Structural Graphical Lasso for Learning Mouse Brain Connectivity

Sen Yang
IDST at Alibaba Group, USA
senyang.sy@alibaba-inc.com

Qian Sun
Arizona State University, USA
qsun21@asu.edu

Shuiwang Ji
Old Dominion University, USA
sji@cs.odu.edu

Peter Wonka
King Abdullah University of Science and Technology, Saudi Arabia
peter.wonka@kaust.edu.sa

Ian Davidson
University of California, USA
davidson@cs.ucdavis.edu

Jieping Ye
University of Michigan, USA
jpye@umich.edu

ABSTRACT

Investigations into brain connectivity aim to recover networks of brain regions connected by anatomical tracts or by functional associations. The inference of brain networks has recently attracted much interest due to the increasing availability of high-resolution brain imaging data. Sparse inverse covariance estimation with lasso and group lasso penalty has been demonstrated to be a powerful approach to discover brain networks. Motivated by the hierarchical structure of the brain networks, we consider the problem of estimating a graphical model with tree-structural regularization in this paper. The regularization encourages the graphical model to exhibit a brain-like structure. Specifically, in this hierarchical structure, hundreds of thousands of voxels serve as the leaf nodes of the tree. A node in the intermediate layer represents a region formed by voxels in the subtree rooted at that node. The whole brain is considered as the root of the tree. We propose to apply the tree-structural regularized graphical model to estimate the mouse brain network. However, the dimensionality of whole-brain data, usually on the order of hundreds of thousands, poses significant computational challenges. Efficient algorithms that are capable of estimating networks from high-dimensional data are highly desired. To address the computational challenge, we develop a screening rule which can quickly identify many zero blocks in the estimated graphical model, thereby dramatically reducing the computational cost of solving the proposed model. It is based on a novel insight on the relationship between screening and the so-called proximal operator that we first establish in this paper. We perform experiments on both synthetic data and real data from the Allen Developing Mouse Brain Atlas; results demonstrate the effectiveness and efficiency of the proposed approach.

KDD'15, August 11-14, 2015, Sydney, NSW, Australia.
 ISBN 978-1-4503-3664-2/15/08 ...$15.00.
DOI: http://dx.doi.org/10.1145/2783258.2783391.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*

General Terms

Algorithms

Keywords

Graphical lasso, tree-structural regularization, screening, second-order method, proximal operator, brain networks

1. INTRODUCTION

The rich behavior of numerous complex systems is rooted in the underlying networks governing element interactions. For example, cells are best described as networks of molecules connected by chemical reactions; brains are commonly represented as networks comprising a set of neurons interconnected by their communication pathways; our society is characterized by a network of individuals connected by social relationships. In reality, it is usually the behaviors of individual elements, rather than their interactions, that are directly measurable. This gives rise to the central problem of how to identify system interaction structures by reasoning backwards from the observed behaviors of the individual elements, a process known as network modeling.

Undirected graphical models explore relationships among a set of random variables through their joint distribution. The estimation of undirected graphical models has applications in many domains, such as biology and medicine [10, 12, 33, 27]. One 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 [6]. 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 interactions. One of the most widely used graphical models is the Gaussian graphical model (GGM). In the GGM, the variables are assumed to follow a Gaussian distribution [4, 35]. Then the problem of learning a graphical model is equivalent to estimating the inverse of the covariance matrix (pre-

cision matrix), since the nonzero off-diagonal elements of the precision matrix represent edges in the graph [4, 35].

The main challenge of estimating a sparse precision matrix for problems with a large number of nodes (variables) is its intensive computation. Witten *et al.* [30] and Mazumder and Hastie [21] independently derived a necessary and sufficient condition for the solution of a single graphical lasso to be block diagonal (subject to some rearrangement of variables). This can be used as a simple screening test to identify the associated blocks, and the original problem can thus be decomposed into a group of smaller sized but independent problems corresponding to these blocks. When the number of blocks is large, it can achieve massive computational speedup. However, these formulations only assume that the graph is sparse. In many applications, domain structural knowledge exists and can potentially be exploited to improve the learning performance; in this case, structural regularization can be used to improve the estimation of graphical model. However, due to the complexity of structural regularization, it is challenging to derive screening rules for general structural regularization.

To attack the above central challenge, we derive a screening rule for structural Graphical Lasso in this paper. Specifically, we show that the derivation of the screening rule is critically dependent on the so-called "proximal operator" associated with the structural regularization, *e.g.*, Lasso, group Lasso, or tree Lasso penalty. In recent years, tremendous efforts have been devoted to the efficient computation of the proximal operators, which plays a central role in structured sparse learning [3, 34]. In many cases, the proximal operators, such as Lasso penalty, group lasso penalty, and tree group penalty, have closed form solutions, thereby leading to very efficient computation [17]. One of our major contributions in this work is to establish a bridge between the computation of the proximal operator associated with a structural regularization and the derivation of a screening rule for structural Graphical Lasso. To the best of our knowledge, our work represents the first attempt to construct screening rules for the **Structural Graphical Lasso** based on general structural regularization.

The major contributions of this work are summarized as follows:

- We propose a structural Graphical Lasso formulation based on a general structural regularization imposed on the nodes, *e.g.*, a tree structure. The proposed formulation estimates a sparse precision matrix which is encouraged to satisfy certain structures.
- We derive a screening rule to identify a block diagonal structure of the resulting network. The original large-size precision matrix can thus be decomposed into a set of smaller sized blocks. Such decomposition can potentially lead to massive computational speedup. One of our key technical contributions of this work is the establishment of the intrinsic relationship between screening rules and proximal operators.
- The proposed screening is safe in the sense the screening does not affect the final solution. In addition, the proposed screening only relies on the data and the parameters, thus it can be combined with any existing algorithms to reduce the computational cost.
- We evaluate the proposed screening rule and the proposed model using both synthetic data and real data from the Allen Developing Mouse Brain Atlas. Results demonstrate the effectiveness and efficiency of the proposed methods.

Notation: In this paper, $\Re$ stands for the set of all real numbers, $\Re^n$ denotes the n-dimensional Euclidean space, and the set of all $m \times n$ matrices with real entries is denoted by $\Re^{m\times n}$. All matrices are presented in bold format. The space of symmetric matrices is denoted by $\mathcal{S}^n$. If $\mathbf{X} \in \mathcal{S}^n$ is positive semidefinite (resp. definite), we write $\mathbf{X} \succeq 0$ (resp. $\mathbf{X} \succ 0$). The cone of positive semidefinite matrices in $\mathcal{S}^n$ is denoted by $\mathcal{S}^n_+$. Given matrices $\mathbf{X}$ and $\mathbf{Y}$ in $\Re^{m\times n}$, the standard inner product is defined by $\langle \mathbf{X}, \mathbf{Y}\rangle := \mathrm{tr}(\mathbf{X}\mathbf{Y}^T)$, where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix. The determinant of a real symmetric matrix $\mathbf{X}$ is denoted by $\det(\mathbf{X})$. Given a matrix $\mathbf{X} \in \Re^{n\times n}$, $\mathrm{diag}(\mathbf{X})$ denotes the vector formed by the diagonal of $\mathbf{X}$, *i.e.*, $\mathrm{diag}(\mathbf{X})_i = \mathbf{X}_{ii}$ for $i = 1, \ldots, n$.

Organization: The rest of this paper is organized as follows. We introduce the structural graphical lasso formulation as well as the screening rule in Section 2. The experimental results for both synthetic data and real data are shown in Section 3. Related work is discussed in Section 4. We conclude the paper in Section 5.

