A Latent Multilayer Graphical Model for Complex, Interdependent Systems

Martin Ondrus
Neuroscience and Mental Health Institute
University of Alberta

Ivor Cribben
Alberta School of Business
University of Alberta

Yang Feng
School of Global Public Health
New York University

Abstract

Networks have been extensively used and have provided novel insights across a wide variety of research areas. However, many real-world systems are, in fact, a "network of networks", or a multilayer network, which interact as components of a larger multimodal system. A major difficulty in this multilayer framework is the estimation of interlayer edges or connections. In this work, we propose a new estimation method, called multilayer sparse + low-rank inverse covariance estimation (multiSLICE), which estimates the interlayer edges. multiSLICE bridges latent variable Gaussian graphical methods with multilayer networks, offering a flexible framework for modeling processes with irregular sampling and heterogeneous graph structures. We develop an effective algorithm to compute the estimator. We also establish theoretical conditions for the recoverability of the joint space, analyze how interlayer interactions influence joint parameter estimation, and provide theoretical bounds on their relationships. Finally, we rigorously evaluate our method on both simulated and multimodal neuroimaging data, demonstrating improvements over state-of-the-art approaches. All the relevant R code implementing the method in the article is available on GitHub.

1 Introduction

Our world is a complex assemblage of many interdependent systems that can be described as sets of interacting components. Networks (or graphs, which we use interchangeably) succinctly represent these systems, wherein individual actors (nodes) are connected through relationships (edges) that often encode some quantitative meaning (Wasserman and Faust, 1994). By representing systems as graphs, local and global graph attributes can describe phenomena in social analysis (Wasserman and Faust, 1994), genomics (Seal et al., 2023; Argelaguet et al., 2018), and neuroscience (Bassett and Sporns, 2017; Bassett et al., 2011), among others.

Gaussian graphical models (GGMs) estimate dependencies in such systems by modeling them as multivariate Gaussian distributions, parameterized by the inverse covariance (precision) matrix (Hastie et al., 2009). However, many real-world systems are in fact a "network of networks" (Craven and Wellman, 1973) that interact as parts of a larger multimodal system. For example, a logistics network may contain multiple transportation types or "layers" (e.g., air, rail, pipeline, road), which have within (intralayer) and between (interlayer) relationships. In neuroscience, genes, cells, tissue, anatomical structure, and functional measures can interact and contribute to behaviors or pathologies. Multilayer graphs naturally extend to such cases and encode rich multimodal data into a unified representation (Kivelä et al., 2014), but learning the dependence structure remains difficult.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

In such systems, estimating the dependence across different modalities presents challenges due to differences in the number of measured variables, as well as the sampling rate. In neuroscience, for example, measurement can be made across multiple modalities, such as magneto/electroencephalography (M/EEG), functional magnetic resonance imaging (fMRI), and structural magnetic resonance imaging (sMRI), among others. Each modality has unique strengths and weaknesses that are typically complementary to each other. For example, M/EEG has high temporal resolution but poor spatial resolution, and vice versa for fMRI. Different layers are often tightly coupled, with structure affecting function and vice versa. The challenge in combining the information across these modalities is that each may differ in the number of variables (nodes) and/or the number of samples. Furthermore, these measurements are taken independently from one another, creating additional issues on how to best structure the "jointness" of these measurements. Ideally, we endeavor to preserve the natural structure of the data by avoiding up- or down-sampling between different modalities. We may wish to incorporate information from other variables not observed across all modalities. As such, there is a clear need for a flexible framework that brings together these multiple modalities.

To this end, we introduce multiSLICE, which bridges multilayer networks (Kivelä et al., 2014) and latent variable GGMs (Chandrasekaran et al., 2012) by providing a flexible framework for modeling processes with irregular sampling and heterogeneous graph structures. In our setting, we consider a sparse + low-rank setup, where each modality has a sparse component, and multiple modalities are allowed to exist over a joint low-rank, latent space. Crucially, in a neuroscience context, the sparse component captures local circuits between different regions of the brain, while the low-rank component depicts common environmental influences across modalities (Yatsenko et al., 2015). To our knowledge, this is the *first* statistical model for estimating interlayer edges in a multilayer network with different numbers of variables and sample sizes, and one that is scalable in the number of modalities it can integrate. Our main contributions can be summarized as follows:

- We link latent variable Gaussian graphical methods and multilayer networks, establishing a flexible framework to model processes with irregular sampling and non-identical nodes.
- We derive an effective algorithm to solve for this estimator.
- We establish theoretical conditions for the recoverability of the joint space, analyze how interlayer interactions influence joint parameter estimation, and provide theoretical bounds on their relationships.
- We rigorously test our method on both simulated and real experimental data, with comparisons to the state-of-the-art, supporting the efficacy of our approach in both settings.

2 Related methods

In general, related methods accommodate either differing numbers of variables (p) between layers or differing sample sizes (n). Table 1 provides a summary. Mohan et al. (2014) suggests a node-based method (CNJGL) which uses a row-column overlap norm based approach for estimation. Lin et al. (2016) propose a multilayer GGM via penalized likelihood estimation (MLGGM). Gan et al. (2019) uses a Bayesian group regularization method and spike-and-slab Lasso priors in the proposed BJEMGM method. Price et al. (2021) suggests a cluster fusion regularization (CFR) based method to estimate multiple precision matrices. The BANS method models hierarchical dependencies using Bayesian node-wise selection (Ha et al., 2021). JMMLE extends this method by decomposing the multilayer problem into two-layer subproblems using neighborhood selection and group-penalized regression (Majumdar and Michailidis, 2022). Chang et al. (2022) considers a graph quilting problem, in which observations of the covariance are missing, and proposes a matrix-completion-based method (LRGQ) to estimate the missing entries. Albanese et al. (2024) suggests a collaborative graphical lasso (coglasso) for the estimation of multi-omics network data.

	CNJGL (2014)	MLGGM (2016)	BJEMGM (2019)	CFR (2021)	BANS (2021)	JMMLE (2022)	LRGQ (2022)	coglasso (2024)	multiSLICE (Ours)
n	**Yes**	No	**Yes**	**Yes**	No	No	**Yes**	No	**Yes**
p	No	**Yes**	No	No	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

Table 1: A summary of methods in columns with year of publication. Entries indicate whether the approach can handle different sample sizes (n) or different node sets (p) across modalities.

3 Preliminaries

3.1 Notation

A graph G is defined as a collection of vertices (or nodes) and edges, $G = (V, E)$. For the edge set of a weighted graph, we denote the edge between nodes i and j with weight w_{ij} by the tuple $(i, j, w_{ij}) \in E$. Alternatively, G can also be described by an adjacency matrix denoted by $\boldsymbol{A}$. The cardinality of a set s, or the number of elements in the set, is denoted by $|s|$. We denote a matrix $\boldsymbol{B}$ by a bold uppercase letter, a vector $\boldsymbol{b}$ with a bold lowercase letter, and a scalar a with a lowercase letter. $\boldsymbol{B}_{ij}$ is the ith row and jth column of a matrix $\boldsymbol{B}$. The vector arising from the jth column in $\boldsymbol{B}$ is denoted by $\boldsymbol{b}_j$. Positive definiteness of a matrix is denoted by $\succ 0$ and positive semidefiniteness by $\succcurlyeq 0$. The rank of a matrix $\boldsymbol{B}$ is denoted by $\mathcal{R}(\boldsymbol{B})$. We define a matrix $\boldsymbol{B} \in \mathbb{R}^{p \times p}$ as low-rank if $\mathcal{R}(\boldsymbol{B}) \ll p$. We denote the upper triangle of a square symmetric matrix by $\mathcal{U}(\boldsymbol{B})$, with the corresponding vector $\boldsymbol{b}_u$. $\mathcal{P}_\Omega$ is a projection operator that selects the indices in Ω, such that

$$[\mathcal{P}_\Omega(\boldsymbol{B})]_{ij} = \begin{cases} \boldsymbol{B}_{ij} & \text{if } (i,j) \in \Omega \\ \emptyset & \text{if } (i,j) \in \Omega^c \end{cases}$$

We denote the identity matrix by $\boldsymbol{I}$, where entries on the main diagonal are 1 and all other entries are 0. We denote the singular value decomposition (SVD) of the matrix $\boldsymbol{B}$ by $\boldsymbol{B} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^T$, where the ith value on the diagonal of $\boldsymbol{\Lambda}$ is denoted by λ_i. We denote a truncated SVD of rank r by $\boldsymbol{B}_r = \mathit{SVD}_r(\boldsymbol{B}) = \boldsymbol{U}_r\boldsymbol{\Lambda}_r\boldsymbol{V}_r^T$. The norm of a matrix is denoted by $\|\cdot\|$.

