of various topics, and it assumes that the generation of a document follows some topic-word distributions which can be found by sampling. Similarly, we could view brain scans as mixtures of latent modes, where each mode is characterized by a Gaussian distribution with different covariance Σ_{M}. Each covariance matrix Σ_{M} corresponds to a specific connectivity among brain nodes. In the generation of each brain node activity, our model chooses a mode M based on the mode distribution π (as LDA chooses a topic), and then it generates a brain node activity $A_i \sim \text{Multinomial}(\mathbf{0}, \Sigma_{\mathrm{M}})$ (as LDA generates a word based on the topic chosen). Figure 2 illustrates the relations and differences between our proposals and LDA, we also compare with traditional edge detection methods Graphical Lasso [14],



(a) Latent Dirichlet Allocation (LDA)[6]



(b) Graphical Lasso[14]



(c) Mixture Graphical Lasso (this paper)

Fig. 2: Comparison of Latent Dirichlet Allocation(LDA), Graphical Lasso (GLasso) and our model in this paper. In each sub-graph, the boxes are "plates" representing replicates, which are repeated entities. The outer plate represents document in LDA or observation subject in brain network study, while the inner plate represents the generative process of word (W) in a given document or brain node activity (A) in a given subject, each of which word or scan is associated with a choice of topic (T) or mode (M) and the parameter of the corresponding word or node activity distribution (ϕ or Σ). π is the topic or mode distribution. N denotes the number of words or scans.

where all brain node activities are assumed being produced by a single unified zero-mean Σ-covariance multivariate Gaussian distribution.

In this paper, our goal is to reveal these structure of underlying sub-network from the observed activities simultaneously. To solve above issues, our main challenges are as follows:

- **Mixture of multiple connectivity networks**: In real-world cases, the proportions and assignments of each mode are not observable. Without the prior knowledge of them, general GLasso only discovers a simple graph for the whole data sets. While our problem setting requires estimating the proportions and assignments of multiple latent cognitive modes as well as the parameters of the network for each mode, with the same input as GLasso, which is much more challenging.
- **Direct connectivity among the nodes**: The finite Gaussian Mixture Model (GMM) [24] seems a straightforward solution to our problem, which incorporates a heterogeneous structure into the graphical model. It fits multivariate normal distributions and treats proportions and assignments as prior and posterior probabilities (estimators) in the Bayesian setting respectively. However, it estimates the covariance of each distribution rather than the inverse covariance, which indicates that the discovered connections could be indirect and make the network unnecessarily complicated. So GMM is inappropriate to distinguish the directed relationships between each pair of nodes.
- **Noisy Observations and Small Sample**: It is already a challenging task to discover a single network given the noisy observations and the small size of the data sample. GLasso employs simple ℓ_1-norm regularization on to alleviate the sensitiveness to noises, but it is not sufficient for our case. Based on the cognitive studies on the human brain [20], [7], each brain sub-network is not only sparse but also has limited overlapping with other sub-network, simply adopting ℓ_1-norm regularization as in GLasso may make the derived sub-network highly intertwined and hard to interpret. So we want to design a new regularization into the model, which can enforce it to discover a set of different sub-graphs, no matter small sample size or noisy data.

To tackle the above challenges, we propose a new model, namely MGL, to discover such mixture connectivity structures of the brain network. Similar to GMM, MGL learns the proportions and assignments of each latent cognitive mode iteratively via an EM framework, with the emphases on inferring the inverse covariance matrix of each latent distribution. A novel regularization approach called Mutual Exclusivity Regularization (MER) is also proposed to differ each inverse covariance matrix, implying that sub-network of different brain regions are activated under different cognitive modes. We introduce our model and algorithm in next two sections. Then in the fourth part, we show Extensive empirical studies on both synthetic data and real brain fMRI data demonstrate the effectiveness of MGL. We also discuss some related works related to edge detection problem in brain network. Finally, we conclude our work.

II. Preliminary

A. Notation

Throughout this paper, $\mathbb{R}$ denotes the set of all real numbers, $\mathbb{R}^n$ stands for the n-dimensional euclidean space. The set of all $m \times n$ matrices with real entries is denoted as $\mathbb{R}^{m \times n}$. All matrices are written in boldface. We write $\mathbf{X} \succ 0$ to denote that matrix $\mathbf{X}$ is positive definite. We write $\mathrm{tr}(\cdot)$ to refer the trace of a matrix, which is defined to be the sum of the elements on the main diagonal of the matrix. We use $|\mathbf{X}|$ to denote the determinant of a real square matrix $\mathbf{X}$. We define a special matrix of $\mathbf{X}$ as follows:

$$\bar{\mathbf{X}} = \begin{bmatrix} 0 & |X_{12}| & \cdots & |X_{1N}| \\ |X_{21}| & 0 & \cdots & |X_{2N}| \\ |X_{N1}| & |X_{N2}| & \cdots & 0 \end{bmatrix} \quad (1)$$

TABLE I: Important Notations.

Symbol	Definition
D	The number of variables or objects
N	The number of samples
K	The number of distribution
$\mathbf{X}$	The observations of D-variate Gaussian distribution
$\mathbf{S}$	The empirical covariance matrix of $\mathbf{X}$
ϕ_k	The prior probability or mixing coefficient of distribution k
r_{ik}	The posterior probability of sample i generated from distribution k
$\mathbf{\Sigma}_k$	The covariance matrix of distribution k
$\mathbf{\Theta}_k$	The precision matrix of distribution k
$\bar{\mathbf{\Theta}}_k$	The non-negative copy of $\mathbf{\Theta}_k$ with zero diagonal elements
λ_1	The Lagrangian multiplier of general lasso regularization
λ_2	The Lagrangian multiplier of mutual exclusivity regularization

$\bar{\mathbf{X}}$ is the non-negative copy of $\mathbf{X}$ removed all diagonal elements. In addition, we collect other important notations in Table(I).

B. Inverse Covariance Estimation

The inverse covariance matrix is defined as $\mathbf{\Theta} = \mathbf{\Sigma}^{-1}$, which can filter the directed links between all relationships.

C. Graphical Lasso

Graphical Lasso (GLasso) or Gaussian Graphical Model (GGM) is usually formulated as the following optimization problem,

$$\min_{\mathbf{\Theta}\succ 0} -\log|\mathbf{\Theta}| + \text{tr}(\mathbf{S\Theta}) + \lambda||\mathbf{\Theta}||_1 \tag{2}$$

where $\mathbf{S} = \frac{1}{n}\mathbf{X}^\top\mathbf{X}$ is the empirical covariance matrix, $||\mathbf{\Theta}||_1$ is the ℓ_1-norm regularization that encourages sparse solutions, and λ is a positive parameter denotes the strength of regularization. Since $\hat{\mathbf{\Theta}}$ is usually a relatively sparse matrix with non-zero entries corresponding to directly connected pairs of nodes, we can use it as a concise representation of the underlying network.

