Journal of Machine Learning Research 25 (2024) 1-40 Submitted 11/22; Revised 2/24; Published 12/24

Transfer learning for tensor Gaussian graphical models

Mingyang Ren MINGYANGREN@SJTU.EDU.CN
School of Mathematical Sciences
Shanghai Jiao Tong University
Minhang, Shanghai, China

Yaoming Zhen YAOMING.ZHEN@UTORONTO.CA
Department of Statistical Sciences
University of Toronto
Toronto, Ontario, Canada

Junhui Wang JUNHUIWANG@CUHK.EDU.HK
Department of Statistics
The Chinese University of Hong Kong
Shatin, Hong Kong, China

Editor: Gal Elidan

Abstract

Tensor Gaussian graphical models (GGMs), interpreting conditional independence structures within tensor data, have important applications in numerous areas. Yet, the available tensor data in one single study is often limited due to high acquisition costs. Although relevant studies can provide additional data, it remains an open question how to pool such heterogeneous data. In this paper, we propose a transfer learning framework for tensor GGMs, which takes full advantage of informative auxiliary domains even when non-informative auxiliary domains are present, benefiting from the carefully designed data-adaptive weights. Our theoretical analysis shows substantial improvement of estimation errors and variable selection consistency on the target domain under much relaxed conditions, by leveraging information from auxiliary domains. Extensive numerical experiments are conducted on both synthetic tensor graphs and brain functional connectivity network data, which demonstrates the satisfactory performance of the proposed method.

Keywords: brain functional connectivity, Gaussian graphical models, precision matrix, tensor data, transfer learning.

1. Introduction

The development of modern science facilitates the collection of high-order tensor data in various research areas, ranging from molecular biology, neurophysiology, to signal processing. For examples, in cancer staging studies, multi-stage, multi-tissue, and multi-omics observations will be analyzed, which are organized as order-3 tensors (Krishnan et al., 2018); in brain functional connectivity analysis, the functional magnetic resonance imaging (fMRI) data is also considered as an order-2 tensor, which includes blood oxygen level signals in different brain regions at different time points (Bellec et al., 2017; Zhang et al., 2019).

In light of the importance of tensor data in modern science, tensor data analysis has received increasing attention in recent years, such as supervised learning represented by tensor regression and classification (Zhou et al., 2013; Sun and Li, 2017; Pan et al., 2018) and unsupervised learning represented by tensor clustering and principal component analysis (Hopkins et al., 2015; Luo and Zhang, 2022). In addition, the Gaussian graphical model (GGM) interpreting conditional independence structures within tensor data is also an essential topic but relatively understudied in literature. A straightforward approach for describing conditional independence in tensor data is to vectorize the tensor and fit multivariate GGMs (Friedman et al., 2008; Lam and Fan, 2009; Zhang and Zou, 2014; Liu and Luo, 2015), which is considered, however, to largely ignore the tensor structure and require almost unrealistic estimation of a tremendous number of parameters (He et al., 2014). For example, in the brain fMRI tensor data, if modeling the vectorized tensor with 200 time points and 116 widely studied brain regions of interest using multivariate GGMs, it requires estimation of $\binom{200\times 116+1}{2}$ parameters, which is more than 269 million. Instead, the proposed tensor normal distribution only needs to estimate $\binom{201}{2}+\binom{117}{2}$ parameters, which is less than 27 thousands, thanks to the Kronecker decomposition of the covariance matrix. More severely, simply vectorizing the tensor data may dilute our concern on the conditional independence between brain regions, corresponding to the functional brain connectivity, which is important for exploring the neurophysiological etiology. Tensor GGMs (He et al., 2014; Lyu et al., 2019) and related efficient algorithms (Min et al., 2022) are proposed in recent literature and have been widely reported their success. The models usually assume that the covariance matrix of the tensor data is separable, in the sense that it can be decomposed as the Kronecker product of multiple much smaller covariance matrices, each corresponding to one mode of the tensor data.

In many medical applications, high-dimensional and high-order tensor data are often extremely limited in one medical institution, due to the high acquisition costs and the rarity of certain diseases (Westin et al., 2002). Fortunately, relevant data may be collected by other institutions, which may be helpful for the tasks studied at the target institution. Our motivation is to investigate the brain fMRI scans of attention deficit hyperactivity disorder (ADHD) patients from various sites, in which the data in NeuroIMAGE site consists only 17 samples, but other sites can further provide more than ten times of relevant data. To pool these heterogeneous data from different sites, transfer learning is a promising solution with growing popularity, which aims at transferring the information from different auxiliary domains to help with the specific task on the target domain of interest (Pan and Yang, 2009).

Transfer learning has been studied in many branches of machine learning, including image recognition (Gao and Mosalam, 2018), natural language processing (Ruder et al., 2019), and drug discovery (Cai et al., 2020). More discussion on transfer learning can be found in Zhuang et al. (2020) and the references therein. Despite significant successes of transfer learning in algorithm developments and real-life applications, it is recognized that the existing studies on their statistical theory guarantees are still insufficient and are also gaining attention. Recently, Cai and Wei (2021) proposes some minimax and adaptive transfer learning-based classifiers, Bastani (2021) derives the estimation error bound of linear models in the single auxiliary domain case. Li et al. (2022a) proposes the Trans-Lasso method under high-dimensional linear models with multiple auxiliary domains and establishes its minimax optimality. This transfer learning framework is extended to high-dimensional generalized linear models (Tian and Feng, 2022), federated learning (Li et al., 2021), and functional linear regression (Lin and Reimherr, 2022). However, transfer learning for unsupervised tasks, such as GGMs, is still in its infancy. It was only until very recently that Li et al. (2022b) proposes a Trans-CLIME method for transfer learning on high-dimensional GGMs and it is subsequently extended to semiparametric graphical models (He et al., 2022), but these approaches are still restricted to vector-value data.

In this paper, we propose a transfer learning framework for tensor GGMs. It introduces a type of divergence matrix to measure the similarity between the target and auxiliary domains for each mode benefiting from the separability of the tensor covariance matrix, as well as some novel data-adaptive weights on the auxiliary domains based on the divergence matrices. The divergence matrix is first estimated based on a carefully designed regularized loss function by combining information from both target and auxiliary domains, and then the estimates of precision matrices can be better constructed based on the auxiliary domain and the well-estimated divergence matrices. The efficient algorithm and rigorous theoretical analysis of the proposed method are also conducted.

This paper advances the current research on transfer learning in a number of ways. First, the proposed transfer learning method provides a more flexible modeling framework for high-order tensor GGMs, which also includes Li et al. (2022b) as a special case. Second, to prevent the negative transfer phenomenon (Shu et al., 2019), data-adaptive weights for auxiliary domains are constructed to minimize the interference from the non-informative auxiliary domains. Third, the established theoretical analysis shows that the estimation error can be improved using the data-adaptive weights as long as there is at least one informative auxiliary domain that is close enough to the target domain. This is significantly different from the results in Li et al. (2022b); He et al. (2022), which require all auxiliary domains to be informative for the improvement of error. Our theoretical analysis also demonstrates that transfer learning can help improve variable selection performance by weakening the regular minimum signal condition in literature (Lyu et al., 2019). Last but not least, the proposed method is applied to analyze the ADHD brain functional connectivity, which provides interesting neurophysiological insights into the pathogenesis.

The rest of the paper is organized as follows. Section 2 introduces some necessary notations and brief backgrounds on tensor GGMs. Section 3 introduces the proposed transfer learning framework for the tensor GGMs and its implementing algorithm. The consistency of estimation and variable selection is established in Section 4. Numerical simulations and the application on ADHD brain fMRI data and breast cancer gene interaction study are conducted in Sections 5 and 6, respectively. Section 7 contains a brief discussion, and all technical details are provided in Appendix.

2. Preliminaries

In this section, we introduce necessary notations that will be used throughout the paper and some brief backgrounds on the tensor graphical model.

2.1 Notations

Denote $\|\boldsymbol{u}\|_q$ as the l_q-norm of a vector $\boldsymbol{u}$, for $q \geqslant 0$. For a matrix $\boldsymbol{A} = [A_{(j_1,j_2)}]_{1\leqslant j_1,j_2\leqslant p}$, let $\boldsymbol{A}_{(j)}$ be its j-th column, $\|\boldsymbol{A}\|_{q,\infty} = \max_{1\leqslant j\leqslant p}\|\boldsymbol{A}_{(j)}\|_q$, $\|\boldsymbol{A}\|_1 = \sum_{j=1}^{p}\|\boldsymbol{A}_{(j)}\|_1$, $\|\boldsymbol{A}\|_{\max} = \max_{1\leqslant j_1,j_2\leqslant p}|A_{(j_1,j_2)}|$, $\|\boldsymbol{A}\|_{1,\mathrm{off}} = \sum_{1\leqslant j_1\neq j_2\leqslant p}|A_{(j_1,j_2)}|$, and $\|\boldsymbol{A}\|_F$ be the Frobenius norm of $\boldsymbol{A}$. When $\boldsymbol{A}$ is symmetric, we further denote $\psi_{\min}(\boldsymbol{A})$ and $\psi_{\max}(\boldsymbol{A})$ as the smallest and largest eigenvalues of $\boldsymbol{A}$, respectively. A multidimensional array $\boldsymbol{\mathcal{X}} = (x_{j_1,\cdots,j_M}) \in \mathbb{R}^{p_1\times\cdots\times p_M}$ is called a tensor of order-M. The vectorization of $\boldsymbol{\mathcal{X}}$ is defined by $\text{vec}(\boldsymbol{\mathcal{X}}) \in \mathbb{R}^p$ with $p = \prod_{m=1}^M p_m$. The mode-m matricization of $\boldsymbol{\mathcal{X}}$ is denoted by $\boldsymbol{\mathcal{X}}_{(m)} \in \mathbb{R}^{p_m\times(p/p_m)}$, which is obtained by arranging the mode-m fibers of $\boldsymbol{\mathcal{X}}$ to be the columns of the resulting matrix. Herein, a mode-m fiber of $\boldsymbol{\mathcal{X}}$ refers to a vector from $\boldsymbol{\mathcal{X}}$ by fixing all the indexes but the m-th mode. The mode-m product between a tensor $\boldsymbol{\mathcal{X}}$ and a matrix $\boldsymbol{\Omega} \in \mathbb{R}^{d\times p_m}$ is defined as $\boldsymbol{\mathcal{X}} \times_m \boldsymbol{\Omega} \in \mathbb{R}^{p_1\times\cdots p_{m-1}\times d\times p_{m+1}\times\cdots\times p_M}$, whose entry is defined as $(\boldsymbol{\mathcal{X}} \times_m \boldsymbol{\Omega})_{j_1,\cdots,j_{m-1},j'_m,j_{m+1}\cdots,j_M} = \sum_{j_m=1}^{p_m} x_{j_1,\cdots,j_M}\boldsymbol{\Omega}_{j'_m,j_m}$. In addition, for a list of matrices $\{\boldsymbol{\Omega}_1,\cdots,\boldsymbol{\Omega}_M\}$ with $\boldsymbol{\Omega}_m \in \mathbb{R}^{d_m\times p_m}$, we define $\boldsymbol{\mathcal{X}} \times \{\boldsymbol{\Omega}_1,\cdots,\boldsymbol{\Omega}_M\} = \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{\Omega}_1 \cdots \times_M \boldsymbol{\Omega}_M$. Similar to the matrix case, the Frobenius norm of $\boldsymbol{\mathcal{X}}$ is denoted as $\|\boldsymbol{\mathcal{X}}\|_F = (\sum_{j_1,\cdots,j_M} x^2_{j_1,\cdots,j_M})^{1/2}$. More detailed tensor algebra can be found in Kolda and Bader (2009).

Next, let card(S) be the cardinality of a set S and $[K] = \{1,\cdots,K\}$ be the K-set for any positive integer K. For sequences a_n and b_n, define $a_n \lesssim b_n$ if there exists a positive constant C such that $a_n \leqslant Cb_n$, and $a_n \asymp b_n$ if $a_n \lesssim b_n$ and $b_n \lesssim a_n$. For two real numbers a and b, define $a \wedge b = \min\{a, b\}$ and $a \vee b = \max\{a, b\}$. The superscript * of the parameter marks its true value.

Finally, some frequently used notations are summarized in the following table

Table 1: Table of Notation

Symbol	Description	Symbol	Description
n	sample size in the target domain	n_k	sample size in the k-th auxiliary domain
K	number of auxiliary domains	N	$\sum_{k=1}^K n_k$
M	the order of the data tensor	p_m	the m-th mode dimension of the data tensor
p	$\Pi_{m=1}^M p_m$	$\bar{p}$	$\max_{m\in[M]} p_m$
$\boldsymbol{\mathcal{X}}$	An tensor normal object in general	$\boldsymbol{\mathcal{X}}_{(j)}^{(m-\textbf{sub})}$	the j-th mode-m sub-tensor of $\boldsymbol{\mathcal{X}}$
$\boldsymbol{\mathcal{X}}_i$	observation i from target domain	$\boldsymbol{\mathcal{X}}_i^{(k)}$	observation i in the k-th auxiliary domain
$\boldsymbol{\Sigma}_m$	Targeted mode-m covariance matrix	$\boldsymbol{\Sigma}^{(k)}$	The k-th auxiliary mode-m covariance matrix
$\boldsymbol{\Omega}_m$	Targeted mode-m precision matrix	$\boldsymbol{\Omega}_m^{(k)}$	The k-th auxiliary mode-m precision matrix
$\{\boldsymbol{\Sigma}^{-1/2}\}$	$\{\boldsymbol{\Sigma}_1^{-1/2},\ldots,\boldsymbol{\Sigma}_M^{-1/2}\}$	$\{\boldsymbol{\Omega}^{1/2}\}$	$\{\boldsymbol{\Omega}_1^{1/2},\ldots,\boldsymbol{\Omega}_M^{1/2}\}$
TN	centered tensor normal distribution	$\text{TN}_{\boldsymbol{\Sigma}}$	TN with covariance matrices $\{\boldsymbol{\Sigma}_1,\ldots,\boldsymbol{\Sigma}_M\}$
α_k	weight associated to the k-th auxiliary domain	$\boldsymbol{\Sigma}_m^{\boldsymbol{\mathcal{A}}}$	$\sum_{k=1}^K \alpha_k\boldsymbol{\Sigma}_m^{(k)}$
$\mathbf{0}$	zero tensor of appropriate dimension	$\boldsymbol{I}_{p_m}$	p_m-dimensional identity matrix
$\boldsymbol{\Delta}_m^{(k)}$	$\boldsymbol{\Omega}_m\boldsymbol{\Sigma}_m^{(k)} - \boldsymbol{I}_{p_m}$	$\boldsymbol{\Delta}_m$	$\sum_{k=1}^K \alpha_k\boldsymbol{\Delta}_m^{(k)}$
$\det(\cdot)$	determinant of a matrix	$\text{tr}(\cdot)$	trace of a matrix
λ_{1m}	turning parameter for estimating $\boldsymbol{\Delta}_m$	λ_{2m}	tuning parameter for estimating $\boldsymbol{\Omega}_m$
h	informative threshold for the auxiliary domains	$\boldsymbol{\mathcal{A}}$	informative auxiliary domains
s_{mj}	sparsity parameter: $\|\boldsymbol{\Omega}^*_{m(j)}\|_0$	$\bar{s}$	$\max_{m\in[M],j\in[p_m]} s_{mj}$

2.2 Tensor GGMs

Suppose that an order-M tensor $\boldsymbol{\mathcal{X}} = (x_{j_1,\cdots,j_M}) \in \mathbb{R}^{p_1\times\cdots\times p_M}$ follows a zero-mean tensor normal distribution, denoted as $\boldsymbol{\mathcal{X}} \sim \text{TN}(\mathbf{0}; \boldsymbol{\Sigma}_1,\cdots,\boldsymbol{\Sigma}_M)$, its probability density function is then defined as

$$p(\boldsymbol{\mathcal{X}} \mid \boldsymbol{\Sigma}_1,\ldots,\boldsymbol{\Sigma}_M) = (2\pi)^{-p/2}\left(\prod_{m=1}^M \det(\boldsymbol{\Sigma}_m)^{-p/(2p_m)}\right)\exp\left(-\frac{1}{2}\left\|\boldsymbol{\mathcal{X}} \times \{\boldsymbol{\Sigma}^{-1/2}\}\right\|_F^2\right), \quad (1)$$

where $\mathbf{\Sigma}_m \in \mathbb{R}^{p_m \times p_m}$ is the mode-m covariance matrix, and $\{\mathbf{\Sigma}^{-1/2}\} = \{\mathbf{\Sigma}_1^{-1/2}, \cdots, \mathbf{\Sigma}_M^{-1/2}\}$. Clearly, the tensor normal distribution extends the multivariate normal distribution (Ghurye and Olkin, 1962; Stein, 1981; Tong and Tong, 1990) with $M = 1$ or matrix normal distribution (Dawid, 1981) with $M = 2$ to a tensor random variable with general order-M. It can be shown that $\boldsymbol{\mathcal{X}} \sim \text{TN}(\mathbf{0}; \mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_M)$ if and only if $\text{vec}(\boldsymbol{\mathcal{X}}) \sim N(\text{vec}(\mathbf{0}); \mathbf{\Sigma}_M \otimes \cdots \otimes \mathbf{\Sigma}_1)$, where $\otimes$ stands for the Kronecker product. We remark that the Kronecker decomposition $\mathbf{\Sigma}_M \otimes \ldots \otimes \mathbf{\Sigma}_1$ is readily identifiable only up to scalar multiplication as the dimensions of the data tensor are given. Precisely, if $\mathbf{\Sigma}_M \otimes \ldots \otimes \mathbf{\Sigma}_1 = \tilde{\mathbf{\Sigma}}_M \otimes \ldots \otimes \tilde{\mathbf{\Sigma}}_1$, we must have $\mathbf{\Sigma}_m = c_m \tilde{\mathbf{\Sigma}}_m$ for $m \in [M]$, and the positive c_m's satisfying $\Pi_{m=1}^M c_m = 1$. Importantly, the graphical structures, or equivalently the conditional independent relationship of the random variables in each mode remain unchanged under such scalar multiplications. To account for such scalar multiplication issue, the naive way is to fix $M - 1$ factor covariance matrices to have unit Frobenius norm, while allowing the remaining one to have varying Frobenius norm.

We consider sparse estimation of $\{\mathbf{\Omega}_m\}_{m=1}^M$ to characterize the conditional independence relation among the features of any given mode of $\boldsymbol{\mathcal{X}}$. Specifically, let $\boldsymbol{\mathcal{X}}_{(j)}^{(m\text{-sub})} \in \mathbb{R}^{p_1 \times \ldots p_{m-1} \times p_{m+1} \times \ldots \times p_M}$ denote the j-th sub-tensor extracted from $\boldsymbol{\mathcal{X}}$ by fixing the index in the m-th mode as j, then $[\mathbf{\Omega}_m]_{(j,j')} = 0$ if and only if $\boldsymbol{\mathcal{X}}_{(j)}^{(m\text{-sub})}$ is independent of $\boldsymbol{\mathcal{X}}_{(j')}^{(m\text{-sub})}$ given all other $\boldsymbol{\mathcal{X}}_{(j'')}^{(m\text{-sub})}$ with $j'' \neq j, j'$. For example, in an order-3 tendor $\boldsymbol{\mathcal{X}}$, x_{j_1,j_2,j_3} denotes the activation level at region j_1 of subject j_2 in the j_3-th fMRI scan over the lateral prefrontal cortex, $[\mathbf{\Omega}_1]_{(j_1,j_1')}$ indicates the regularity strength of regions j_1 and j_1' given the activation levels of all other regions of interests across different subjects and scans, and the activation levels of the region j_1 and j_1' are conditional independent if and only if $[\mathbf{\Omega}_1]_{(j_1,j_1')} = 0$.

Estimation of $\mathbf{\Omega}_m$ amounts to maximizing the likelihood function of $\{\boldsymbol{\mathcal{X}}_i\}_{i=1}^n$ that are independently sampled from (1), which is block multi-convex (Lyu et al., 2019) with respect to $\{\boldsymbol{\Omega_m}\}_{m=1}^M$. Leveraging the multi-convex property, Lyu et al. (2019) proposed to alternatively update one precision matrix with others fixed. Specifically, one can minimize

$$\ell(\mathbf{\Omega}_m) = -\frac{1}{p_m} \log[\det(\mathbf{\Omega}_m)] + \frac{1}{p_m} \text{tr}(\boldsymbol{S}_m \mathbf{\Omega}_m) + \lambda_m \|\mathbf{\Omega}_m\|_{1,\text{off}}, \tag{2}$$

where $\boldsymbol{S}_m = \frac{p_m}{np} \sum_{i=1}^n \boldsymbol{V}_{i(m)} \boldsymbol{V}_{i(m)}^\top$, $\boldsymbol{V}_{i(m)} = [\boldsymbol{\mathcal{X}}_i]_{(m)} (\mathbf{\Omega}_M^{1/2} \otimes \cdots \otimes \mathbf{\Omega}_{m+1}^{1/2} \otimes \mathbf{\Omega}_{m-1}^{1/2} \otimes \cdots \otimes \mathbf{\Omega}_1^{1/2})$, and $\det(\mathbf{\Omega}_m)$ is the determinant of $\mathbf{\Omega}_m$. This optimization task can be efficiently solved via the graphical lasso algorithm (Friedman et al., 2008), and the obtained estimates of $\mathbf{\Omega}_m$'s enjoy the asymptotic consistency following standard treatment of penalized maximum likelihood estimation (Lyu et al., 2019). Yet, the applicability of such consistency results requires a sufficiently large sample size, which is usually not realistic in practice. To this end, we propose a transfer learning method to leverage information from auxiliary domains so as to enhance the learning performance in the target domain.