3.2 Multilayer Graph Structure

We define a multilayer graph using the principles of Kivelä et al. (2014). An l-layer multilayer graph is defined as $G_M = (V_M, E_M, V, L)$, where V is the set of vertices or nodes, L is the set of layers, $V_M \subseteq V \times L$ is the set of node-layer tuples, and $E_M \subseteq V_M \times V_M$ defines the edge set describing connections between node-layer tuples. A node, u, of a specific layer, $\boldsymbol{\alpha}$, is encoded as $(u, \boldsymbol{\alpha})$. The set of *intralayer* edges, is defined as $E_A = \{((u, \boldsymbol{\alpha}), (v, \boldsymbol{\beta})) \in E_M | \boldsymbol{\alpha} = \boldsymbol{\beta}\}$ while *interlayer* edges are defined as $E_C = \{((u, \boldsymbol{\alpha}), (v, \boldsymbol{\beta})) \in E_M | \boldsymbol{\alpha} \neq \boldsymbol{\beta}\}$ where $E_C = E_M \setminus E_A$.

A supra-adjacency matrix $\boldsymbol{A}$ is a two-dimensional representation of G_M obtained through a mapping process called "flattening" or "matricization" (Kivelä et al., 2014). For multiSLICE, detailed in Section 3.3, we consider a layer-disjoint multilayer network, where each node occupies only one layer. To reduce notation clutter, when $\boldsymbol{\alpha} = \boldsymbol{\beta}$, we simplify the subscript to $\boldsymbol{\alpha}$. Hence, an intralayer adjacency matrix is defined as $\boldsymbol{A}_{\boldsymbol{\alpha}}$, whereas an interlayer adjacency matrix is defined as $\boldsymbol{A}_{\boldsymbol{\alpha}\boldsymbol{\beta}} | \boldsymbol{\alpha} \neq \boldsymbol{\beta}$. $\boldsymbol{A}_{\boldsymbol{\alpha}, \setminus \boldsymbol{\alpha}}$ defines the submatrix formed by the intersection of all rows of $\boldsymbol{\alpha}$ and all columns not in $\boldsymbol{\alpha}$. In general, we use bold subscripts to denote a layer-specific quantity.

Remark 1. *For a multilayer graph G_M with a symmetric supra-adjacency matrix $\boldsymbol{A} = \boldsymbol{A}^T$, we make the following connections to the projection operator $\mathcal{P}_\Omega$.*

- *For layer $\boldsymbol{\alpha}$, the intralayer edges $E_{\boldsymbol{\alpha}}$ are encoded by projection of that layer, $\mathcal{P}_{\Omega_{\boldsymbol{\alpha}}}(\boldsymbol{A})$, since we have that $E_{\boldsymbol{\alpha}} = \{((u, \boldsymbol{\alpha}), (v, \boldsymbol{\alpha})) \in E_M \mid u, v \in \omega_{\boldsymbol{\alpha}}\}$.*
- *For l layers, Ω is the union of individual layers' observed indices, $\Omega = \bigcup_{\boldsymbol{\alpha}=1}^{l}(\Omega_{\boldsymbol{\alpha}}) = \Omega_1 \cup \Omega_2 \cup, \ldots, \cup \Omega_l$. The intralayer edges E_A across all layers are encoded by $\mathcal{P}_\Omega(\boldsymbol{A})$.*
- *The interlayer edges, $E_C = \{((u, \boldsymbol{\alpha}), (v, \boldsymbol{\beta})) \in E_M \mid u \in \omega_{\boldsymbol{\alpha}}$ and $v \in \omega_{\boldsymbol{\beta}}$, $\boldsymbol{\alpha} \neq \boldsymbol{\beta}\}$ are obtained through the complement of $\mathcal{P}_\Omega$, denoted by $\mathcal{P}_{\Omega^c}$. $\mathcal{P}_{\Omega^c}(\boldsymbol{A})$ corresponds to the hidden or unobserved edges of $\boldsymbol{A}$.*

The problem of multilayer graph estimation can then be seen as a special case of matrix completion in which $\boldsymbol{A}$ is observed in blocks along the main diagonal. More formally, we observe $\mathcal{P}_\Omega(\boldsymbol{A})$ and we wish to recover the full adjacency matrix $\boldsymbol{A}$ by learning the mapping $\mathcal{P}_\Omega^{-1}$.

To provide some insight, Figure 1 shows a toy example of a 3-layer system with its associated adjacency matrix. In our multilayer system, we are interested in recovering a *weighted* supra-adjacency matrix, in which intralayer edges are organized along the main diagonal in a block-like structure and interlayer edges are in the remaining off-diagonal elements.

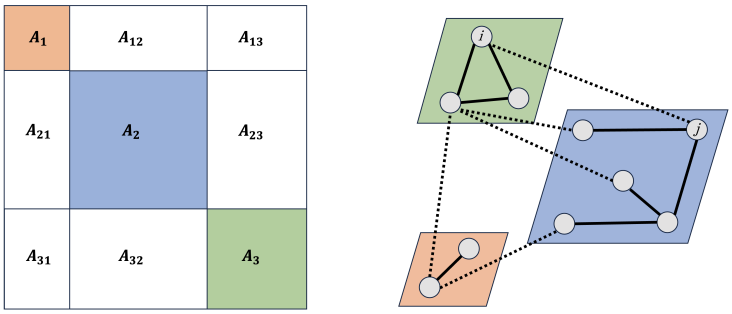

Figure 1: A graphic of a weighted adjacency matrix (left), and the associated multilayer graph (right). Intralayer edges are shown in the graph as solid black lines and color-coded along the main diagonal in a block-like structure ($\boldsymbol{A_1}, \boldsymbol{A_2}, \boldsymbol{A_3}$ in the supra-adjacency matrix). Interlayer edges are encoded with dashed lines in the graph (right) and in the matrix (left) as the remaining off-diagonal blocks (uncolored). Note that this matrix is symmetric, which enforces an undirected structure and ensures that interlayer pairs are transposes of each other ($\boldsymbol{A_{12}} = \boldsymbol{A_{21}}^T, \boldsymbol{A_{13}} = \boldsymbol{A_{31}}^T, \boldsymbol{A_{23}} = \boldsymbol{A_{32}}^T$). The edge between nodes $i, j \in V$ in the graph (right) is expressed as $\{(i, \mathbf{1}), (j, \mathbf{2})\} \subset E_M$.

3.3 Problem Setting

Assume that $\boldsymbol{L}^* \in \mathbb{R}^{p \times p}$ is a weighted supra-adjacency matrix that is composed of square blocks along the main diagonal, corresponding to the observed variables. For matrix $\boldsymbol{L}^*$, we observe structures related only to intralayer edges and denote the projection into the observed space by $\mathcal{P}_\Omega$. $\boldsymbol{L}^*$ and its projection $\mathcal{P}_\Omega(\boldsymbol{L}^*)$ are defined as follows:

$$\boldsymbol{L}^* = \begin{bmatrix} \boldsymbol{L_1^*} & \boldsymbol{L_{12}^*} & \cdots & \boldsymbol{L_{1l}^*} \\ \boldsymbol{L_{21}^*} & \boldsymbol{L_2^*} & \cdots & \boldsymbol{L_{2l}^*} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{L_{l1}^*} & \boldsymbol{L_{l2}^*} & \cdots & \boldsymbol{L_l^*} \end{bmatrix}, \quad \mathcal{P}_\Omega(\boldsymbol{L}^*) = \begin{bmatrix} \boldsymbol{L_1^*} & \emptyset & \cdots & \emptyset \\ \emptyset & \boldsymbol{L_2^*} & \cdots & \emptyset \\ \vdots & \vdots & \ddots & \vdots \\ \emptyset & \emptyset & \cdots & \boldsymbol{L_l^*} \end{bmatrix}.$$

Note that the low-rank latent component corresponding to layer $\boldsymbol{\alpha}$ is given by $\boldsymbol{L_\alpha^*} = \mathcal{P}_{\Omega_\alpha}(\boldsymbol{L}^*)$ for $\boldsymbol{\alpha} \in \{1, \cdots, l\}$. There are also l sparse intralayer matrices, the set of which is denoted by $\{\boldsymbol{S}^*\}_{\boldsymbol{\alpha}=1}^{l}$.