D. The Adaptive Lasso

[33] proposed a special penalty to achieve the desirable properties, which is called the adaptive Lasso. It requires different weights for each component in the Lasso penalty. So putting the adaptive Lasso penalty into Eq. 2, we can get:

$$\min_{\mathbf{\Theta}\succ 0} -\log|\mathbf{\Theta}| + \text{tr}(\mathbf{S\Theta}) + \lambda\sum_{i=1}^{N}\sum_{k=1}^{K}\omega_{ik}|\mathbf{\Theta}_{(ik)}| \tag{3}$$

where $|\mathbf{\Theta}_{(ij)}|$ denotes the ij-th element of $\mathbf{\Theta}$. We will propose a similar idea in our model to deal with the non-overlapping problem.

E. Gaussian Mixture Model

One of the most popular mixture model is Gaussian Mixture Model (GMM), where each base distribution in the mixture is a multivariate Gaussian (MVG) with mean $\boldsymbol{\mu}_k$ and covariance matrix $\mathbf{\Sigma}_k$, the probability of data sample $\boldsymbol{x}_i$ is as follows:

$$p(\boldsymbol{x}_i|\boldsymbol{\theta}) = \sum_{k=1}^{K}\phi_k\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k) \tag{4}$$

where $\boldsymbol{\theta}$ is the model parameters, $0 \le \phi_k \le 1$ is the prior probability of the k-th base distribution chosen to generate a sample and $\sum_{k=1}^{K}\phi_k = 1$.

III. MGL Method

A. Gaussian Mixture Graphical Lasso

Given the number of base distributions K and the number of node N, we assume the observed sample of each node is a mixture of the K distributions. Thus, the joint probability of all observations $\mathbf{X} = (\boldsymbol{x}_1^\top, \cdots, \boldsymbol{x}_N^\top) \in \mathbb{R}^{N\times D}$ is given by

$$\begin{aligned} p(\mathbf{X}|\mathbf{\Theta}_k, \boldsymbol{\mu}_k, \phi_k) &= \prod_{i=1}^{N}\sum_{k=1}^{K}\phi_k\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k) \\ &= \prod_{i=1}^{N}\sum_{k=1}^{K}\phi_k\frac{\exp\big(-\frac{1}{2}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)^\top\mathbf{\Sigma}_k(\boldsymbol{x}_i-\boldsymbol{\mu}_k)\big)}{(2\pi)^{D/2}|\mathbf{\Sigma}_k|^{1/2}} \end{aligned}$$

We could assume $\boldsymbol{\mu}_k = \mathbf{0}$ without losing generality, so the negative log likelihood (NLL) in terms of $\{\mathbf{\Theta}_k\}$ is given by,

$$\begin{aligned} \text{NLL}(\boldsymbol{\theta}) &= -\log\Big(\prod_{i=1}^{N}\sum_{k=1}^{K}\phi_k\mathcal{N}(\boldsymbol{x}_i|\mathbf{0}, \mathbf{\Sigma}_k)\Big) \\ &= -\sum_{i=1}^{N}\log\Big(\sum_{k=1}^{K}\phi_k\mathcal{N}(\boldsymbol{x}_i|\mathbf{0}, \mathbf{\Theta}_k^{-1})\Big) \end{aligned} \tag{5}$$

where $\boldsymbol{\theta} = \{\phi_1, \cdots, \phi_k, \mathbf{\Theta}_1, \cdots \mathbf{\Theta}_k\}$ is the model parameters.

B. The Mutual Exclusivity Regularization

Similar to Eq. (3), we also need to impose regularization on our mixture model to obtain interpretable results, which means non overlapping edges exist among all estimators of precision matrices. However, be different with adaptive lasso

or fused lasso, the intuitions are two folds: (1) we want each $\mathbf{\Theta}_k$ to be sparse; (2) we want each $\mathbf{\Theta}_k$ to be fairly different from other $\mathbf{\Theta}_{k'}$. Towards this end, we propose to the mutual exclusivity regularization as follows,

$$\ell_{\lambda_1,\lambda_2}(\{\mathbf{\Theta}_k\}) = \lambda_1 \sum_{k=1}^{K} \|\mathbf{\Theta}_k\|_1 + \lambda_2 \sum_{i\neq j} \text{tr}(\bar{\mathbf{\Theta}}_i \bar{\mathbf{\Theta}}_j) \tag{6}$$

where $\bar{\mathbf{\Theta}} = \begin{bmatrix} |\Theta_{11}| & \cdots & |\Theta_{1N}| \\ & \cdots & \\ |\Theta_{N1}| & \cdots & |\Theta_{NN}| \end{bmatrix}$ is the non-negative copy of $\mathbf{\Theta}$. The first term is identical to graphical lasso, which imposes sparsity controlled by $\lambda_1 > 0$ on each $\mathbf{\Theta}_k$. The second term is the summation of the approximate divergence measure between each pair $(\mathbf{\Theta}_i, \mathbf{\Theta}_j)$. It is easy to see when there is no overlapping non-zero entities between each $\mathbf{\Theta}_k$, this term reaches its minimal value 0. $\lambda_2 > 0$ is employed to tune the strength of the second regularization. So it makes sense that we can use this term to force each estimation of $\mathbf{\Theta}_k$ in the result to have as few over-lapping elements as possible.

Hence, we formally present the objective of our MGL as follows,

$$\min_{\{\mathbf{\Theta}_k \succ 0\}} \text{NLL}(\{\mathbf{\Theta}_k\}) + \ell_{\lambda_1,\lambda_2}(\{\mathbf{\Theta}_k\}) \tag{7}$$

C. The Latent States

Since there are K separate latent distributions, so each data sample $\boldsymbol{x}_i$ could come from one of the K distributions, we denote the corresponding state as $z_i \in \{1, \cdots, K\}$. Thus, the NLL function could be rewritten as follows,

$$\begin{aligned} \text{NLL}(\boldsymbol{\theta}) &= -\sum_{i=1}^{N} \log \sum_{k=1}^{K} \Big(\frac{\mathbf{Q}(z_{ik}) p\big(\boldsymbol{x}_i | \mathbf{\Theta}_k\, \phi_k\big)}{\mathbf{Q}(z_{ik})} \Big) \\ &= -\sum_{i=1}^{N} \log \sum_{k=1}^{K} \Big(\frac{p\big(\boldsymbol{x}_i, z_{ik} | \mathbf{\Theta}_k\, \phi_k\big)}{\mathbf{Q}(z_{ik})} \Big) \end{aligned} \tag{8}$$

Here $\mathbf{Q}(z_{ik})$ is the latent variable and $\sum_{k=1}^{K} \mathbf{Q}(z_{ik}) = 1$. In fact, we can treat this item as the posterior probability of the i-th observation generated by the k-th distribution, which will be proved in the next section.