3. Proposed method

Suppose that besides observations $\{\boldsymbol{\mathcal{X}}_i\}_{i=1}^n$ from the target domain, observations $\{\boldsymbol{\mathcal{X}}_i^{(k)}\}_{i=1}^{n_k}$; $k \in [K]$ from some auxiliary domains are also available. For example, in the ADHD brain functional network dataset, $\{\boldsymbol{\mathcal{X}}_i\}_{i=1}^n$ are the dynamic activation levels of many brain regions of interests collected from some fMRI scans at one neuroscience institute, and $\{\boldsymbol{\mathcal{X}}_i^{(k)}\}_{i=1}^{n_k}$ are collected from $K = 6$ other neuroscience institutes for better data analysis in the target institute. That is, $\boldsymbol{\mathcal{X}}_i$'s are independently generated from $\mathrm{TN}(\mathbf{0}; \boldsymbol{\Sigma}_1, \cdots, \boldsymbol{\Sigma}_M)$ and $\boldsymbol{\mathcal{X}}_i^{(k)}$'s are independently generated from $\mathrm{TN}(\mathbf{0}; \boldsymbol{\Sigma}_1^{(k)}, \cdots, \boldsymbol{\Sigma}_M^{(k)})$ with $\Sigma_m \in \mathbb{R}^{p_m \times p_m}$ and $\boldsymbol{\Sigma}_m^{(k)} \in \mathbb{R}^{p_m \times p_m}$. Particularly, we are interested in estimating the precision matrix $\boldsymbol{\Omega}_m = (\boldsymbol{\Sigma}_m)^{-1}$ in the target domain for $m \in [M]$ via transfer learning on the tensor GGMs.

3.1 Divergence matrix

The key to transfer learning is to construct a similarity measure between parameters of interest in the auxiliary and target domains. Particularly, let $TN_{\Sigma^{(k)}}$ and TN_{Σ} denote $\mathrm{TN}(\mathbf{0}; \boldsymbol{\Sigma}_1^{(k)}, \cdots, \boldsymbol{\Sigma}_M^{(k)})$ and $\mathrm{TN}(\mathbf{0}; \boldsymbol{\Sigma}_1, \cdots, \boldsymbol{\Sigma}_M)$ for short, and we consider the Kullback–Leibler (KL) divergence between $TN_{\Sigma^{(k)}}$ and TN_{Σ},

$$
\begin{aligned}
KL(TN_{\Sigma^{(k)}} || TN_{\Sigma}) = & -\sum_{m=1}^{M} \frac{p}{2p_m} \log[\det(\boldsymbol{\Omega}_m \boldsymbol{\Sigma}_m^{(k)})] \\
& + \frac{1}{2}\left\{ \mathbb{E}\left(\|\boldsymbol{\mathcal{X}}^{(k)} \times \{\boldsymbol{\Omega}^{1/2}\}\|_F^2 \right) - \mathbb{E}\left(\|\boldsymbol{\mathcal{X}}^{(k)} \times \{(\boldsymbol{\Omega}^{(k)})^{1/2}\}\|_F^2 \right) \right\},
\end{aligned}
$$

where $\{\boldsymbol{\Omega}^{1/2}\} = \{\boldsymbol{\Omega}_1^{1/2}, \cdots, \boldsymbol{\Omega}_M^{1/2}\}$ and $\{(\boldsymbol{\Omega}^{(k)})^{1/2}\} = \{(\boldsymbol{\Omega}_1^{(k)})^{1/2}, \cdots, (\boldsymbol{\Omega}_M^{(k)})^{1/2}\}$.

Define the *divergence matrix* as $\boldsymbol{\Delta}_m^{(k)} = \boldsymbol{\Omega}_m \boldsymbol{\Sigma}_m^{(k)} - \boldsymbol{I}_{p_m}$, where $\boldsymbol{I}_{p_m}$ is the p_m-dimensional identity matrix. Clearly, it gets closer to $\mathbf{0}$ when $\boldsymbol{\Sigma}_m^{(k)}$ gets closer to $\boldsymbol{\Sigma}_m$, and thus it provides a natural measure of the similarity between $\boldsymbol{\Sigma}_m^{(k)}$ and $\boldsymbol{\Sigma}_m$. More interestingly, if $\boldsymbol{\Omega}_{m'} = \boldsymbol{\Omega}_{m'}^{(k)}$ for all $m' \neq m$, it follows that

$$
KL(TN_{\Sigma^{(k)}} || TN_{\Sigma}) = -\frac{p}{2p_m} \log[\det(\boldsymbol{\Delta}_m^{(k)} + \boldsymbol{I}_{p_m})] + \frac{p}{2p_m} \mathrm{tr}[\boldsymbol{\Delta}_m^{(k)}],
$$

which is solely parametrized by $\boldsymbol{\Delta}_m^{(k)}$.

To leverage information of all auxiliary domains, we consider the weighted average of the covariance and divergence matrices as follows,

$$
\boldsymbol{\Sigma}_m^{\mathcal{A}} = \sum_{k=1}^{K} \alpha_k \boldsymbol{\Sigma}_m^{(k)} \text{ and } \boldsymbol{\Delta}_m = \sum_{k=1}^{K} \alpha_k \boldsymbol{\Delta}_m^{(k)}, \text{ with } \sum_{k=1}^{K} \alpha_k = 1,
$$

where the choice of weights $\{\alpha_k\}_{k=1}^K$ shall depend on the contribution of each auxiliary domain and will be discussed in detail in Section 3.3. Also, it holds true that $\boldsymbol{\Omega}_m \boldsymbol{\Sigma}_m^{\mathcal{A}} - \boldsymbol{\Delta}_m - \boldsymbol{I}_{p_m} = \mathbf{0}$.

3.2 Separable transfer estimation

For each $m \in [M]$, we first estimate $\boldsymbol{\Delta}_m$ via samples from both the auxiliary and target domains and then estimate $\boldsymbol{\Omega}_m$ by leveraging only the auxiliary samples. Accordingly, we design two specific loss functions for $\boldsymbol{\Delta}_m$ and $\boldsymbol{\Omega}_m$ as

$$\mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Omega}_m) = \frac{1}{2}\operatorname{tr}\{\boldsymbol{\Delta}_m^\top \boldsymbol{\Delta}_m\} - \operatorname{tr}\{\left(\boldsymbol{\Omega}_m \boldsymbol{\Sigma}_m^{\mathcal{A}} - \boldsymbol{I}_{p_m}\right)^\top \boldsymbol{\Delta}_m\},$$
$$\mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Delta}_m) = \frac{1}{2}\operatorname{tr}\{\boldsymbol{\Omega}_m^\top \boldsymbol{\Sigma}_m^{\mathcal{A}} \boldsymbol{\Omega}_m\} - \operatorname{tr}\{(\boldsymbol{\Delta}_m^\top + \boldsymbol{I}_{p_m})\boldsymbol{\Omega}_m\},$$

where $\boldsymbol{\Sigma}_m^{\mathcal{A}} = \sum_{k=1}^K \alpha_k \boldsymbol{\Sigma}_m^{(k)}$ for any $\{\alpha_k\}_{k=1}^K$ satisfying $\sum_{k=1}^K \alpha_k = 1$. The two loss functions are expressed as the difference of two trace operators, which share similar spirit with the D-trace loss (Zhang and Zou, 2014).

Lemma 1 *Both loss functions $\mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Omega}_m)$ and $\mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Delta}_m)$ are convex with respect to $\boldsymbol{\Delta}_m$ and $\boldsymbol{\Omega}_m$, respectively. Furthermore, $\boldsymbol{\Delta}_m^*$ and $\boldsymbol{\Omega}_m^*$ are unique minimizers of $\mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)*}\}_{k=1}^K, \boldsymbol{\Omega}_m^*)$ and $\mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)*}\}_{k=1}^K, \boldsymbol{\Delta}_m^*)$, respectively.*

By Lemma 1, the two empirical loss functions are suitable to get an accurate estimation of $\boldsymbol{\Delta}_m^*$ and $\boldsymbol{\Omega}_m^*$. Furthermore, both empirical losses can be equipped with various regularization terms if additional structures are desired.

In view of the above discussion, for each mode, a multi-step method can be proposed to realize the transfer learning of tensor graphical models.

Step 1. Initialization. Estimate $\{\widehat{\boldsymbol{\Omega}}_m^{(0)}\}_{m=1}^M$ based on target samples $\{\boldsymbol{\mathcal{X}}_i\}_{i=1}^n$, and $\{\widehat{\boldsymbol{\Omega}}_m^{(k)}\}_{m=1}^M$ based on auxiliary samples $\{\boldsymbol{\mathcal{X}}_i^{(k)}\}_{i=1}^{n_k}$, for $k \in [K]$, using the separable estimation approach (Lyu et al., 2019). Then, define

$$\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} = \sum_{k=1}^K \alpha_k \widehat{\boldsymbol{\Sigma}}_m^{(k)}, \quad \text{where } \widehat{\boldsymbol{\Sigma}}_m^{(k)} = \frac{p_m}{n_k p}\sum_{i=1}^{n_k} \widehat{\boldsymbol{V}}_{i,m}^{(k)} \widehat{\boldsymbol{V}}_{i,m}^{(k)\top},$$
$$\widehat{\boldsymbol{V}}_{i,m}^{(k)} = [\boldsymbol{\mathcal{X}}_i^{(k)}]_{(m)} \left[(\widehat{\boldsymbol{\Omega}}_M^{(k)})^{1/2} \otimes \cdots \otimes (\widehat{\boldsymbol{\Omega}}_{m+1}^{(k)})^{1/2} \otimes (\widehat{\boldsymbol{\Omega}}_{m-1}^{(k)})^{1/2} \otimes \cdots \otimes (\widehat{\boldsymbol{\Omega}}_1^{(k)})^{1/2}\right].$$

Step 2. For each $m \in [M]$, perform the following two estimation steps separately.

(a). Estimate the divergence matrix of mode-m,

$$\widehat{\boldsymbol{\Delta}}_m = \arg\min \mathcal{Q}_1(\boldsymbol{\Delta}_m), \tag{3}$$

where $\mathcal{Q}_1(\boldsymbol{\Delta}_m) = \frac{1}{2}\operatorname{tr}\{\boldsymbol{\Delta}_m^\top \boldsymbol{\Delta}_m\} - \operatorname{tr}\left\{(\widehat{\boldsymbol{\Omega}}_m^{(0)} \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{I}_{p_m})^\top \boldsymbol{\Delta}_m\right\} + \lambda_{1m}\|\boldsymbol{\Delta}_m\|_1$.

(b). Estimate the precision matrix of mode-m,

$$\widehat{\boldsymbol{\Omega}}_m = \arg\min \mathcal{Q}_2(\boldsymbol{\Omega}_m), \tag{4}$$

where $\mathcal{Q}_2(\boldsymbol{\Omega}_m) = \frac{1}{2}\operatorname{tr}\{\boldsymbol{\Omega}_m^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} \boldsymbol{\Omega}_m\} - \operatorname{tr}\{(\widehat{\boldsymbol{\Delta}}_m^\top + \boldsymbol{I}_{p_m})\boldsymbol{\Omega}_m\} + \lambda_{2m}\|\boldsymbol{\Omega}_m\|_{1,\text{off}}$.

In Step 2(a), $\widehat{\boldsymbol{\Delta}}_m$ can be considered as an adaptive thresholding of a naive estimate, $\widehat{\boldsymbol{\Omega}}_m^{(0)} \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{I}_{p_m}$, which is inspired by the definition of $\widehat{\boldsymbol{\Delta}}_m$. If the difference between the target and auxiliary domains in mode-m precision matrices are small enough, some elements of $\widehat{\boldsymbol{\Delta}}_m$ can shrink to zero with appropriate λ_{1m}. The thresholding can improve the estimation of $\widehat{\boldsymbol{\Delta}}_m$ with the help of the auxiliary samples. Correspondingly, $\boldsymbol{\Omega}_m$ can also be better estimated via $\widehat{\boldsymbol{\Delta}}_m$ by leveraging only the auxiliary samples in Step 2(b). We note that in the proposed transfer learning procedure, the separable transfer estimation is performed for each mode in turn. This procedure is mainly designed for high-order tensor GGMs, but can be naturally applied to vector-valued data thanks to the fact that a vector can be regarded as the simplest tensor with only one mode. Therefore, the transfer learning method for vector-valued GGMs (Li et al., 2022b) can be accommodated as a special case.

Moreover, the similarity between the target and auxiliary domains may be weak in some scenarios, so that the learning performance in the target domain may be deteriorated due to information transfer, which is so-called "negative transfer". Under supervised cases, some excellent solutions utilizing label information have been proposed (Shu et al., 2019). However, handling the negative transfer in unsupervised models remains challenging. One practical solution is to further perform a model selection step following Li et al. (2022b), which guarantees that transfer learning is no less effective than using only the target domain.

To this end, we can first randomly split the data from the target domain into two folds $\mathcal{N}$ and $\mathcal{N}^C$, such that $\mathcal{N} \bigcup \mathcal{N}^C = \{1, \cdots, n\}$ and $\mathrm{card}(\mathcal{N}) = cn$, for some fraction $0 < c < 1$. As suggested in Li et al. (2022b), the value of c might not be sensitive in practice, and we thus set $c = 0.6$ in all of our numerical experiments. Second, we use the subjects in $\mathcal{N}$ to construct the initialization of the separable transfer estimation according to Step 1, and we denote the resulting initializers as $\{\widetilde{\mathbf{\Omega}}_m^{(0)}\}_{m=1}^M$. Third, we apply the data in $\mathcal{N}^C$ for model selection. Specifically, we compute $\widetilde{\mathbf{\Sigma}}_m = \frac{p_m}{(1-c)np} \sum_{i \in \mathcal{N}^C} \widetilde{\boldsymbol{V}}_{i,m} \widetilde{\boldsymbol{V}}_{i,m}^\top$ with $\widetilde{\boldsymbol{V}}_{i,m} = [\boldsymbol{\mathcal{X}}_i]_{(m)} \left[(\widetilde{\mathbf{\Omega}}_M^{(0)})^{1/2} \otimes \cdots \otimes (\widetilde{\mathbf{\Omega}}_{m+1}^{(0)})^{1/2} \otimes (\widetilde{\mathbf{\Omega}}_{m-1}^{(0)})^{1/2} \otimes \cdots \otimes (\widetilde{\mathbf{\Omega}}_1^{(0)})^{1/2} \right]$, for $i \in \mathcal{N}^C$ and $m \in [M]$. For each $j \in [p_m]$, the j-th column selector $\widehat{\omega}_{m,j}$ for the mode-m precision matrix is defined as

$$\widehat{w}_{m,j} = \underset{w \in \{(0,1)^\top, (1,0)^\top\}}{\arg\min} \|\widetilde{\mathbf{\Sigma}}_m (\widehat{\mathbf{\Omega}}_{m(j)}^{(0)}, \widehat{\mathbf{\Omega}}_{m(j)}) w - \boldsymbol{I}_{p_m(j)}\|_2^2,$$

where $\widehat{\mathbf{\Omega}}_m^{(\mathbf{0})}$ is the initialization result from Step 1 before transfer learning and $\widehat{\mathbf{\Omega}}_m$ is the transfer learning estimator from (4), both of which are estimated using all the data in $\mathcal{N} \bigcup \mathcal{N}^C$, and $\widehat{\mathbf{\Omega}}_{m(j)}^{(0)}$, $\widehat{\mathbf{\Omega}}_{m(j)}$, and $\boldsymbol{I}_{p_m(j)}$ are the j-th columns of $\widehat{\mathbf{\Omega}}_m^{(0)}$, $\widehat{\mathbf{\Omega}}_m$, and $\boldsymbol{I}_{p_m}$, respectively. Then the j-th column of the final estimator is constructed as

$$\widehat{\mathbf{\Omega}}_{m(j)}^{(f)} = (\widehat{\mathbf{\Omega}}_{m(j)}^{(0)}, \widehat{\mathbf{\Omega}}_{m(j)}) \widehat{w}_{m(j)}, \tag{5}$$

for $j \in [p_m]$ and $m \in [M]$. Note that $\widehat{\mathbf{\Omega}}_m^{(f)}$ is not symmetric in general, and $(\widehat{\mathbf{\Omega}}_m^{(f)} + [\widehat{\mathbf{\Omega}}_m^{(f)}]^\top)/2$ can be used as a symmetrization estimate.

The selection step realizes a model selection between the $\widehat{\mathbf{\Omega}}_{m(j)}^{(0)}$ and $\widehat{\mathbf{\Omega}}_{m(j)}$, which yields satisfactory theoretical and numerical performance (Li et al., 2022b). Furthermore, it can be theoretically guaranteed that the final estimate is positive definite (Liu and Luo, 2015; Li et al., 2022b).

3.3 Construction of weights

How to aggregate multiple auxiliary domains is an important initial part of transfer learning. A natural choice of the weights is to set

$$\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} = \sum_{k=1}^{K} \alpha_k \widehat{\boldsymbol{\Sigma}}_m^{(k)}, \text{ with } \alpha_k = n_k/N \text{ and } N = \sum_{k=1}^{K} n_k, \tag{6}$$

following from the fact that the auxiliary domain with larger sample size shall be more important, which is similar to Li et al. (2022b). Yet, it does not take into account the similarities between the target and auxiliary domains. If there are some large non-informative auxiliary domains that are extremely different from the target domain, the final model selection step can force the initial estimator using only the target domain to be selected. In this sense, although the model selection can guarantee that transfer learning is no less effective than using the target domain only, it may also offset the potential improvement benefiting from the informative auxiliary domains with a positive impact.

To address this challenge, we further design some data-adaptive weights for auxiliary covariance matrices, in which weights are constructed by combining both sample sizes and the estimated differences between the target and auxiliary domains. Particularly, we set

$$\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} = \sum_{k=1}^{K} \alpha_k \widehat{\boldsymbol{\Sigma}}_m^{(k)}, \text{ with } \alpha_k = \frac{n_k/\widehat{h}_k}{\sum_{k=1}^{K}(n_k/\widehat{h}_k)}, \tag{7}$$

where $\widehat{h}_k = \max_{m\in[M]} \|\widehat{\boldsymbol{\Delta}}_m^{(k)}\|_{1,\infty}$ and $\widehat{\boldsymbol{\Delta}}_m^{(k)} = \widehat{\boldsymbol{\Omega}}_m^{(0)}\widehat{\boldsymbol{\Sigma}}_m^{(k)} - \boldsymbol{I}_{p_m}$. Clearly, for auxiliary domains with similar sample sizes, the weight for the one with a smaller difference from the target domain is larger, while the weight for the one with an extremely large difference can tend to zero to adaptively the negative transfer. Here we note that the type of norm for measuring similarity is not critical, and the specified L_1-norm is only for keeping with the form of theoretical analysis and may be replaced by other norms with slight modification. It is also interesting to note that even with such data-adaptive weights, the model selection step in (5) is still necessary to safeguard the extreme case where all the auxiliary domains are non-informative.

3.4 Computing algorithm

For Step 2(a), define $\widehat{\boldsymbol{B}}_m = \widehat{\boldsymbol{\Omega}}_m^{(0)}\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{I}_{p_m}$ for each $m \in [M]$, and then (3) can be rewritten as

$$\mathcal{Q}_1(\boldsymbol{\Delta}_m) = \frac{1}{2}\sum_{1\leqslant i,j\leqslant p_m} [\boldsymbol{\Delta}_m]_{(i,j)}^2 - \sum_{1\leqslant i,j\leqslant p_m} [\widehat{\boldsymbol{B}}_m]_{(i,j)}[\boldsymbol{\Delta}_m]_{(i,j)} + \lambda_{1m}\sum_{1\leqslant i,j\leqslant p_m} |[\boldsymbol{\Delta}_m]_{(i,j)}|,$$

where $[\boldsymbol{\Delta}_m]_{(i,j)}$ and $[\widehat{\boldsymbol{B}}_m]_{(i,j)}$ are the (i,j) entries of $\boldsymbol{\Delta}_m$ and $\widehat{\boldsymbol{B}}_m$, respectively. It can be separated into p_m^2 lasso-type optimizations; that is, for any i and j,

$$[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} = \arg\min_{\Delta}\left\{\frac{1}{2}(\Delta - [\widehat{\boldsymbol{B}}_m]_{(i,j)})^2 + \lambda_{1m}|\Delta|\right\} = \mathcal{T}([\widehat{\boldsymbol{B}}_m]_{(i,j)}, \lambda_{1m}),$$

where $\mathcal{T}(z, \lambda) = \mathrm{sign}(z)\max(0, |z| - \lambda)$.