2. STRUCTURAL GRAPHICAL LASSO

Suppose we are given a data set $\mathbf{X} \in \Re^{n\times p}$ with n samples, and p features (or variables). The n samples are independently and identically distributed with a p-variate Gaussian distribution with zero mean and positive definite covariance matrix $\mathbf{\Sigma}$. Even if all features are correlated, there are usually many conditional independences among these features. In other words, a sparse precision matrix $\mathbf{\Theta} = \mathbf{\Sigma}^{-1}$ is of interest in most cases. This Gaussian graphical model (GMM) is also referred to as *Gaussian Markov Random Field (GMRF)*. The negative log likelihood for the data $\mathbf{X}$ takes the form of

$$\mathcal{L}(\mathbf{\Theta}) := -\log\det(\mathbf{\Theta}) + \mathrm{tr}(\mathbf{S}\mathbf{\Theta}), \tag{1}$$

where $\mathbf{S}$ is the sample covariance matrix given by $\mathbf{S} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$. Minimizing (1) leads to the maximum likelihood estimation (MLE) $\mathbf{\Theta}^* = \mathbf{S}^{-1}$. However, there are some issues with MLE. In particular, it fails in the high-dimensional setting ($n < p$), where MLE $\mathbf{\Theta}^*$ does not exist due to the singularity of $\mathbf{S}$. To handle this issue, regularization is usually employed, resulting in penalized maximum likelihood estimation. For applications with prior domain knowledge, different regularization terms can be employed to encourage the estimated model to satisfy the desired structural property. For example, a common assumption is that the graphical model is sparse. In this case, the ℓ_1 regularization has been employed to encourage sparsity [8]. In this paper, we consider the general structural graphical lasso, which integrates the structural regularization as follows:

$$\min_{\mathbf{\Theta}\succ 0} -\log\det(\mathbf{\Theta}) + \mathrm{tr}(\mathbf{S}\mathbf{\Theta}) + \phi(\mathbf{\Theta}), \tag{2}$$

where $\phi(\mathbf{\Theta})$ is the convex structural regularization. We refer to problem (2) as structural graphical lasso (SGL). Examples include but are not limited to

- Sparsity: $\phi(\mathbf{\Theta}) = \lambda\|\mathbf{\Theta}\|_1$

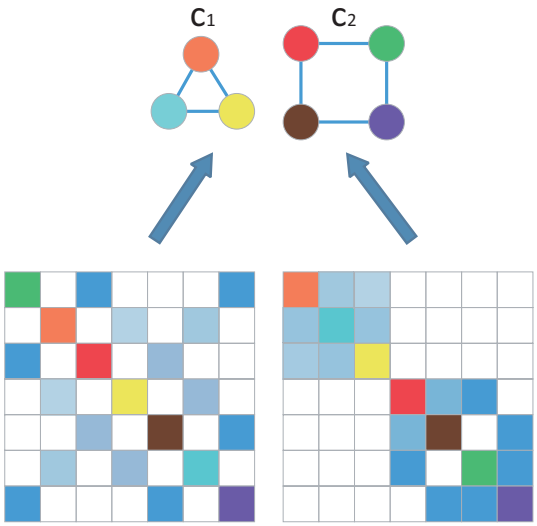

Figure 1: Illustration of two precision matrices (bottom) whose nodes are in different order correspond to the same graph with two connected components (top). The white color in the precision matrices represents a zero entry.

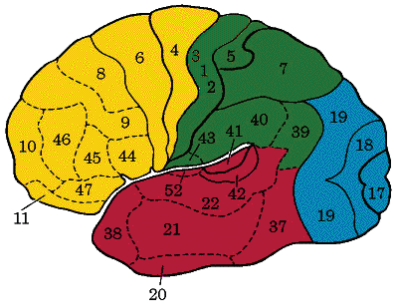

Figure 2: Illustration of the brain[2]. Yellow: frontal lobe; green: parietal lobe; red: temporal lobe; blue: occipital lobe. Number represents brain regions within lobes.

- Group sparsity: $\phi(\mathbf{\Theta}) = \lambda \sum_{i,j} \|\mathbf{\Theta}_{G_i,G_j}\|_F$.

The penalized log likelihood function with a convex regularizer, *i.e.*, problem (2), is strictly convex, however, the minimum of problem (2) may not be achievable. This is usually dependent on the property of the sample covariance matrix $\mathbf{S}$. For example, $\text{diag}(\mathbf{S}) > 0$ is a sufficient condition for problem (2) to have a unique solution [32] when the ℓ_1 regularization exists. For simplicity of presentation, we assume throughout the paper that the minimum of problem (2) can be achieved, *i.e.*, problem (2) has a unique solution.

The remainder of this section is organized as follows. We introduce a Tree-Guided Graphical Lasso formulation in Section 2.1. In Section 2.2, we present a second-order method to efficiently solve the proposed model. In addition, we derive a sufficient condition for estimating many zero blocks in the graph in Section 2.3. Based on this property, we propose a simple screening rule which significantly reduces the complexity of the optimization problem, thus improving the computational efficiency. The proposed screening only relies on the data and the parameters, thus it can be combined with any existing algorithms to reduce the computational cost. We discuss two special cases in Section 2.4.

[2] `http://www.umich.edu/~cogneuro/jpg/Brodmann.html`

2.1 Tree-Guided Graphical Lasso Formulation

In this subsection, we present a hierarchical graphical model framework where the features exhibit a hierarchical structure. A motivating example is the estimation of brain networks. The brain is a multi-level system, and the brain network has a native hierarchical structure as shown in Figure 2: hundreds of thousands of voxels form regions, and regions form systems.

We employ the tree-structural group regularization to encourage the estimated graph to have a hierarchical structure. Specifically, in this hierarchical structure, hundreds of thousands of voxels serve as the leaf nodes of the tree. A node in the intermediate layer represents a region formed by voxels in the subtree rooted at that node. The whole brain is considered as the root of the tree. Mathematically, we solve the following formulation:

$$\min_{\mathbf{\Theta} \succ 0} -\log\det(\mathbf{\Theta}) + \text{tr}(\mathbf{S}\mathbf{\Theta}) + \phi(\mathbf{\Theta}) \tag{3}$$

where

$$\phi(\mathbf{\Theta}) = \sum_j \left(\sum_{i \neq i'} w^j_{ii'} \|\mathbf{\Theta}_{G^j_i, G^j_{i'}}\|_F + w^j_{ii} \|\mathbf{\Theta}_{G^j_i, G^j_i, off}\|_F \right),$$

G^j_i is the i-th group at depth j (the groups of a tree are defined in Definition 1 below; see Figure 3 for an illustration), $\mathbf{\Theta}_{G^j_i, G^j_{i'}}$ denotes the submatrix of $\mathbf{\Theta}$ consisting of features in $G^j_i, G^j_{i'}$, and $w^j_{ii'} = w^j_{i'i}$ is a positive weight for $\mathbf{\Theta}_{G^j_i, G^j_{i'}}$. $\mathbf{\Theta}_{\cdot,\cdot,off}$ represents the matrix $\mathbf{\Theta}_{\cdot,\cdot}$ excluding the diagonal elements. We do not penalize the diagonal elements of $\mathbf{\Theta}$ since $\mathbf{\Theta}$ is required to be positive definite. For simplicity of notation, we use $\mathbf{\Theta}^j_{ii'}$ to represent $\mathbf{\Theta}_{G^j_i, G^j_{i'}}$, and $\mathbf{\Theta}^j_{ii/}$ to represent $\mathbf{\Theta}_{G^j_i, G^j_i, off}$. It is clear that $\mathbf{\Theta}^j_{ii'} = (\mathbf{\Theta}^j_{i'i})^T$, thus we require $w^j_{ii'} = w^j_{i'i}$. The regularization $\phi(\mathbf{\Theta})$ encourages the estimated precision matrix to be tree-structural (see Figure 4 for an example). We formally define a tree structure as follows:

DEFINITION 1. *[18] For an index tree T of depth U, we let $T_u = \{G_1, \ldots, G_{n_i}\}$ contain all the nodes corresponding to depth u, where $n_0 = 1$, $G^0_1 = \{1, \ldots, K\}$ and $n_i \geq 1, i = 1, \ldots, U$. The nodes satisfy the following conditions:*

- *The nodes from the same depth level have non-overlapping indices,* i.e., *$G^u_j \cap G^u_k = \emptyset, \forall u = 1, \ldots, U, j \neq k, 1 \leq j, k \neq n_i$;*
- *Let G^{u-1}_{j0} be the parent node of a non-root node G^u_j, then $G^u_j \subseteq G^{u-1}_{j0}$.*

2.2 Algorithm

We propose to employ the second-order method to solve the tree-guided graphical lasso problem in (3) as it has been shown to be quite efficient for solving the Graphical Lasso formulation with ℓ_1 regularization [11]. Let $f(\mathbf{\Theta})$ be the smooth function in (3) such that

$$f(\mathbf{\Theta}) = -\log\det(\mathbf{\Theta}) + \text{tr}(\mathbf{S}\mathbf{\Theta}).$$

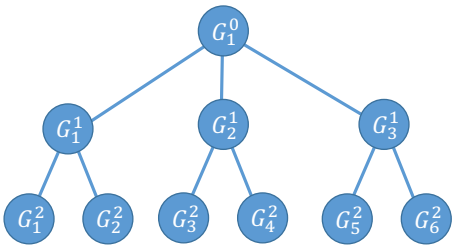

Figure 3: A sample index tree. Root: $G_1^0 = \{1,2,3,4,5,6,7,8\}$. **Depth 1:** $G_1^1 = \{1,2\}$, $G_2^1 = \{3,4,5,6\}$, $G_3^1 = \{7,8\}$. **Depth 2:** $G_1^2 = \{1\}$, $G_2^2 = \{2\}$, $G_3^2 = \{3,4,5\}$, $G_4^2 = \{6\}$, $G_5^2 = \{7\}$, $G_6^2 = \{8\}$.

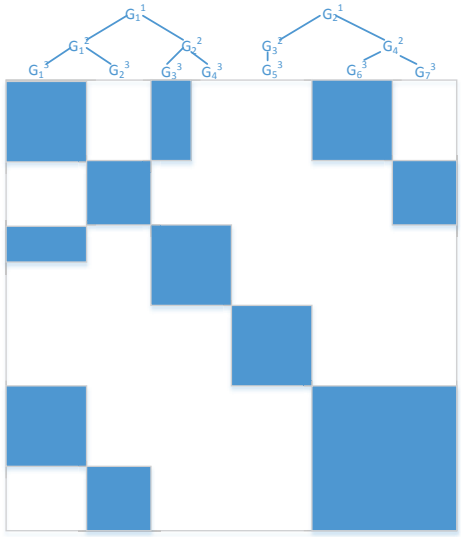

Figure 4: Illustration of a hierarchical graphical model. The features exhibit a hierarchical structure specified by tree groups $\{G_i^j\}$. The blue blocks represent the nonzero blocks in the precision matrix.

(3) can be rewritten as

$$\min_{\mathbf{\Theta} \succ 0} f(\mathbf{\Theta}) + \phi(\mathbf{\Theta}). \tag{4}$$

In the second-order method, we solve a "quadratic" model of (3) at each iteration defined by

$$\min_{\mathbf{\Theta}} \frac{1}{2}\mathrm{tr}(\mathbf{W}_t\mathbf{D}\mathbf{W}_t\mathbf{D}) + \mathrm{tr}((\mathbf{S}-\mathbf{W}_t)\mathbf{D}) + \phi(\mathbf{\Theta}), \tag{5}$$

where $\mathbf{W}_t = \mathbf{\Theta}_t^{-1}$ and $\mathbf{D} = \mathbf{\Theta} - \mathbf{\Theta}_t$, and t represents the t-th Newton iteration.

The subproblem (5) can be solved by non-monotone spectral projected gradient (NSPG) method [31]. When applied to (5), NSPG needs to solve the proximal subproblem in the form of

$$\min_{\mathbf{\Theta}} \frac{1}{2}\|\mathbf{\Theta} - \mathbf{G}_r\|_F^2 + \alpha\phi(\mathbf{\Theta}), \tag{6}$$

where

$$\mathbf{G}_r = \mathbf{\Theta}_r - \alpha(\mathbf{S} - 2\mathbf{W}_t + \mathbf{W}_t\mathbf{\Theta}_r\mathbf{W}_t)$$

Algorithm 1: Tree-Guided Graphical Lasso (TGL)

Input: $\mathbf{S}, \{G_i^j\}, \{w_{ii'}^j\}$
Output: $\mathbf{\Theta}$

```
Initialization: Θ_0 = (Diag(S))^{-1};
while Not Converged do
    Compute the Newton direction D by solving (5) and (7).
    Choose Θ_{t+1} by performing the Armijo backtracking line search along Θ_t + βD.
end
return Θ_{t+1};
```

and r denotes the r-th inner iteration in NSPG. Denote

$$\mathbf{R} = \mathbf{\Theta}_r - \mathbf{\Theta}_{r-1}$$

and

$$\bar{\alpha} = \mathrm{tr}(\mathbf{R}\mathbf{W}_t\mathbf{R}\mathbf{W}_t)/\|\mathbf{R}\|_F^2,$$

then α is given by

$$\alpha = \max(\alpha_{min}, \min(1/\bar{\alpha}, \alpha_{max})),$$

where $[\alpha_{min}, \alpha_{max}]$ is a predefined safeguard.

After obtaining the optimal solution of (5) $\mathbf{\Theta}^*$, the Newton direction $\mathbf{D}$ can be computed as

$$\mathbf{D} = \mathbf{\Theta}^* - \mathbf{\Theta}_t. \tag{7}$$

Once the Newton direction is obtained, we need to find an appropriate step size $\beta \in (0,1]$ to ensure a sufficient reduction in the objective function in (4). Because of the positive definite constraint in (4), we need to ensure the next iterate $\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t + \beta\mathbf{D}$ to be positive definite. It is not hard to show that such step size satisfying the above requirements always exits [11]. Thus, we can adopt the Amrmijo's backtracking line search rule to select a step length $\beta \in (0,1]$. We use the Cholesky decomposition to check the positive definiteness of $\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t + \beta\mathbf{D}$. In addition, the $\log\det(\mathbf{\Theta}_{t+1})$ and $\mathbf{\Theta}_{t+1}^{-1}$ can be efficiently computed as a byproduct of the Cholesky decomposition of $\mathbf{\Theta}_{t+1}$. The algorithm is summarized in Algorithm 1.