According to the expression in the Equation (8), it can not be directly computed because the expression in log is a sum term. So we use the Expectation Maximization (EM) algorithm to optimize the above NLL *w.r.t.* $\{\mathbf{\Theta}_k\}$.

D. The E Step

Firstly, according to the Jensen inequality, we known that when the optimal function is convex:

$$f(E(x)) \geqslant E(f(x)) \tag{9}$$

Because NLL is convex, and $\sum_{k=1}^{K} \Big(\frac{p\left(\boldsymbol{x}_i, z_i | \mathbf{\Theta}_k\, \phi_k\right)}{\mathbf{Q}(z_i)} \Big)$ can be treated as the expectation of $p\big(\boldsymbol{x}_i, z_i | \mathbf{\Theta}_k\, \phi_k\big)$. So we apply Jensen inequality here to find a lower bound of it. Based on this principle, we get the lower bound of NLL such as follows,

$$\text{NLL}(\boldsymbol{\theta}) = -\sum_{i=1}^{N} \log \sum_{k=1}^{K} \Big(\frac{p\big(\boldsymbol{x}_i, z_i | \mathbf{\Theta}_k\, \phi_k\big)}{\mathbf{Q}(z_i)} \Big) \tag{10}$$

$$\leqslant -\sum_{i=1}^{N} \sum_{k=1}^{K} \mathbf{Q}(z_i) \log(p\big(\boldsymbol{x}_i, z_i | \mathbf{\Theta}_k\, \phi_k\big)) \tag{11}$$

The equal sign is approved only when the following is true,

$$\frac{p(\boldsymbol{x}_i, z_{ik})}{\mathbf{Q}(z_{ik})} = C \tag{12}$$

where C is a constant. So simply we have,

$$\sum_{k=1}^{K} p(\boldsymbol{x}_i, z_{ik}) = C \sum_{k=1}^{K} \mathbf{Q}(z_{ik}) = C \tag{13}$$

$$\mathbf{Q}(z_{ik}) = \frac{p(\boldsymbol{x}_i, z_{ik})}{\sum_{k=1}^{K} p(\boldsymbol{x}_i, z_{ik})} = r_{ik} \tag{14}$$

The $\text{NLL}(\boldsymbol{\theta}) = -\sum_{i=1}^{N} \sum_{k=1}^{K} \mathbf{Q}(z_i) \log(p\big(\boldsymbol{x}_i, z_i | \mathbf{\Theta}_k\, \phi_k\big))$ is correct only when the constraint of $\mathbf{Q}(z_{ik})$ is true. So we can get the conclusion that the latent variable we created is the posterior probability of the i-th observation generated by the k-th distribution. Then we can compute each r_{ik} based on the initialization or update results of $\mathbf{\Theta}_k$ and ϕ_k.

E. The M Step

After we obtain the $\boldsymbol{r}_i^{(t)}$ in the E step, we could update ϕ_k accordingly:

$$\phi_k^{(t)} = \frac{1}{N} \sum_{i=1}^{N} r_{ik}^{(t)} \tag{15}$$

However, $\frac{1}{N} \sum_{i=1}^{N} r_{ik}^{(t)} - \phi_k^{(t-1)}$ is a feasible descent direction. So we update ϕ_k based on the follows:

$$\phi_k^{(t)} = \phi_k^{(t-1)} + \delta^k (\frac{1}{N} \sum_{i=1}^{N} r_{ik}^{(t)} - \phi_k^{(t-1)}) \tag{16}$$

where $\{\delta^k | k = 0, 1, 2, \ldots; \delta \in (0, 1)\}$ is a learning rate and we find $\delta = 0.1$ works well in our experiments.

The remaining problem is to find $\mathbf{\Theta}_k$ that maximizes the expectation we obtain in the E step, which is equivalent to minimize the following function:

$$\min_{\{\mathbf{\Theta}_k \succ 0\}} -\sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik}^{(t)} \log p(\boldsymbol{x}_i | \mathbf{\Theta}_k) \tag{17}$$

$$\iff \min_{\{\mathbf{\Theta}_k \succ 0\}} -\sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik}^{(t)} \log \Big(\frac{\exp(-\frac{1}{2} \boldsymbol{x}_i^\top \mathbf{\Theta}_k \boldsymbol{x}_i)}{(2\pi)^{D/2} |\mathbf{\Theta}_k|^{-1/2}} \Big) \tag{18}$$

$$\iff \min_{\{\mathbf{\Theta}_k \succ 0\}} \sum_{i=1}^{N} \sum_{k=1}^{K} \frac{r_{ik}^{(t)}}{2} (\boldsymbol{x}_i^\top \mathbf{\Theta}_k \boldsymbol{x}_i + D \log 2\pi - \log |\mathbf{\Theta}_k|) \tag{19}$$

By dropping terms do not rely on $\mathbf{\Theta}_k$, we obtain:

$$\min_{\{\mathbf{\Theta}_k \succ 0\}} \sum_{i=1}^{N} \sum_{k=1}^{K} -r_{ik}^{(t)} \big(\log|\mathbf{\Theta}_k| - \boldsymbol{x}_i^\top \mathbf{\Theta}_k \boldsymbol{x}_i\big) \tag{20}$$

Thus, deriving $\mathbf{\Theta}_k$ in M-step is equivalent to solve the following problem,

$$\min_{\mathbf{\Theta_1} \succ 0, \cdots, \mathbf{\Theta_K} \succ 0} \sum_{i=1}^{N} \sum_{k=1}^{K} -r_{ik}^{(t)} (\log|\mathbf{\Theta}_k| - \boldsymbol{x}_i^\top \mathbf{\Theta}_k \boldsymbol{x}_i) \tag{21}$$

Intuitively, above problem is equivalent to K separate conventional graphical lasso sub-problem weighted by $r_{ik}^{(t)}$, where each sub-problem has the form of

$$\min_{\mathbf{\Theta}_k \succ 0} -s_k \log|\mathbf{\Theta}_k| + \boldsymbol{r}_k^\top \mathbf{X}^\top \mathbf{\Theta}_k \mathbf{X} \tag{22}$$

where $\boldsymbol{r}_k = (r_{1k}, \cdots, r_{Nk})^\top$ and $s_k = \sum_{i=1}^{N} r_{ik}$. If we set $\tilde{\mathbf{X}}_k = (\sqrt{r_{1k}/s_k}\boldsymbol{x}_1^\top, \cdots, \sqrt{r_{Nk}/s_k}\boldsymbol{x}_N^\top)$, the above problem can be reformed as follows,

$$\min_{\mathbf{\Theta}_k \succ 0} -\log|\mathbf{\Theta}_k| + \mathrm{tr}(\tilde{\mathbf{X}}_k^\top \mathbf{\Theta}_k \tilde{\mathbf{X}}_k) \tag{23}$$

which takes a similar form to Eq. (2). Then we bring in the adaptive regularization to obtain the final problem for M step,

$$\min_{\{\mathbf{\Theta}_k \succ 0\}} \sum_{k=1}^{K} \big(-\log|\mathbf{\Theta}_k| + \mathrm{tr}(\tilde{\mathbf{X}}_k^\top \mathbf{\Theta}_k \tilde{\mathbf{X}}_k)\big) + \ell_{\lambda_1,\lambda_2}(\{\mathbf{\Theta}_k\}) \tag{24}$$