For Step 2(b), note that (4) can be rewritten as

$$\mathcal{Q}_2(\boldsymbol{\Omega}_m) = \sum_{1 \leqslant j \leqslant p_m} \left\{ \frac{1}{2} \boldsymbol{\Omega}_{m(j)}^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} \boldsymbol{\Omega}_{m(j)} - \boldsymbol{\Omega}_{m(j)}^\top (\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) + \lambda_{2m} \|\boldsymbol{\Omega}_{m(j)}\|_1 - \lambda_{2m} |[\boldsymbol{\Omega}_m]_{(j,j)}| \right\},$$

where $\boldsymbol{\Omega}_{m(j)}$ and $\boldsymbol{I}_{p_m(j)}$ are the j-th columns of $\boldsymbol{\Omega}_m$ and $\boldsymbol{I}_{p_m}$, respectively. It can be separated into p_m optimizations; that is, for any j,

$$\widehat{\boldsymbol{\Omega}}_{m(j)} = \arg\min_{\boldsymbol{\theta}} \left\{ \frac{1}{2} \boldsymbol{\theta}^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} \boldsymbol{\theta} - \boldsymbol{\theta}^\top (\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) + \lambda_{2m} \|\boldsymbol{\theta}_{(-j)}\|_1 \right\}, \tag{8}$$

where $\boldsymbol{\theta}_{(-j)}$ is the sub-vector of $\boldsymbol{\theta}$ with the j-th component removed.

For the optimization of (8), we adopt the coordinate descent algorithm. Particularly, at iteration $t+1$, the updating formula of θ_i, i-th component of $\boldsymbol{\theta}$, with other components $\{\theta_{i'}^{(t+1)}, i' < i;\ \theta_{i'}^{(t)}, i' > i\}$ fixed, are

$$\theta_i^{(t+1)} = [\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,i)}^{-1} \mathcal{T}(\xi^{(t)}, \lambda_{2m} I(i \neq j)), \text{ for } i = 1, \cdots, p_m,$$

where $\xi^{(t)} = [\widehat{\boldsymbol{\Delta}}_m + \boldsymbol{I}_{p_m}]_{(i,j)} - \sum_{i' < i} \theta_{i'}^{(t+1)} [\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,i')} - \sum_{i' > i} \theta_{i'}^{(t)} [\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,i')}$.

As computational remarks, explicit solutions can be derived in each step of the algorithm, which makes it very efficient. The initial values of $\boldsymbol{\theta}$ are set as $\widehat{\boldsymbol{\Omega}}_{m(j)}^{(0)}$. Note that these developments are specifically for the Lasso penalty, and optimization with other penalties may require minor modifications. Convergence properties of the algorithm can be guaranteed, thanks to the convexity of the objective function. As for the tuning parameter selection, we set $\lambda_{1m} = 2\|\widehat{\boldsymbol{\Omega}}_m^{(0)}\|_{1,\infty} \sqrt{\frac{p_m \log p_m}{np}}$ for mode-m, following Li et al. (2022b). For λ_{2m}, it is suggested to be determined via minimizing a BIC-type criterion, $\frac{1}{2}\,\mathrm{tr}\{\widehat{\boldsymbol{\Omega}}_m^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} \widehat{\boldsymbol{\Omega}}_m\} - \mathrm{tr}\{(\widehat{\boldsymbol{\Delta}}_m^\top + \boldsymbol{I}_{p_m})\widehat{\boldsymbol{\Omega}}_m\} + \frac{\log N}{N}\|\widehat{\boldsymbol{\Omega}}_m\|_0$, for each mode.

4. Statistical properties

In this section, we establish some theoretical properties of the proposed transfer learning method. The following technical conditions are made.

Condition 1 *For each $m \in [M]$ and $k \in [K]$, $\|\boldsymbol{\Omega}_m^*\|_{1,\infty}$ and $\|\boldsymbol{\Omega}_m^{(k)*}\|_{1,\infty}$ are bounded, and there is a constant C_1, satisfying $1/C_1 \leqslant \psi_{\min}(\boldsymbol{\Sigma}_m^*) \leqslant \psi_{\max}(\boldsymbol{\Sigma}_m^*) \leqslant C_1$ and $1/C_1 \leqslant \psi_{\min}(\boldsymbol{\Sigma}_m^{(k)*}) \leqslant \psi_{\max}(\boldsymbol{\Sigma}_m^{(k)*}) \leqslant C_1$.*

Condition 2 *Denote $\boldsymbol{\Gamma}_m^* = \boldsymbol{\Sigma}_m^* \otimes \boldsymbol{\Sigma}_m^*$, $S_m = \{(i,j) : [\boldsymbol{\Omega}_m^*]_{(i,j)} \neq 0\}$, and $[\boldsymbol{\Gamma}_m^*]_{(S_m,S_m)}$ the sub-matrix with rows and columns of $\boldsymbol{\Gamma}_m^*$ indexed by S_m and S_m, respectively. For each $m \in [M]$, $\|\boldsymbol{\Sigma}_m^*\|_{1,\infty}$ and $\|([\boldsymbol{\Gamma}_m^*]_{(S_m,S_m)})^{-1}\|_{1,\infty}$ are bounded, and there exists some constant $C_2 \in (0,1]$ such that $\max_{e \in S_m^C} \|[\boldsymbol{\Gamma}_m^*]_{(e,S_m)}([\boldsymbol{\Gamma}_m^*]_{(S_m,S_m)})^{-1}\|_1 \leqslant 1 - C_2$.*

Condition 1 has been commonly assumed in the literature of Gaussian graphical models (Lam and Fan, 2009; Zhang and Zou, 2014). Condition 2 limits the influence of the non-connected terms in S_m^C on the connected edges in S_m, which is also widely assumed to establish theoretical properties of lasso-type estimators (Ravikumar et al., 2011; Zhang and Zou, 2014; Lyu et al., 2019). Denote $\overline{p} = \max_{m\in[M]} p_m$, and $\overline{s} = \max_{m\in[M],j\in[p_m]} s_{mj}$ with $s_{mj} = \|\mathbf{\Omega}^*_{m(j)}\|_0$ that may diverge with n. We first state some existing result in Lyu et al. (2019), which quantifies the asymptotic behavior of the initial estimate $\widehat{\mathbf{\Omega}}^{(0)}_m$.

Lemma 2 *(Lyu et al., 2019) If Condition 1 holds, then* $\|\widehat{\mathbf{\Sigma}}_m - \mathbf{\Sigma}^*_m\|_{\max} = O_p\left(\sqrt{\frac{p_m \log p_m}{np}}\right)$, *for* $m \in [M]$. *If Condition 2 holds,* $\overline{s}\sqrt{\frac{p_m \log p_m}{np}} \ll 1$, *and* $p_1 \asymp \cdots \asymp p_m$, *then* $\|\widehat{\mathbf{\Omega}}^{(0)}_m - \mathbf{\Omega}^*_m\|_{\max} = O_p\left(\sqrt{\frac{p_m \log p_m}{np}}\right)$. *Furthermore, for* $m \in [M]$, *if the minimal signal of* $\mathbf{\Omega}^*_m$ *satisfies that* $\sqrt{\frac{p_m \log p_m}{np}} \lesssim \min_{(i,j)\in S_m} |[\mathbf{\Omega}^*_m]_{(i,j)}|$, *then with probability tending to 1,* $\widehat{S}^{(0)}_m = \{(i,j) : [\widehat{\mathbf{\Omega}}^{(0)}_m]_{(i,j)} \neq 0\} = S_m$.

To describe the similarity between the target domain and auxiliary domains, we define the set of *informative* auxiliary domains as $\mathcal{A}_h = \{k : \max_{m\in[M]}\{\|\mathbf{\Delta}^{(k)*}_m\|_{1,\infty} + \|(\mathbf{\Delta}^{(k)*}_m)^\top\|_{1,\infty}\} \leqslant h\}$ for some positive h. Clearly, h measures the difference between the precision matrices of each mode in the target and the k-th auxiliary domain.

4.1 All auxiliary domains are informative

We first consider an ideal scenario where all available auxiliary domains are informative.

Condition 3 *Assume that* $\mathcal{A}_h = [K]$.

Theorem 1 *Suppose all the conditions of Lemma 2 and Condition 3 are met,* $n \leqslant N$ *with* $N = \sum_{k=1}^K n_k$, *and* $\lambda_{1m} = C(1+h)\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$ *for a sufficiently large constant* C. *For* $\widehat{\mathbf{\Sigma}}^{\mathcal{A}}_m$ *in (6), it holds true that* $\|\widehat{\mathbf{\Delta}}_m - \mathbf{\Delta}^*_m\|^2_{2,\infty} = O_p(\delta_h)$ *for* $m \in [M]$, *where* $\delta_h = (1+h)h\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h^2$.

Theorem 2 *If the conditions of Theorem 1 hold, and* $\lambda_{2m} = C\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right)$ *for a sufficiently large constant* C, *then* $\|\widehat{\mathbf{\Omega}}_m - \mathbf{\Omega}^*_m\|^2_{2,\infty} \vee \frac{1}{p_m}\|\widehat{\mathbf{\Omega}}_m - \mathbf{\Omega}^*_m\|^2_F = O_p\left(\frac{\overline{s}\overline{p}\log\overline{p}}{(N+n)p} + \delta_h\right)$ *for* $m \in [M]$.

Remark 1 *Note that Lemma 2 implies that* $\|\widehat{\mathbf{\Omega}}^{(0)}_m - \mathbf{\Omega}^*_m\|^2_{2,\infty} \vee \frac{1}{p_m}\|\widehat{\mathbf{\Omega}}^{(0)}_m - \mathbf{\Omega}^*_m\|^2_F = O_p(\frac{\overline{s}\overline{p}\log\overline{p}}{np})$ *for* $m \in [M]$. *It is thus clear that the proposed transfer learning method achieves a faster convergence rate when* $N \gg n$ *and* $h \ll \overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$.

We next establish the variable selection consistency of the proposed transfer learning method in terms of exactly recovering the tensor graphical model. Some additional conditions are necessary.

Condition 4 *Define* $\mathbf{\Sigma}^{\mathcal{A}*}_m = \sum_{k=1}^K \alpha_k \mathbf{\Sigma}^{(k)*}_m$ *with* $\sum_{k=1}^K \alpha_k = 1$, *then for each* $m \in [M]$, $\|\mathbf{\Sigma}^{\mathcal{A}*}_m\|_{1,\infty}$ *and* $\max_{j\in[p_m]} \|([\mathbf{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}$ *are bounded, and there exists some constant* $C_3 \in (0,1]$ *such that* $\max_{j\in[p_m],e\in S^C_{mj}} \|[\mathbf{\Sigma}^{\mathcal{A}*}_m]_{(e,S_{mj})}([\mathbf{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_1 \leqslant 1 - C_3$, *where* $S_{mj} = \{i \in [p_m] : [\mathbf{\Omega}^*_m]_{(i,j)} \neq 0\}$ *and* $S^C_{mj} = \{i \in [p_m] : [\mathbf{\Omega}^*_m]_{(i,j)} = 0\}$.

Condition 5 *Assume that* $\max_{m\in[M],k\in[K]} \|\mathbf{\Delta}_m^{(k)*}\|_{\max} \lesssim h/\overline{s}$.

Condition 4 imposes the irrepresentability condition on the auxiliary domains, to quantify the behavior of $\mathcal{Q}_2(\mathbf{\Omega}_m)$ in Step 2(b). Condition 5 is necessary for establishing estimation error of $\widehat{\mathbf{\Omega}}_m$ in max norm, which is mild due to the fact that $\overline{s} < \overline{p}$.

Theorem 3 *If the conditions of Theorem 2 and Conditions 4 to 5 hold, and h is bounded, then* $\|\widehat{\mathbf{\Omega}}_m - \mathbf{\Omega}_m^*\|_{\max} = O_p\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{(N+n)p}}\right)$ *for* $m \in [M]$*. Furthermore, if* $\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{(N+n)p}} \lesssim \min_{(i,j)\in S_m} |[\mathbf{\Omega}_m^*]_{(i,j)}|$*, then with probability tending to 1,* $\widehat{S}_m = \{(i,j) : [\widehat{\mathbf{\Omega}}_m]_{(i,j)} \neq 0\} = S_m$ *for* $m \in [M]$*.*

It is interesting to note that $\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{(N+n)p}} \ll \sqrt{\frac{\overline{p}\log\overline{p}}{np}}$, if $N \gg n$ and $h \ll \overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$, thus it can be concluded that the minimum signal condition required for eatablishing the variable selection consistency for the proposed transfer learning method is much weaker than that when using the target domain only.

4.2 At least one informative auxiliary domain

We now turn to a more complex case where some non-informative auxiliary domains dominates, so that the model selection step may force the final estimator to become the initial estimate based on the target domain only, and then another part of information on the informative auxiliary domains will be offset. At this point, it only ensures that the transfer learning does not deteriorate, but does not make full use of positive information from informative auxiliary domains. Therefore, we further consider the theoretical properties of the proposed method based on $\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}$ constructed by the data-adaptive weights.

Condition 6 *There exists a* $h \lesssim \overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$ *such that the positive set* $\mathcal{A}_h \subseteq [K]$ *is non-empty.*

Condition 7 *Re-define* $\mathbf{\Sigma}_m^{\mathcal{A}*} = \sum_{k\in\mathcal{A}_h} \alpha_k \mathbf{\Sigma}_m^{(k)*}$ *with* $\sum_{k\in\mathcal{A}_h} \alpha_k = 1$*, then for each* $m \in [M]$*,* $\|\mathbf{\Sigma}_m^{\mathcal{A}*}\|_{1,\infty}$ *and* $\max_{j\in[p_m]} \|([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}$ *are bounded, and there exists some constant* $C_3 \in (0,1]$ *such that* $\max_{j\in[p_m],e\in S_{mj}^C} \|[\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(e,S_{mj})}([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_1 \leqslant 1 - C_3$*, where* $S_{mj} = \{i \in [p_m] : [\mathbf{\Omega}_m^*]_{(i,j)} \neq 0\}$ *and* $S_{mj}^C = \{i \in [p_m] : [\mathbf{\Omega}_m^*]_{(i,j)} = 0\}$*.*

Condition 8 *Assume that* $\max_{m\in[M],k\in\mathcal{A}_h} \|\mathbf{\Delta}_m^{(k)*}\|_{\max} \lesssim h/\overline{s}$.

Conditions 6 to 8 are weakened forms of Conditions 3 to 5, respectively, in which the assumption of similarity is only imposed on informative auxiliary domains.

Theorem 4 *Suppose all the conditions of Lemma 2 and Condition 6 are met,* $n_1 \asymp \cdots \asymp n_K$*,* $n \leqslant N_{\mathcal{A}}$ *with* $N_{\mathcal{A}} = \sum_{k\in\mathcal{A}_h} n_k$*,* $K = O(1)$*,* $\lambda_{1m} = C(1+h)\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$*, and* $\lambda_{2m} = C\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{pN_{\mathcal{A}}}}\right)$ *for a sufficiently large constant C. For* $\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}$ *in (7), it holds true that* $\|\widehat{\mathbf{\Omega}}_m - \mathbf{\Omega}_m^*\|_{2,\infty}^2 \vee \frac{1}{p_m}\|\widehat{\mathbf{\Omega}}_m - \mathbf{\Omega}_m^*\|_F^2 = O_p\left(\frac{\overline{s}\overline{p}\log\overline{p}}{(N_{\mathcal{A}}+n)p} + \delta_h\right)$ *for* $m \in [M]$*, where* $\delta_h =$

$(1+h)h\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h^2$. *Furthermore, if Conditions 7 to 8 hold, and assume that* $\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{(N_{\mathcal{A}}+n)p}} \lesssim \min_{(i,j)\in S_m} |[\mathbf{\Omega}^*_m]_{(i,j)}|$, *then with probability tending to 1,* $\|\widehat{\mathbf{\Omega}}_m - \mathbf{\Omega}^*_m\|_{\max} = O_p\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{(N_{\mathcal{A}}+n)p}}\right)$ *and* $\widehat{S}_m = S_m$ *for* $m \in [M]$.

It is clear that as long as there is at least one informative auxiliary domain, satisfying $N_{\mathcal{A}} \gg n$ and $h \ll \overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$, the proposed transfer learning method based on data-adaptively defined $\widehat{\mathbf{\Sigma}}^{\mathcal{A}}_m$ can improve estimation errors benefiting from its information, and is not affected by the possible presence of non-informative auxiliary domains, which shows the powerful robustness to complex scenarios.

Remark 2 *If the ideal assumption about informative auxiliary domains is violated in practice, the transfer learning may be counterproductive. As suggested in Li et al. (2022b), the selection step (5) can theoretically guarantee that the final estimator* $\widehat{\mathbf{\Omega}}^{(f)}_m$ *is as effective as* $\widehat{\mathbf{\Omega}}_m$ *if* $h \lesssim \overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$, *and* $\widehat{\mathbf{\Omega}}^{(f)}_m$ *is still no less effective than* $\widehat{\mathbf{\Omega}}^{(0)}_m$ *if* $h \gg \overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$ *for* $\widehat{\mathbf{\Sigma}}^{\mathcal{A}}_m$ *in (6), or the informative set* $\mathcal{A}_h$ *is empty for* $\widehat{\mathbf{\Sigma}}^{\mathcal{A}}_m$ *in (7).*

5. Simulation

We consider two types of target graphs.

- Chain graph. For each $m \in [M]$, the (i,j)-th entry of $\mathbf{\Omega}^*_m$ is set as 1 if $i = j$; $\exp(-\rho_{ij}/2)$ with $\rho_{ij} = \rho_{ji}$ generated from Unif$(0.5, 1)$, if $|i-j| = 1$; and 0, if $|i-j| > 1$.
- Nearest neighbor graph. For each $m \in [M]$, we randomly generate p_m points in a unit square and locate four nearest neighbors for each point. The corresponding entries in $\mathbf{\Omega}_m$ are uniformly sampled from $[-1,-0.5] \cup [0.5,1]$. The final precision matrix is generated as $\mathbf{\Omega}^*_m = \mathbf{\Omega}_m + |\psi_{\min}(\mathbf{\Omega}_m) + 0.2|\boldsymbol{I}_{p_m}$ to ensure the positive definiteness.

For each target graph, we set $M = 3$ with dimensions $(p_1, p_2, p_3) = (10, 10, 20)$ or $M = 2$ with dimensions $(p_1, p_2) = (100, 100)$, and set the size of the target graph as $n = 50$. We also consider two different simulation scenarios. The subscript h of $\mathcal{A}_h$ is removed for simplicity in this section.

Scenario 1. We consider $\mathcal{A} = [K]$ and vary $K \in \{1, \cdots, 5\}$, that is, all auxiliary domains are informative with size $n_k = 80$ for $k \in [K]$, where $[\mathbf{\Delta}^{(k)}_m]_{(i,j)} = 0$ with probability 0.9 or randomly generated from Unif$[-h_{01}, h_{01}]$ with probability 0.1, and $h_{01} = \sqrt{\frac{\overline{p}\log\overline{p}}{np}}$.

Scenario 2. We fix $K = 5$ with size $n_k = 100$ for $k \in [K]$ and vary card$(\mathcal{A}) \in \{0, 1, \cdots, K\}$. The informative auxiliary domains with $k \in \mathcal{A}$ are generated similarly as Scenario 1. For $k \notin \mathcal{A}$, $[\mathbf{\Delta}^{(k)}_m]_{(i,j)} = 0$ with probability 0.75, or randomly generated from Unif$[-h_{02}, h_{02}]$ with probability 0.25, where $h_{02} = 10\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$.

We compare three methods in Scenario 1, including the single task tensor graphical model using the target domain only, which is implemented in the R package "Tlasso", and the proposed methods with the auxiliary covariance in (6) and (7), denoted as "proposed" and "proposed.v", respectively. In Scenario 2, we further consider another "oracle" method, which applies "proposed" on the target domain and the known informative auxiliary domains.

The performances of the competing methods are measured by a number of metrics: (1) estimation error in Frobenius norm of Kronecker product of precision matrices, defined as $\|\widehat{\mathbf{\Omega}}^{(K)} - \mathbf{\Omega}^{(K)*}\|_F$, where $\widehat{\mathbf{\Omega}}^{(K)} = \widehat{\mathbf{\Omega}}_1^{(f)} \otimes \cdots \otimes \widehat{\mathbf{\Omega}}_M^{(f)}$ and $\mathbf{\Omega}^{(K)*} = \mathbf{\Omega}_1^* \otimes \cdots \otimes \mathbf{\Omega}_M^*$; (2) averaged estimation errors in Frobenius norm of all modes $\frac{1}{M}\sum_{m=1}^{M} \|\widehat{\mathbf{\Omega}}_m^{(f)} - \mathbf{\Omega}_m^*\|_F$; (3) averaged estimation errors in max norm of all modes $\frac{1}{M}\sum_{m=1}^{M} \|\widehat{\mathbf{\Omega}}_m^{(f)} - \mathbf{\Omega}_m^*\|_{\max}$; (4) the true positive rate (TPR) and the true negative rate (TNR) of the Kronecker product of precision matrices; (5) the averaged TPRs and TNRs of all modes. All metrics are averaged based on 100 independent replications.