We assume that the inverse covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{*-1}$ originates from the sum of a sparse component $\boldsymbol{S_\alpha}^*$ and a low-rank (or latent, used interchangeably) component $\boldsymbol{L_\alpha}^*$ such that $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{*-1} = \boldsymbol{S}_{\boldsymbol{\alpha}}^* + \boldsymbol{L}_{\boldsymbol{\alpha}}^*$, where $\boldsymbol{L}_{\boldsymbol{\alpha}}^* = \mathcal{P}_{\Omega_\alpha}(\boldsymbol{L}^*)$. We also assume $\boldsymbol{X_1} \dots \boldsymbol{X_l} \in \mathbb{R}^{n_\alpha \times p_\alpha}$ are i.i.d. draws from a multivariate normal distribution, $\boldsymbol{X_\alpha} \sim \mathcal{N}(\mu_{\boldsymbol{\alpha}}, \boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^*)$. For the population covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^* \succ 0$ and population mean $\mu_{\boldsymbol{\alpha}}$, the finite sample realization is the sample covariance matrix,

$$\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}} = \frac{1}{n_{\boldsymbol{\alpha}} - 1} \sum_{i=1}^{n_{\boldsymbol{\alpha}}} (\boldsymbol{x}_{\boldsymbol{\alpha}i} - \bar{\boldsymbol{x}}_{\boldsymbol{\alpha}})(\boldsymbol{x}_{\boldsymbol{\alpha}i} - \bar{\boldsymbol{x}}_{\boldsymbol{\alpha}})^T,$$

where $\boldsymbol{x}_{\boldsymbol{\alpha}i}$ and $\bar{\boldsymbol{x}}_{\boldsymbol{\alpha}}$ are $p_{\boldsymbol{\alpha}}$ dimensional vectors of the ith sample and sample means, respectively, from the $\boldsymbol{\alpha}$th layer. Ignoring the $\mu_{\boldsymbol{\alpha}}$ term, the log-likelihood function is given by

$$\mathcal{L}(\boldsymbol{S}_{\boldsymbol{\alpha}} + \boldsymbol{L}_{\boldsymbol{\alpha}}; \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}) = \log\det(\boldsymbol{S}_{\boldsymbol{\alpha}} + \boldsymbol{L}_{\boldsymbol{\alpha}}) - \operatorname{tr}(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}(\boldsymbol{S}_{\boldsymbol{\alpha}} + \boldsymbol{L}_{\boldsymbol{\alpha}})).$$

The joint multilayer likelihood function, then, is given by the sum of individual layerwise likelihoods

$$\mathcal{L}_M(\{\boldsymbol{S}\}_{\boldsymbol{\alpha}=1}^{l}, \boldsymbol{L}; \{\tilde{\boldsymbol{\Sigma}}\}_{\boldsymbol{\alpha}=1}^{l}) = \sum_{\boldsymbol{\alpha}=1}^{l} \mathcal{L}(\boldsymbol{S}_{\boldsymbol{\alpha}}, \mathcal{P}_{\Omega_{\boldsymbol{\alpha}}}(\boldsymbol{L}); \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}). \tag{1}$$

Remark 2. *By imposing a joint latent matrix $\boldsymbol{L}^*$, the number of nodes ($p_{\boldsymbol{\alpha}}$) between layers can vary. By focusing on the inverse covariance (precision) matrix rather than the layerwise data vectors (which may be different lengths depending on the number of samples), we do not impose the constraint that the number of samples ($n_{\boldsymbol{\alpha}}$) be identical across layers.*

In short, we model a multilayer network as a product of layer-specific Gaussian densities whose precision matrices share a joint parameterization in the latent space. We set this up by decomposing the

precision matrix of each modality into a sparse component unique to that modality and a shared latent component that captures both the within-modality and the cross-modality edges of the multilayer network. Our objective is to maximize (1) by estimating both the sparse layerwise components as well as the low-rank joint latent component. In Figure 2, we show both the forward data-generating process and the proposed reverse estimation method. The data-generating process comprises two steps: the first step is a projection of the latent space into the observed space via $\mathcal{P}$, and the second step is mixing with sparse components to generate the precision matrix. This two-step decomposition is important in the next section, where we propose estimating parameters via the reverse process.

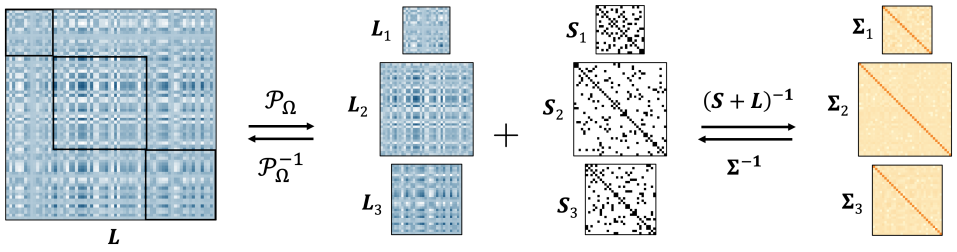

Figure 2: An illustration of the data-generating (forward) and parameter estimation (reverse) processes for a three-layer system. $\boldsymbol{L}$ denotes the joint latent space parameterized by a weighted supra-adjacency matrix, $\mathcal{P}_\Omega$ denotes the projection into observed layers, and its inverse is $\mathcal{P}_\Omega^{-1}$. The observed $\boldsymbol{S}$ and $\boldsymbol{L}$ correspond to edge weights from the intralayer components.

4 Methodology

In (1), the summation over l layers decomposes the objective into independent latent variable GGMs. To this end, we introduce the multilayer sparse + low-rank inverse covariance estimator (multiSLICE), which minimizes the following function,

$$\underbrace{\sum_{\boldsymbol{\alpha}=1}^{l}}_{\text{layers}} \underbrace{(-\mathcal{L}(\hat{\boldsymbol{S}}_{\boldsymbol{\alpha}}; (\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}^{-1} - \hat{\boldsymbol{L}}_{\boldsymbol{\alpha}})^{-1}) + \rho\|\hat{\boldsymbol{S}}_{\boldsymbol{\alpha}}\|_1}_{\text{penalized negative log likelihood}} + \underbrace{\|\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}(\hat{\boldsymbol{S}}_{\boldsymbol{\alpha}} + \hat{\boldsymbol{L}}_{\boldsymbol{\alpha}}) - \boldsymbol{I}\|_F^2)}_{\text{covariance fidelity}} \tag{2}$$

$$\text{s.t. } \mathcal{R}(\hat{\boldsymbol{L}}) = r, \text{ where } 0 < r < p\,.$$

Here, ρ is a tuning parameter for the sparsity in $\hat{\boldsymbol{S}}$ and r is the pre-specified rank of the latent matrix $\hat{\boldsymbol{L}}$. The objective function is decomposed into a penalized negative log-likelihood term and a covariance fidelity term. We propose a two-stage algorithm to solve (2), which follows Figure 2. In the first stage, individual latent variable GGMs are estimated, and in the second stage, matrix completion is applied to the latent estimates using the block singular value decomposition algorithm (Bishop and Yu, 2014). Algorithm 1 describes an alternating descent algorithm which we refer to as sparse + low-rank inverse covariance estimation (SLICE). Notice that it can also be used with various other penalties on $\hat{\boldsymbol{S}}$ (e.g., SCAD; Fan et al. 2009) or estimators (e.g., CLIME; Cai et al. 2011) by simply substituting in a different estimator for $\hat{\boldsymbol{S}}$ (see the Supplementary Materials for more details).