This problem is not convex *w.r.t.* $\{\mathbf{\Theta}_k\}$, but we could solve it alternatively for each $\mathbf{\Theta}_k$ by regarding other $\mathbf{\Theta}_{k' \neq k}$ fixed. Each sub-problem of $\mathbf{\Theta}_k$ is exactly in the form of Eq. (23) plus the adaptive regularization terms. Thus it could be solved by any existing method for solving Graphical Lasso such as QUIC without significant modifications. In each iteration of M-step, the alternating optimization repeats until all estimated $\mathbf{\Theta}_k$ become stable or reaches the maximal number of iterations.

The final solutions to Eq. (24) and updated $\{\phi_k\}$ obtained using Eq. (15) are used in the upcoming iteration of E-step to update the responsibility weights $\{\boldsymbol{r}_i\}$. This loop of E step and M step repeats until the loss function converges.

The MGL algorithm is also summarized in Algorithm (1).

F. Initialization

As we know from the Algorithm (1), we need to give starting values of each estimators. In the process of comparative experiments, we found that the initialization of the parameters will largely affects the performance of our model. The following scheme we found empirically works well in our experiments. For each observation $i = 1, \ldots, N$, we distribute it randomly a class $k \in \{1, \ldots, K\}$. Then we assign a weight $\hat{r}_{ik} = 0.9$ for this observation i and distribution k and $\hat{r}_{ij} = \frac{0.1}{K-1}$ for all other distributions. In the M-step, we update $\mathbf{\Theta}_k$ from the initial values $\hat{\mathbf{\Theta}}_k^{(0)}$ computed by GLasso based on the whole samples. and ϕ_k from the initial values $\hat{\phi}_k = \frac{1}{K}$.

Algorithm 1 Algorithm for `MGL`

Require: i: $\mathbf{X}$: The observations of D-variate Gaussian distribution
ii: k: the number of Gaussian distributions
iii: λ_1: the Lagrangian multiplier of sparsity constraint
iv: λ_2: The Lagrangian multiplier of mutual exclusivity constraint
v: iter$_{max}$: the maximum number of iteration
Output: $\hat{\mathbf{\Theta}}_\mathbf{k}$, $\hat{\phi}_\mathbf{k}$

Initialization: initialize $\phi_k^{(0)}$, $\mathbf{\Theta}_k^{(0)}$ and $r_{ik}^{(0)}$
repeat
  E step: Update the latent variable $r_{ik}^{(t)}$ with given $\phi_k^{(t-1)}$ and $\mathbf{\Theta}_k^{(t-1)}$
  M step: Update $\phi_k^{(t)}$, $\mathbf{\Theta}_k^{(t)}$ with $r_{ik}^{(t-1)}$
until $iter = iter_{max}$ or convergence

IV. Empirical Study

In this part, we demonstrate the performance of our proposed model through extensive comparative experiments. We evaluate our proposed model in synthetic datasets at first. To comprehensively evaluate proposed model, we conduct experiment to answer the following research questions:

- **RQ 1**: How does MGL perform compared with state-of-the-art models in the consideration of the effect of sample size?
- **RQ 2**: Does our model still show robustness under noise? If the MER regularization term has positive influence on the performance under noise?
- **RQ 3**: How do hyper-parameters in comparative experiments impact each model performance?
- **RQ 4**: Is there a problem with mixture brain network structure in real ADHD-200 datasets?

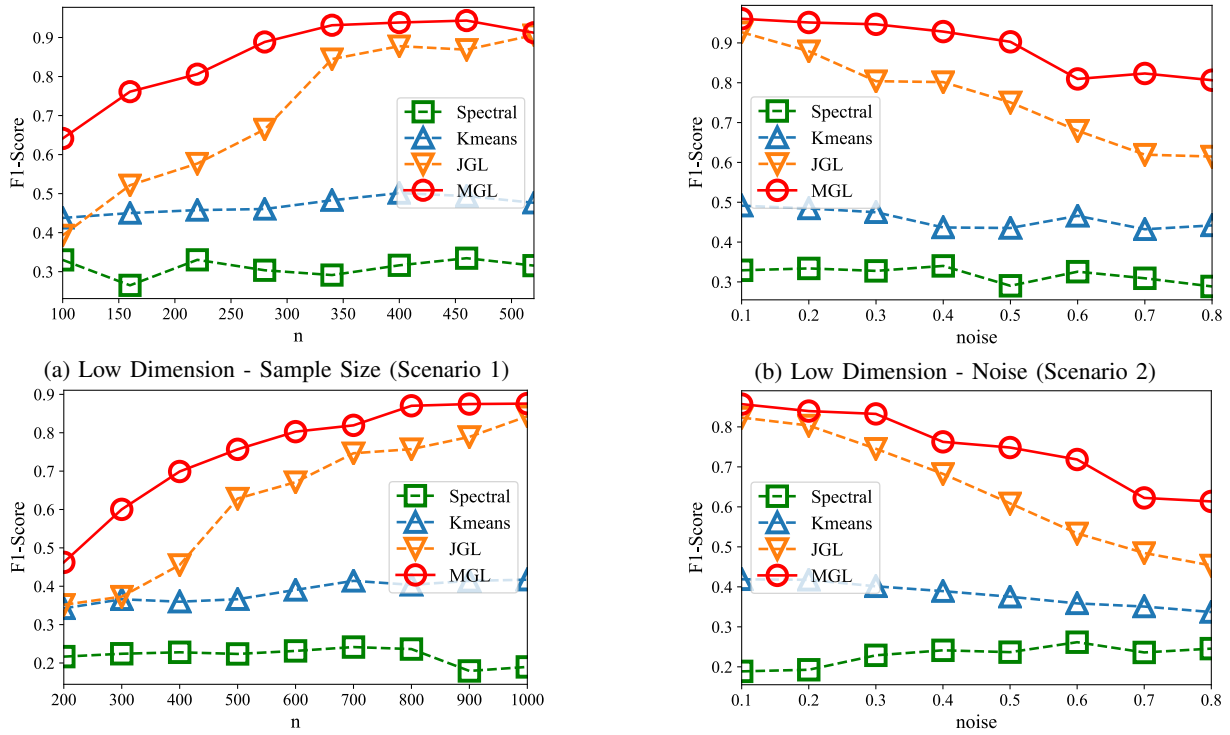
A. Compared Baselines

To demonstrate the effectiveness of our proposed method, we test against several variations of the state-of-art method Graphical Lasso:

- **GLasso + Spectral Clustering**: GLasso algorithm that assumes all data samples are drawn from the same Gaussian distribution, then using Spectral Clustering divide the whole network into several sub-graph.
- ***k*-means + GLasso**: This is a pipeline method that first employs k-means to assign each $\boldsymbol{x}_i$ to different groups, then using GLasso for each group to obtain the final $\mathbf{\Theta}_k$.
- **JGL [16]**: This is the Joint Graphical Model with fused lasso, which is proposed in [16]. It is equivalent to our proposed model without MER term. So it can work as the comparative method for assessing the performance of MER.