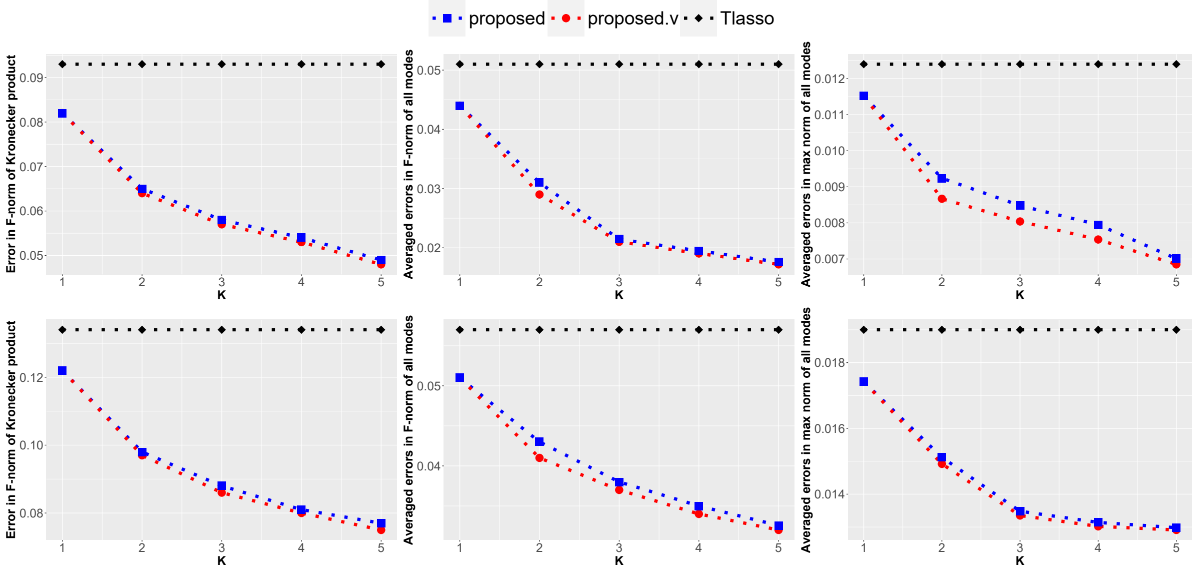

Figure 1: Averaged metrics of estimation errors over 100 replications for Scenario 1 with $M = 3$. The top and bottom rows correspond to the chain and nearest neighbor graph, respectively.

The first three estimation errors are summarized in Figures 1 to 4, whereas the parameter selection metrics are summarized in Tables A1 and A2 in Appendix. Observations made under different settings are very similar. For example, in Scenario 1 where all auxiliary domains are informative, as the number of auxiliary domains K increases, all estimation errors of the two proposed transfer learning-based methods decrease with no significant difference from each other and both are better than Tlasso as expected. In Scenario 2, the two proposed methods are not significantly inferior to Tlasso thanks to the model selection step, when there is no informative auxiliary domain. It is interesting to remark that the two proposed methods have different performance paths as the number of informative

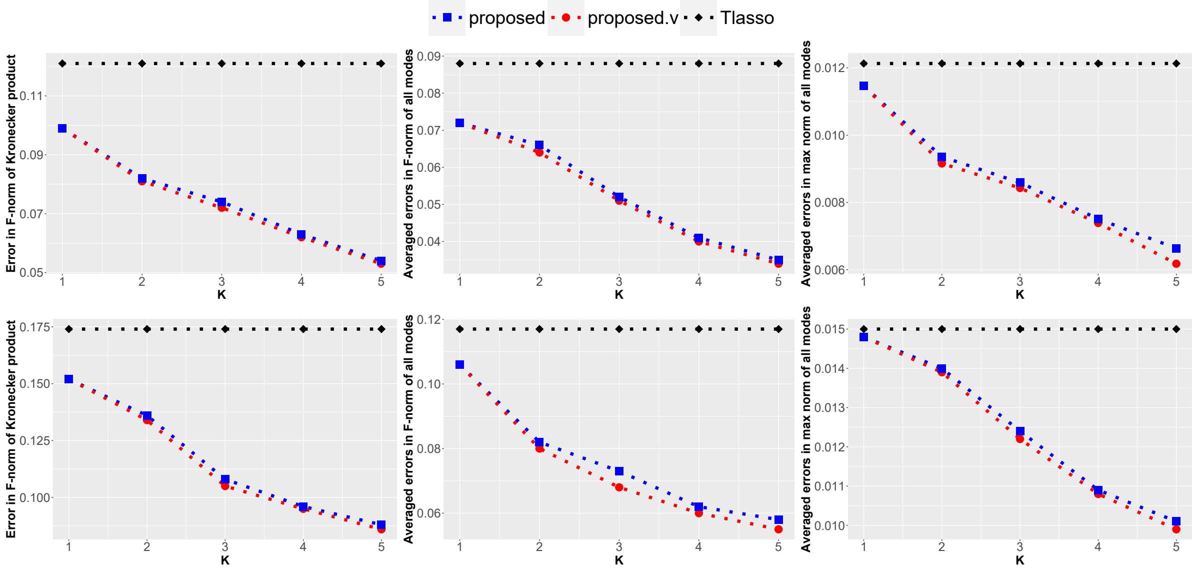

Figure 2: Averaged metrics of estimation errors over 100 replications for Scenario 1 with $M = 2$. The top and bottom rows correspond to the chain and nearest neighbor graph, respectively.

auxiliary domains card($\mathcal{A}$) increases. Specifically, the estimation errors of "proposed.v", whose weights are constructed based on both sample sizes and differences between the target and auxiliary domains, decrease so fast that it can dominate Tlasso even when there is only one informative auxiliary domain, and its overall performance is comparable to "oracle". However, "proposed" is more affected by the non-informative auxiliary domains, whose estimation errors are much larger than "proposed.v" and sometimes only outperform Tlasso when there are relatively large number of informative auxiliary domains. As for the performances of variable selection, all methods have achieved 100% TPR in all settings, and the TNRs of the two proposed methods are significantly improved compared with Tlasso, thanks to the informative auxiliary domains.

6. Real data analyses

6.1 ADHD brain functional networks

In this section, we apply the proposed method to study functional connectivity behaviors among brain regions in the attention deficit hyperactivity disorder (ADHD) disease datasets across multiple sites. In the brain functional network, typically, a node corresponds to an anatomically defined brain region, and the present of connectivity between a pair of nodes to a measure of inter-regional dependency. Resting-state functional magnetic resonance imaging (rs-fMRI) is widely used to measure spontaneous low-frequency blood oxygen level dependent (BOLD) signal fluctuations within several minutes in some brain regions, so

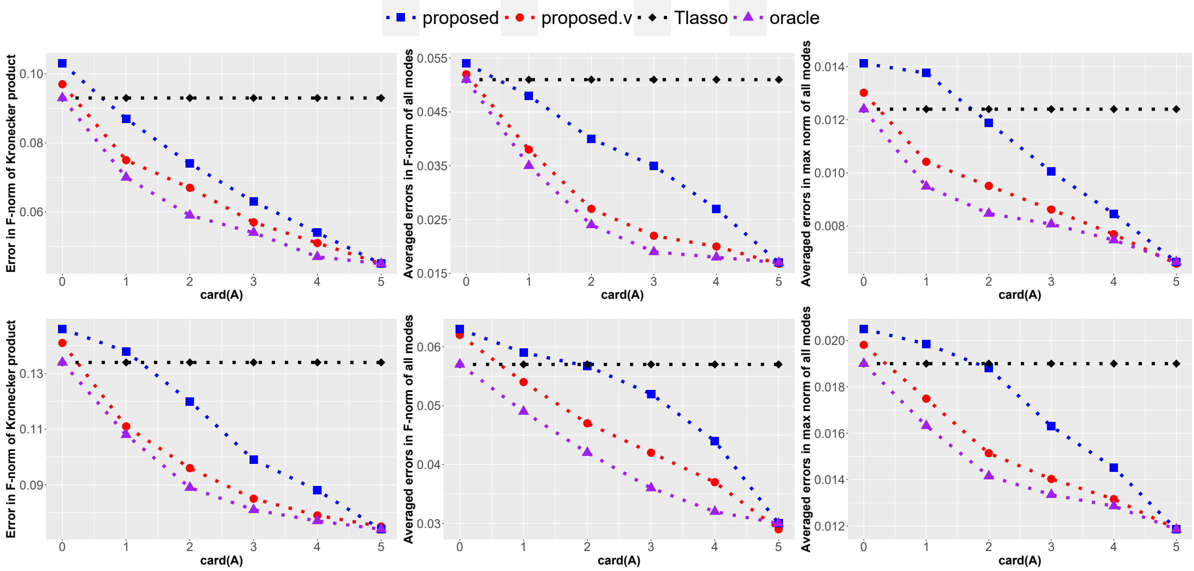

Figure 3: Averaged metrics of estimation errors over 100 replications for Scenario 2 with $M = 3$. The top and bottom rows correspond to the chain and nearest neighbor graph, respectively.

that the functional synchronization of brain systems, that is, the connections of the brain network, can be characterized. There is growing evidence that the brain functional connectivity network is altered in response to ADHD and is important to explore the pathogenesis and diagnosis, while Gaussian graphical model is an important statistical tool to detect this brain functional connectivity (Zhu and Li, 2018).

The analyzed dataset is part of the ADHD-200 repository (Bellec et al., 2017), which is collected from seven sites, containing demographic information, phenotypic data, and rs-fMRI of two groups consisting of typically developing controls (TDC) and ADHD. The names of the seven sites and their sample sizes of TDC and ADHD groups are summarized in Table 2. Only those rs-fMRI scans that pass the quality control are included in our analysis. All rs-fMRIs are pre-processed following the standard Athena pipeline Bellec et al. (2017), and the processed data is publicly available at `https://www.nitrc.org/plugins/mwiki/index.php/neurobureau:AthenaPipeline#Whole_Brain_Data`. Readers may refer to Bellec et al. (2017) for more details about the raw brain imaging data. In the processed data, following the Anatomical Automatic Labeling (AAL) atlas, each brain image is parcellated into 116 regions of interest (ROIs), so that each sample has been reorganized into a $T\times$ 116 matrix, representing BOLD signal fluctuations of 116 ROIs at T time points. Note that the number of time points T varies in different sites. To explore brain functional connectivity, we are only interested in the second mode, that is, the spatial mode corresponding to ROIs. Here we note that although the first mode, which is the temporal mode corresponding to the time series, is not the target of the analysis, its existence leads

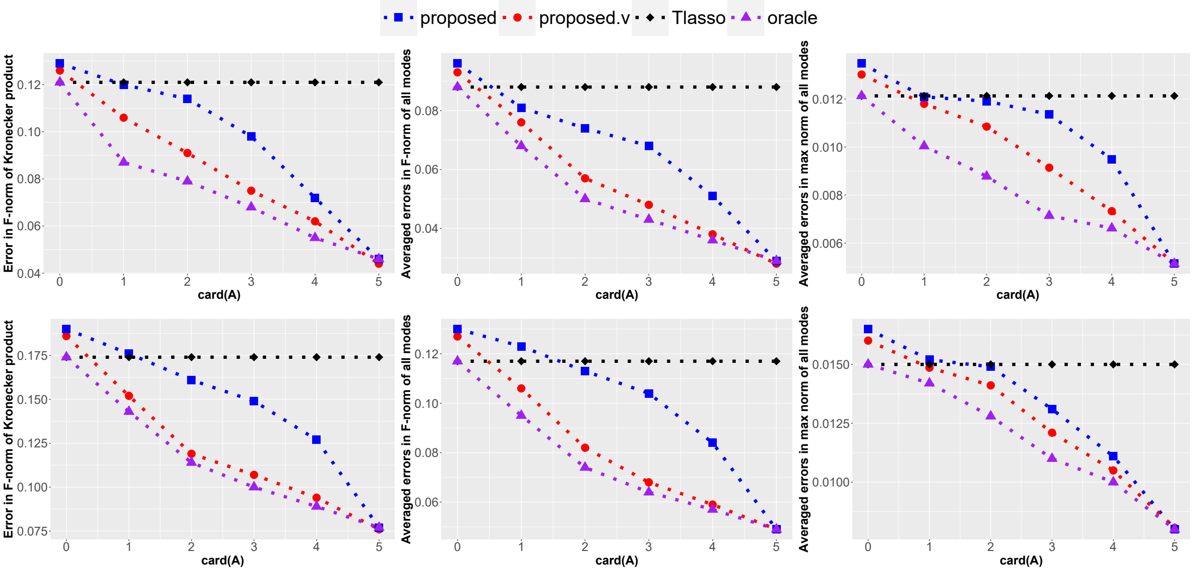

Figure 4: Averaged metrics of estimation errors over 100 replications for Scenario 2 with $M = 2$. The top and bottom rows correspond to the chain and nearest neighbor graph, respectively.

to the necessity of tensor instead of multivariate Gaussian graphical model (Zhu and Li, 2018).

Table 2: Sample sizes and test errors when different data sites become the target domain

			The target domain						
			KKI	NeuroIMAGE	Peking	Pittsburgh	NYU	**OHSU**	WashU
TDC	sample size		58	22	114	66	91	40	37
	absolute error	TransCLIME	2.5381	2.7226	3.3326	3.2099	0.0835	1.8884	3.6907
		Tlasso	0.6444	0.2960	0.4638	0.3793	0.0106	0.4653	0.3762
		proposed	0.5816	0.2913	0.4506	0.3568	0.0144	0.4179	0.3428
		proposed.v	0.5829	0.2910	0.4498	0.3514	0.0112	0.4074	0.2939
	relative error	proposed	0.9026	0.9843	0.9715	0.9406	1.3673	0.8981	0.9113
		proposed.v	0.9046	0.9833	0.9699	0.9265	1.0575	0.8756	0.7813
ADHD	sample size		20	17	90	0	96	30	0
	absolute error	TransCLIME	1.2891	0.5517	3.2591	-	0.1477	2.4494	-
		Tlasso	0.4769	0.3506	0.578	-	0.0119	0.335	-
		proposed	0.4461	0.3415	0.5754	-	0.0166	0.3072	-
		proposed.v	0.4465	0.3423	0.5754	-	0.0165	0.3059	-
	relative error	proposed	0.9354	0.9742	0.9955	-	1.3911	0.9169	-
		proposed.v	0.9362	0.9763	0.9955	-	1.3833	0.9132	-

To compare competitors fairly and comprehensively, we rotated one site as the target domain and other sites as auxiliary domains, and TDC and ADHD groups are considered separately. In addition to the proposed methods and Tlasso, we also consider a naive baseline by applying TransCLIME (Li et al., 2022b) after flattening the tensor into a vector.

Note that the underlying true parameters of precision matrices are unavailable, so we use the negative log-likelihood based on five-fold cross-validation as an indicator to evaluate the performance of all competitors when a site is fixed as the target domain. Specifically, samples of the target domain are randomly divided into five parts, one of which is used as the test sample to calculate the covariance matrices of all modes $\{\widehat{\mathbf{\Sigma}}_m^{\text{test}}\}_{m=1}^M$ and the rest is the training sample. The out-of-sample absolute prediction error of an arbitrary estimator $\widehat{\mathbf{\Omega}}_m^o$ for the m-th mode, estimated using the training sample, is defined as

$$\mathrm{PE}(\widehat{\mathbf{\Omega}}_m^o) = -\frac{1}{p_m}\log[\det(\widehat{\mathbf{\Omega}}_m^o)] + \frac{1}{p_m}\operatorname{tr}(\widehat{\mathbf{\Sigma}}_m^{\text{test}}\widehat{\mathbf{\Omega}}_m^o).$$

Note that the negative log-likelihood are widely used to evaluate method effectiveness for unsupervised graph model problems (Li et al., 2022b), especially when the underlying network structure is unknown. For the estimator of proposed method $\widehat{\mathbf{\Omega}}_m$ and its variant $\widehat{\mathbf{\Omega}}_m^v$, their relative prediction errors are defined as $\frac{\mathrm{PE}(\widehat{\mathbf{\Omega}}_m)}{\mathrm{PE}(\widehat{\mathbf{\Omega}}_m^{(0)})}$ and $\frac{\mathrm{PE}(\widehat{\mathbf{\Omega}}_m^v)}{\mathrm{PE}(\widehat{\mathbf{\Omega}}_m^{(0)})}$, respectively, where $\widehat{\mathbf{\Omega}}_m^{(0)}$ is Tlasso estimator. Here we only consider the $m = 2$-th mode corresponding to ROIs of interest. Average errors of five-fold cross-validation are summarized in Table 2. Here, considering the possible sensitivity of the TLasso to hyperparameters, we vary hyperparameters and report its minimum absolute prediction error under all hyperparameters (that is, the minimum test error) for a more convincing comparison. It is clear that the two proposed methods outperform Tlasso under almost all sites as target domains.

As a byproduct of the above procedure, we are also able to reasonably select OHSU, the target site with the lowest relative prediction error in both TDC and ADHD groups, as the target domain to further demonstrate the performance of the proposed transfer learning-based method by conducting more in-depth biomedical exploration. The detected brain networks of TDC and ADHD groups using the proposed method are provided in Figure A1 of Appendix, and it is clear that the two groups are substantially different. To scrutinize their differences, we plot the differential networks between TDC and ADHD groups in Figure 5, with ROIs labeled as the SRI24 code. A cross-reference between the SRI24 code and full names of ROIs can be found in Table A4 of Appendix. The top 10% important hub nodes and their degrees in differential networks are placed in Table A3 and highlighted in Figure 5, many of which have been widely recognized as relevant to ADHD.

It is evident that the superior frontal gyrus, labeled as 25, has more connections in the TDC group. In fact, it has been found that in the ADHD group, reduced gray matter volumes occurred in this region, and there was a decrease in functional connections between the superior frontal gyrus and other brain regions comparing with the TDC group (Zhao et al., 2020). The functional connectivity mechanism of the inferior occipital gyrus, labeled as 53 and 54, has been recognized significantly different between TDC and ADHD groups, and inattention improvement is related to increased intrinsic brain activity in this region (Zhang et al., 2020). It has been reported that disturbed microstructure of the supramarginal gyrus, labeled as 64, in children with ADHD (Griffiths et al., 2021). In the detected brain network, the cerebellum inferior, labeled as 108, has more connections in the ADHD group. Actually, the cerebellum has been recognized as an important structure in ADHD pathophysiology, and its abnormalities have been reported in patients with ADHD (Stoodley, 2016). In addition, the decreased cerebellar activation in ADHD has been revealed in many cognitive tasks (Valera et al., 2010). Moreover, it has been reported that patients with ADHD have

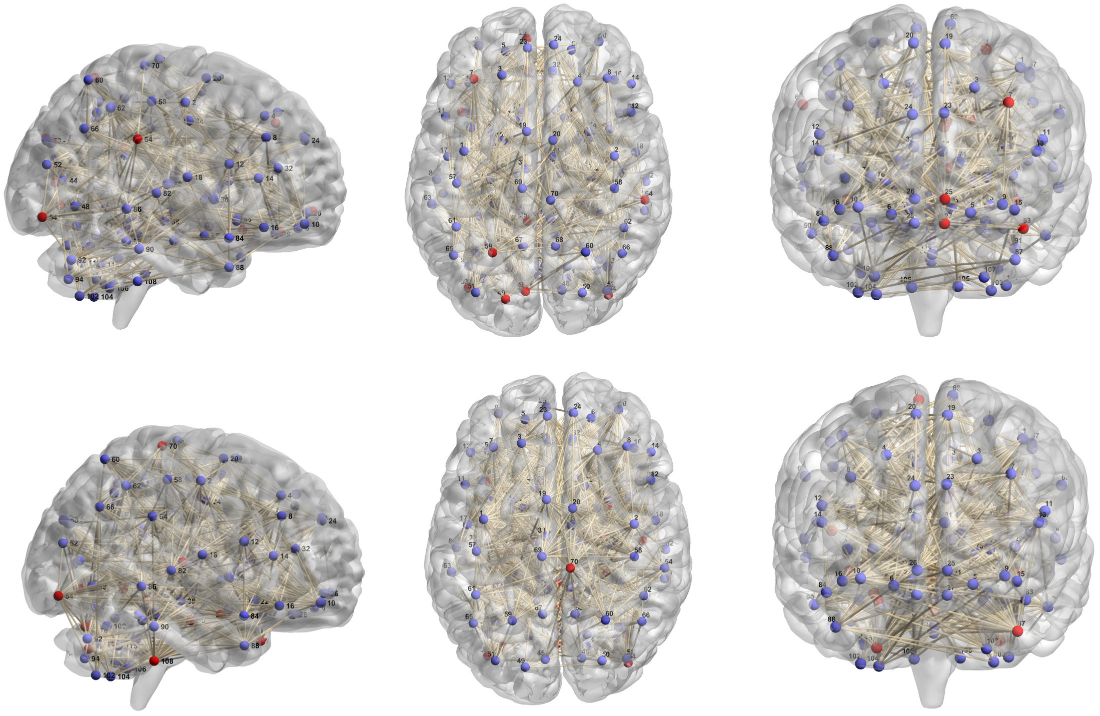

Figure 5: The differential networks of brain functional connectivity between ADHD and TDC groups. The top rows shows the the edges in the TDC group but not the ADHD group, whereas the bottom rows shows the edges in the ADHD group but not the TDC group. In each row, three different views are also provided: sagittal (left), axial (middle), and coronal (right). Nodes with the top 10% of degrees are marked by red.

a larger probability of activation in the paracentral lobule compared to TDC (Dickstein et al., 2006), and this region plays an important role in brain functional networks by controlling sensory nerves of the contralateral lower limb. In conclusion, the analysis results are basically consistent with the evidence of a large number of neuroscience studies.