Algorithm 1 Sparse + low-rank inverse covariance estimation (SLICE) with GLASSO

Inputs: $\tilde{\boldsymbol{\Sigma}}$, ρ, r, $maxiter$, tol
$\hat{\boldsymbol{L}}^0 = \hat{\boldsymbol{S}}^0 = 0$
  for $i = 1$ to $maxiter$ **do**
    $\hat{\boldsymbol{L}}^{(i)} := SVD_r(\tilde{\boldsymbol{\Sigma}}^{-1} - \hat{\boldsymbol{S}}^{(i-1)})$
    $\hat{\boldsymbol{S}}^{(i)} := \text{GLASSO}((\tilde{\boldsymbol{\Sigma}}^{-1} - \hat{\boldsymbol{L}}^{(i)})^{-1},\ \rho)$
  end for $\Delta\hat{\boldsymbol{L}}$ AND $\Delta\hat{\boldsymbol{S}} < tol$ OR $\Delta\mathcal{L}(\hat{\boldsymbol{S}} + \hat{\boldsymbol{L}}) < tol$
Outputs: $\hat{\boldsymbol{S}}$, $\hat{\boldsymbol{L}}$

Algorithm 2 Multilayer SLICE (multiSLICE) estimation

Inputs: $\{\hat{\boldsymbol{\Sigma}}\}_{\boldsymbol{\alpha}=1}^{l}$, ρ, r, $maxiter$, tol
Apply SLICE on $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\alpha}}$ for all $\boldsymbol{\alpha}$
Initialize $\boldsymbol{H}^{p\times r} = 0$
  for $k = 1$ to l **do**
    $\boldsymbol{U}_r\boldsymbol{\Lambda}_r\boldsymbol{V}_r^T := SVD_r(\hat{\boldsymbol{L}}_{\boldsymbol{\alpha}})$
    $\boldsymbol{H}_{\boldsymbol{\alpha}} := \boldsymbol{U}_r\boldsymbol{\Lambda}_r^{1/2}$
  end for
$\hat{\boldsymbol{L}} := \boldsymbol{H} \times \boldsymbol{H}^T$
Outputs: $\{\hat{\boldsymbol{S}}\}_{i=1}^{l}$, $\hat{\boldsymbol{L}}$

We apply SLICE to multilayer networks through a block-coordinate approach, using it independently for each layer. The estimated $\hat{\boldsymbol{L}}_{\boldsymbol{\alpha}}$ for each layer is then used for the matrix completion step in the second part of the multiSLICE algorithm. The complete multiSLICE method is detailed in Algorithm 2. For the matrix completion portion, Algorithm 2 first computes per-layer SVDs, $\boldsymbol{L}_i = \boldsymbol{U}_i \boldsymbol{\Lambda}_i \boldsymbol{U}_i^T, \quad i = 1, \ldots, l$, then intersects each pair of latent subspaces by forming $\boldsymbol{L}_{ij} = \boldsymbol{U}_i \boldsymbol{\Lambda}_i^{1/2} \boldsymbol{\Lambda}_j^{1/2} \boldsymbol{U}_j^T$, and finally assembles all blocks $\{\boldsymbol{L}_{ij}\}$ in the global low-rank matrix, $\boldsymbol{L}$. As an illustrative example, we can consider a two-layer case, where we reconstruct the interlayer latent adjacency matrix $\boldsymbol{L}_{12} = \boldsymbol{U}_1 \boldsymbol{\Lambda}_1^{1/2} \boldsymbol{\Lambda}_2^{1/2} \boldsymbol{U}_2^T$ before assembling them into $\boldsymbol{L}$.

Given assumptions for exact recovery in Theorem 5.1, the rank of each submatrix $\boldsymbol{L}^*_{\boldsymbol{\alpha}}$ must equal that of the overall $\boldsymbol{L}^*$ across all $\boldsymbol{\alpha}$. From Remark 2, we know that depending on $p_{\boldsymbol{\alpha}}$ and $n_{\boldsymbol{\alpha}}$ in each layer, different values of ρ can be specified for each independent SLICE model. To select ρ and r, we suggest a k-fold cross-validation over a grid, where the combined values are based on log-likelihood. Simulation experiments elucidating the sensitivity of multiSLICE to these choices are provided in the Supplementary Materials.

5 Theoretical analysis

We establish consistency conditions for the sparse components and the low-rank joint latent component for our multiSLICE approach. Our proof considers each independent SLICE and matrix completion step separately. Our objective is to show the conditions under which we can exactly recover $\boldsymbol{L}^*$ as the sample size in each layer, $n_{\boldsymbol{\alpha}} \to \infty$. To recover the full $\boldsymbol{L}^*$, we must *exactly* recover $\boldsymbol{L}^*$ from only the observed entries denoted by $\mathcal{P}_\Omega(\boldsymbol{L}^*)$. To do so, we use a result from Liu et al. (2017), that describe the general conditions under which exact matrix completion is possible. More formally, Liu et al. (2017) describe the conditions where $\mathcal{P}_\Omega^{-1}$ exists, which leads us to our first result in Theorem 5.1.

Theorem 5.1 (Recovery of $\boldsymbol{L}^*$)**.** *Let $\boldsymbol{L}^* \succcurlyeq 0$ with rows and columns indexed by $\omega = \{1, \ldots, p\}$. Let $\Omega = \cup_{\boldsymbol{\alpha}=1}^{l} \Omega_{\boldsymbol{\alpha}}$, where $\Omega_{\boldsymbol{\alpha}} = \omega_{\boldsymbol{\alpha}} \times \omega_{\boldsymbol{\alpha}}$ and $\omega_{\boldsymbol{\alpha}} \subseteq \omega$. If $\mathcal{R}(\boldsymbol{L}^*_{\boldsymbol{\alpha}}) = \mathcal{R}(\boldsymbol{L}^*)\ \forall \boldsymbol{\alpha} = 1, \ldots, l$ and $|\cup_{\boldsymbol{\alpha}=1}^{l} \omega_{\boldsymbol{\alpha}}| \geq |\omega|$, then we have that $\mathcal{P}_\Omega(\boldsymbol{L}^*)$ is invertible.*

This result indicates that for exact recovery we only require that each submatrix formed by $\mathcal{P}_{\Omega_{\boldsymbol{\alpha}}}(\boldsymbol{L}^*)$ be of the same rank as $\boldsymbol{L}^*$. Next, we require standard assumptions for sub-Gaussianity, appropriate regularization, and bounds on the eigenvalues of $\boldsymbol{S}^*_{\boldsymbol{\alpha}}$ and $\boldsymbol{L}^*_{\boldsymbol{\alpha}}$ to recover $\boldsymbol{S}^*_{\boldsymbol{\alpha}}$ and $\boldsymbol{L}^*_{\boldsymbol{\alpha}}$ (Theorem 5.2).

Remark 3. *By Lemma 6.8 of Liu et al. (2017), the Ω/Ω^T-isomeric condition is equivalent to requiring the operators $\mathcal{P}_{U^*}\,\mathcal{P}_\Omega\,\mathcal{P}_{U^*}$ and $\mathcal{P}_{V^*}\,\mathcal{P}_\Omega\,\mathcal{P}_{V^*}$ be invertible. Hence, these invertibility conditions are exactly the identifiability requirements for the parameter matrix $\boldsymbol{L}^*$.*

Theorem 5.2 (multiSLICE joint $\{\hat{\boldsymbol{S}}\}_{\boldsymbol{\alpha}=1}^{l}$ and $\hat{\boldsymbol{L}}$ consistency)**.** *Let $\boldsymbol{L}^* \succcurlyeq 0$ with rows and columns indexed by $\omega = \{1, \ldots, p\}$. Let $\Omega = \cup_{\boldsymbol{\alpha}=1}^{l} \Omega_{\boldsymbol{\alpha}}$, where $\Omega_{\boldsymbol{\alpha}} = \omega_{\boldsymbol{\alpha}} \times \omega_{\boldsymbol{\alpha}}$ and $\omega_{\boldsymbol{\alpha}} \subseteq \omega$. Additionally, let $\{\boldsymbol{S}^*\}_{\boldsymbol{\alpha}=1}^{l}$ be the set of true sparse matrices. Then, we have $\hat{\boldsymbol{S}}_{\boldsymbol{\alpha}off} \xrightarrow{P} \boldsymbol{S}^*_{\boldsymbol{\alpha}off}$ and $\hat{\boldsymbol{L}} \xrightarrow{P} \boldsymbol{L}^*$ as $n_{\boldsymbol{\alpha}} \to \infty$, $\forall \boldsymbol{\alpha} = 1 \ldots l$.*

Given the independence of SLICE models, we show the model selection consistency of the sparse component in a similar manner. We require standard assumptions for the minimum signal strength of $\boldsymbol{S}^*$ and irrepresentability. This leads us to the result of Theorem 5.3.

Theorem 5.3 (Model Selection Consistency of $\hat{\boldsymbol{S}}_{\boldsymbol{\alpha}}$)**.** *For the multiSLICE estimator with L1 regularization for $\hat{\boldsymbol{S}}$ in layer $\boldsymbol{\alpha}$, we have*

$$\mathbb{P}\left(sign(\hat{S}^{\lambda}_{\boldsymbol{\alpha}ij}) = sign(S^*_{\boldsymbol{\alpha}ij}), \forall i, j \in \hat{\boldsymbol{S}}^{\lambda}_{\boldsymbol{\alpha}}\right) \geq 1 - \frac{1}{p^{\tau-2}} \to 1.$$

There are further considerations that arise when p and/or n vary between layers.