B. Synthetic Simulations

Due to the lack of ground truth in many real-world data, we first compare our proposed method against other competitors on several carefully designed synthetic data sets.



(a) Low Dimension - Sample Size (Scenario 1)

(b) Low Dimension - Noise (Scenario 2)

(c) High Dimension - Sample Size (Scenario 3)

(d) High Dimension - Noise (Scenario 4)

Fig. 3: Comparison of each model on edge detection. In scenario 1 and 2, we fix $p = 8$ and $k = 2$; in scenario 3 and 4, we fix $p = 20$ and $k = 2$. In the first column, sample size N is controlled from 100 to 520 in scenario 1, from 200 to 1000 in scenario 3; in the second column, noise is indicates by the standard error σ, controlled from 0.1 to 0.8 in scenario 2 and 4. Each figure shows the results of F1-score. The dark blue line indicates GLasso + Spectral; the light blue indicates k-means + GLasso; the orange one shows the result of MGL without Mutual Exclusivity Regularization and the green one shows the result of MGL.

1) Data Set: In this sub-section, we design some synthetic data sets purposefully. Firstly, we generate k diagonal matrices (k is the number of distribution, which is given in advance), then divide it into several equal-scale blocks. It makes sense for two reasons: we need to control each sub-graphs $\mathbf{\Theta}_k$ with non-overlapping edges on off-diagonal areas; by making edges of each sub-graph more concentrated, it is helpful for making results conductive to visualization. Secondly, we choose different off-diagonal blocks on each $\mathbf{\Theta}_k$, giving connectivities for these chosen blocks with a high density. Following the above steps, we generate each $\mathbf{\Theta}_k$ without overlapping edges on off-diagonal areas. Based on $\mathbf{\Theta}_k$, we compute each $\mathbf{\Sigma}_k$, then select N_k samples ($\sum_{k=1}^{K} N_k = N$) randomly from each Gaussian distribution. In the next subsection, in order to evaluate the stability of our model, we also add noise into the samples. To exclusive the system randomness, we sample 10 times for all experiments, calculate the average of each experiments. So we can evaluate the precision and stability of our model at the same time.

2) Experimental Settings: We simulate four scenarios by controlling one parameter and holding on the others. In these situations, we select sample size N and the standard error of noise σ as the controlled parameters.

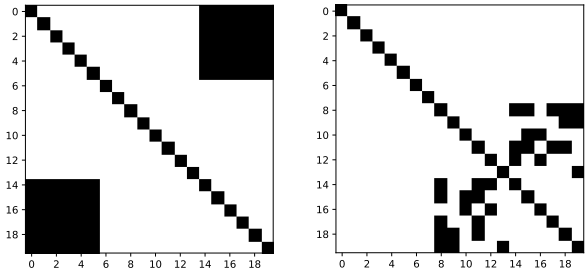
- **Scenario 1**: We fix $p = 8$ (the number of variables), $k = 2$ (the number of Gaussian distributions), and $\sigma = 0$ (the standard error of noise), and then control sample size N from 100 to 520.
- **Scenario 2**: We fix $p = 8$, $k = 2$, and $N = 500$, and then control noise σ from 0.1 to 0.8.
- **Scenario 3**: We fix $p = 20$, $k = 2$, and $\sigma = 0$, and then control sample size N from 200 to 1000.
- **Scenario 3**: We fix $p = 20$, $k = 2$, and $N = 1000$, and then control noise σ from 0.1 to 0.8.

3) Evaluation: To evaluate the quality of each sub-graph, we follow the method of [30] to define the F1-score of edge detection as follows,

$$F1 = \frac{2N_d^2}{N_a N_d + N_g N_d} \tag{25}$$

where N_d is the number of true edges detected by the model, N_g is the number of true edges and N_a is the total number of edges detected. According to the expression, higher F1-score indicates better quality of edge detection.

Figure 3 shows the comparison between MGL and other baseline models. The results in the figure answer the first three **RQ** mentioned at the beginning of this section. The



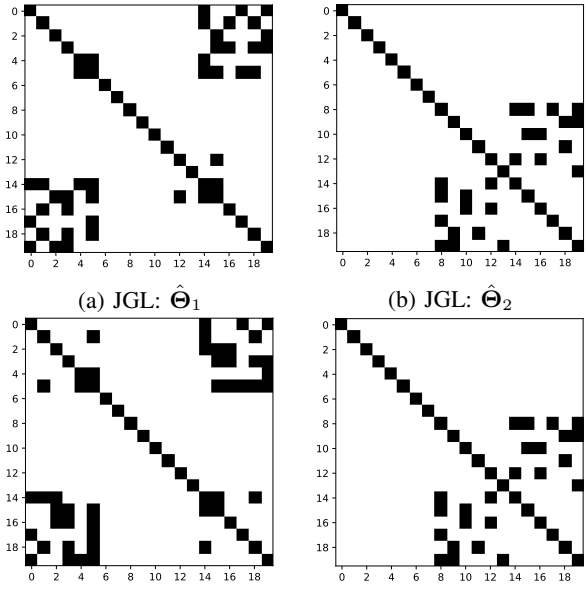
(a) Precision matrix $\mathbf{\Theta}_1$ (b) Precision matrix $\mathbf{\Theta}_2$

Fig. 4: True precision matrices of data sets ($p = 20$ and $k = 2$) in scenario 3 and 4. In the consideration of elements in off-diagonal area, white indicates no directed relationship and black indicates directed relationship between each variable.

first column shows the results when we control sample size N and hold on the others, which corresponds to **RQ1**. It is obvious that k-means and Spectral models are useless when the ground truth data sets are drawn from mixture Gaussian distribution. Meanwhile, when the sample size is not large enough, the precision of JGL is lower than that with MGL. The second column shows the results when we control noise, which corresponds to **RQ2**. We fix the sample size N on 500, so when $\sigma = 0$, JGL is as good as MGL. According to the results, The louder the noise, the worse JGL performs, which means sensitive to the noise.So the result demonstrates that MER regularization can improve the performance of our proposed model. Compared to the others, MGL shows robustness in this scenario. To answer **RQ3**, we can figure out the answer from both column in this figure. Since our experiments are setting in low-dimensional and high-dimensional space separately, we can see from all comparison results that the issue of hyper-parameters does not affect the performance of MGL. In contrast, the performance of JGL in high-dimensional space isn't as well as that in the low-dimensional space, no matter in the scenario of sample size or noise. In summary, in the comparative experiment of synthetic datasets with ground-truth, our proposed method MGL shows better accuracy and robustness than that of other comparison methods.