Under the assumption that the subproblem (5) is solved exactly, the convergence rate of the second-order method is locally quadratic when the exact Hessian is used [11, 14, 28]. If the subproblem (5) is solved inexactly, the convergence rate of the second method is locally superlinear by adopting an adaptive stopping criterion in NSPG [14]. Due to the use of Cholesky decomposition and the need of computing $\mathrm{tr}(\mathbf{W}_t\mathbf{D}\mathbf{W}_t\mathbf{D})$ in (5), the complexity of Algorithm 1 is $O(p^3)$.

2.3 Screening

Due to the existence of the log determination, it is computationally expensive to solve the penalized log likelihood model (3) by applying Algorithm 1 directly. The screening strategy has commonly been employed to reduce the size of optimization problems so that a massive computational gain can be achieved. In this section, we derive a sufficient condition for the solution of SGL to be block diagonal (subject to some rearrangement of features; see Figure 1 for illustration), thus significantly reducing the complexity of the problem.

Let $C_1, \ldots, C_L$ be a partition of the p features into L non-overlapping sets such that

$$C_l \cap C_{l'} = \emptyset, \ \forall l \neq l'.$$

We say that the solution $\widehat{\mathbf{\Theta}}$ of SGL (2) is block diagonal (subject to some rearrangement of features) with L known blocks $C_l, \ l = 1, \ldots, L$ if $\widehat{\mathbf{\Theta}}_{ij} = \widehat{\mathbf{\Theta}}_{ji} = 0$ for $i \in C_l, \ j \in C_{l'}, \ l \neq l'$. Without loss of generality, we assume that a block diagonal solution $\widehat{\mathbf{\Theta}}$ with L blocks $C_l, \ l = 1, \ldots, L$ takes the form of

$$\widehat{\mathbf{\Theta}} = \begin{pmatrix} \widehat{\mathbf{\Theta}}_1 & & \\ & \ddots & \\ & & \widehat{\mathbf{\Theta}}_L \end{pmatrix}, \tag{8}$$

where $\widehat{\mathbf{\Theta}}_l$ is the $|C_l| \times |C_l|$ symmetric submatrix of $\widehat{\mathbf{\Theta}}$ consisting of features in C_l.

Since the elements in off diagonal blocks are zero, the original optimization problem can thus be reduced to a much smaller problem restricted to the elements in the diagonal blocks, resulting in a great computational gain. Our main result is summarized in the following theorem:

THEOREM 1. *Suppose $\mathbf{U}^{d+1} = -\mathbf{S}$, where d is the depth of the tree structure. For different groups G_i^j at the depth j, define U^j recursively as follows:*

$$\mathbf{U}^j_{ii'} = \begin{cases} 0 & \|\mathbf{U}^{j+1}_{ii'}\|_F \leq w^j_{ii'} \\ \frac{\|\mathbf{U}^{j+1}_{ii'}\|_F - w^j_{ii'}}{\|\mathbf{U}^{j+1}_{ii'}\|_F} \mathbf{U}^{j+1}_{ii'} & \|\mathbf{U}^{j+1}_{ii'}\|_F > w^j_{ii'} \end{cases} \tag{9}$$

A sufficient condition for the solution of SGL to be block diagonal with blocks $C_1, \ldots, C_L$ is that $\mathbf{U}^j$ at some layer j has the same block diagonal structure such that

$$\mathbf{U}^j_{ii'} = 0, \forall G^j_i \subseteq C_l, G^j_{i'} \subseteq C_{l'}, l \neq l'$$

and there is no group G_i^j across two blocks, that is, there do not exist $x_1 \in C_l$, and $x_2 \in C_{l'}$, such that $\{x_1, x_2\} \subseteq G_i^j$.

PROOF. By the first-order optimality condition, $\widehat{\mathbf{\Theta}}$ is the optimal solution of problem (2) if and only if it satisfies

$$-(\widehat{\mathbf{\Theta}})^{-1} + \mathbf{S} + \partial\phi(\widehat{\mathbf{\Theta}}) = 0. \tag{10}$$

Suppose that $\mathbf{U}^j$ at layer j has the block diagonal structure $C_1, \ldots, C_L$ such that

$$\mathbf{U}^j_{ii'} = 0, \forall G^j_i \subseteq C_l, G^j_{i'} \subseteq C_{l'}, l \neq l'$$

and there is no group G_i^j across two blocks. According to [18], it is not hard to show that $\mathbf{U}^j$ is the solution of the following problem:

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{X} + \mathbf{S}\|_F^2 + \tilde{\phi}(\mathbf{X}), \tag{11}$$

where

$$\tilde{\phi}(\mathbf{\Theta}) = \sum_{k=j}^{d} \left(\sum_{i \neq i'} w^k_{ii'} \|\mathbf{\Theta}^k_{ii'}\|_F + w^k_{ii} \|\mathbf{\Theta}^k_{ii/}\|_F \right),$$

and $\mathbf{U}^0$ is the solution of

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{X} + \mathbf{S}\|_F^2 + \phi(\mathbf{X}).$$

According to Theorem 1 in [18], we have $\mathbf{U}^k_{G^j_i, G^j_{i'}} = 0, k = 0, \ldots, j-1$ if $\mathbf{U}^j_{ii'} = 0$. Thus, we have

$$\mathbf{S}_{G^j_i, G^j_{i'}} + \partial\tilde{\phi}(0)_{G^j_i, G^j_{i'}} = 0.$$

As $\mathbf{U}^k_{G^j_i, G^j_{i'}} = 0, k = 0, \ldots, j-1$ if $\mathbf{U}^j_{ii'} = 0$, it can be shown that $0 \in \phi(\mathbf{\Theta})^k_{G^j_i, G^j_i}$, for $k = 0, \ldots, j-1$ since the minimum of $\|\cdot\|_F$ is achieved at 0. Then, we have

$$\mathbf{S}_{G^j_i, G^j_{i'}} + \partial\phi(0)_{G^j_i, G^j_{i'}} = 0,$$

since $0 \in \phi(\mathbf{\Theta})^k_{G^j_i, G^j_i}$, for $k = 0, \ldots, j-1$. Therefore, the first-order optimality condition holds for the elements in off diagonal blocks.

Next we show how to construct a $\widehat{\mathbf{\Theta}}$ which satisfies the first optimality condition. Let $\widehat{\mathbf{\Theta}}$ be a block diagonal matrix with blocks $C_l, \ l = 1 \ldots, L$. It is clear that the optimality condition of (2) for off diagonal elements are satisfied. We can let the elements in the diagonal blocks of $\widehat{\mathbf{\Theta}}$ be the solution of the following problem:

$$\min_{\mathbf{\Theta}_l, l=1,\ldots,L} \sum_{l=1}^{L} (-\log\det(\mathbf{\Theta}_l) + \mathrm{tr}(\mathbf{S}_l\mathbf{\Theta}_l)) + \phi(\mathbf{\Theta})$$
$$s.t. \quad \mathbf{\Theta}_{i,j} = 0, \forall i \in C_l, j \in C_{l'}, \ l \neq l'.$$

Since $\mathbf{U}^k_{G^j_i, G^j_{i'}} = 0$, for $k = 0, \ldots, j-1$ if $\mathbf{U}^j_{ii'} = 0$, the first optimality condition (10) holds for $\widehat{\mathbf{\Theta}}$, thus $\widehat{\mathbf{\Theta}}$ is the optimal solution of (2). This completes the proof of the theorem. □

Theorem 1 can be used as a screening rule to determine the elements in the identified off-diagonal blocks to be zero in advance. Assume that there are L blocks of the same size identified by the screening rule, $p^2(1 - \frac{1}{L})$ elements do not need to be computed as the optimal values for these elements are determined as 0 by the screening. Recall that the complexity of the proposed second-order method is $O(p^3)$ due to Cholesky decomposition and computation of $\mathrm{tr}(\mathbf{W}_t\mathbf{D}\mathbf{W}_t\mathbf{D})$. The complexity of solving the proximal operator (11) for the tree group structural regularization is $O(p^2)$ [18]. By applying the screening rule, the complexity of Cholesky decomposition and computation of $\mathrm{tr}(\mathbf{W}_t\mathbf{D}\mathbf{W}_t\mathbf{D})$ are reduced to $O(p^3/L^2)$, and the complexity of solving (11) is reduced to $O(p^2/L)$. Therefore, the complexity of the second-order method with screening is $O(p^3/L^2)$ since $L \leq p$. When L is large, application of the screening rule can achieve a great computational gain.