6.2 Breast cancer gene interaction study

In this example, we consider the breast cancer gene expression data, which can be downloaded using R package *brca.data* (`https://github.com/averissimo/brca.data/releases/download/1.0/brca.data_1.0.tar.gz`). The breast cancer samples can be divided into 6 domains based on race (Asian, Black, and White) and age (> 60 and $\leqslant 60$), and the sample size for each domain is summarized in Table 3. We aim to conduct the tensor GGMs for gene interaction data. Specifically, we focus on interactions formed by two gene pathways, hsa05224 and hsa04310, which contain 147 and 119 genes respectively and have been reported to be related to breast cancer. In other words, each sample is re-organized as 147×119-dimensional tensor.

Similar to ADHD real data analysis, we rotated one domain as the target domain and other domains as auxiliary domains, and use the negative log-likelihood based on five-fold cross-validation as an indicator to evaluate the performance of all competitors when a site is fixed as the target domain. Average errors of five-fold cross-validation are summarized in Table 3. Clearly, the two proposed transfer learning methods outperform Tlasso under all cases.

Table 3: Summary of test errors for each domain as the target domain and their sample sizes in breast cancer gene interaction study.

		The target domain: race(age)					
		ASIAN(> 60)	ASIAN($\leqslant 60$)	BLACK(> 60)	BLACK($\leqslant 60$)	WHITE(> 60)	WHITE($\leqslant 60$)
sample size		10	37	45	71	293	314
absolute error	Tlasso	0.680396	0.601356	0.992349	0.964175	0.773465	0.713554
	proposed	0.667352	0.578745	0.922713	0.920674	0.744267	0.691949
	proposed.v	0.645611	0.574723	0.922197	0.917455	0.742158	0.68931
relative error	proposed	0.980829	0.962399	0.929827	0.954882	0.96225	0.969722
	proposed.v	0.948875	0.955711	0.929307	0.951544	0.959524	0.966024

In order to explore the biological significance of the detected gene network, some hub genes are observed. For example, the gene LEF-1 in the hsa05224 pathway is an important hub node, and there is biological evidence to suggest a pivotal role of LEF-1 in the regulation of proliferation in breast cancer cells. Moreover, it has been reported that the hub gene Sp1 may participate in the invasion and metastasis of breast cancer and is one of the valuable markers indicating poor prognosis of breast cancer.

7. Discussion

This paper proposes a transfer learning method for tensor GGMs leveraging the separability of its covariance. For each mode, a two-step algorithm is performed to improve the estimation in the target domain by making full use of the information from auxiliary domains, in which we design data-adaptive weights on auxiliary domains that can detect informative auxiliary domains and free from the interference of non-informative auxiliary domains. Theoretically, it has been shown that the estimation error of the proposed transfer learning method can be improved with the increasing sample size from informative auxiliary domains. The condition required for the recovery of graph structures has been relaxed in terms of variable selection. Numerical simulations have been performed to verify the statistical theory and to demonstrate the dominant advantages of the proposed method. The conclusions of real data analysis are also consistent with the existing biological knowledge.

This work has some potential extensions. The development of semi-parametric tensor graphical models is an important refinement to address the non-Gaussian property frequently found in biomedical tensor data. Moreover, it is also worthwhile to explore the tensor-valued differential network model to perform inferential analysis on different edges between two networks, which can replace the current descriptive contrastive patterns between two groups in the ADHD brain network analysis.

Acknowledgment

We thank the action editor and two anonymous referees, whose constructive comments and suggestions have led to significant improvements of the article. This research is supported in part by HK RGC grants GRF-11311022, GRF-14306523, GRF-14303424, CUHK Startup Grant 4937091, CUHK Direct Grant 4053588, Shanghai Rising-Star Program 24YF2721900, and SJTU Startup Grant WH220407121.

Appendix

A1. Technical proofs

We use c to denote a universal constant whose value may vary from place to place.

Proof of Lemma 1.

Note that

$$\begin{aligned}
\frac{\partial \mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Omega}_m)}{\partial \boldsymbol{\Delta}_m} &= \boldsymbol{\Delta}_m - \boldsymbol{\Omega}_m \boldsymbol{\Sigma}_m^{\mathcal{A}} + \boldsymbol{I}_{p_m}, \\
\frac{\partial \mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Delta}_m)}{\partial \boldsymbol{\Omega}_m} &= \boldsymbol{\Sigma}_m^{\mathcal{A}} \boldsymbol{\Omega}_m - (\boldsymbol{\Delta}_m + \boldsymbol{I}_{p_m})^\top, \\
\frac{\partial^2 \mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Omega}_m)}{\partial \mathrm{vec}^2(\boldsymbol{\Delta_m})} &= \boldsymbol{I}_{p_m} \otimes \boldsymbol{I}_{p_m}, \\
\frac{\partial^2 \mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Delta}_m)}{\partial \mathrm{vec}^2(\boldsymbol{\Omega_m})} &= \boldsymbol{I}_{p_m} \otimes \boldsymbol{\Sigma}_m^{\mathcal{A}}.
\end{aligned}$$

By the definition of $\boldsymbol{\Delta}_m^*$, $\frac{\partial \mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)*}\}_{k=1}^K, \boldsymbol{\Omega}_m^*)}{\partial \boldsymbol{\Delta}_m}|_{\boldsymbol{\Delta}_m = \boldsymbol{\Delta}_m^*} = \boldsymbol{0}$, $\frac{\partial \mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)*}\}_{k=1}^K, \boldsymbol{\Delta}_m^*)}{\partial \boldsymbol{\Omega}_m}|_{\boldsymbol{\Omega}_m = \boldsymbol{\Omega}_m^*} = \boldsymbol{0}$, and the Hessian matrices $\frac{\partial^2 \mathcal{L}_\Delta(\boldsymbol{\Delta}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Omega}_m)}{\partial \mathrm{vec}^2(\boldsymbol{\Delta_m})}$ and $\frac{\partial^2 \mathcal{L}_\Omega(\boldsymbol{\Omega}_m; \{\boldsymbol{\Sigma}_m^{(k)}\}_{k=1}^K, \boldsymbol{\Delta}_m)}{\partial \mathrm{vec}^2(\boldsymbol{\Omega_m})}$ are positive definite for any fixed positive definite $\boldsymbol{\Sigma}_m^{\mathcal{A}}$. Therefore, Lemma 1 holds.

Proof of Theorem 1.

Considering that $\{\lambda_{1m}\}_{m=1}^M$ for all modes are of the same order, we omit the subscript m and denote λ_{1m} by λ_1 for simplicity. Define $\widehat{\boldsymbol{B}}_m = \widehat{\boldsymbol{\Omega}}_m^{(0)} \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{I}_{p_m}$, and recall that

$$\begin{aligned}
\mathcal{Q}_1(\boldsymbol{\Delta}_m) &= \frac{1}{2} \operatorname{tr}\{\boldsymbol{\Delta}_m^\top \boldsymbol{\Delta}_m\} - \operatorname{tr}\left\{\widehat{\boldsymbol{B}}_m^\top \boldsymbol{\Delta}_m\right\} + \lambda_1 \|\boldsymbol{\Delta}_m\|_1 \\
&= \frac{1}{2} \sum_{1 \leqslant i,j \leqslant p_m} [\boldsymbol{\Delta}_m]_{(i,j)}^2 - \sum_{1 \leqslant i,j \leqslant p_m} [\widehat{\boldsymbol{B}}_m]_{(i,j)} [\boldsymbol{\Delta}_m]_{(i,j)} + \lambda_1 \sum_{1 \leqslant i,j \leqslant p_m} |[\boldsymbol{\Delta}_m]_{(i,j)}|,
\end{aligned}$$

where $[\boldsymbol{\Delta}_m]_{(i,j)}$ is the (i,j) entry of the matrix $\boldsymbol{\Delta}_m$. It can be separated into p_m^2 independent single-lasso optimizations, that is, for any i and j,

$$[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} = \arg \min_{[\boldsymbol{\Delta}_m]_{(i,j)}} \left\{ \frac{1}{2} ([\boldsymbol{\Delta}_m]_{(i,j)} - [\widehat{\boldsymbol{B}}_m]_{(i,j)})^2 + \lambda_1 |[\boldsymbol{\Delta}_m]_{(i,j)}| \right\}.$$

Here, we note that $\boldsymbol{\Delta}_m$ is not necessarily symmetric by its definition. Then it can be obtained that

$$[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} = \operatorname{sign}([\widehat{\boldsymbol{B}}_m]_{(i,j)}) \max(0, |[\widehat{\boldsymbol{B}}_m]_{(i,j)}| - \lambda_1). \tag{A.1}$$

We first consider $\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{\max}$, which needs to establish the bound of $\|\boldsymbol{\Delta}_m^* - \widehat{\boldsymbol{B}}_m\|_{\max}$.

$$\begin{aligned}
\|\boldsymbol{\Delta}_m^* - \widehat{\boldsymbol{B}}_m\|_{\max} &= \|\boldsymbol{\Delta}_m^* - (\widehat{\boldsymbol{\Omega}}_m^{(0)}\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{I}_{p_m})\|_{\max} \\
&\leqslant \|\boldsymbol{\Delta}_m^* - (\widehat{\boldsymbol{\Omega}}_m^{(0)}\boldsymbol{\Sigma}_m^{\mathcal{A}*} - \boldsymbol{I}_{p_m})\|_{\max} + \|\widehat{\boldsymbol{\Omega}}_m^{(0)}(\boldsymbol{\Sigma}_m^{\mathcal{A}*} - \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}})\|_{\max} \\
&\leqslant \|(\widehat{\boldsymbol{\Omega}}_m^{(0)} - \boldsymbol{\Omega}_m^*)\boldsymbol{\Sigma}_m^{\mathcal{A}*}\|_{\max} + \|\widehat{\boldsymbol{\Omega}}_m^{(0)}(\boldsymbol{\Sigma}_m^{\mathcal{A}*} - \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}})\|_{\max} \\
&:= q_1 + q_2,
\end{aligned}$$

where $\boldsymbol{\Sigma}_m^{\mathcal{A}*} = \sum_{k=1}^K \alpha_k \boldsymbol{\Sigma}_m^{(k)*}$. For q_2, we have

$$\begin{aligned}
q_2 &\leqslant \|\widehat{\boldsymbol{\Omega}}_m^{(0)}\|_{1,\infty}\|\boldsymbol{\Sigma}_m^{\mathcal{A}*} - \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\|_{\max} \\
&\leqslant (\|\boldsymbol{\Omega}_m^*\|_{1,\infty} + \|\widehat{\boldsymbol{\Omega}}_m^{(0)} - \boldsymbol{\Omega}_m^*\|_{1,\infty})\|\boldsymbol{\Sigma}_m^{\mathcal{A}*} - \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\|_{\max} \\
&\leqslant (\|\boldsymbol{\Omega}_m^*\|_{1,\infty} + \|\widehat{\boldsymbol{\Omega}}_m^{(0)} - \boldsymbol{\Omega}_m^*\|_{1,\infty})\|\boldsymbol{\Sigma}_m^{\mathcal{A}*} - \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\|_{\max} \\
&\leqslant (c + \bar{s}\sqrt{\frac{p_m \log p_m}{np}})\sqrt{\frac{p_m \log p_m}{Np}},
\end{aligned}$$

with probability tending to 1. The last inequality is from Condition 1 and Lemma 2. It is clear that $q_2 = O_p\left(\sqrt{\frac{p_m \log p_m}{np}}\right)$.

For q_1, it can be further decomposed as follows.

$$\begin{aligned}
q_1 &= \|(\widehat{\boldsymbol{\Omega}}_m^{(0)} - \boldsymbol{\Omega}_m^*)\boldsymbol{\Sigma}_m^*(\boldsymbol{\Delta}_m^* + \boldsymbol{I}_{p_m})\|_{\max} \\
&\leqslant \|\widehat{\boldsymbol{\Omega}}_m^{(0)}\boldsymbol{\Sigma}_m^* - \boldsymbol{I}_{p_m}\|_{\max}\|\boldsymbol{\Delta}_m^* + \boldsymbol{I}_{p_m}\|_{1,\infty} \\
&\leqslant \left[\|\widehat{\boldsymbol{\Omega}}_m^{(0)}\widehat{\boldsymbol{\Sigma}}_m - \boldsymbol{I}_{p_m}\|_{\max} + \|\boldsymbol{\Omega}_m^*(\boldsymbol{\Sigma}_m^* - \widehat{\boldsymbol{\Sigma}}_m)\|_{\max} + \|(\widehat{\boldsymbol{\Omega}}_m^{(0)} - \boldsymbol{\Omega}_m^*)(\boldsymbol{\Sigma}_m^* - \widehat{\boldsymbol{\Sigma}}_m)\|_{\max}\right]\|\boldsymbol{\Delta}_m^* + \boldsymbol{I}_{p_m}\|_{1,\infty} \\
&\leqslant c\sqrt{\frac{p_m \log p_m}{np}}(1+h).
\end{aligned}$$

with probability tending to 1. The last inequality is from Condition 1 and Lemma 2. Therefore, $\|\boldsymbol{\Delta}_m^* - \widehat{\boldsymbol{B}}_m\|_{\max} \leqslant c(1+h)\sqrt{\frac{p_m \log p_m}{np}} := \delta_h' \leqslant \lambda_1$. Note that $\|\widehat{\boldsymbol{\Delta}}_m - \widehat{\boldsymbol{B}}_m\|_{\max} \leqslant \lambda_1$ by (A.1), then,

$$\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{\max} \leqslant \|\widehat{\boldsymbol{\Delta}}_m - \widehat{\boldsymbol{B}}_m\|_{\max} + \|\widehat{\boldsymbol{B}}_m - \boldsymbol{\Delta}_m^*\|_{\max} \leqslant 2\lambda_1. \tag{A.2}$$

Then we consider $\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{1,\infty}$.

(i) If $\|\boldsymbol{\Delta}_m^*\|_{\max} \lesssim \delta_h'$, we have $\|\widehat{\boldsymbol{B}}_m\|_{\max} \leqslant \|\boldsymbol{\Delta}_m^*\|_{\max} + \|\widehat{\boldsymbol{B}}_m - \boldsymbol{\Delta}_m^*\|_{\max} \leqslant \lambda_1$. Therefore, $\widehat{\boldsymbol{\Delta}}_m = \boldsymbol{0}$, then $\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{1,\infty} \leqslant \|\boldsymbol{\Delta}_m^*\|_{1,\infty} \leqslant h$.

(ii) If $\|\boldsymbol{\Delta}_m^*\|_{\max} \gg \delta_h'$, we have $|[\widehat{\boldsymbol{B}}_m]_{(i,j)}| \leqslant |[\boldsymbol{\Delta}_m^*]_{(i,j)}| + \|\widehat{\boldsymbol{B}}_m - \boldsymbol{\Delta}_m^*\|_{\max} \leqslant |[\boldsymbol{\Delta}_m^*]_{(i,j)}| + \delta_h'$. For any fixed $1 \leqslant i, j \leqslant p$, if $|[\boldsymbol{\Delta}_m^*]_{(i,j)}| \lesssim \delta_h'$, then $|[\widehat{\boldsymbol{B}}_m]_{(i,j)}| \leqslant \lambda_1$, so $|[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)}| = 0$; if $|[\boldsymbol{\Delta}_m^*]_{(i,j)}| \gg \delta_h'$, then $|[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)}| \leqslant |\widehat{B}_{m(i,j)}| \leqslant c|[\boldsymbol{\Delta}_m^*]_{(i,j)}|$. Therefore, for any fixed $1 \leqslant j \leqslant p$, $\|\widehat{\boldsymbol{\Delta}}_{m(j)}\|_1 \leqslant c\|\boldsymbol{\Delta}_{m(j)}^*\|_1 \leqslant ch$. It follows that $\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{1,\infty} \leqslant \|\boldsymbol{\Delta}_m^*\|_{1,\infty} + \|\widehat{\boldsymbol{\Delta}}_m\|_{1,\infty} \leqslant (c+1)h$.

Combining (i) and (ii),

$$\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{1,\infty} \leqslant (c+1)h. \tag{A.3}$$

Note that

$$\begin{aligned} &\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{2,\infty}^2 \leqslant \|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{1,\infty}\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{\max}, \\ &\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{2,\infty}^2 \leqslant \|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{1,\infty}^2. \end{aligned} \tag{A.4}$$

By (A.2), (A.3), (A.4), and $\lambda_1 = C(1+h)\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$, we have

$$\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{2,\infty}^2 = O_p\left((1+h)h\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h^2\right).$$

It concludes Theorem 1.

Proof of Theorem 2.

Considering that $\{\lambda_{2m}\}_{m=1}^M$ for all modes are of the same order, we omit the subscript m and denote λ_{2m} by λ_2 for simplicity. Recall that

$$\begin{aligned} \mathcal{Q}_2(\boldsymbol{\Omega}_m) &= \frac{1}{2}\operatorname{tr}\{\boldsymbol{\Omega}_m^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} \boldsymbol{\Omega}_m\} - \operatorname{tr}\{(\widehat{\boldsymbol{\Delta}}_m^\top + \boldsymbol{I}_{p_m})\boldsymbol{\Omega}_m\} + \lambda_2\|\boldsymbol{\Omega}_m\|_1 \\ &= \sum_{1\leqslant j\leqslant p_m}\left\{\frac{1}{2}\boldsymbol{\Omega}_{m(j)}^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\boldsymbol{\Omega}_{m(j)} - \boldsymbol{\Omega}_{m(j)}^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) + \lambda_2\|\boldsymbol{\Omega}_{m(j)}\|_1\right\}, \end{aligned}$$

where $\boldsymbol{\Omega}_{m(j)}$ and $\boldsymbol{I}_{p_m(j)}$ are the j-th columns of $\boldsymbol{\Omega}_m$ and $\boldsymbol{I}_{p_m}$, respectively. It can be separated into p_m independent optimizations, that is, for any j,

$$\widehat{\boldsymbol{\Omega}}_{m(j)} = \arg\min_{\boldsymbol{\Omega}_{m(j)}}\left\{\frac{1}{2}\boldsymbol{\Omega}_{m(j)}^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\boldsymbol{\Omega}_{m(j)} - \boldsymbol{\Omega}_{m(j)}^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) + \lambda_2\|\boldsymbol{\Omega}_{m(j)}\|_1\right\}.$$

Note that

$$\begin{aligned} \frac{1}{2}\widehat{\boldsymbol{\Omega}}_{m(j)}^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\widehat{\boldsymbol{\Omega}}_{m(j)} \leqslant \frac{1}{2}(\boldsymbol{\Omega}_{m(j)}^*)^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\boldsymbol{\Omega}_{m(j)}^* &+ (\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}_{m(j)}^*)^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) \\ &- \lambda_2\|\widehat{\boldsymbol{\Omega}}_{m(j)}\|_1 + \lambda_2\|\boldsymbol{\Omega}_{m(j)}^*\|_1, \end{aligned} \tag{A.5}$$

$$\operatorname{pr}\left(\|\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\boldsymbol{\Omega}_{m(j)}^* - (\boldsymbol{I}_{p_m(j)} + \boldsymbol{\Delta}_{m(j)}^*)\|_\infty \leqslant \lambda_2/2\right) \to 1. \tag{A.6}$$

The inequality (A.5) is from the definition of $\widehat{\boldsymbol{\Omega}}_{m(j)}$, and the (A.6) is from Condition 1, the definition of λ_2, and

$$\begin{aligned} \|\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\boldsymbol{\Omega}_{m(j)}^* - (\boldsymbol{I}_{p_m(j)} + \boldsymbol{\Delta}_{m(j)}^*)\|_\infty &= \|\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\boldsymbol{\Omega}_{m(j)}^* - \boldsymbol{\Sigma}_m^{\mathcal{A}*}\boldsymbol{\Omega}_{m(j)}^*\|_\infty = \|(\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{\Sigma}_m^{\mathcal{A}*})\boldsymbol{\Omega}_{m(j)}^*\|_\infty \\ &= \|\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} - \boldsymbol{\Sigma}_m^{\mathcal{A}*}\|_{\max}\|\boldsymbol{\Omega}_{m(j)}^*\|_1 = O_p\left(\sqrt{\frac{p_m\log p_m}{Np}}\right). \end{aligned}$$

