# Joint Semi-Symmetric Tensor PCA for Integrating Multi-modal Populations of Networks

Jiaming Liu[†1], Lili Zheng[†*2], Zhengwu Zhang[3], and Genevera I. Allen[1,2,4]

[1]*Department of Statistics, Rice University*
[2]*Department of Electrical and Computer Engineering, Rice University*
[3]*Department of Statistics and Operations Research, University of North Carolina at Chapel Hill*
[4]*Neurological Research Institue, Baylor College of Medicine*

**Abstract**

Multi-modal populations of networks arise in many scenarios including in large-scale multi-modal neuroimaging studies that capture both functional and structural neuroimaging data for thousands of subjects. A major research question in such studies is how functional and structural brain connectivity are related and how they vary across the population. we develop a novel PCA-type framework for integrating multi-modal undirected networks measured on many subjects. Specifically, we arrange these networks as semi-symmetric tensors, where each tensor slice is a symmetric matrix representing a network from an individual subject. We then propose a novel Joint, Integrative Semi-Symmetric Tensor PCA (JisstPCA) model, associated with an efficient iterative algorithm, for jointly finding low-rank representations of two or more networks across the same population of subjects. We establish one-step statistical convergence of our separate low-rank network factors as well as the shared population factors to the true factors, with finite sample statistical error bounds. Through simulation studies and a real data example for integrating multi-subject functional and structural brain connectivity, we illustrate the advantages of our method for finding joint low-rank structures in multi-modal populations of networks.

*Keywords:* Tensor PCA, multi-modal network analysis, data integration, semi-symmetric tensor, brain connectivity, joint factorization

## 1 Introduction

Large-scale network data arises frequently from a wide range of biomedical and social science applications, such as brain connectomes (Yao et al., 2015; Rubinov and Sporns, 2010; Bullmore

*Corresponding author: lili.zheng@rice.edu
†: Equal contribution.

and Sporns, 2009), social networks (Hoff et al., 2002), and gene co-expression networks (Li et al., 2011). Many of these applications also have different types of networks measured on the same set of nodes from the same subject, called multi-modal networks. For example in neuroimaging, we may have functional connectivity derived from functional magnetic resonance imaging (fMRI) data and structural connectivity derived from diffusion MRI (dMRI) data measured for the same subject (Yao et al., 2015; Cole et al., 2021). We can have a large population of these multi-modal brain networks; the Human Connectome Project is an example of one such multi-modal population study (Van Essen et al., 2013). Although there are many developed techniques for analyzing multi-modal or multi-view networks (Han et al., 2015; D'Angelo et al., 2019; Gao et al., 2022) and separately populations of networks (Paul and Chen, 2020a; MacDonald et al., 2022), there is limited work that can analyze both aspects simultaneously. Some existing works, e.g. Murden et al. (2022), vectorize networks into vectors and ignore intrinsic structures in networks. This raises the question: can we jointly analyze and perform dimension reduction for multi-modal populations of networks? Such a joint analysis has many benefits, as it allows researchers to extract shared structures and relationships between different network modalities, reflect commonalities and variations amongst subjects, detect outliers, and identify clustering patterns within the population. In the example of multi-modal population neuroimaging studies, this approach can discover connectivity patterns shared between function and structure across many subjects, paving the way to demystifying how the brain works. Moreover, such analysis can highlight how these connections vary across different populations and how they relate to demographic or genomic traits. Motivated by these applications, we propose to structure multi-modal populations of networks as tensors and develop a novel dimension reduction approach: Joint-Integrative Semi-Symmetric Tensor PCA (JisstPCA).

### 1.1 Joint, Integrative Semi-Symmetric Tensor PCA

Tensors are natural tools for modeling a collection of networks (Wu et al., 2019; Jing et al., 2021; Zhang et al., 2020b), as one can stack the matrix representation of networks along an extra mode. We specifically take inspiration from recent works (Weylandt and Michailidis, 2022; Zhang et al., 2019) on semi-symmetric tensor modeling of populations of networks, where each slice of the tensor is a symmetric positive semi-definite matrix representing a network, e.g. an adjacency or Laplacian matrix for undirected networks. This modeling framework and its associated tensor PCA algorithms (Weylandt and Michailidis, 2022; Zhang et al., 2019) can help extract principal network factors across the whole population and achieve simultaneous dimension reduction for both the networks and the population. For our goal of analyzing multi-modal populations of networks, we consider the natural idea of integrating multiple semi-symmetric tensors.

For simplicity, we focus on the case with two modalities of networks, while it is straightforward to extend our framework and algorithms to more general cases. We can arrange the two modalities of networks into tensors $\mathcal{X} \in \mathbb{R}^{p\times p\times N}$ and $\mathcal{Y} \in \mathbb{R}^{q\times q\times N}$, where $p$, $q$ are the network dimensions, and $N$ is the sample size. We allow the network sizes to be different ($p \neq q$) to account for possibly varying resolutions across modalities. In order to capture the shared low-rank structures from the

two tensors, we consider the following joint, integrative semi-symmetric tensor PCA (JisstPCA) model:

$$\mathcal{X} = \sum_{k=1}^{K} d^*_{x,k} \cdot \boldsymbol{V}^*_k \boldsymbol{V}^{*\prime}_k \circ \boldsymbol{u}^*_k + \mathcal{E}_x, \quad \mathcal{Y} = \sum_{k=1}^{K} d^*_{y,k} \cdot \boldsymbol{W}^*_k \boldsymbol{W}^{*\prime}_k \circ \boldsymbol{u}^*_k + \mathcal{E}_y. \tag{1}$$