Remark 4. *We apply SLICE models independently, and the conditions for model selection consistency in each layer for the value of ρ may be different if the number of observed variables in the layer p and/or the sample size n is different. In such cases, the bounds for $\|\hat{\boldsymbol{S}}_{\boldsymbol{\alpha}off} - \boldsymbol{S}^*_{\boldsymbol{\alpha}off}\|_F, \|\hat{\boldsymbol{L}}_{\boldsymbol{\alpha}} - \boldsymbol{L}^*_{\boldsymbol{\alpha}}\|_\infty$ are also different. Layers with lower $p_{\boldsymbol{\alpha}}$ and higher $n_{\boldsymbol{\alpha}}$ therefore require less regularization (through ρ) and have tighter bounds on $\hat{\boldsymbol{S}}$ and $\hat{\boldsymbol{L}}$.*

Theorem 5.2 establishes the consistency of multiSLICE. However, to establish a rate, we require a further assumption. In particular, we assume that $\|[\hat{\boldsymbol{L}} - \boldsymbol{L}^*]_{\setminus\boldsymbol{\alpha},\boldsymbol{\alpha}}\|_\infty = \mathcal{O}(\|\hat{\boldsymbol{L}}_{\boldsymbol{\alpha}} - \boldsymbol{L}^*_{\boldsymbol{\alpha}}\|)$, where for some layer $\boldsymbol{\alpha}$, the rate of the unobserved portions is the same as the rate of the observed portions, up to a constant. With this, we can prove the result in Theorem 5.4.

Theorem 5.4 (Rate for joint low-rank, latent space)**.** *Let $\boldsymbol{L}^* \succcurlyeq 0$ with rows and columns indexed by $\omega = \{1, \ldots, p\}$. Let $\Omega = \cup_{\boldsymbol{\alpha}=1}^{l} \Omega_{\boldsymbol{\alpha}}$, where $\Omega_{\boldsymbol{\alpha}} = \omega_{\boldsymbol{\alpha}} \times \omega_{\boldsymbol{\alpha}}$ and $\omega_{\boldsymbol{\alpha}} \subseteq \omega$. If $\|[\hat{\boldsymbol{L}} - \boldsymbol{L}^*]_{\setminus\boldsymbol{\alpha},\boldsymbol{\alpha}}\|_\infty = \mathcal{O}(\|\hat{\boldsymbol{L}}_{\boldsymbol{\alpha}} - \boldsymbol{L}^*_{\boldsymbol{\alpha}}\|_\infty)$, then we have that*

$$\|\hat{\boldsymbol{L}} - \boldsymbol{L}^*\|_\infty \lesssim \sum_{\boldsymbol{\alpha}=1}^{l} C\sqrt{\frac{\log p_{\boldsymbol{\alpha}}}{n_{\boldsymbol{\alpha}}}} \quad \textit{for some constant } C.$$

Remark 5. *From Theorem 5.4, we have that the ∞-norm of the joint space is bounded by the sum of the layerwise L_∞ norms. For a two-layer system, the following is implied: The overall bounds on $\hat{\boldsymbol{L}}$ improve as we increase the sample size from one domain, $\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$, even while holding the other number of samples constant. However, we can only recover $\boldsymbol{L}^*$ exactly when we have exactly recovered $\boldsymbol{L}^*_{\boldsymbol{\alpha}}$ and $\boldsymbol{L}^*_{\boldsymbol{\beta}}$.*

This two-layer intuition can be extended to a network of any arbitrary size by considering any pair of layers $\boldsymbol{\alpha}, \boldsymbol{\beta} \in 1, \ldots, l$. These phenomena are made evident in the simulation studies in Section 6.

6 Simulated data study

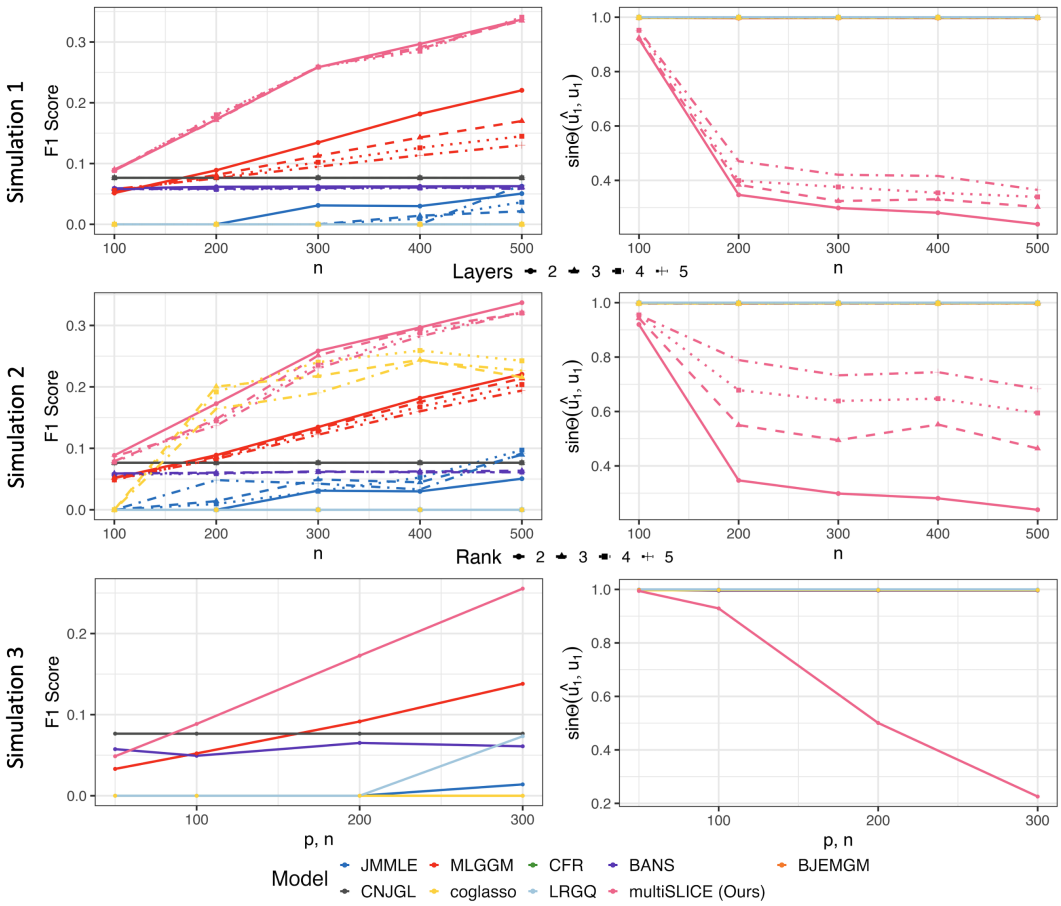

Figure 3: The effects of changing l, the number of layers (simulation 1) and $\mathcal{R}(\boldsymbol{L}^*)$, the rank of the latent parameter matrix (simulation 2), and scaling p and n at the same rate (simulation 3). Each simulation has 100 iterations.

We consider two challenging simulation studies, which are inspired by structures in neuroimaging (Yatsenko et al., 2015). For each simulation, we generate $\boldsymbol{S}^*_{\boldsymbol{\alpha}}$ and $\boldsymbol{L}^*$. To generate the sparse component for the $\boldsymbol{\alpha}$th layer, $\boldsymbol{S}^*_{\boldsymbol{\alpha}}$, we first use an initial value for the main diagonal, denoted by β and a decay rate, denoted by ζ. The ijth element is defined as

$$S^*_{\boldsymbol{\alpha}ij} = \begin{cases} \beta e^{\zeta|i-j|} & \text{if } \beta e^{\zeta|i-j|} \geq \psi, \\ 0 & \text{if } \beta e^{\zeta|i-j|} < \psi. \end{cases}$$

We then permute $\boldsymbol{S}^*_{\boldsymbol{\alpha}}$ over rows and columns to randomize the structure. We set $\beta = 1.5, \zeta = 2, \psi = 0.01,$ and $p_{\boldsymbol{\alpha}} = 100$ for all simulations. To generate $\boldsymbol{L}^*$, we construct a $p \times r$ binary matrix Z by, for each row i, selecting one of the r columns uniformly at random and setting $Z_{ij} = 1$, with all other entries in row i equal to zero. Then, we obtain $\boldsymbol{L}^* \leftarrow \beta \times \boldsymbol{Z}\boldsymbol{Z}^T$. For simulation 1, we fix $\mathcal{R}(\boldsymbol{L}^*) = 2$ and vary l; for simulation 2, we fix $l = 2$ and vary $\mathcal{R}(\boldsymbol{L}^*)$; for simulation 3, we fix $\mathcal{R}(\boldsymbol{L}^*) = 2$ and $l = 2$, and vary p and n jointly. We evaluate estimates of $\hat{\boldsymbol{S}}$ using the F1 score for sparsity, which is typical in the sparse GGM literature (Wang and Allen, 2023). We evaluate $\hat{\boldsymbol{L}}$ using the angle between the first eigenvector $\hat{\boldsymbol{L}}$ and $\boldsymbol{L}^*$, denoted by $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}^*_1)$, which is a natural choice for comparing the estimated and true low-rank parameters (Athreya et al., 2018).