Therefore, with probability tending to 1, we have

$$
\begin{aligned}
&\frac{1}{2}\left(\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\right)^\top \widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m \left(\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\right)\\
=& \frac{1}{2}\widehat{\boldsymbol{\Omega}}^\top_{m(j)}\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\widehat{\boldsymbol{\Omega}}_{m(j)} - \frac{1}{2}(\boldsymbol{\Omega}^*_{m(j)})^\top\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\widehat{\boldsymbol{\Omega}}_{m(j)} - \frac{1}{2}\widehat{\boldsymbol{\Omega}}^\top_{m(j)}\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\boldsymbol{\Omega}^*_{m(j)} + \frac{1}{2}(\boldsymbol{\Omega}^*_{m(j)})^\top\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\boldsymbol{\Omega}^*_{m(j)}\\
\leqslant& \left(\boldsymbol{\Omega}^*_{m(j)} - \widehat{\boldsymbol{\Omega}}_{m(j)}\right)^\top \widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\boldsymbol{\Omega}^*_{m(j)} + (\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)})^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) - \lambda_2\|\widehat{\boldsymbol{\Omega}}_{m(j)}\|_1 + \lambda_2\|\boldsymbol{\Omega}^*_{m(j)}\|_1\\
\leqslant& \left|\left(\boldsymbol{\Omega}^*_{m(j)} - \widehat{\boldsymbol{\Omega}}_{m(j)}\right)^\top [\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\boldsymbol{\Omega}^*_{m(j)} - (\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)})]\right| - \lambda_2\|\widehat{\boldsymbol{\Omega}}_{m(j)}\|_1 + \lambda_2\|\boldsymbol{\Omega}^*_{m(j)}\|_1\\
\leqslant& \left|(\boldsymbol{\Omega}^*_{m(j)} - \widehat{\boldsymbol{\Omega}}_{m(j)})^\top[\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m\boldsymbol{\Omega}^*_{m(j)} - (\boldsymbol{\Delta}^*_{m(j)} + \boldsymbol{I}_{p_m(j)})]\right| + \left|(\boldsymbol{\Omega}^*_{m(j)} - \widehat{\boldsymbol{\Omega}}_{m(j)})^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} - \boldsymbol{\Delta}^*_{m(j)})\right|\\
&- \lambda_2\|\widehat{\boldsymbol{\Omega}}_{m(j)}\|_1 + \lambda_2\|\boldsymbol{\Omega}^*_{m(j)}\|_1\\
\leqslant& \frac{\lambda_2}{2}\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_1 + R_{m,j} + \lambda_2\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1 - \lambda_2\|[\widehat{\boldsymbol{\Omega}}_m]_{(S^C_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S^C_{mj},j)}\|_1\\
\leqslant& R_{m,j} + \frac{3\lambda_2}{2}\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1 - \frac{\lambda_2}{2}\|[\widehat{\boldsymbol{\Omega}}_m]_{(S^C_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S^C_{mj},j)}\|_1.
\end{aligned}
$$

where the first inequality is from (A.5), the fourth inequality is from (A.6), $R_{m,j} = |(\boldsymbol{\Omega}^*_{m(j)} - \widehat{\boldsymbol{\Omega}}_{m(j)})^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} - \boldsymbol{\Delta}^*_{m(j)})|$, $S_{mj} = \{i \in [p_m] : [\boldsymbol{\Omega}^*_m]_{(i,j)} \neq 0\}$, $S^C_{mj} = \{i \in [p_m] : [\boldsymbol{\Omega}^*_m]_{(i,j)} = 0\}$, and $[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)}$ is a vector consisting of the elements in $\widehat{\boldsymbol{\Omega}}_{m(j)}$ labeled by S_{mj}.

(i) If $R_{m,j} \leqslant \frac{3\lambda_2}{2}\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1$, then

$$
\begin{aligned}
3\lambda_2\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1 - \frac{\lambda_2}{2}\|[\widehat{\boldsymbol{\Omega}}_m]_{(S^C_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S^C_{mj},j)}\|_1&\\
\geqslant \frac{1}{2}\left(\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\right)^\top \widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m \left(\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\right) \geqslant 0,& \qquad (A.7)\\
\|[\widehat{\boldsymbol{\Omega}}_m]_{(S^C_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S^C_{mj},j)}\|_1 \leqslant 6\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1.& \qquad (A.8)
\end{aligned}
$$

By (A.8) and the restricted strong convexity property of $\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m$ (Raskutti et al., 2010), for a positive constant ϕ_0,

$$
\left(\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\right)^\top \widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m \left(\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\right) \geqslant \phi_0\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_2^2. \qquad (A.9)
$$

Then by (A.7) and (A.9),

$$
\begin{aligned}
\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_2^2 &\leqslant 6\phi_0^{-1}\lambda_2\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1\\
&\leqslant 6\phi_0^{-1}\sqrt{s_{mj}}\lambda_2\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_2\\
&\leqslant 6\phi_0^{-1}\sqrt{s_{mj}}\lambda_2\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_2.
\end{aligned}
$$

It follows that

$$
\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_2 \leqslant 6\phi_0^{-1}\sqrt{s_{mj}}\lambda_2. \qquad (A.10)
$$

(ii) If

$$
\frac{3\lambda_2}{2}\|[\widehat{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - [\boldsymbol{\Omega}^*_m]_{(S_{mj},j)}\|_1 \leqslant R_{m,j}, \qquad (A.11)
$$

then

$$2R_{m,j} - \frac{\lambda_2}{2}\|[\widehat{\mathbf{\Omega}}_m]_{(S_{mj}^C,j)} - [\mathbf{\Omega}_m^*]_{(S_{mj}^C,j)}\|_1 \geqslant \frac{1}{2}\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right)^\top \widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right) \geqslant 0.$$

It follows that

$$\lambda_2\|[\widehat{\mathbf{\Omega}}_m]_{(S_{mj}^C,j)} - [\mathbf{\Omega}_m^*]_{(S_{mj}^C,j)}\|_1 \leqslant 4R_{m,j}, \tag{A.12}$$

$$\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right)^\top \widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right) \leqslant 4R_{m,j}, \tag{A.13}$$

$$\lambda_2\|\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\|_1 \leqslant \frac{14}{3}R_{m,j}, \tag{A.14}$$

where (A.14) is from (A.11) and (A.12). Therefore,

$$\begin{aligned}
&\|\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\|_2^2 \leqslant c\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right)^\top \mathbf{\Sigma}_m^{\mathcal{A}*}\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right)\\
&\leqslant c\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right)^\top \widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}\left(\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\right) + c\|\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}} - \mathbf{\Sigma}_m^{\mathcal{A}*}\|_{\max}\|\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\|_1^2\\
&\leqslant c_1R_{m,j} + c_2\frac{\|\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}} - \mathbf{\Sigma}_m^{\mathcal{A}*}\|_{\max}}{\lambda_2^2}R_{m,j}^2,
\end{aligned} \tag{A.15}$$

where the last inequality is from (A.13) and (A.14). Applying Cauchy-Schwartz inequality to $R_{m,j}$, we have $R_{m,j} \leqslant \|\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\|_2\|\widehat{\mathbf{\Delta}}_{m(j)} - \mathbf{\Delta}_{m(j)}^*\|_2$. Then, combining (A.15) and the fact that $c_2\delta_h\|\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}} - \mathbf{\Sigma}_m^{\mathcal{A}*}\|_{\max} \leqslant \lambda_2^2$ when $\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} \lesssim 1$, we have

$$\|\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\|_2 \leqslant c\|\widehat{\mathbf{\Delta}}_{m(j)} - \mathbf{\Delta}_{m(j)}^*\|_2. \tag{A.16}$$

Combining (A.10) and (A.16), it can be obtained that

$$\|\widehat{\mathbf{\Omega}}_{m(j)} - \mathbf{\Omega}_{m(j)}^*\|_2^2 \leqslant c\left[s_{mj}\lambda_2^2 + \|\widehat{\mathbf{\Delta}}_{m(j)} - \mathbf{\Delta}_{m(j)}^*\|_2^2\right] = c(s_{mj}\lambda_2^2 + \delta_h).$$

with probability tending to 1. It concludes Theorem 2.

Proof of Theorem 3.

The following two lemmas are useful for the proof of theorem 3, and their proofs are placed after this section.

Lemma A3 *If Conditons 1 and 4 hold, and* $\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} \lesssim 1$, *then for each* $m \in [M]$,
(i)

$$\max_{j\in[p_m]}\|([\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - ([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{\max} = O_p\left(\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right),$$

$$\max_{j\in[p_m]}\|([\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - ([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty} = O_p\left(\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right).$$

(ii)

$$\max_{j\in[p_m],i\in S_{mj}^C} \|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_\infty = O_p\left(\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right),$$

$$\max_{j\in[p_m],i\in S_{mj}^C} \|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_1 = O_p\left(\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right).$$

(iii) $\mathrm{pr}\left(\max_{j\in[p_m],i\in S_{mj}^C} \|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}\|_1 \leqslant 1 - C_3/2\right) \to 1.$

Lemma A4 *If the conditions of Theorem 1 hold, and h is bounded, then* $\|\widehat{\boldsymbol{\Delta}}_m - \boldsymbol{\Delta}_m^*\|_{\max} = O_p\left(\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h_0\right)$*, where* $h_0 = \max_{m\in[M],k\in[K]}\|\boldsymbol{\Delta}_m^{(k)*}\|_{\max}$.

Now we begin the proof of Theorem 3. First, we verify that $\widehat{S}_m \subseteq S_m$. Recall that Step 2(b) can be separated into p_m independent optimizations, that is, for any j,

$$\widehat{\boldsymbol{\Omega}}_{m(j)} = \arg\min_{\boldsymbol{\Omega}_{m(j)}} \mathcal{Q}_{2j}(\boldsymbol{\Omega}_{m(j)}), \tag{A.17}$$

where $\mathcal{Q}_{2j}(\boldsymbol{\Omega}_{m(j)}) = \frac{1}{2}\boldsymbol{\Omega}_{m(j)}^\top \widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}} \boldsymbol{\Omega}_{m(j)} - \boldsymbol{\Omega}_{m(j)}^\top(\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) + \lambda_2\|\boldsymbol{\Omega}_{m(j)}\|_1$. Consider

$$\widetilde{\boldsymbol{\Omega}}_{m(j)} = \mathop{\arg\min}_{[\boldsymbol{\Omega}_m]_{(S_{mj}^C,j)}=\mathbf{0}} \mathcal{Q}_{2j}(\boldsymbol{\Omega}_{m(j)}), \tag{A.18}$$

where $[\boldsymbol{\Omega}_m]_{(S_{mj}^C,j)}$ is a vector consisting of the elements in $\widehat{\boldsymbol{\Omega}}_{m(j)}$ labeled by S_{mj}^C. By its directional derivative, we obtain the equality

$$\left[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}\widetilde{\boldsymbol{\Omega}}_{m(j)} - (\widehat{\boldsymbol{\Delta}}_{m(j)} + \boldsymbol{I}_{p_m(j)}) + \lambda_2\widehat{\boldsymbol{Z}}_{m(j)}\right]_{S_{mj}} = 0, \tag{A.19}$$

where

$$[\widehat{\boldsymbol{Z}}_m]_{(i,j)}\begin{cases} = 0, & i \in S_{mj}^C,\\ = \mathrm{sign}\left([\widetilde{\boldsymbol{\Omega}}_m]_{(i,j)}\right), & i \in S_{mj}, [\widetilde{\boldsymbol{\Omega}}_m]_{(i,j)} \neq 0,\\ \in [-1,1], & i \in S_{mj}, [\widetilde{\boldsymbol{\Omega}}_m]_{(i,j)} = 0.\end{cases}$$

Note that $[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj}^C,j)} = \mathbf{0}$, then (A.19) is equivalent to $[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})}[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - ([\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}) + \lambda_2[\widehat{\boldsymbol{Z}}_m]_{(S_{mj},j)} = 0$, and the explicit solution to (A.18) is

$$[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} = ([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}\{[\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)} - \lambda_2[\widehat{\boldsymbol{Z}}_m]_{(S_{mj},j)}\}. \tag{A.20}$$

Now we verify that $\widetilde{\boldsymbol{\Omega}}_{m(j)}$ is also the solution to (A.17). Since the objective function in (A.17) is convex, combining (A.19), it is sufficient to check that, for any $i \in S_{mj}^C$,

$$\left|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - ([\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} + [\boldsymbol{I}_{p_m}]_{(i,j)})\right| \leqslant \lambda_2. \tag{A.21}$$

According to the fact that $[\boldsymbol{I}_{p_m}]_{(S_{mj}^C,j)} = \boldsymbol{0}$, $[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})}[\boldsymbol{\Omega}_m^*]_{(S_{mj},j)} = [\boldsymbol{\Delta}_m^*]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}$, and $[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}[\boldsymbol{\Omega}_m^*]_{(S_{mj},j)} = [\boldsymbol{\Delta}_m^*]_{(i,j)} + [\boldsymbol{I}_{p_m}]_{(i,j)}$, then for $i \in S_{mj}^C$,

$$[\boldsymbol{\Delta}_m^*]_{(i,j)} = [\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}([\boldsymbol{\Delta}_m^*]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}). \tag{A.22}$$

Combining (A.20) and (A.22), then the left part of (A.21) can be decomposed as

$$\begin{aligned}
&\left|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - ([\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} + [\boldsymbol{I}_{p_m}]_{(i,j)})\right| \\
\leqslant &\Big|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}([\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}) \\
&-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}([\boldsymbol{\Delta}_m^*]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}) \\
&-\lambda_2[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}[\widehat{\boldsymbol{Z}}_m]_{(S_{mj},j)} - [\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} + [\boldsymbol{\Delta}_m^*]_{(i,j)}\Big| \\
\leqslant &\left|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}([\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} - [\boldsymbol{\Delta}_m^*]_{(S_{mj},j)})\right| \\
&+\left|\left\{[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\right\}([\boldsymbol{\Delta}_m^*]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)})\right| \\
&+\lambda_2\left|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}[\widehat{\boldsymbol{Z}}_m]_{(S_{mj},j)}\right| + \left|[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} - [\boldsymbol{\Delta}_m^*]_{(i,j)}\right|.
\end{aligned} \tag{A.23}$$

By the fact that $\|[\widehat{\boldsymbol{Z}}_m]_{(S_{mj},j)}\|_\infty \leqslant 1$, then

$$\begin{aligned}
&\left|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - ([\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} + [\boldsymbol{I}_{p_m}]_{(i,j)})\right| \\
\leqslant &\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}\|_1\|[\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} - [\boldsymbol{\Delta}_m^*]_{(S_{mj},j)}\|_\infty \\
&+\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}(([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_1\|[\boldsymbol{\Delta}_m^*]_{(S_{mj},j)}\|_\infty \\
&+\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_\infty \\
&+\lambda_2\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}\|_1 + \|[\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} - [\boldsymbol{\Delta}_m^*]_{(i,j)}\|_\infty.
\end{aligned} \tag{A.24}$$

By Lemma A3, Lemma A4, and (A.24), we have

$$\begin{aligned}
&\left|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}[\widetilde{\boldsymbol{\Omega}}_m]_{(S_{mj},j)} - ([\widehat{\boldsymbol{\Delta}}_m]_{(i,j)} + [\boldsymbol{I}_{p_m}]_{(i,j)})\right| \\
\leqslant &c(1 - C_3/2)\left(\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h_0\right) + ch_0\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} \\
&+ c\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} + \lambda_2(1 - C_3/2) + c\left(\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h_0\right),
\end{aligned}$$

with probability tending to 1. By Condition 5 (i.e., $h_0 = O(h/\overline{s})$), the assumption that h is bounded, and $\lambda_2 = C\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right)$, we have $c(2 - C_3/2)\left(\sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h_0\right) + c(1 + h_0\overline{s})\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} \leqslant \lambda_2 C_3/2$ for a sufficiently large constant C. Therefore, (A.21) holds with probability tending to 1. It concludes that $\widehat{S}_m \subseteq S_m$.

Next, we verify that $S_m \subseteq \widehat{S}_m$, which requires establishing the bound of $\|\widehat{\boldsymbol{\Omega}}_m - \boldsymbol{\Omega}^*_m\|_{\max}$. By the fact that $\widetilde{\boldsymbol{\Omega}}_{m(j)} = \widehat{\boldsymbol{\Omega}}_{m(j)}$, (A.20), and $[\boldsymbol{\Omega}^*_m]_{(S_{mj},j)} = ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}([\boldsymbol{\Delta}^*_m]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)})$, we have

$$
\begin{aligned}
&\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_\infty \\
=& \Big\|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1}([\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} - [\boldsymbol{\Delta}^*_m]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)} - [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}) \\
&+\{([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\}([\boldsymbol{\Delta}^*_m]_{(S_{mj},j)} + [\boldsymbol{I}_{p_m}]_{(S_{mj},j)}) \\
&-\lambda_2([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1}[\widehat{\boldsymbol{Z}}_m]_{(S_{mj},j)}\Big\|_\infty \\
\leqslant& \|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}\|[\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} - [\boldsymbol{\Delta}^*_m]_{(S_{mj},j)}\|_\infty \\
&+ \|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}\|[\boldsymbol{\Delta}^*_m]_{(S_{mj},j)}\|_\infty \\
&+ \|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{\max} + \lambda_2\|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}.
\end{aligned}
$$

Note that $\|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty} = \|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty} + \|([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}$, then

$$
\begin{aligned}
&\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_\infty \\
\leqslant& \|([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}(\lambda_2 + \|[\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} - [\boldsymbol{\Delta}^*_m]_{(S_{mj},j)}\|_\infty) \\
&+ \|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}(\lambda_2 + \|[\widehat{\boldsymbol{\Delta}}_m]_{(S_{mj},j)} - [\boldsymbol{\Delta}^*_m]_{(S_{mj},j)}\|_\infty + \|[\boldsymbol{\Delta}^*_m]_{(S_{mj},j)}\|_\infty) \\
&+ \|([\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})})^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\|_{\max}.
\end{aligned}
$$

By Lemma A3, Lemma A4, and Condition 4, we have

$$
\begin{aligned}
&\|\widehat{\boldsymbol{\Omega}}_{m(j)} - \boldsymbol{\Omega}^*_{m(j)}\|_\infty \\
\leqslant& c\left(\lambda_2 + \sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h_0\right) + c\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\left(\lambda_2 + \sqrt{\frac{\overline{p}\log\overline{p}}{np}} \wedge h_0 + h_0\right) + c\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} \\
\leqslant& c'\lambda_2
\end{aligned}
$$

for a sufficiently large constant c', with probability tending to 1. The last line is from the fact that $\lambda_2 = C\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\right)$. Note that $n \leqslant N$, therefore, for any $m \in [M]$,

$$
\|\widehat{\boldsymbol{\Omega}}_m - \boldsymbol{\Omega}^*_m\|_{\max} = O_p\left(\sqrt{\frac{\delta_h}{\overline{s}}} + \sqrt{\frac{\overline{p}\log\overline{p}}{(N+n)p}}\right). \tag{A.25}
$$

Combining (A.25) and the minimal signal condition of $\boldsymbol{\Omega}^*_m$, we have $S_m \subseteq \widehat{S}_m$. It concludes Theorem 3.

Proof of Lemma A3.

First, we consider (i) of Lemma A3. For any $m \in [M]$ and $j \in [p_m]$, define

$$
\begin{aligned}
\boldsymbol{D} =& \{[\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})} + [\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})} - [\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})}\}^{-1} - ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1} \\
&+ ([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}\{[\widehat{\boldsymbol{\Sigma}}^{\mathcal{A}}_m]_{(S_{mj},S_{mj})} - [\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})}\}([\boldsymbol{\Sigma}^{\mathcal{A}*}_m]_{(S_{mj},S_{mj})})^{-1}.
\end{aligned}
$$

Following the proof of Lemma 5 in Ravikumar et al. (2011), it can be obtained that

$$\begin{aligned}&\|\boldsymbol{D}\|_{\max}\\ \leqslant\ &\frac{3}{2}\|[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})}\|_{1,\infty}^3\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})}\|_{\max}\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})}\|_{1,\infty}.\end{aligned}\tag{A.26}$$

Therefore,

$$\begin{aligned}&\|([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{\max}\\ \leqslant\ &\|\boldsymbol{D}\|_{\max}+\|([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\{[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})}\}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{\max}\\ \leqslant\ &\|\boldsymbol{D}\|_{\max}+\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})}\|_{\max}\|([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}^2\\ \leqslant\ &c_{D_1}\overline{s}\frac{\overline{p}\log\overline{p}}{Np}+c_{D_2}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\\ \leqslant\ &c_D\sqrt{\frac{\overline{p}\log\overline{p}}{Np}},\end{aligned}\tag{A.27}$$

for some constants c_{D_1} and c_{D_2}, with probability tending to 1. The penultimate line is from (A.26), Condition 4, and Lemma 2. The last line is from the assumption that $\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}\lesssim 1$. It follows that

$$\|([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}\leqslant c_D\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}},\tag{A.28}$$

with probability tending to 1. It concludes (i) of Lemma A3 by (A.27) and (A.28).