2.4 Discussions

We want to emphasize that Theorem 1 provides a screening rule for a large family of graphical model problems. Several examples in the literature can be reformulated into problem (2) with specific constraints. In the following, we provide several examples as follows.

ℓ_1 regularization: When the ℓ_1 regularization is used, SGL degenerates to standard graphical lasso [4, 8] given by:

$$\min_{\mathbf{\Theta} \succ 0} -\log\det(\mathbf{\Theta}) + \mathrm{tr}(\mathbf{S}\mathbf{\Theta}) + \lambda\|\mathbf{\Theta}\|_1.$$

The proximal operator in (11) can be written as

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{X} + \mathbf{S}\|_F^2 + \lambda\|\mathbf{X}\|_1. \tag{12}$$

According to Theorem 1, the sufficient condition for the optimal solution (*i.e.*, the solution of graphical lasso based on the ℓ_1 regularization) to have a block structure $C_1, \ldots, C_L$ is that the optimal solution $\widehat{\mathbf{X}}$ of (12) has the same block diagonal structure, *i.e.*, $C_1, \ldots, C_L$. It is not hard to see that the following first order optimality condition is satisfied

$$-\lambda \leq \mathbf{S}_{ij} \leq \lambda,\ \forall i \in C_l,\ j \in C_{l'},\ l \neq l',$$

which is exactly the same as the screening condition for graphical lasso proposed in [21, 30]:

$$|\mathbf{S}_{ij}| \leq \lambda,\ \forall i \in C_l,\ j \in C_{l'},\ l \neq l'.$$

Thus, the screening rule in [21, 30] is a special case of the proposed rule.

Group regularization: The graphical lasso with group regularization has been studied in [13]. The formulation of group graphical lasso is given by

$$\min_{\mathbf{\Theta} \succ 0} -\log\det(\mathbf{\Theta}) + \mathrm{tr}(\mathbf{S}\mathbf{\Theta}) + \lambda \sum_{i,j} \|\mathbf{\Theta}_{G_i,G_j}\|_F,$$

where $\mathbf{\Theta}_{G_i,G_j}$ is a submatrix of $\mathbf{\Theta}$, and G_i is the i-th group of features. Note that $\cup G_i = \{1, \ldots, p\}$ and different groups do not overlap. In [13], Kolar *et al.* proposed a sufficient condition for the solution of group graphical lasso to be block diagonal, which is given by

$$\|\mathbf{S}_{G_i,G_j}\|_F \leq \lambda,\ \forall\ G_i \subseteq C_l, G_j \subseteq C_{l'},\ l \neq l'. \qquad (13)$$

It is clear that condition (13) is the first-order optimality condition for the solution of (11) to have the block diagonal solution $C_1, \ldots, C_L$. Thus, the screening rule in [21, 30] is also a special case of the proposed rule.

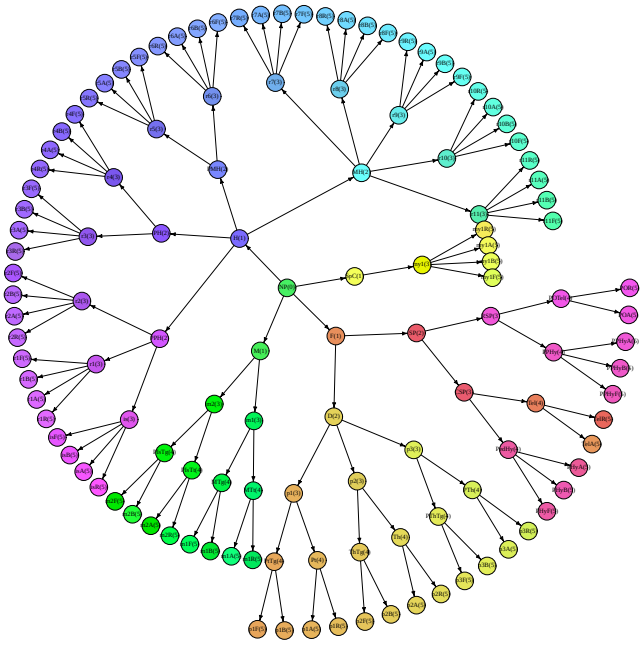

Figure 5: The ontology hierarchy of the Allen Developing Mouse Brain Atlas from level 0 to level 5. Each brain region is colored using the color code of the Allen Developing Mouse Brain Reference Atlas.

3. EXPERIMENTAL EVALUATION

In this section, we conduct experiments to demonstrate the effectiveness and efficiency of the proposed screening rule and the proposed tree-guided graphical lasso (TGL). We used both synthetic and real mouse brain gene expression data to evaluate our methods. The experiments are performed on a PC with quad-core Intel i7 3.4GHz CPU with 16GB of RAM. The TGL formulation is implemented in MATLAB, while the sub-routine for solving the subproblem (6) is implemented in C. We compare TGL with standard graphical lasso (GLasso) in the experiments.

3.1 Synthetic Data

We first evaluate our method using synthetic data. We follow [32] in generating the synthetic covariance matrix. Specifically, we first generate the ground truth precision matrix $\mathbf{\Theta}$ with random block nonzero patterns. Each nonzero block has a random sparse structure. Given the precision matrix $\mathbf{\Theta}$, we sample from a Gaussian distribution to compute the sample covariance matrix. The weights for tree-structural group regularization take the form of

$$w^j_{ii'} = \rho / \sqrt{|\mathbf{\Theta}^j_{ii'}|},$$

where ρ is a given positive parameter and $|\mathbf{\Theta}^j_{ii'}|$ is the number of elements in $\mathbf{\Theta}^j_{ii'}$. To make a fair comparison between different methods, we control the regularization parameters of TGL and GLasso to ensure the numbers of edges obtained from both estimations to be the same.

Figure 6 shows the comparison between TGL and GLasso in terms of edge detection. The first column of Figure 6 shows the nonzero patterns (*i.e.*, edges) of two ground truth precision matrices. In both cases, the same index tree is used, which is given by

$$\begin{cases} G^3_i = \{i\}, i = 1, \ldots, 100, \\ G^2_i = \{20i + 1 : 20(i+1)\}, i = 0, \ldots, 4, \\ G^1_1 = \{1 : 60\}, \\ G^1_2 = \{61 : 100\}. \end{cases} \qquad (14)$$

We can observe from Figure 6 that the nonzero patterns of the precision matrices estimated by TGL are more similar to the ground truth than GLasso. These results demonstrate that TGL outperforms GLasso in terms of detecting true edges in the precision matrices.