Figure 4: Recovery of $\boldsymbol{L}^*$ with $l = 2, \mathcal{R}(\boldsymbol{L}^*) = 2, n = 500$ for methods with non-zero estimates.

Figure 3 shows the results of our simulation study. Across all simulations, we find that multiSLICE outperforms all other methods in the F1 score, indicating improved estimation of $\boldsymbol{S}^*$. In addition, it is the only method that converges toward the true $\boldsymbol{L}^*$, as indicated by the decreasing $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}^*_1)$ over increasing n. Also, for multiSLICE, the number of layers does not appear to greatly affect the F1 score and $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}^*_1)$, while $\mathcal{R}(\boldsymbol{L}^*)$ has a stronger influence. In Simulation 1 and 2, for all methods, increasing l or $\mathcal{R}(\boldsymbol{L}^*)$ has a negative impact on performance. From the related methods, we find that MLGGM has the second-best performance in simulation 1, and coglasso has the second-best performance in simulation 2. There is a vast difference in coglasso's performance between Simulations 1 and 2, suggesting that it is more sensitive to the number of layers than $\mathcal{R}(\boldsymbol{L}^*)$. Figure 4 shows the recovered latent parameter matrices.

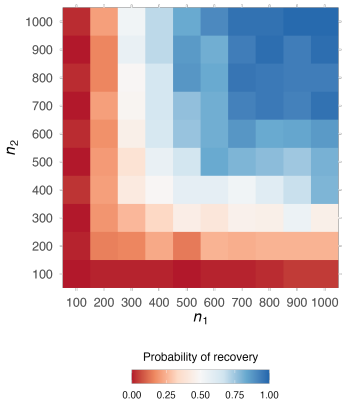

Figure 5: $\boldsymbol{L}$ recovery for different layer sample sizes n_1, n_2 with constant $l = 2, \mathcal{R}(\boldsymbol{L}^*) = 2$.

Figure 5 shows the results of simulating a two-layer system with $\mathcal{R}(\boldsymbol{L}^*) = 2$, with varying n_1 and n_2, and applying multiSLICE. Our results suggest that increasing the sample size in one layer improves recovery of the joint latent structure as n_1 or n_2 increases.

Lastly, we conduct simulation studies following Wainwright (2009) to validate our theoretical work. In all 100 iterations, we simulate a two-layer system, apply multiSLICE, then compute the relevant error norm, and declare recovery successful if that norm falls below a constant, as predicted by Lemma 9.1. In Figure 6, we plot the empirical success probability against the raw sample size $n_{\boldsymbol{\alpha}}$, and scaled sample size, defined as

$$n'_{\boldsymbol{S}} = \frac{n_{\boldsymbol{\alpha}}}{s_{\boldsymbol{\alpha}} \log p_{\boldsymbol{\alpha}}}, \qquad n'_{\boldsymbol{L}} = \frac{n_{\boldsymbol{\alpha}}}{C_1 \log p_{\boldsymbol{\alpha}}},$$

where $s_{\boldsymbol{\alpha}}$ and $p_{\boldsymbol{\alpha}}$ are the number of nonzeros and variables in modality $\boldsymbol{\alpha}$, respectively, and C_1 is defined in Lemma 9.1. We find that by scaling the sample sizes by the appropriate factors, the probability of recovery curves collapse toward a common transition point as predicted by our theory. Details of hyperparameter selection in all simulations and additional information are in the Supplementary Materials.

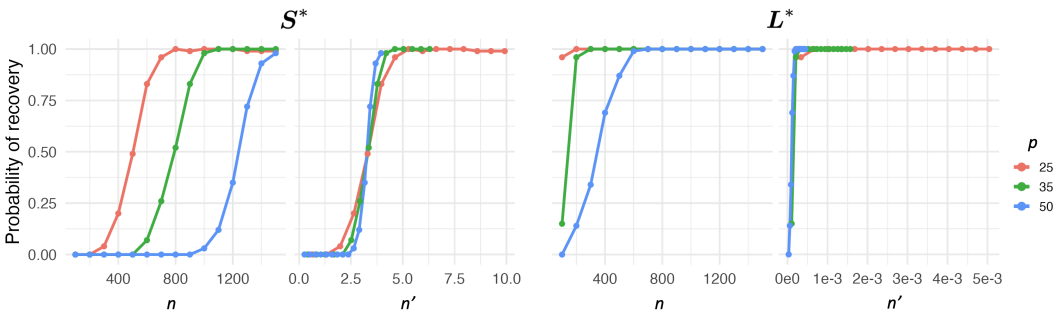

Figure 6: The probability of successful recovery of $\boldsymbol{S}^*$ (left panel) and $\boldsymbol{L}^*$ (right panel). In each panel, the left subplot x-axis uses the sample size n, whereas the right x-axis uses the scaled sample sizes for $\boldsymbol{S}^*$ and $\boldsymbol{L}^*$, respectively.

7 Multimodal neuroimaging data study

We apply multiSLICE and competitor methods to a multimodal neuroimaging dataset from Wakeman and Henson (2015). In this dataset, 16 subjects are scanned during the presentation of three different facial stimuli. "Famous" and "Unfamiliar" faces are those of people who are publicly well known and those of people who are not, respectively. The "Scrambled" group is a set of images that have the outline or general shape of a face but are filled in with white noise; these images serve as a control stimulus. Each subject has data related to structure (sMRI) and function (MEG, fMRI). Pre-processing steps and further details are in the Supplementary Materials, and all experiments are run on a M1 MacBook Pro with 16GB of RAM using R 4.4.3.

Each modality $\boldsymbol{\alpha}$ has different $p_{\boldsymbol{\alpha}}$ and $n_{\boldsymbol{\alpha}}$, which is typical in neuroimaging. We exclude methods that assume an equal sample size $n_{\boldsymbol{\alpha}}$ between layers (MLGGM, BANS, JMMLE, coglasso), as this assumption is violated in our dataset. Due to excessive run-time, CFR was omitted. For methods requiring the same $p_{\boldsymbol{\alpha}}$ across modalities (CNJGL, BJEMGM), we apply SVD to project the data into the smallest joint subspace, corresponding to MEG ($p_{\text{MEG}} = 52$). We then apply CNJGL, BJEMGM, LRGQ, multiSLICE to estimate networks independently for each subject and stimulus.

	Famous		Unfamiliar		Scrambled	
	$Q(\hat{\boldsymbol{S}})$	$H(\hat{\boldsymbol{L}})$	$Q(\hat{\boldsymbol{S}})$	$H(\hat{\boldsymbol{L}})$	$Q(\hat{\boldsymbol{S}})$	$H(\hat{\boldsymbol{L}})$
CNJGL (2014)	0.107 (9.65e-03)	N/A	0.106 (1.01e-02)	N/A	0.104 (1.08e-02)	N/A
BJEMGM (2019)	0.084 (1.25e-02)	N/A	0.084 (9.09e-03)	N/A	0.079 (1.27e-02)	N/A
LRGQ (2022)	0.112 (1.21e-01)	1.29 (6.63e-02)	0.122 (1.18e-01)	1.29 (6.38e-02)	0.130 (1.07e-01)	1.28 (7.09e-02)
multiSLICE (ours)	**0.170 (1.86e-03)**	**0.626 (4.53e-02)**	**0.171 (1.64e-03)**	**0.631 (4.02e-02)**	**0.170 (1.28e-03)**	**0.660 (1.17e-01)**

Table 2: The estimated modularity of the sparse intralayer graphs, $Q(\hat{\boldsymbol{S}})$, and the multilayer von Neumann entropy for the latent supra adjacency matrix, $H(\hat{\boldsymbol{L}})$. Bold values are the best results across methods, highest for $Q(\hat{\boldsymbol{S}})$ and lowest for $H(\hat{\boldsymbol{L}})$, for each visual stimulus (Famous, Unfamiliar, and Scrambled). Entries show the mean (standard deviation) across the 16 subjects.