Figure 3 has exhibited the overall performance of all methods. To better understand the effectiveness of MGL, we also show the visualization of these scenarios, which can more intuitively reflect the problem of multiple mixture network and the performance of each method. Figure 4 shows the true precision matrices of data sets in scenario 3 and 4. It indicates that these data sets are drawn from two independent Gaussian distributions, which is consistent with the defined problem in the paper. According to the color bar, in the consideration of elements in off-diagonal area, white indicates no directed relationship and black indicates directed relationship between each variable. So it is obvious that the precision matrices of them have non-overlapping area with each others.

Figure 5 shows the comparison when sample size $N = 1000$ and no noise exists in the data sets. We filter the precision matrix into a matrix with only 0 and 1 by a threshold close



(a) JGL: $\hat{\mathbf{\Theta}}_1$ (b) JGL: $\hat{\mathbf{\Theta}}_2$

(c) MGL: $\hat{\mathbf{\Theta}}_1$ (d) MGL: $\hat{\mathbf{\Theta}}_2$

Fig. 5: Estimated precision matrices of data sets ($p = 20$ and $k = 2$) when $N = 1000$ and $\sigma = 0$ (no noise).

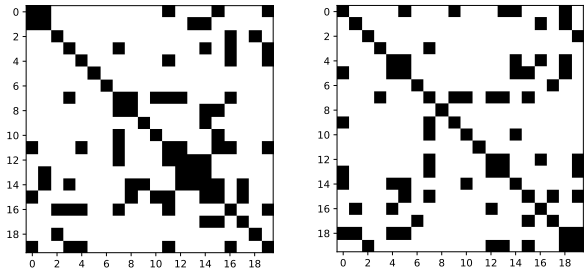
to zero. So according to the figure, as long as the sample is large enough and no noise exists, these two models can figure out ground truth mixture distribution, no matter whether considering the mutual exclusivity term. However, when the sample size is insufficient or there is noise in the datasets (Figure 6 and 7), JGL begins to show malfunction, which indicates that this model is sensitive to sample size N or noise. According to the second row of them, the mutual exclusivity regularization can solve this problem to some extent. MGL can still get results that are very close to the real situation, which demonstrate that MER improves the performance of MGL under small sample size. In summary, we believe that MGL is more accurate and robust to mixture Gaussian data sets with non-overlapping constraint. The mutual exclusivity regularization works well on small samples or noisy data sets.

C. Real fMRI Data

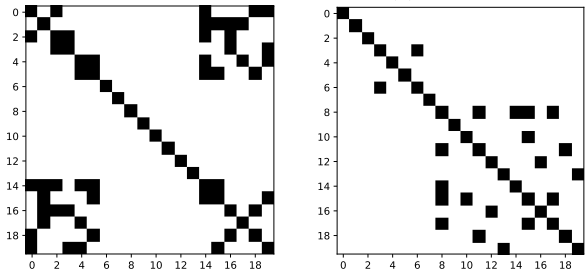
In the subsection, we evaluate our proposed method on fMRI dataset from ADHD-200 project[1]. ADHD, which is also called Attention Deficit Hyperactivity Disorder, is a chronic condition. This condition has been happened on 5% - 10% of school-age children. Through this paper, we discover the network discovery from a collection of fMRI scans, in which each sample corresponds to a 4D brain image (a sequence of 3D images) of a subject. These scans are usually transferred into time series of voxels in the 3D images space. Consequently in practice with real fMRI cases, node is a set of voxels in 3D brain images that are similar to each other in function. Our real world dataset is distributed by nilearn[2]. Specifically, there are 40 subjects in total. Among them, 20 subjects are labeled

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

[2]http://nilearn.github.io/

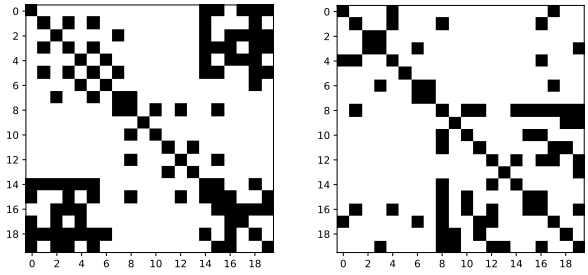


(a) JGL: $\hat{\mathbf{\Theta}}_1$ (b) JGL: $\hat{\mathbf{\Theta}}_2$

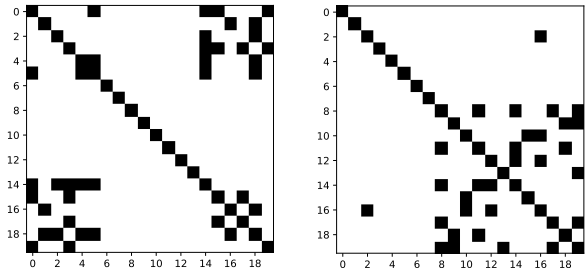


(c) MGL: $\hat{\mathbf{\Theta}}_1$ (d) MGL: $\hat{\mathbf{\Theta}}_2$

Fig. 6: Estimated precision matrices of data sets ($p = 20$ and $k = 2$) when $N = 400$ and $\sigma = 0$(no noise).



(a) JGL: $\hat{\mathbf{\Theta}}_1$ (b) JGL: $\hat{\mathbf{\Theta}}_2$

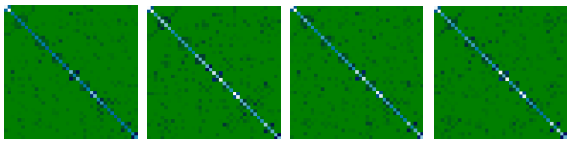


(c) MGL: $\hat{\mathbf{\Theta}}_1$ (d) MGL: $\hat{\mathbf{\Theta}}_2$

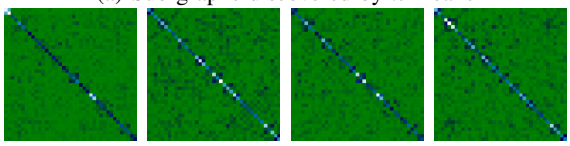
Fig. 7: Estimated precision matrices of data sets ($p = 20$ and $k = 2$) when $N = 1000$ and $\sigma = 0.5$.

as ADHD, and the others are labeled as TDC. The fMRI scan of each subject in the dataset is a series of snapshots of 3D brain images of size $61 \times 76 \times 61$ over $\sim$176 time steps. In our experiment, we only choose the subjects which are labeled as ADHD.