We conduct experiments to demonstrate the effectiveness of the proposed screening rule. We terminate NSPG using the following stopping criterion:

$$\frac{\|\mathbf{\Theta}^{(k)}_r - \mathbf{\Theta}^{(k)}_{r-1}\|_\infty}{\|\mathbf{\Theta}^{(k)}_{r-1}\|_\infty} \leq 1e\text{-}6.$$

Additionally, TGL is terminated when the relative error of the objective value is smaller than $1e$-5. The used index tree is given by

$$\begin{cases} G^3_i = \{i\}, i = 1, \ldots, p, \\ G^2_i = \{\frac{ip}{2L} + 1 : \frac{(i+1)p}{2L}\}, i = 0, \ldots, 2L-1, \\ G^1_i = \{\frac{ip}{L} + 1 : \frac{(i+1)p}{L}\}, i = 0, \ldots, L-1. \end{cases} \qquad (15)$$

Table 1: Timing comparison of the proposed TGL with and without screening in terms of average computational time (seconds). TGL-S denotes TGL with screening. The computational time of TGL-S is the summation of screening and TGLs. p stands for the dimension, and L is the number of blocks. $\|\mathbf{\Theta}\|_0$ represents the total number of nonzero entries in the ground truth precision matrix $\mathbf{\Theta}$, and $\|\mathbf{\Theta}^*\|_0$ is the number of nonzeros in the solution.

Data setting				Computational time (seconds)		
				TGL-S		TGL
p	L	$\|\mathbf{\Theta}\|_0$	$\|\mathbf{\Theta}^*\|_0$	screening	TGLs	
1000	5	11442	11914	**0.0109**	**0.1715**	2.8219
2000		23694	23854	**0.0395**	**1.0839**	12.2679
1000	10	11142	9782	**0.0105**	**0.2286**	6.481
2000		23308	23862	**0.0366**	**0.4257**	19.1117

where L is the number of blocks. The time comparison results are given in Table 1. We can observe that the computational time of screening is negligible compared with that of solving the TGL. Since the complexity of identifying the connected components is $O(\|\mathbf{\Theta}^*\|_0)$, the computational time of screening is almost linear with respect to $\|\mathbf{\Theta}^*\|_0$. Results in Table 1 demonstrate that the screening rule can achieve very significant computational gain. The larger the L is, the higher the speedup is. These results demonstrate the potential of our method for identifying structured networks for large-scale data.

3.2 Allen Developing Mouse Brain Atlas Data

We also evaluate our methods using the Allen Developing Mouse Brain Atlas data. The Allen Developing Mouse Brain Atlas contains spatiotemporal *in situ* hybridization (ISH) gene expression data across multiple stages of mouse brain development [26, 1]. The primary data consist of 3-D, cellular resolution ISH expression patterns of approximately 2000 genes in sagittal plane across four embryonic (E11.5, E13.5, E15.5, and E18.5) and three early postnatal ages (P4, P14, and P28). The ISH image series are passed through an informatics data processing pipeline by which they are converted to grid-level expression summaries in the same coordinate space [2].

After the ISH image series are mapped to the reference space, a gridding module is applied to divide the 3-D reference space into regular grids, creating a low resolution 3-D summary of the gene expression. The resolution of the data grids varies with age. For each grid voxel, expression density is the number of expressing pixels divided by the number of image pixels in the voxel; expression intensity is the averaged inverted ISH gray-scale value at expressing pixels within the span of the grid voxel; expression energy is defined as the product of expression intensity with expression density. Our analysis in this work is also based on the grid-level expression energy. In this work, we use data from the first three developmental ages with 7796, 9963, and 8258 structural voxels, respectively. We use a data set of 1724 genes.

We apply the proposed TGL method to the voxel-level gene expression data to demonstrate the effectiveness of TGL and the proposed screening rule. In the Allen Developing Mouse Brain Atlas, the brain regions are organized into a tree-structural hierarchy as shown in Figure 5. This provides an ideal setting for evaluating our proposed tree-structural graphical Lasso formulation. We use such hierarchical structure as the input prior knowledge to our algorithm TGL. We compare TGL with the standard GLasso on this data. Figure 7 shows the comparison between the precision matrices estimated by TGL and GLasso. We can observe that, although the data inherently exhibits certain tree structures, the results obtained by GLasso do not recover these structures clearly. In contrast, our proposed TGL method successfully recovers the hierarchical structures. Nevertheless, GLasso recovers some overall structures that are largely consistent with the hierarchy with the corruption of some noises.

To demonstrate the power of the proposed screening, we report the running time of the TGL with and without screening. We use the data from the first stage for our evaluation. We stop the computation of the algorithm after we obtain a solution with precision $1e$-6. The computational time of TGL without screening is 57189.6 seconds. With the screening, the total computational time of TGL-S including the time for screening is reduced to 2781.5 seconds, demonstrating the superiority of the proposed screening rule.

4. RELATED WORK

Brain connectivity describes how the brain regions are connected, thereby providing information pathways in the brain. Graphical modeling is a statistical tool to capture the connectivity between multiple random variables. Thus graphical models are natural tools for brain connectivity analysis. However, the dimensionality is usually very large for brain data, and this prohibits the direct application of many existing methods. Therefore, large-scale brain network estimation is considered as a big data problem and has raised several challenges and opportunities [20].

The task of estimating the whole brain connectivity is important, but also very challenging. There are two major types of connectivity analysis; namely functional connectivity and effective connectivity. There are a few simple approaches for estimating functional connectivity, *i.e.*, these based on pair-wise correlations, clustering, and independent component analysis (ICA) [9]. Effective connectivity aims to find directional relationships between brain regions. Popular approaches for effective connectivity include dynamic causal modeling, structural equation models, and Granger causality. These tools are complex in computation and modeling, thus they are usually applicable for a small number (e.g. < 100) of preselected voxels or regions. Recently, voxel correlations are used to provide more accurate selection of voxels for a certain region of the brain. However, the results may be sensitive to the selection of regions, and the network inference can be biased if the influence from other omitted