To compare estimates of $\hat{\boldsymbol{S}}$ from each method, we use modularity (Bullmore and Sporns, 2009), which measures how well communities are separated within the intralayer graphs. This is calculated using $Q(\hat{\boldsymbol{S}}) = \frac{1}{2m}\ \text{tr}\big(\boldsymbol{C}^T \boldsymbol{B}\, \boldsymbol{C}\big)$, where $\boldsymbol{B} = \hat{\boldsymbol{S}} - \frac{\mathbf{k}\mathbf{k}^T}{2m}$, where $\boldsymbol{C}$ is a matrix of community assignments, m is the total number of edges and $\mathbf{k}$ is a vector of node degrees. To compare estimates of $\hat{\boldsymbol{L}}$ from each method, we use the multilayer von Neumann entropy, H, (De Domenico et al., 2015), where lower values are preferred, indicating higher order in the network. The von Neumann entropy is

defined as $H(\hat{\boldsymbol{L}}) = -\sum_{i=1}^{p} \lambda_{\phi_i} \ln(\lambda_{\phi_i})$, where ϕ is $\hat{\boldsymbol{L}}$, normalized to a trace of 1, $\phi = \frac{\hat{\boldsymbol{L}}}{\text{tr}(\hat{\boldsymbol{L}})}$. Table 2 shows the results. Across all measures and face stimuli, multiSLICE has the best performance, supporting the efficacy of our method. LRGQ has the second-best performance across all measures, followed by CNJGL and BJEMGM, suggesting that the multiSLICE model is more realistic and yields more favorable estimates of both $\hat{\boldsymbol{S}}$ and $\hat{\boldsymbol{L}}$. Figure 7 shows the multiSLICE adjacency estimates for the different face stimulus conditions. Although they all share a low-rank structure, capturing the common task-related variance, each stimulus produces a unique pattern that highlights stimulus-specific multilayer networks.

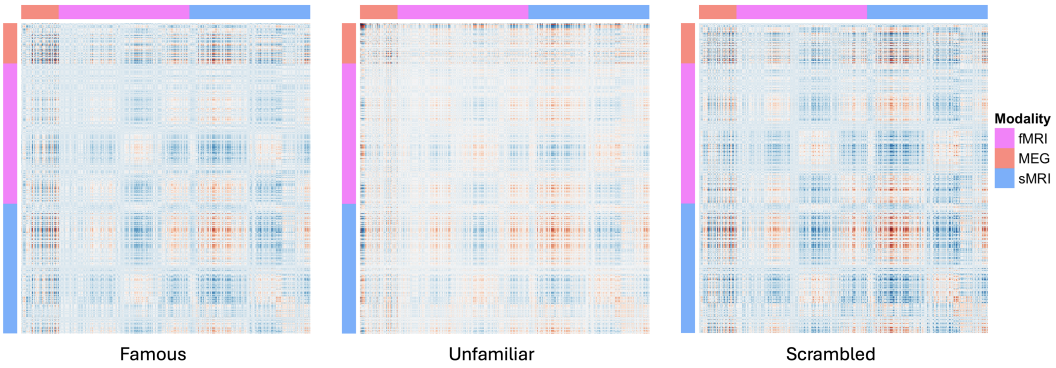

Figure 7: The estimated supra adjacency matrices, $\hat{\boldsymbol{L}}$, obtained from multiSLICE for each visual stimuli presented to subject 1 of the Wakeman and Henson (2015) dataset.

8 Discussion

We highlight three key regimes in which our model's guarantees may not hold in practice. The first is that if measurements from different modalities are gathered simultaneously, our assumption of independent modality measurements is violated. Second, any form of temporal or spatial correlation across observations breaches the i.i.d. requirement underpinning our per-modality covariance estimators. Lastly, while our sparse, joint-latent parameterization performs well for neuroimaging data, it remains untested in other application areas. As such, practitioners should validate its suitability before applying it to novel domains.

Future studies could adapt multiSLICE to capture sample-to-sample dependencies via matrix-variate extensions that regularize the row precision matrix in a joint manner. Time-varying and longitudinal domains are also noteworthy, as it is well known that brain imaging data can be non-stationary (Fox et al., 2005; Eichele et al., 2008; Doucet et al., 2012). For example, for the time-varying case, one could formulate the objective function to jointly estimate $\{\boldsymbol{L}^{(t)}\}_{t=1}^{T}$, with penalties enforcing smoothness or sparsity in time as in Hallac et al. (2017). For strictly repeated (longitudinal) measurements, one only needs to treat each time point as a separate "layer" in multiSLICE, ordering covariance inputs by time so that the interlayer estimates $\{\boldsymbol{L}_{t,t+1}\}$ capture the within-subject evolution directly.

We also consider extending multiSLICE into end-to-end deep-learning workflows by using its outputs, namely the sparse $\boldsymbol{S}$ and low-rank $\boldsymbol{L}$ estimates to define graph inputs for GNNs. For example, Do et al. (2023) showed that decoupling graph construction (via Graphical LASSO) from GNN training both accelerates convergence and improves performance, while Sriramulu et al. (2023)'s adaptive dependency learning graph neural networks (ADLNN) uses Graphical LASSO as a structural prior before refining edges within a GNN. In neuroimaging, Wang et al. (2021); Yu et al. (2022); Thapaliya et al. (2025) utilize graphs representations of regional brain activity for downstream prediction tasks. These single-modality methods underscore how graphical models yield effective, interpretable graphs from multivariate data; multiSLICE generalizes this to a multimodal setting, filling a clear gap.

The iterative steps of Algorithm 2 can be unrolled into a module and embedded directly into differentiable architectures. Alternatively, one can amortize the mapping from sample covariances to $(\boldsymbol{S}, \boldsymbol{L})$ via a small neural network, similar to Belilovsky et al. (2017), and feed its output into downstream GNNs. In the case of non-Gaussian data, such as raw images, audio, or video, multiSLICE could be used downstream of modality-specific feature extractors (e.g. a CNN).

Acknowledgements

We would like to sincerely thank the reviewers for their helpful feedback and comments throughout the submission, which have led to several improvements of the paper. The first author was supported by the H. Jean McDiarmaid scholarship provided by the Faculty of Medicine and Dentistry, University of Alberta. The second author was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant RGPIN-2024-06102. The third author was partially supported by the National Science Foundation (NSF) Grant DMS-2324489.

References

Alessio Albanese, Wouter Kohlen, and Pariya Behrouzi. Collaborative graphical lasso. *arXiv preprint arXiv:2403.18602*, 2024.

Ricard Argelaguet, Britta Velten, Damien Arnol, Sascha Dietrich, Thorsten Zenz, John C Marioni, Florian Buettner, Wolfgang Huber, and Oliver Stegle. Multi-omics factor analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular Systems Biology*, 14(6):e8124, 2018.

Avanti Athreya, Donniell E Fishkind, Minh Tang, Carey E Priebe, Youngser Park, Joshua T Vogelstein, Keith Levin, Vince Lyzinski, Yichen Qin, and Daniel L Sussman. Statistical inference on random dot product graphs: a survey. *Journal of Machine Learning Research*, 18(226):1–92, 2018.

Danielle S Bassett and Olaf Sporns. Network neuroscience. *Nature Neuroscience*, 20(3):353–364, 2017.

Danielle S Bassett, Nicholas F Wymbs, Mason A Porter, Peter J Mucha, Jean M Carlson, and Scott T Grafton. Dynamic reconfiguration of human brain networks during learning. *Proceedings of the National Academy of Sciences*, 108(18):7641–7646, 2011.

Eugene Belilovsky, Kyle Kastner, Gaël Varoquaux, and Matthew B Blaschko. Learning to discover sparse graphical models. In *International Conference on Machine Learning*, pages 440–448. PMLR, 2017.

William E Bishop and Byron M Yu. Deterministic symmetric positive semidefinite matrix completion. *Advances in Neural Information Processing Systems*, 27, 2014.

Ed Bullmore and Olaf Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3):186–198, 2009.

Tony Cai, Weidong Liu, and Xi Luo. A constrained l1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607, 2011.

Emmanuel Candes and Benjamin Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.

Venkat Chandrasekaran, Pablo A. Parrilo, and Alan S. Willsky. Latent variable graphical model selection via convex optimization. *The Annals of Statistics*, 40(4):1935 – 1967, 2012.

Andersen Chang, Lili Zheng, and Genevera I Allen. Low-rank covariance completion for graph quilting with applications to functional connectivity. *arXiv preprint arXiv:2209.08273*, 2022.

Paul Craven and Barry Wellman. The network city. *Sociological Inquiry*, 43(3-4):57–88, 1973.

Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora. Structural reducibility of multilayer networks. *Nature Communications*, 6(1):6864, 2015.