For (ii) of Lemma A3, note that

$$\begin{aligned}&[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\\ =\ &([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})})([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\\ &+[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}\{([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\}\\ &+([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})})\{([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\},\end{aligned}\tag{A.29}$$

for any $m\in[M]$ and $j\in[p_m]$. It follows that

$$\begin{aligned}&\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{\infty}\\ \leqslant\ &\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}\|_{\infty}\|([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{1,\infty}\\ &+(\|[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}\|_1+\|[\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}-[\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}\|_1)\|([\widehat{\boldsymbol{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}-([\boldsymbol{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_{\max}\\ \leqslant\ &c_{\mathcal{A}_1}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}},\end{aligned}\tag{A.30}$$

for some constant $c_{\mathcal{A}_1}$, with probability tending to 1. The last line is from Condition 4, Lemma 2, (i) of Lemma A3, and the assumption that $\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} \lesssim 1$. It follows that

$$\|[\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_1 \leqslant c_{\mathcal{A}_1}\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}}, \quad \text{(A.31)}$$

with probability tending to 1. It concludes (ii) of Lemma A3 by (A.30) and (A.31).

As for (iii) of Lemma A3, for any $m \in [M]$ and $j \in [p_m]$, we have

$$\begin{aligned}
&\|[\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1}\|_1 \\
\leqslant\ &\|[\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(i,S_{mj})}([\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}}]_{(S_{mj},S_{mj})})^{-1} - [\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_1 \\
&+ \|[\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(i,S_{mj})}([\mathbf{\Sigma}_m^{\mathcal{A}*}]_{(S_{mj},S_{mj})})^{-1}\|_1 \qquad \text{(A.32)} \\
\leqslant\ &c_{\mathcal{A}_1}\overline{s}\sqrt{\frac{\overline{p}\log\overline{p}}{Np}} + 1 - C_3 \leqslant 1 - C_3/2,
\end{aligned}$$

with probability tending to 1. The last line is from Condition 4 and (ii) of Lemma A3. It concludes Lemma A3.

Proof of Lemma A4.

By (A.2) in the proof of Theorem 1, with probability tending to 1,

$$\|\widehat{\mathbf{\Delta}}_m - \mathbf{\Delta}_m^*\|_{\max} \leqslant c_1'\sqrt{\frac{p_m\log p_m}{np}},$$

for a sufficiently large constant c_1'. Similar to the discussion of (i) and (ii) in the proof of Theorem 1, it can be obtained that $\|\widehat{\mathbf{\Delta}}_m - \mathbf{\Delta}_m^*\|_{\max} \leqslant c_2'h_0$ with probability tending to 1, for a sufficiently large constant c_2'. It follows that

$$\|\widehat{\mathbf{\Delta}}_m - \mathbf{\Delta}_m^*\|_{\max} \leqslant O_p\left(\sqrt{\frac{p_m\log p_m}{np}} \wedge h_0\right).$$

Proof of Theorem 4.

Recall that

$$\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}} = \sum_{k=1}^{K}\alpha_k\widehat{\mathbf{\Sigma}}_m^{(k)}, \text{ with } \alpha_k = \frac{n_k/\widehat{h}_k}{\sum_{k=1}^{K}(n_k/\widehat{h}_k)},$$

where $\widehat{h}_k = \max_{m\in[M]}\|\widehat{\mathbf{\Delta}}_m^{(k)}\|_{1,\infty}$ and $\widehat{\mathbf{\Delta}}_m^{(k)} = \widehat{\mathbf{\Omega}}_m^{(0)}\widehat{\mathbf{\Sigma}}_m^{(k)} - \boldsymbol{I}_{p_m}$. Combing $K = O(1)$ and the proofs of Theorems 2 to 3, it is sufficient to show that

$$\alpha_k \to 0, \text{ if } k \notin \mathcal{A}_h. \quad \text{(A.33)}$$

In fact, for $k \in \mathcal{A}_h$, with probability tending to 1,

$$\begin{aligned}
\widehat{h}_k &= \max_{m\in[M]} \|\widehat{\mathbf{\Omega}}_m^{(0)}\widehat{\mathbf{\Sigma}}_m^{(k)} - \boldsymbol{I}_{p_m}\|_{1,\infty} \\
&\leqslant h + \max_{m\in[M]} \|\widehat{\mathbf{\Omega}}_m^{(0)}\widehat{\mathbf{\Sigma}}_m^{(k)} - \mathbf{\Omega}_m^{*}\mathbf{\Sigma}_m^{(k)*}\|_{1,\infty} \\
&\leqslant h + \max_{m\in[M]} \|(\widehat{\mathbf{\Omega}}_m^{(0)} - \mathbf{\Omega}_m^{*})\widehat{\mathbf{\Sigma}}_m^{(k)}\|_{1,\infty} + \max_{m\in[M]} \|\mathbf{\Omega}_m^{*}(\widehat{\mathbf{\Sigma}}_m^{(k)} - \mathbf{\Sigma}_m^{(k)*})\|_{1,\infty} \\
&\leqslant c\bar{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}},
\end{aligned}$$

for a sufficiently large constant c, where the last inequality is from Lemma 2 and Condition 6. Note that $\widehat{h}_{k'} \gg \bar{s}\sqrt{\frac{\overline{p}\log\overline{p}}{np}}$ for $k' \notin \mathcal{A}_h$, so $\widehat{h}_k^{-1} \gg \widehat{h}_{k'}^{-1}$ for any $k \in \mathcal{A}_h$, $k' \notin \mathcal{A}_h$. Combining $n_1 \asymp \cdots \asymp n_K$, it concludes (A.33).

A2. Additional numerical results

A2.1 Additional simulation results: Tables A1 and A2

Table A1: Averaged TNRs over 100 replications and their standard deviation in parenthesis for Scenario 1.

		$M=3$				$M=2$			
		TNRs of $\widehat{\mathbf{\Omega}}^{(K)}$		Averaged TNRs of all modes		TNRs of $\widehat{\mathbf{\Omega}}^{(K)}$		Averaged TNRs of all modes	
Methods	K	chain	nearest	chain	nearest	chain	nearest	chain	nearest
proposed.v	1	0.939(0.006)	0.932(0.012)	0.746(0.027)	0.735(0.035)	0.973(0.008)	0.997(0.000)	0.851(0.004)	0.949(0.002)
	2	0.952(0.008)	0.937(0.009)	0.763(0.029)	0.744(0.032)	0.985(0.006)	0.998(0.000)	0.879(0.004)	0.951(0.002)
	3	0.961(0.007)	0.944(0.009)	0.778(0.027)	0.760(0.031)	0.986(0.006)	0.998(0.000)	0.889(0.005)	0.952(0.002)
	4	0.968(0.007)	0.950(0.009)	0.801(0.030)	0.777(0.030)	0.986(0.005)	0.999(0.000)	0.892(0.005)	0.956(0.002)
	5	0.977(0.007)	0.955(0.011)	0.811(0.030)	0.796(0.033)	0.986(0.005)	0.999(0.000)	0.899(0.003)	0.956(0.001)
proposed	1	0.939(0.006)	0.932(0.012)	0.746(0.027)	0.735(0.035)	0.973(0.008)	0.997(0.000)	0.851(0.004)	0.949(0.002)
	2	0.953(0.006)	0.938(0.008)	0.764(0.029)	0.742(0.034)	0.984(0.006)	0.998(0.000)	0.877(0.003)	0.950(0.002)
	3	0.960(0.007)	0.945(0.011)	0.780(0.027)	0.763(0.028)	0.984(0.006)	0.999(0.000)	0.886(0.004)	0.952(0.002)
	4	0.969(0.008)	0.949(0.010)	0.798(0.029)	0.776(0.034)	0.985(0.006)	0.999(0.000)	0.891(0.003)	0.955(0.002)
	5	0.976(0.008)	0.953(0.010)	0.809(0.028)	0.793(0.031)	0.986(0.004)	0.999(0.000)	0.898(0.004)	0.956(0.001)
Tlasso	1	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	2	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	3	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	4	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	5	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
oracle	1	0.939(0.006)	0.932(0.012)	0.746(0.027)	0.735(0.035)	0.973(0.008)	0.997(0.000)	0.851(0.004)	0.949(0.002)
	2	0.953(0.006)	0.938(0.008)	0.764(0.029)	0.742(0.034)	0.984(0.006)	0.998(0.000)	0.877(0.003)	0.950(0.002)
	3	0.960(0.007)	0.945(0.011)	0.780(0.027)	0.763(0.028)	0.984(0.006)	0.999(0.000)	0.886(0.004)	0.952(0.002)
	4	0.969(0.008)	0.949(0.010)	0.798(0.029)	0.776(0.034)	0.985(0.006)	0.999(0.000)	0.891(0.003)	0.955(0.002)
	5	0.976(0.008)	0.953(0.010)	0.809(0.028)	0.793(0.031)	0.986(0.004)	0.999(0.000)	0.898(0.004)	0.956(0.001)

* All methods have achieved 100% TPR and hence not shown in the table.

A2.2 Additional results and information for ADHD brain network data: Tables A3 and A4 and Figure A1

Table A2: Averaged TNRs over 100 replications and their standard deviation in parenthesis for Scenario 2.

		$M = 3$				$M = 2$			
		TNRs of $\widehat{\mathbf{\Omega}}^{(K)}$		Averaged TNRs of all modes		TNRs of $\widehat{\mathbf{\Omega}}^{(K)}$		Averaged TNRs of all modes	
Methods	card($\mathcal{A}$)	chain	nearest	chain	nearest	chain	nearest	chain	nearest
proposed.v	0	0.926(0.010)	0.869(0.015)	0.717(0.023)	0.639(0.029)	0.969(0.002)	0.994(0.001)	0.847(0.006)	0.950(0.006)
	1	0.928(0.008)	0.883(0.016)	0.756(0.020)	0.674(0.031)	0.974(0.001)	0.994(0.000)	0.858(0.001)	0.953(0.001)
	2	0.954(0.005)	0.914(0.01)	0.787(0.017)	0.725(0.025)	0.976(0.001)	0.994(0.000)	0.863(0.004)	0.953(0.001)
	3	0.964(0.004)	0.936(0.008)	0.823(0.013)	0.770(0.023)	0.984(0.001)	0.997(0.000)	0.897(0.003)	0.961(0.001)
	4	0.969(0.003)	0.947(0.007)	0.841(0.013)	0.794(0.021)	0.986(0.001)	0.998(0.000)	0.897(0.003)	0.964(0.001)
	5	0.974(0.003)	0.953(0.007)	0.860(0.013)	0.810(0.022)	0.989(0.001)	0.999(0.000)	0.903(0.004)	0.966(0.002)
proposed	0	0.925(0.010)	0.846(0.018)	0.718(0.025)	0.628(0.033)	0.969(0.002)	0.993(0.000)	0.846(0.006)	0.935(0.001)
	1	0.928(0.011)	0.862(0.014)	0.741(0.027)	0.662(0.026)	0.972(0.001)	0.994(0.000)	0.858(0.000)	0.936(0.001)
	2	0.933(0.011)	0.879(0.016)	0.775(0.026)	0.709(0.031)	0.974(0.000)	0.994(0.000)	0.861(0.002)	0.948(0.001)
	3	0.942(0.01)	0.883(0.016)	0.808(0.024)	0.743(0.031)	0.981(0.002)	0.996(0.000)	0.891(0.002)	0.960(0.001)
	4	0.954(0.011)	0.913(0.016)	0.822(0.024)	0.774(0.031)	0.982(0.001)	0.996(0.000)	0.897(0.003)	0.962(0.002)
	5	0.974(0.003)	0.958(0.006)	0.861(0.013)	0.815(0.021)	0.990(0.004)	0.998(0.000)	0.905(0.003)	0.963(0.001)
Tlasso	0	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	1	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	2	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	3	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	4	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	5	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
oracle	0	0.923(0.017)	0.863(0.017)	0.724(0.026)	0.718(0.029)	0.965(0.004)	0.985(0.002)	0.842(0.002)	0.938(0.003)
	1	0.941(0.007)	0.936(0.003)	0.752(0.024)	0.739(0.015)	0.979(0.004)	0.999(0.000)	0.854(0.003)	0.950(0.001)
	2	0.956(0.006)	0.941(0.004)	0.772(0.022)	0.748(0.018)	0.984(0.003)	0.999(0.000)	0.881(0.005)	0.951(0.001)
	3	0.968(0.008)	0.948(0.005)	0.793(0.022)	0.772(0.018)	0.986(0.004)	0.999(0.000)	0.890(0.004)	0.955(0.001)
	4	0.974(0.008)	0.955(0.005)	0.812(0.023)	0.799(0.018)	0.987(0.005)	0.999(0.000)	0.900(0.004)	0.957(0.001)
	5	0.979(0.006)	0.958(0.006)	0.861(0.023)	0.815(0.021)	0.990(0.004)	0.998(0.000)	0.905(0.003)	0.963(0.001)

* All methods have achieved 100% TPR and hence not shown in the table.

Table A3: The top 10% important hub nodes and their degrees in differential networks of brain functional connectivity between ADHD and TDC groups.

	SRI24 code	Full name	Degree
TDC-ADHD	25	Superior frontal gyrus, medial orbital	20
	27	Gyrus rectus	16
	43	Calcarine fissure and surrounding cortex	15
	45	Cuneus	15
	49	Superior occipital gyrus	15
	35	Posterior cingulate gyrus	14
	53	Inferior occipital gyrus	14
	54	Inferior occipital gyrus	14
	59	Superior parietal gyrus	13
	64	Supramarginal gyrus	13
	83	Temporal pole: superior temporal gyrus	13
	7	Middle frontal gyrus	12
ADHD-TDC	87	Temporal pole: middle temporal gyrus	29
	108	Cerebellum Inferior	26
	113	Vermis	25
	54	Inferior occipital gyrus	23
	80	Heschl gyrus	23
	93	Cerebellum Inferior	23
	40	Parahippocampal gyrus	21
	110	Vermis	21
	111	Vermis	21
	42	Amygdala	20
	53	Inferior occipital gyrus	20
	70	Paracentral lobule	20

*TDC-ADHD: the differential network consisting of the edges in the TDC group but not the ADHD group; ADHD-TDC: the differential network consisting of the edges in the ADHD group but not the TDC group.

Table A4: The detailed information of 116 ROIs defined by the AAL atlas.

SRI24 code	Tzourio Mazoyer name*	Full name	SRI24 code	Tzourio Mazoyer name*	Full name
1	Precentral-L	Precentral gyrus	59	Parietal-Sup-L	Superior parietal gyrus
2	Precentral-R	Precentral gyrus	60	Parietal-Sup-R	Superior parietal gyrus
3	Frontal-Sup-L	Superior frontal gyrus, dorsolateral	61	Parietal-Inf-L	Inferior parietal, but supramarginal and angular
4	Frontal-Sup-R	Superior frontal gyrus, dorsolateral	62	Parietal-Inf-R	Inferior parietal, but supramarginal and angular
5	Frontal-Sup-Orb-L	Superior frontal gyrus, orbital part	63	SupraMarginal-L	Supramarginal gyrus
6	Frontal-Sup-Orb-R	Superior frontal gyrus, orbital part	64	SupraMarginal-R	Supramarginal gyrus
7	Frontal-Mid-L	Middle frontal gyrus	65	Angular-L	Angular gyrus
8	Frontal-Mid-R	Middle frontal gyrus	66	Angular-R	Angular gyrus
9	Frontal-Mid-Orb-L	Middle frontal gyrus, orbital part	67	Precuneus-L	Precuneus
10	Frontal-Mid-Orb-R	Middle frontal gyrus, orbital part	68	Precuneus-R	Precuneus
11	Frontal-Inf-Oper-L	Inferior frontal gyrus, opercular part	69	Paracentral-Lobule-L	Paracentral lobule
12	Frontal-Inf-Oper-R	Inferior frontal gyrus, opercular part	70	Paracentral-Lobule-R	Paracentral lobule
13	Frontal-Inf-Tri-L	Inferior frontal gyrus, triangular part	71	Caudate-L	Caudate nucleus
14	Frontal-Inf-Tri-R	Inferior frontal gyrus, triangular part	72	Caudate-R	Caudate nucleus
15	Frontal-Inf-Orb-L	Inferior frontal gyrus, orbital part	73	Putamen-L	Lenticular nucleus, putamen
16	Frontal-Inf-Orb-R	Inferior frontal gyrus, orbital part	74	Putamen-R	Lenticular nucleus, putamen
17	Rolandic-Oper-L	Rolandic operculum	75	Pallidum-L	Lenticular nucleus, pallidum
18	Rolandic-Oper-R	Rolandic operculum	76	Pallidum-R	Lenticular nucleus, pallidum
19	Supp-Motor-Area-L	Supplementary motor area	77	Thalamus-L	Thalamus
20	Supp-Motor-Area-R	Supplementary motor area	78	Thalamus-R	Thalamus
21	Olfactory-L	Olfactory cortex	79	Heschl-L	Heschl gyrus
22	Olfactory-R	Olfactory cortex	80	Heschl-R	Heschl gyrus
23	Frontal-Sup-Medial-L	Superior frontal gyrus, medial	81	Temporal-Sup-L	Superior temporal gyrus
24	Frontal-Sup-Medial-R	Superior frontal gyrus, medial	82	Temporal-Sup-R	Superior temporal gyrus
25	Frontal-Mid-Orb-L	Superior frontal gyrus, medial orbital	83	Temporal-Pole-Sup-L	Temporal pole: superior temporal gyrus
26	Frontal-Mid-Orb-R	Superior frontal gyrus, medial orbital	84	Temporal-Pole-Sup-R	Temporal pole: superior temporal gyrus
27	Rectus-L	Gyrus rectus	85	Temporal-Mid-L	Middle temporal gyrus
28	Rectus-R	Gyrus rectus	86	Temporal-Mid-R	Middle temporal gyrus
29	Insula-L	Insula	87	Temporal-Pole-Mid-L	Temporal pole: middle temporal gyrus
30	Insula-R	Insula	88	Temporal-Pole-Mid-R	Temporal pole: middle temporal gyrus
31	Cingulum-Ant-L	Anterior cingulate and paracingulate gyri	89	Temporal-Inf-L	Inferior temporal gyrus
32	Cingulum-Ant-R	Anterior cingulate and paracingulate gyri	90	Temporal-Inf-R	Inferior temporal gyrus
33	Cingulum-Mid-L	Median cingulate and paracingulate gyri	91	Cerebelum-Crus1-L	Cerebellum-Superior
34	Cingulum-Mid-R	Median cingulate and paracingulate gyri	92	Cerebelum-Crus1-R	Cerebellum-Superior
35	Cingulum-Post-L	Posterior cingulate gyrus	93	Cerebelum-Crus2-L	Cerebellum-Inferior
36	Cingulum-Post-R	Posterior cingulate gyrus	94	Cerebelum-Crus2-R	Cerebellum-Inferior
37	Hippocampus-L	Hippocampus	95	Cerebelum-3-L	Cerebellum-Superior
38	Hippocampus-R	Hippocampus	96	Cerebelum-3-R	Cerebellum-Superior
39	ParaHippocampal-L	Parahippocampal gyrus	97	Cerebelum-4-5-L	Cerebellum-Superior
40	ParaHippocampal-R	Parahippocampal gyrus	98	Cerebelum-4-5-R	Cerebellum-Superior
41	Amygdala-L	Amygdala	99	Cerebelum-6-L	Cerebellum-Superior
42	Amygdala-R	Amygdala	100	Cerebelum-6-R	Cerebellum-Superior
43	Calcarine-L	Calcarine fissure and surrounding cortex	101	Cerebelum-7b-L	Cerebellum-Inferior
44	Calcarine-R	Calcarine fissure and surrounding cortex	102	Cerebelum-7b-R	Cerebellum-Inferior
45	Cuneus-L	Cuneus	103	Cerebelum-8-L	Cerebellum-Inferior
46	Cuneus-R	Cuneus	104	Cerebelum-8-R	Cerebellum-Inferior
47	Lingual-L	Lingual gyrus	105	Cerebelum-9-L	Cerebellum-Inferior
48	Lingual-R	Lingual gyrus	106	Cerebelum-9-R	Cerebellum-Inferior
49	Occipital-Sup-L	Superior occipital gyrus	107	Cerebelum-10-L	Cerebellum-Inferior
50	Occipital-Sup-R	Superior occipital gyrus	108	Cerebelum-10-R	Cerebellum-Inferior
51	Occipital-Mid-L	Middle occipital gyrus	109	Vermis-1-2	Vermis
52	Occipital-Mid-R	Middle occipital gyrus	110	Vermis-3	Vermis
53	Occipital-Inf-L	Inferior occipital gyrus	111	Vermis-4-5	Vermis
54	Occipital-Inf-R	Inferior occipital gyrus	112	Vermis-6	Vermis
55	Fusiform-L	Fusiform gyrus	113	Vermis-7	Vermis
56	Fusiform-R	Fusiform gyrus	114	Vermis-8	Vermis
57	Postcentral-L	Postcentral gyrus	115	Vermis-9	Vermis
58	Postcentral-R	Postcentral gyrus	116	Vermis-10	Vermis

*L: left hemisphere, R: right hemisphere.