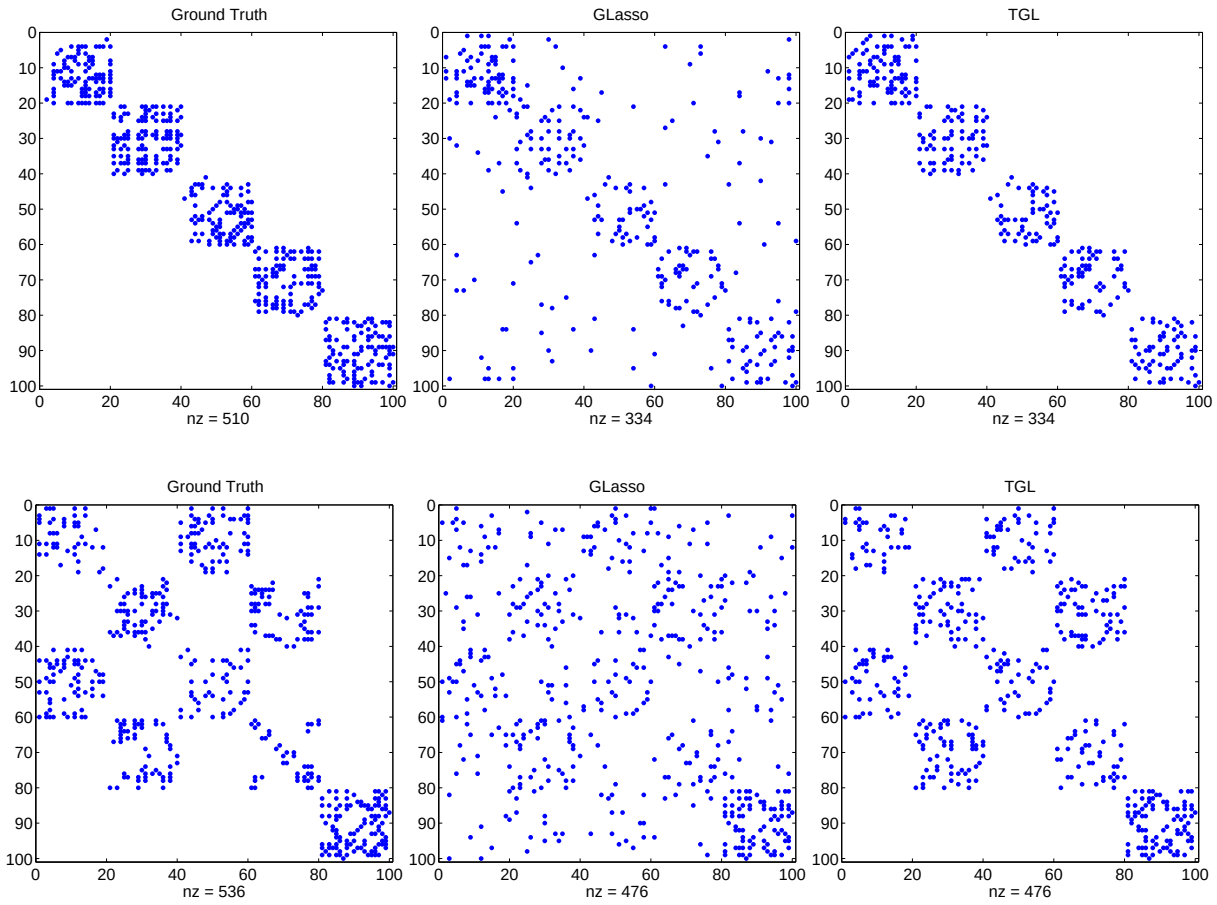

Figure 6: Comparison between TGL and GLasso in terms of edge detection. Left: the ground truth precision matrix; middle: the precision matrix estimated by GLasso; right: the precision matrix estimated by TGL.

regions is large [20]. To date, several challenges remain in inferring large-scale direct connectivity.

Sparse Gaussian graphical models (sGGM) [4, 8, 21, 22, 35] are proposed to estimate large-scale brain connectivity. This type of models has a solid probabilistic foundation for distinguishing direct connections from indirect connections. Suppose we have a multivariate variable X following a p-variate normal distribution $N(\mu, \mathbf{\Sigma})$, and we are given n i.i.d observations. sGGM represents the relationships between the p variables by a network of p nodes, where each node represents a variable and there are connections between nodes. Formally, inference of the connections between the p nodes is reduced to estimating a sparse inverse covariance $\mathbf{\Theta} = \mathbf{\Sigma}^{-1}$, where a nonzero off-diagonal entry in $\mathbf{\Theta}$ indicates that the corresponding row and column variables are connected. Similarly, a zero entry indicates the absence of connection. The sGGM approach performs well on a simulation study using a small number of regions [20].

In recent years, considerable research efforts have been devoted to estimating the precision matrix and the corresponding sGGM [11, 12, 15, 16, 19, 23, 24]. Numerous methods have been developed for solving this model. For example, Banerjee *et al.* [4] and Friedman *et al.* [8] proposed block coordinate ascent methods for solving the dual problem. The latter method [8] is widely referred to as Graphical lasso (GLasso). Yuan [36] and Scheinberg *et al.* [25] applied the alternating direction method of multipliers (ADMM) [5] to this problem. Wang *et al.* [29], Hsieh *et al.* [11], Olsen *et al.* [24], and Dinh *et al.* [7] applied the Newton method for solving this model.

The brain network system is complex and structured. For example, brain regions are usually organized into a hierarchy in which a large region includes multiple sub-regions. We propose a tree-structural graphical model to represent the multi-level brain network in this paper. Specifically, voxels are represented as the leaf nodes of the tree. The nodes in the intermediate layer represents the regions. This way, the entire brain is considered as the root of the tree. Our model is different from the model in [20] in multiple ways, and our proposed model is more general. Specifically, the nodes in [20] can only connect with each other via the hub nodes, while the nodes can connect in arbitrary ways in our model. In [20] an alternating update algorithm is proposed to solve the model, and much computational efforts have been devoted to computing the determinant of $\mathbf{\Theta}$. This prohibits the direct application of graphical models from large-scale brain datasets. The contributions of this paper lie in two folds: (1) we propose a tree-structural graphical model to incorporate the multi-level brain structure; and (2) we develop a sufficient screening rule to dramatically reduce the computational cost for computing the determinant of $\mathbf{\Theta}$ in general structural graphical model.