Ngoc-Dung Do, Truong Son Hy, and Duy Khuong Nguyen. Sparsity exploitation via discovering graphical models in multi-variate time-series forecasting. *arXiv preprint arXiv:2306.17090*, 2023.

Gaëlle Doucet, Mikaël Naveau, Laurent Petit, Laure Zago, Fabrice Crivello, Gaël Jobard, Nicolas Delcroix, Emmanuel Mellet, Nathalie Tzourio-Mazoyer, Bernard Mazoyer, and Others. Patterns of hemodynamic low-frequency oscillations in the brain are modulated by the nature of free thought during rest. *NeuroImage*, 59(4):3194–3200, 2012.

Tom Eichele, Stefan Debener, Vince D Calhoun, Karsten Specht, Andreas K Engel, Kenneth Hugdahl, D Yves von Cramon, and Markus Ullsperger. Prediction of human errors by maladaptive changes in event-related brain networks. *Proceedings of the National Academy of Sciences*, 105(16): 6173–6178, 2008.

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

Michael D Fox, Abraham Z Snyder, Justin L Vincent, Maurizio Corbetta, David C Van Essen, and Marcus E Raichle. The human brain is intrinsically organized into dynamic, anticorrelated functional networks. *Proceedings of the National Academy of Sciences*, 102(27):9673–9678, 2005.

Lingrui Gan, Xinming Yang, Naveen Narisetty, and Feng Liang. Bayesian joint estimation of multiple graphical models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Min Jin Ha, Francesco Claudio Stingo, and Veerabhadran Baladandayuthapani. Bayesian structure learning in multilayered genomic networks. *Journal of the American Statistical Association*, 116 (534):605–618, 2021.

David Hallac, Youngsuk Park, Stephen Boyd, and Jure Leskovec. Network inference via the time-varying graphical lasso. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 205–213, 2017.

Trevor Hastie, Robert Tibshirani, and Jerome H Friedman. *The Elements of Statistical Learning: Data mining, Inference, and Prediction*, volume 2. Springer, 2009.

Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P Gleeson, Yamir Moreno, and Mason A Porter. Multilayer networks. *Journal of Complex Networks*, 2(3):203–271, 2014.

Jiahe Lin, Sumanta Basu, Moulinath Banerjee, and George Michailidis. Penalized maximum likelihood estimation of multi-layered gaussian graphical models. *Journal of Machine Learning Research*, 17(146):1–51, 2016.

Guangcan Liu, Qingshan Liu, and Xiaotong Yuan. A new theory for matrix completion. *Advances in Neural Information Processing Systems*, 30, 2017.

Subhabrata Majumdar and George Michailidis. Joint estimation and inference for data integration problems based on multiple multi-layered gaussian graphical models. *Journal of Machine Learning Research*, 23(1):1–53, 2022.

Karthik Mohan, Palma London, Maryam Fazel, Daniela Witten, and Su-In Lee. Node-based learning of multiple gaussian graphical models. *Journal of Machine Learning Research*, 15(1):445–488, 2014.

Bradley S Price, Aaron J Molstad, and Ben Sherwood. Estimating multiple precision matrices with cluster fusion regularization. *Journal of Computational and Graphical Statistics*, 30(4):823–834, 2021.

Pradeep Ravikumar, Martin J. Wainwright, Garvesh Raskutti, and Bin Yu. High-dimensional covariance estimation by minimizing L1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935 – 980, 2011.

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

Souvik Seal, Qunhua Li, Elle Butler Basner, Laura M Saba, and Katerina Kechris. Rcfgl: Rapid condition adaptive fused graphical lasso and application to modeling brain region co-expression networks. *PLOS Computational Biology*, 19(1):e1010758, 2023.

Jakob Seidlitz, František Váša, Maxwell Shinn, Rafael Romero-Garcia, Kirstie J Whitaker, Petra E Vértes, Konrad Wagstyl, Paul Kirkpatrick Reardon, Liv Clasen, Siyuan Liu, et al. Morphometric similarity networks detect microscale cortical organization and predict inter-individual cognitive variation. *Neuron*, 97(1):231–247, 2018.

Abishek Sriramulu, Nicolas Fourrier, and Christoph Bergmeir. Adaptive dependency learning graph neural networks. *Information Sciences*, 625:700–714, 2023.

Bishal Thapaliya, Esra Akbas, Jiayu Chen, Ram Sapkota, Bhaskar Ray, Pranav Suresh, Vince D Calhoun, and Jingyu Liu. Brain networks and intelligence: A graph neural network based approach to resting state fMRI data. *Medical Image Analysis*, 101:103433, 2025.

Martin J Wainwright. Sharp thresholds for high-dimensional and noisy sparsity recovery using l1-constrained quadratic programming (lasso). *IEEE transactions on information theory*, 55(5): 2183–2202, 2009.

Martin J Wainwright, John Lafferty, and Pradeep Ravikumar. High-dimensional graphical model selection using l1-regularized logistic regression. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

Daniel G Wakeman and Richard N Henson. A multi-subject, multi-modal human neuroimaging dataset. *Scientific Data*, 2(1):1–10, 2015.

Lebo Wang, Kaiming Li, and Xiaoping P Hu. Graph convolutional network for fMRI analysis based on connectivity neighborhood. *Network Neuroscience*, 5(1):83–95, 2021.

Minjie Wang and Genevera I Allen. Thresholded graphical lasso adjusts for latent variables. *Biometrika*, 110(3):681–697, 2023.

Stanley Wasserman and Katherine Faust. *Social Network Analysis in the Social and Behavioral Sciences*, page 3–27. Structural Analysis in the Social Sciences. Cambridge University Press, 1994.

Pan Xu, Jian Ma, and Quanquan Gu. Speeding up latent variable gaussian graphical model estimation via nonconvex optimization. *Advances in Neural Information Processing Systems*, 30, 2017.

Dimitri Yatsenko, Krešimir Josić, Alexander S Ecker, Emmanouil Froudarakis, R James Cotton, and Andreas S Tolias. Improved estimation and interpretation of correlations in neural circuits. *PLOS Computational Biology*, 11(3):e1004083, 2015.

Yue Yu, Xuan Kan, Hejie Cui, Ran Xu, Yujia Zheng, Xiangchen Song, Yanqiao Zhu, Kun Zhang, Razieh Nabi, Ying Guo, et al. Learning task-aware effective brain connectivity for fMRI analysis with graph neural networks. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 4995–4996. IEEE, 2022.

Peng Zhao and Bin Yu. On model selection consistency of lasso. *Journal of Machine Learning Research*, 7(Nov):2541–2563, 2006.

Tuo Zhao, Han Liu, Kathryn Roeder, John Lafferty, and Larry Wasserman. The huge package for high-dimensional undirected graph estimation in r. *Journal of Machine Learning Research*, 13(1): 1059–1062, 2012.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our approach in the conclusion. Furthermore, in our theoretical analysis, formal assumptions are outlined, and therefore directly state the constraints of our modelling approach.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide assumptions and complete proofs for theoretical results in the Supplementary Materials.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all the steps taken for generating simulations, as well as preprocessing the open source neuroimaging data. We provide all code as an Github repository.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Github

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: For the real data analysis, we provide standard deviation in brackets next to the average values obtained, for each experiment.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experiments are run on consumer-grade hardware, and we include timing experiments for our proposed method in the Supplementary Material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Our method provides more accurate and robust estimates of brain connectivity compared to other state-of-the-art methods, which can accelerate fundamental neuroscience research, improve biomarkers for neurological and psychiatric disorders, and ultimately inform better diagnostics and therapies. However, there are potential negative impacts. For one, there is a risk of misinterpretation of estimates. Treating the edges of a correlational network as causal may prompt unsafe interventions. Another concern is privacy. High-resolution connectomes estimated via multilayer networks can, in principle, carry individual-specific signatures. Sharing or pooling data without adequate safeguards risks misuse of

participants' brain data. There are also risks in using this method in unintended ways, such as outside clinical or research contexts (e.g., surveillance of cognitive states). Lastly, there are considerations regarding fairness. If the method is applied to heterogeneous populations without proper care, estimates can systematically misrepresent under-studied groups (e.g., age, ethnicity), leading to biased conclusions.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We include appropriate citations for all code and data sets used for the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We include all code and experiments as an Github.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: Although we did not collect any human data as part of our neuroimaging experiments, we did use an open source data set which includes human subjects. Details are available in the Supplementary Materials.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

Justification: Although our real data experiments include human subjects, we were not involved in the data collection process at all, and are using the open source and de-identified version of the data set.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.