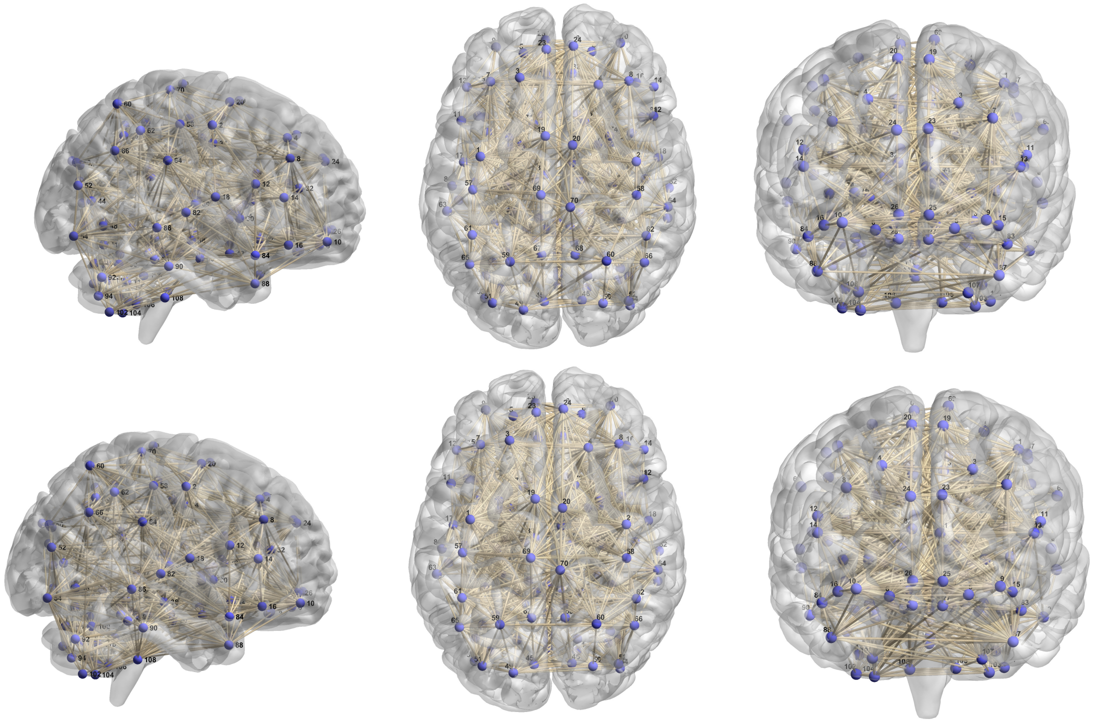

Figure A1: The networks of brain functional connectivity of TDC (top) and ADHD (bottom) groups. In each row, three different views are also provided: sagittal (left), axial (middle), and coronal (right).

A2.3 Additional comparative results on Tlasso and TransCLIME

1. Sensitivity analysis for Tlasso. In simulations of the main text, we followed the treatment in Lyu et al. (2019) for selecting the tuning parameters $\{\lambda_m; m = 1, \ldots, M\}$ in TLasso. Specifically, it is set as $\lambda_m = C\sqrt{(p_m \log p_m)/(np)}$ with $C = 20$ as suggested in all the numerical experiments in Lyu et al. (2019). Moreover, we also conduct a sensitivity analysis on the choice of the tuning parameter under $M = 3$ with dimensions $(p_1, p_2, p_3) = (10, 10, 20)$ and $M = 2$ with dimensions $(p_1, p_2) = (100, 100)$, which is shown in Figure A2. A similar observation has also been made in Lyu et al. (2019). More importantly, the smallest error of TLasso is still larger than that of the proposed method with at least one informative auxiliary domain.

2. Additional comparative results on TransCLIME. We conduct a comparison with vector-valued methods, including the transfer learning method TransCLIME (Li et al., 2022b) and the CLIME method (Cai et al., 2011) using only the target domain, in two ways.

In the first way, we still follow Scenario 1 in the main text to generate tensor data (all auxiliary domains are informative by setting the informative set $\mathcal{A} = [K]$ and varying $K \in \{1, \cdots, 5\}$), then we can apply the TransCLIME and CLIME after flattening the

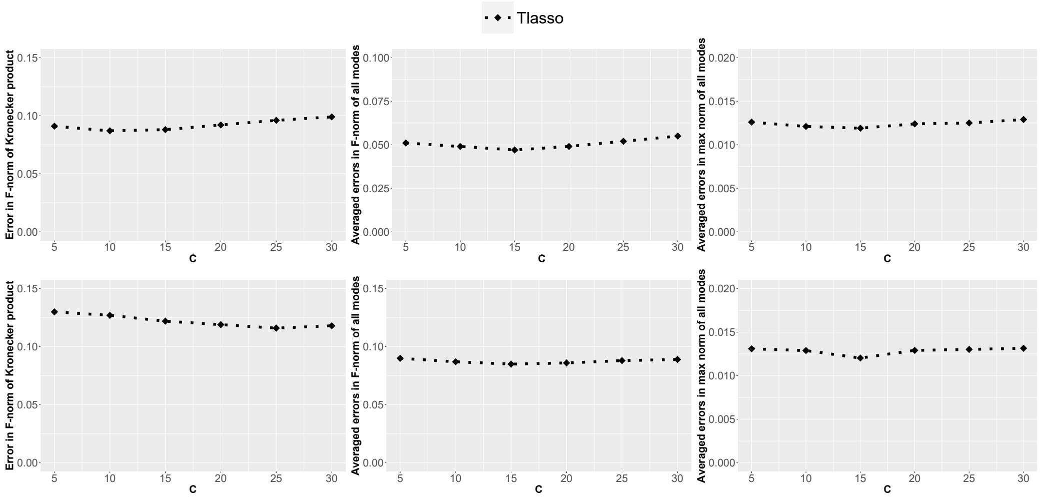

Figure A2: Estimation errors of Tlasso under the choices of $C \in \{5, 10, 15, 20, 25, 30\}$. The top and bottom rows correspond to the $M = 3$ and $M = 2$, respectively.

tensor into a vector. We modify the dimensions $(p_1, p_2, p_3) = (5, 5, 10)$ considering the huge computational cost after flattening the tensor. The estimation error in Frobenius norm of Kronecker product of precision matrices, defined as $\|\widehat{\mathbf{\Omega}}^{(K)} - \mathbf{\Omega}^{(K)*}\|_F$, where $\widehat{\mathbf{\Omega}}^{(K)} = \widehat{\mathbf{\Omega}}_1 \otimes \cdots \otimes \widehat{\mathbf{\Omega}}_M$ and $\mathbf{\Omega}^{(K)*} = \mathbf{\Omega}_1^* \otimes \cdots \otimes \mathbf{\Omega}_M^*$, of all competitors are shown in Figure A3. Note that the estimation errors in Frobenius or max norm of all modes are not available using TransCLIME and CLIME with the flattened data.

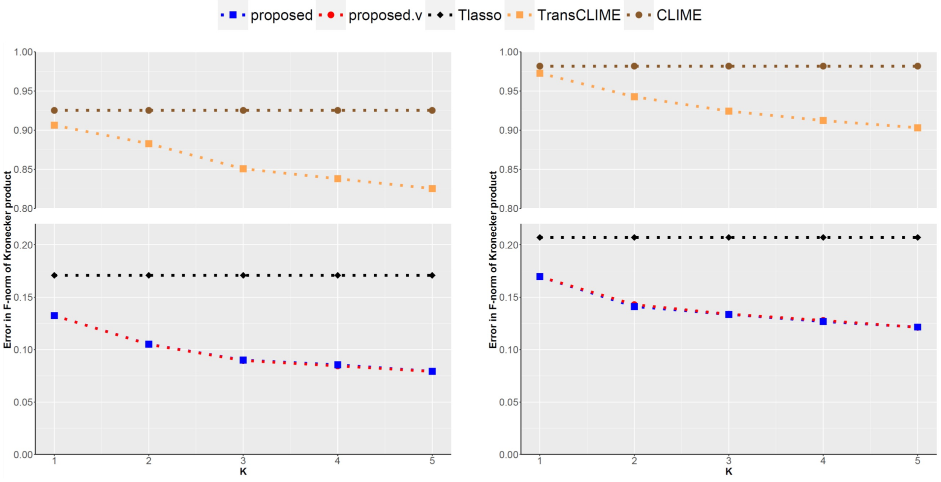

Figure A3: Averaged estimation errors in F-norm of Kronecker product for Scenario 1 with $(p_1, p_2, p_3) = (5, 5, 10)$. The left and right columns correspond to the chain and nearest neighbor graph, respectively.

Some interesting results can be observed. First, CLIME is far inferior to TLasso, which is consistent with the existing literature Lyu et al. (2019). More interestingly, as the number of informative auxiliary domains increases, even though TransCLIME is gradually improved compared to CLIME, it is still inferior to TLasso. This observation reflects that when the tensor structure is ignored, even with auxiliary information (while the data in the auxiliary domains is also flattened), the transfer learning method with the flattened data cannot improve the estimation efficiency sufficiently. Despite being a popular treatment for handling tensorial data, flattening data may suffer from information loss and can completely destroy certain intrinsic structures of the tensor data, which seriously reduces estimation efficiency as well as computational efficiency.

In the second way, we directly generate vector-valued data (i.e., $M = 1$), which is the main focus of the TransCLIME (Li et al., 2022b). We fix the data dimension $p = 100$ and the number of auxiliary domains $K = 5$ with size $n_k = 200$ for $k \in [K]$, and vary the numbers of informative auxiliary domains $\text{card}(\mathcal{A}) \in \{1, \cdots, K\}$. The informative auxiliary domains with $k \in \mathcal{A}$ are generated similarly as Scenario 2. To perform a positive comparison to demonstrate the powerful effectiveness of our method in dealing with negative transfer problems, we consider non-informative auxiliary domains that are further away from the target domain and more challenging than Scenario 2. Specifically, for $k \notin \mathcal{A}$, $[\mathbf{\Delta}^{(k)}]_{(i,j)} = 0$ with probability 0.5, or randomly generated from $\text{Unif}[-h_{02}, h_{02}]$ with probability 0.5, where $h_{02} = 20s\sqrt{\frac{\log p}{n}}$. In this simulation, we also consider the CLIME using only the target domain as a naive benchmark and the "oracle" method by applying TransCLIME on the target domain and the known informative auxiliary domains. The performances of competitors are shown in Figure A4.

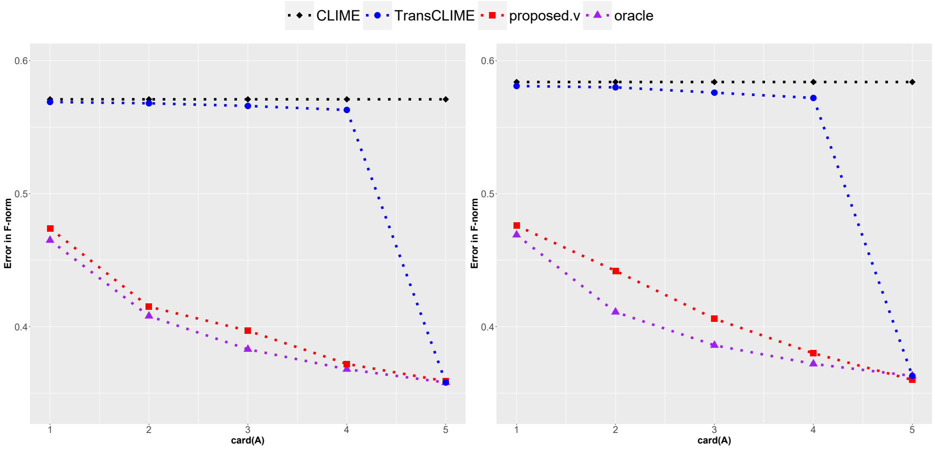

Figure A4: Averaged estimation errors in F-norm for vector-valued data with varying $\text{card}(\mathcal{A})$. The left and right columns correspond to the chain and nearest neighbor graph, respectively.

Some important results can be observed. If there are some non-informative auxiliary domains with strong influence (even if only one), the performance of TransCLIME is similar to that of the benchmark CLIME using only the target domain, and it hardly shows the improvement of transfer learning. This is due to the fact that TransCLIME adopts naive sample-size-based weights in the aggregation step for multiple auxiliary domains, that is, $\widehat{\mathbf{\Sigma}}_m^{\mathcal{A}} = \sum_{k=1}^{K} \alpha_k \widehat{\mathbf{\Sigma}}_m^{(k)}$, with $\alpha_k = n_k/N$ and $N = \sum_{k=1}^{K} n_k$. Yet, it does not take into account the similarities between the target and auxiliary domains. If there are some non-informative auxiliary domains that are extremely different from the target domain, it will result in the negative transfer. Although Li et al. (2022b) adopts the model selection between the transfer learning estimator and the initial estimator, this step may force the initial estimator to be selected. In this sense, TransCLIME can guarantee that transfer learning is no less effective than using the target domain only, but it may also offset the potential improvement benefiting from the informative auxiliary domains with a positive impact. Naturally, the significant improvement of TransCLIME compared to CLIME may rely on the fact that all auxiliary domains are informative. In sharp contrast, the estimation errors of the proposed method decrease so fast that it can dominate TransCLIME even when there is only one informative auxiliary domain, and its overall performance is comparable to "oracle", thanks to the data-adaptive weighting scheme for the auxiliary domains combining both sample sizes and the similarities between the target and auxiliary domains.

References

Hamsa Bastani. Predicting with proxies: Transfer learning in high dimension. *Management Science*, 67(5):2964–2984, 2021.

Pierre Bellec, Carlton Chu, Francois Chouinard-Decorte, Yassine Benhajali, Daniel S Margulies, and R Cameron Craddock. The neuro bureau adhd-200 preprocessed repository. *Neuroimage*, 144:275–286, 2017.

Chenjing Cai, Shiwei Wang, Youjun Xu, Weilin Zhang, Ke Tang, Qi Ouyang, Luhua Lai, and Jianfeng Pei. Transfer learning for drug discovery. *Journal of Medicinal Chemistry*, 63(16):8683–8694, 2020.

T Tony Cai and Hongji Wei. Transfer learning for nonparametric classification: Minimax rate and adaptive classifier. *The Annals of Statistics*, 49(1):100–128, 2021.

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

A Philip Dawid. Some matrix-variate distribution theory: notational considerations and a bayesian application. *Biometrika*, 68(1):265–274, 1981.

Steven G Dickstein, Katie Bannon, F Xavier Castellanos, and Michael P Milham. The neural correlates of attention deficit hyperactivity disorder: An ale meta-analysis. *Journal of Child Psychology and Psychiatry*, 47(10):1051–1062, 2006.

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

Yuqing Gao and Khalid M Mosalam. Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, 33(9):748–768, 2018.

SG Ghurye and Ingram Olkin. A characterization of the multivariate normal distribution. *The Annals of Mathematical Statistics*, 33(2):533–541, 1962.

Kristi R Griffiths, Taylor A Braund, Michael R Kohn, Simon Clarke, Leanne M Williams, and Mayuresh S Korgaonkar. Structural brain network topology underpinning adhd and response to methylphenidate treatment. *Translational psychiatry*, 11(1):1–9, 2021.

Shiyuan He, Jianxin Yin, Hongzhe Li, and Xing Wang. Graphical model selection and estimation for high dimensional tensor data. *Journal of Multivariate Analysis*, 128:165–185, 2014.

Yong He, Qiushi Li, Qinqin Hu, and Lei Liu. Transfer learning in high-dimensional semiparametric graphical models with application to brain connectivity analysis. *Statistics in medicine*, 41(21):4112–4129, 2022.

Samuel B Hopkins, Jonathan Shi, and David Steurer. Tensor principal component analysis via sum-of-square proofs. In *Conference on Learning Theory*, pages 956–1006. PMLR, 2015.

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

Karthickeyan Chella Krishnan, Zeyneb Kurt, Rio Barrere-Cain, Simon Sabir, Aditi Das, Raquel Floyd, Laurent Vergnes, Yuqi Zhao, Nam Che, Sarada Charugundla, et al. Integration of multi-omics data from mouse diversity panel highlights mitochondrial dysfunction in non-alcoholic fatty liver disease. *Cell systems*, 6(1):103–115, 2018.

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

Sai Li, Tianxi Cai, and Rui Duan. Targeting underrepresented populations in precision medicine: A federated transfer learning approach. *arXiv:2108.12112*, pages 1–25, 2021.

Sai Li, T Tony Cai, and Hongzhe Li. Transfer learning for high-dimensional linear regression: Prediction, estimation, and minimax optimality. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, pages 1–26, 2022a.

Sai Li, T Tony Cai, and Hongzhe Li. Transfer learning in large-scale gaussian graphical models with false discovery rate control. *Journal of the American Statistical Association*, pages 1–13, 2022b.

Haotian Lin and Matthew Reimherr. On transfer learning in functional linear regression. *arXiv: 2206.04277*, pages 1–31, 2022.

Weidong Liu and Xi Luo. Fast and adaptive sparse precision matrix estimation in high dimensions. *Journal of multivariate analysis*, 135:153–162, 2015.

Yuetian Luo and Anru R Zhang. Tensor clustering with planted structures: Statistical optimality and computational limits. *The Annals of Statistics*, 50(1):584–613, 2022.

Xiang Lyu, Will Wei Sun, Zhaoran Wang, Han Liu, Jian Yang, and Guang Cheng. Tensor graphical model: Non-convex optimization and statistical inference. *IEEE transactions on pattern analysis and machine intelligence*, 42(8):2024–2037, 2019.

Keqian Min, Qing Mai, and Xin Zhang. Fast and separable estimation in high-dimensional tensor gaussian graphical models. *Journal of Computational and Graphical Statistics*, 31 (1):294–300, 2022.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

Yuqing Pan, Qing Mai, and Xin Zhang. Covariate-adjusted tensor classification in high dimensions. *Journal of the American statistical association*, 114(527):1305–1319, 2018.

Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Restricted eigenvalue properties for correlated gaussian designs. *The Journal of Machine Learning Research*, 11:2241–2259, 2010.

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

Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. Transfer learning in natural language processing. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Tutorials*, pages 15–18, 2019.

Yang Shu, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Transferable curriculum for weakly-supervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4951–4958, 2019.

Charles M Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, pages 1135–1151, 1981.

Catherine J Stoodley. The cerebellum and neurodevelopmental disorders. *The Cerebellum*, 15(1):34–37, 2016.

Will Wei Sun and Lexin Li. Store: sparse tensor response regression and neuroimaging analysis. *The Journal of Machine Learning Research*, 18(1):4908–4944, 2017.

Ye Tian and Yang Feng. Transfer learning under high-dimensional generalized linear models. *Journal of the American Statistical Association*, pages 1–30, 2022.

Yung Liang Tong and YL Tong. *Fundamental properties and sampling distributions of the multivariate normal distribution*. Springer, 1990.

Eve M Valera, Rebecca MC Spencer, Thomas A Zeffiro, Nikos Makris, Thomas J Spencer, Stephen V Faraone, Joseph Biederman, and Larry J Seidman. Neural substrates of impaired sensorimotor timing in adult attention-deficit/hyperactivity disorder. *Biological psychiatry*, 68(4):359–367, 2010.

C-F Westin, Stephan E Maier, Hatsuho Mamata, Arya Nabavi, Ferenc A Jolesz, and Ron Kikinis. Processing and visualization for diffusion tensor mri. *Medical image analysis*, 6 (2):93–108, 2002.

Huayu Zhang, Yue Zhao, Weifang Cao, Dong Cui, Qing Jiao, Weizhao Lu, Hongyu Li, and Jianfeng Qiu. Aberrant functional connectivity in resting state networks of adhd patients revealed by independent component analysis. *BMC neuroscience*, 21(1):1–11, 2020.

Teng Zhang and Hui Zou. Sparse precision matrix estimation via lasso penalized d-trace loss. *Biometrika*, 101(1):103–120, 2014.

Xiang Zhang, Lexin Li, Hua Zhou, Yeqing Zhou, Dinggang Shen, et al. Tensor generalized estimating equations for longitudinal imaging analysis. *Statistica Sinica*, 29(4):1977, 2019.

Yue Zhao, Dong Cui, Weizhao Lu, Hongyu Li, Huayu Zhang, and Jianfeng Qiu. Aberrant gray matter volumes and functional connectivity in adolescent patients with adhd. *Journal of Magnetic Resonance Imaging*, 51(3):719–726, 2020.

Hua Zhou, Lexin Li, and Hongtu Zhu. Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, 108(502):540–552, 2013.

Yunzhang Zhu and Lexin Li. Multiple matrix gaussian graphs estimation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):927–950, 2018.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.