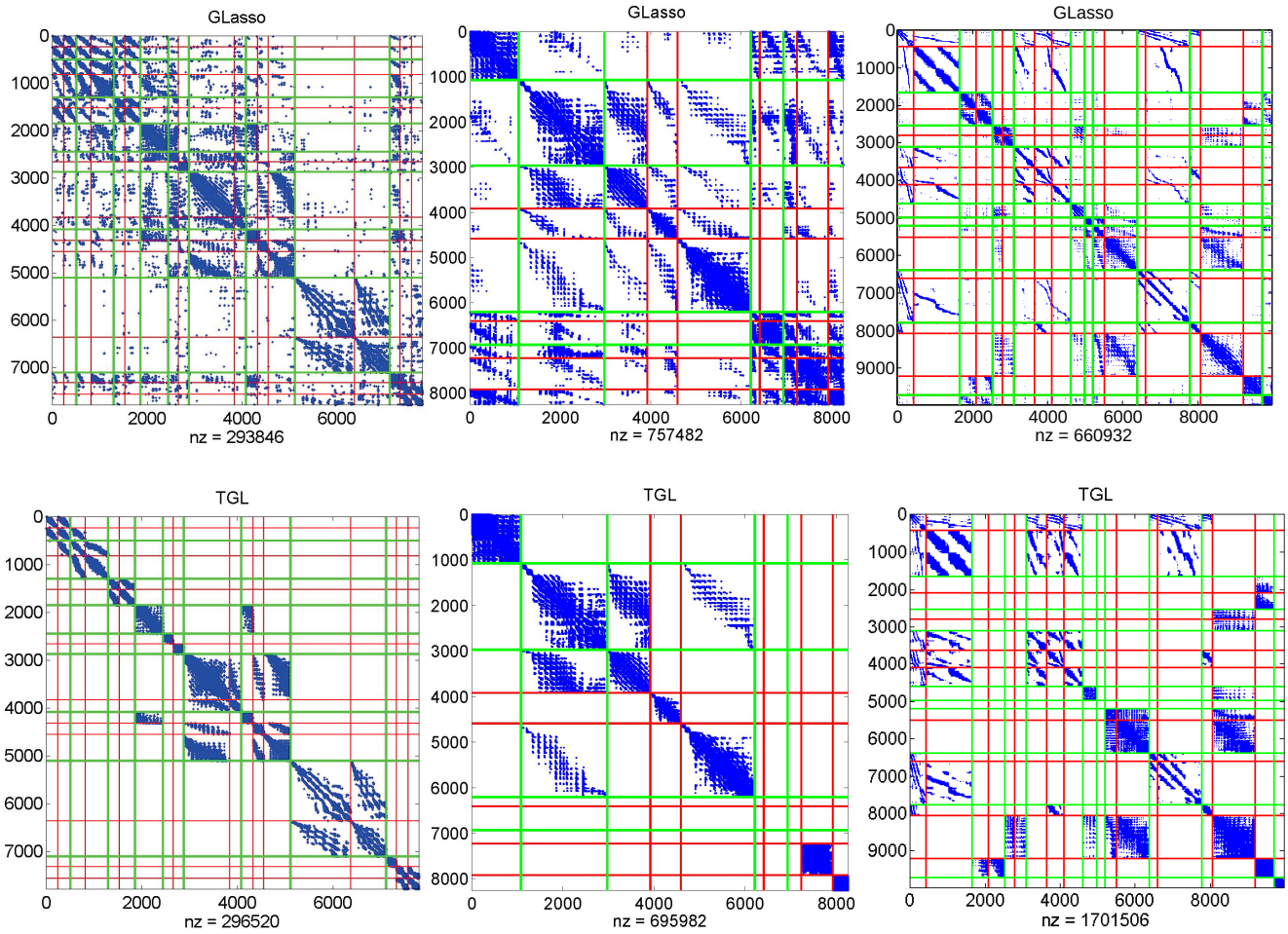

Figure 7: Comparison between TGL and GLasso in terms of edge detection on Allen developing mouse brain atlas data. Upper: the precision matrix estimated by GLasso; bottom: the precision matrix estimated by TGL. Left to right: Mouse brain networks in the 1st, 2nd, and 3rd development stages. The red and green grids visualize the tree-structural groups in two layers.

5. CONCLUSION AND FUTURE WORK

In this work, we propose a hierarchical graphical model framework known as the tree-guided graphical lasso. In order to scale the proposed formulation to large-scale network inference, we develop a screening rule to dramatically speedup the computation. Specifically, we employ the second-order method to solve the proposed formulation. In addition, we derive a sufficient condition for the TGL solution to be block diagonal. Based on this condition, a simple screening rule has been developed to scale our methods to large-scale problems. We apply the proposed methods to infer the large-scale mouse brain connectivity. Numerical experiments on synthetic and real data demonstrate the efficiency and effectiveness of the proposed method and the proposed screening rule.

This work focuses on the inference of mouse brain networks. On the other hand, human brain networks are more complex and involve more structures. We plan to apply the proposed methods to the human brain networks in the future. This work represents the first attempt to investigate screening rules for structural graphical Lasso, and many theoretical problems in this direction remain unexplored. We plan to derive a necessary and sufficient condition for screening the TGL solution, thereby enhancing the theoretical guarantee of our algorithm. In addition, we plan to explore the convergence properties of the second-order method using the inexact Newton direction.

6. ACKNOWLEDGMENTS

This work was supported in part by research grants from NIH (R01 LM010730) and NSF (IIS-0953662, DBI-1147134, DBI-1350258, III-1421057, and III-1421100).

7. REFERENCES

[1] Allen Developing Mouse Brain Atlas. http://developingmouse.brain-map.org. 2013.

[2] Allen Institute for Brain Science. Technical White Paper: Informatics Data Processing for the Allen Developing Mouse Brain Atlas, 2012.

[3] Francis Bach, Rodolphe Jenatton, Julien Mairal, Guillaume Obozinski, et al. Structured sparsity through convex optimization. *Statistical Science*, 27(4):450–468, 2012.

[4] 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.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[6] 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.

[7] Q. Dinh, A. Kyrillidis, and V. Cevher. A proximal newton framework for composite minimization: Graph learning without cholesky decompositions and matrix inversions. *arXiv preprint arXiv:1301.1459*, 2013.

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

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

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

[11] C.J. Hsieh, M. A. Sustik, I. S. Dhillon, and P. Ravikumar. Sparse inverse covariance matrix estimation using quadratic approximation. *Advances in Neural Information Processing Systems (NIPS)*, 24, 2011.

[12] S. Huang, J. Li, L. Sun, J. Liu, T. Wu, K. Chen, A. Fleisher, E. Reiman, and J. Ye. Learning brain connectivity of alzheimer's disease from neuroimaging data. In *Advances in Neural Information Processing Systems (NIPS)*, pages 808–816, 2009.

[13] M. Kolar, H. Liu, and E. Xing. Markov network estimation from multi-attribute data. In *Proceedings of The 30th International Conference on Machine Learning*, pages 73–81, 2013.

[14] J. D. Lee, Y. Sun, and M. A. Saunders. Proximal newton-type methods for minimizing convex objective functions in composite form. *arXiv preprint arXiv:1206.1623*, 2012.

[15] L. Li and K.C. Toh. An inexact interior point method for l 1-regularized sparse covariance selection. *Mathematical Programming Computation*, 2(3):291–315, 2010.

[16] H. Liu, K. Roeder, and L. Wasserman. Stability approach to regularization selection (StARS) for high dimensional graphical models. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

[17] J. Liu, S. Ji, and J. Ye. *SLEP: Sparse Learning with Efficient Projections*. Arizona State University, 2009.

[18] J. Liu and J. Ye. Moreau-yosida regularization for grouped tree structure learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1459–1467, 2010.

[19] Z. Lu. Smooth optimization approach for sparse covariance selection. *SIAM Journal on Optimization*, 19(4):1807–1827, 2009.

[20] X. Luo. A hierarchical graphical model for big inverse covariance estimation with an application to fmri. *arXiv preprint arXiv:1403.4698*, 2014.

[21] R. Mazumder and T. Hastie. Exact covariance thresholding into connected components for large-scale graphical lasso. *The Journal of Machine Learning Research*, 13:781–794, 2012.

[22] R. Mazumder and T. Hastie. The graphical lasso: New insights and alternatives. *Electronic Journal of Statistics*, 6:2125–2149, 2012.

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

[24] P. A. Olsen, F. Oztoprak, J. Nocedal, and S. J. Rennie. Newton-like methods for sparse inverse covariance estimation. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[25] K. Scheinberg, S. Ma, and D. Goldfarb. Sparse inverse covariance selection via alternating linearization methods. In *Advances in Neural Information Processing Systems (NIPS)*, 2010.

[26] Carol L Thompson, Lydia Ng, Vilas Menon, Salvador Martinez, Chang-Kyu Lee, Katie Glattfelder, Susan M Sunkin, Alex Henry, Christopher Lau, Chinh Dang, et al. A high-resolution spatiotemporal atlas of gene expression of the developing mouse brain. *Neuron*, 83(2):309–323, 2014.

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

[28] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1):387–423, 2009.

[29] C. Wang, D. Sun, and K.C. Toh. Solving log-determinant optimization problems by a newton-cg primal proximal point algorithm. *SIAM Journal on Optimization*, 20(6):2994–3013, 2010.

[30] 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.

[31] S. J. Wright, R. D. Nowak, and M.A.T. Figueiredo. Sparse reconstruction by separable approximation. *Signal Processing, IEEE Transactions on*, 57(7):2479–2493, 2009.

[32] S. Yang, Z. Lu, X. Shen, P. Wonka, and J. Ye. Fused multiple graphical lasso. *arXiv preprint arXiv:1209.2139v2*, 2013.

[33] S. Yang, L. Yuan, Y.C. Lai, X. Shen, P. Wonka, and J. Ye. Feature grouping and selection over an undirected graph. In *KDD*, pages 922–930. ACM, 2012.

[34] J. Ye and J. Liu. Sparse methods for biomedical data. *ACM SIGKDD Explorations Newsletter*, 14(1):4–15, 2012.

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

[36] X. Yuan. Alternating direction method for covariance selection models. *Journal of Scientific Computing*, 51(2):261–273, 2012.