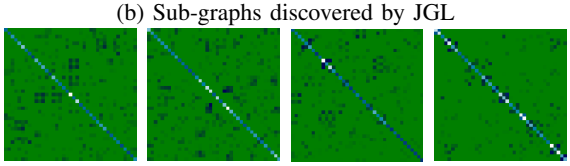
Because real fMRI data lacks ground-truth as a reference to measure the accuracy and robustness of the model. We



(a) Sub-graphs discovered by k-means



(b) Sub-graphs discovered by JGL



(c) Sub-graphs discovered by MGL

Fig. 8: Comparison of k-means + GLasso, JGL and MGL on ADHD dataset. The results show how to estimate a mixture connectivity structure on a group of subjects using different group sparse inverse covariance estimation models from real fMRI data set. The closer the color of elements in off-diagonal is to blue, the bigger probability the directed edges between corresponding nodes.

are more concerned with the interpretability and rationality of the results. Specific to our proposed model, we are more concerned about whether our model can mine different connectivity structures among nodes from the fMRI datasets. Throughout this subsection, we still make horizontal comparisons of the models mentioned in synthetic datasets, in order to compare the different results of each method on this fMRI datasets.

In our experiment, we only choose the subjects which are labeled as ADHD. We focus on the multiple connectivity structures among the same subjects, in order to provide evidence on feature selection between different subjects in further study. Rather than discover the brain network on the level of voxels, we extracts the signal on regions defined via a probabilistic atlas, to construct the data sets. So it is more conventional for visualization of the results. The data sets is a 1899×39 data sets and we consider that they are drawn from a mixture Gaussian distribution. However, the number k of it is unknown, which need to be given in advance. Through repeated experimental observations, we found that $k = 4$ can provide the most reasonable results on the data sets.

In the Figure 8, the first row indicates the results of k-means plus GLasso; the second one indicates JGL and the third one indicates MGL. According to the results, we can find that there are almost no differences among four sub-graphs discovered by k-means plus GLasso. It indicates that this method is useless for mining sub-graphs in ADHD data sets. JGL shows four different sub-graphs, however, so many overlapped areas among them. These results seem not to

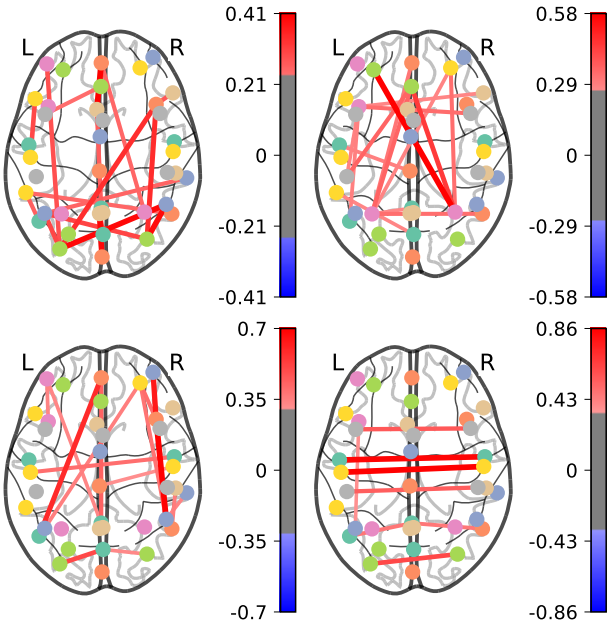


Fig. 9: We turn the results of Fig. 8 into connectome for visualization. Each precision matrix is displayed on glass brain on extracted coordinates. These graphs of precision matrices discovered by MGL in ADHD dataset. The closer the color of edge is to red, the stronger the directed relationship between corresponding nodes.

be sparse matrices, which indicates that the corresponding connectivity structure is not very clear through this method. Compared to it, sub-graphs discovered by MGL is clearer and the number of overlapped areas is less. Therefore, although lacking the ground truth in ADHD data, we can still believe that the inferred results of MGL is consistent with the defined problem in this paper, especially in the consideration of mixture Gaussian distribution with non-overlapping areas among their precision matrices. The Figure 9 shows the corresponding connectivity structure of the results discovered by MGL. Here we only choose the axial direction of the cuts to show. The closer the color is to red, the stronger the directed relationship between the corresponding nodes. We highlight the stronger edges by adjusting the threshold of colorbar. According to the visualization of results, we can see that different sub-graphs highlight different relationships among all nodes. Different sub-graphs emphasize the relationships of different nodes, which means that subjects present different network structures on the time-line. This phenomenon is more obvious between the nodes related to DMN (default mode network), which includes the Parietal, Occipital Lobes, the Cingulum Region Posterior and the Frontal Cortex. Although the hypothesis about non-overlapped areas among each connectivity structure may not exist in real ADHD subjects, we believe that MGL with MER regularization can more prominently show the difference between each connectivity structures discovered, so that we can have a better understanding of the association between cognitive network and human activities.

According to the analysis above, despite the lack of ground-truth, we believe that the existing results are still consistent with the problem defined in this paper. So the result shows that there is a mixture connectivity structure among nodes in the fMRI datasets, and our proposed model MGL can effectively mine this mixture connectivity structure.

V. Related Work

In the section, we introduce some existing related methods from two perspectives: edge detection and mixture network discovery.

In the consideration of edge detection, this issue has two major branches: effective connectivity estimation and functional connectivity estimation. For the first branch, scholars pay more attention on obtaining a directed network from fMRI data through structure learning method for Bayesian networks [18]. In contrast, the second branch focuses on some approaches such as hierarchical clustering, pairwise correlations and independent component analysis, which can be found in [15] for more details. In this paper, we focus on sparse gaussian graphic models [14], [4], which are a very useful for discovering connectivity of brain network based on large-scale dataset by using sparse inverse covariance estimation. The main ideas of these methods are that they can distinguish direct links from indirect connections due to their solid probabilistic foundation. However, in the task of edge detection, these methods focus on unimodal distributions, where it is usually assumed that the observed samples are drawn from a single Gaussian distribution, which is opposed to some recent studies [20], [3]. The Joint Graphical Model with fused lasso, which is proposed in [16], is in the framework of multivariate Gaussian mixture modeling. However, this method has shown to be sensitive to the noise and small size of the data sample.

Secondly, we consider the Gaussian Mixture Model (GMM), which is introduced by Pearson in [24]. It models the distribution of data observations as a weighted sum of parameterized Gaussian distribution. In later extended researches, an obvious issue related to GMM is estimating the parameters given observations. EM algorithm in [10] has proven to be a powerful algorithm for the maximum-likelihood estimation of GMM. In addition, [25], [2] consider the issue of the number of mixture components in model, which can lead to over-fitting in practice. Nowadays, GMM has been widely used in many areas, especially in network discovery [22], [21], [32]. The majority of the existing studies on mixture modeling focus on regularizing only the mean parameters with diagonal covariance matrices [23], [27], [29], though some [31], [17], [28] have started considering regularization of the covariance parameters too, all of which, however do not touch on the key issue of identifying the varying sparse structures of the precision matrices across the components of a mixture model in brain network discovery. [16] propose a joint graphical model called JGL, which is combined with fused lasso, to deal with cluster-specific networks. This method aims at discovering both the commonalities and diversities across the

multiple precision matrices through possibly nonconvex fusion regularization. However, this method does not fully highlight the non-overlapping areas of the substructures, meanwhile it is unstable to noise and small data set.

Based on the above discussion, we find out that the existing models related to GMM or GLasso are not suitable for the problem we define in this paper.

VI. Conclusion

Through this paper, we aims at addressing the question of interest here: how to discover different connectivity substructures between a set o nodes based upon the observed node activities in brain network discovery. Existing sparse Gaussian graphical models always give the same network for all populations unless the parcellation of the data set has been finished before. On the other hand, the methods related to mixture brain network discovery ignore the direct connectivity among the nodes meanwhile show lack of robustness to noisy observations and small sample. We propose embedding one of the current methods of estimating multiple Gaussian graphical models in the framework of Gaussian mixture modeling, then design a new regularization term, called mutual exclusivity regularization, to make sub-graphs un-overlapped with each other. Meanwhile, we develop the EM algorithm on our model. Through extensive controlled experiments, we demonstrate that our proposed model MGL shows more effectiveness than other baseline models, meanwhile, MGL shows more robustness than JGL, especially in the consideration of small samples or noisy data sets. In addition, this conclusion is also demonstrated in the experiment of real fMRI brain scanning datasets from ADHD subjects. So we have reason to believe that, our method can also be applied in other domains when network connectivity structure is very complex.

References

[1] Mehran Ahmadlou, Hojjat Adeli, and Amir Adeli. Graph theoretical analysis of organization of functional brain networks in adhd. *Clinical EEG and Neuroscience*, 43(1):5–13, 2012.

[2] Hirotugu Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[3] Alana J Anderson and Sammy Perone. Developmental change in the resting state electroencephalogram: insights into cognition and the brain. *Brain and Cognition*, 126:40–52, 2018.

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

[5] Baker J.R. Buchbinder B.R. Stern C.E. Kwong K.K. Cosgrove G.R. Alpert N.M. Belliveau, J.W. and B.R. Rosen. fmri brain mapping: Clinical applications of functional magnetic resonance imaging. In *Cerebrovascular Diseases*, pages 519–539. 1995.

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[7] Ed Bullmore and Olaf Sporns. The economy of brain network organization. *Nature Reviews Neuroscience*, 13(5):336, 2012.

[8] Miao Cao, Ni Shu, Qingjiu Cao, Yufeng Wang, and Yong He. Imaging functional and structural brain connectomics in attention-deficit/hyperactivity disorder. *Molecular Neurobiology*, 50(3):1111–1123, 2014.

[9] Ian Davidson, Sean Gilpin, Owen Carmichael, and Peter Walker. Network discovery via constrained tensor analysis of fmri data. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 194–202, 2013.

[10] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (methodological)*, pages 1–38, 1977.

[11] Thomas Martin Deserno. *Biomedical image processing*. 2011.

[12] Ibai Diez and Jorge Sepulcre. Neurogenetic profiles delineate large-scale connectivity dynamics of the human brain. *Nature Communications*, 9(1):1–10, 2018.

[13] Anna Fabijanska and Dominik Sankowski. Edge detection in brain images. In *2008 International Conference on Perspective Technologies and Methods in MEMS Design*, pages 60–62, 2008.

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

[15] Karl J Friston. Functional and effective connectivity: a review. *Brain Connectivity*, 1(1):13–36, 2011.

[16] Chen Gao, Yunzhang Zhu, Xiaotong Shen, and Wei Pan. Estimation of multiple networks in gaussian mixture models. *Electronic Journal of Statistics*, 10:1133, 2016.

[17] Steven M Hill and Sach Mukherjee. Network-based clustering with mixtures of l1-penalized gaussian graphical models: an empirical investigation. *arXiv preprint arXiv:1301.2194*, 2013.

[18] Shuai Huang, Jing Li, Jieping Ye, Adam Fleisher, Kewei Chen, Teresa Wu, and Eric Reiman. Brain effective connectivity modeling for alzheimer's disease by sparse gaussian bayesian network. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 931–939, 2011.

[19] Xiangnan Kong and Philip S Yu. Brain network analysis: a data mining perspective. *ACM SIGKDD Explorations Newsletter*, 15(2):30–38, 2014.

[20] Pan Lin, Yong Yang, Junfeng Gao, Nicola De Pisapia, Sheng Ge, Xiang Wang, Chun S Zuo, James Jonathan Levitt, and Chen Niu. Dynamic default mode network across different brain states. *Scientific Reports*, 7:46088, 2017.

[21] Geoffrey McLachlan and David Peel. Finite mixture models, willey series in probability and statistics, 2000.

[22] Mark EJ Newman and Elizabeth A Leicht. Mixture models and exploratory analysis in networks. *Proceedings of the National Academy of Sciences*, 104(23):9564–9569, 2007.

[23] Wei Pan and Xiaotong Shen. Penalized model-based clustering with application to variable selection. *Journal of Machine Learning Research*, 8(May):1145–1164, 2007.

[24] Karl Pearson. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, 185:71–110, 1894.

[25] Gideon Schwarz et al. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

[26] Liang Sun, Rinkal Patel, Jun Liu, Kewei Chen, Teresa Wu, Jing Li, Eric Reiman, and Jieping Ye. Mining brain region connectivity for alzheimer's disease study via sparse inverse covariance estimation. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1335–1344, 2009.

[27] Sijian Wang and Ji Zhu. Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, 64(2):440–448, 2008.

[28] Meng Yun Wu, Dao Qing Dai, Xiao Fei Zhang, and Yuan Zhu. Cancer subtype discovery and biomarker identification via a new robust network clustering algorithm. *PloS One*, 8(6), 2013.

[29] Benhuai Xie, Wei Pan, and Xiaotong Shen. Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. *Electronic Journal of Statistics*, 2:168, 2008.

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

[31] Hui Zhou, Wei Pan, and Xiaotong Shen. Penalized model-based clustering with unconstrained covariance matrices. *Electronic Journal of Statistics*, 3:1473, 2009.

[32] Yunzhang Zhu, Xiaotong Shen, and Wei Pan. Structural pursuit over multiple undirected graphs. *Journal of the American Statistical Association*, 109(508):1683–1696, 2014.

[33] Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.