Here, both tensors are decomposed into $K$ network factors plus observational noise: $\mathcal{E}_x \in \mathbb{R}^{p\times p\times N}$, $\mathcal{E}_y \in \mathbb{R}^{q\times q\times N}$ are random zero-mean semi-symmetric noise tensors. Note that we do not assume $\mathcal{X}$ and $\mathcal{Y}$ to be binary so that our approach generalizes to weighted networks. The orthogonal matrices $\boldsymbol{V}^*_k \in \mathcal{O}_{p,r_{x,k}}$, $\boldsymbol{W}^*_k \in \mathcal{O}_{q,r_{y,k}}$ are the $k$th principal network factors for the two modalities respectively, and represent the major network patterns. The unit vector $\boldsymbol{u}^*_k \in \mathbb{S}^{N-1}$ is the joint population factor, indicating the weight of the $k$th network factor for each subject; this is similar to the sample loading in classical PCA and represents population-level patterns affiliated with each network pattern. One can view $(\boldsymbol{V}^*_k \boldsymbol{V}^{*\top}_k, \boldsymbol{W}^*_k \boldsymbol{W}^{*\top}_k)$ as a pair of network prototypes for the two modalities, and for each subject $i \in [N]$, the corresponding networks $\mathcal{X}_{:,:,i}$, $\mathcal{Y}_{:,:,i}$ are noisy realizations of a mixture of the $K$ prototypes with weights $(\boldsymbol{u}^*_{1,i}, \ldots, \boldsymbol{u}^*_{K,i})$. To ensure identifiability, we assume the factors across different layers are linearly independent*. Our goal is to *find the ground truth factors* $\boldsymbol{V}^*_k$, $\boldsymbol{W}^*_k$, $\boldsymbol{u}^*_k$ *for* $k \in [K]$ *from noisy network data* $\mathcal{X}$ *and* $\mathcal{Y}$. Our main contributions and the paper organization are summarized as follows.

**Main contributions:** We propose the first framework for simultaneously analyzing multi-modal populations of networks: the joint, integrative semi-symmetric tensor PCA (JisstPCA) model (1), with associated algorithms for extracting the population and network factors. Our JisstPCA algorithm, introduced in Section 2, consists of efficient joint power iteration and sequential deflation schemes. We additionally propose and study several important extensions of our basic JisstPCA model (1), including integration of networks and vector-valued covariates, and a more general model with multiple eigenvalues associated with each factor. In Section 3, we prove the first statistical theory for integrative tensor PCA; under the single-factor case, we establish one-step convergence for both individual and joint factors to a small neighborhood of the true factors with corresponding statistical errors under provable initialization conditions. Our theoretical guarantees improve upon or are comparable to prior works for single tensor PCA (Weylandt and Michailidis, 2022; Zhang and Xia, 2018). Furthermore, we validate our algorithms via empirical studies in Section 4 and demonstrate its superior performance compared to baseline methods. The real data study in Section 5 showcases how JisstPCA can derive insightful connections between functional and structural networks in human brain and detect outliers in the population.

### 1.2 Related Works

Here, we discuss prior works that most closely relate to ours; a more detailed literature review is included in the Appendix. There has been an extensive literature on tensor decompositions and tensor PCA (Kolda and Bader, 2009), covering various tensor low-rank structures including the CP

*$\text{rank}\left(\boldsymbol{V}^*_i, \boldsymbol{V}^*_j\right) = r_{x,i} + r_{x,j}, \text{rank}\left(\boldsymbol{W}^*_i, \boldsymbol{W}^*_j\right) = r_{y,i} + r_{y,j}$ and $\text{rank}\left(\boldsymbol{u}^*_i, \boldsymbol{u}^*_j\right) = 2$ for $i \neq j = 1, \cdots, K$.

decomposition (Carroll and Chang, 1970; Anandkumar et al., 2014; Han and Zhang, 2022), Tucker decomposition (De Lathauwer et al., 2000b; Zhang and Xia, 2018; Luo et al., 2021), and tensor-train low-rank structures (Zhou et al., 2022); many of these recent works have also established statistical consistency results. Our JisstPCA framework (1), however, is built upon the prior literature on semi-symmetric tensor PCA (Weylandt and Michailidis, 2022; Zhang et al., 2019; Winter et al., 2020) for analyzing populations of networks. Amongst these, our basic model (1) is most similar to, and reduces to the semi-symmetric tensor PCA model in (Weylandt and Michailidis, 2022) when focusing on one modality of networks. Different from our work and Weylandt and Michailidis (2022), however, Zhang et al. (2019); Winter et al. (2020); Wang et al. (2014) utilize the CP decomposition, a sum of rank-1 tensors, to model populations of networks. Although the CP decomposition is widely studied and there also exists joint factorization methods for integrating multiple CP low-rank tensors (Acar et al., 2011, 2014; Wu et al., 2018; Schenker et al., 2020; Lu et al., 2020; Farias et al., 2016; Fu et al., 2015), it is not the most appropriate model for our analysis of multi-modal populations of networks. First, rank-one factors cannot sufficiently capture complex network patterns, hence limiting the expressivity of the model. Second, standard CP decompositions cannot enforce symmetry in the factors. Finally, although one can write our model as a special CP decomposition with the same $\boldsymbol{u}_k^*$ factors repeated $r_k$ times, existing joint CP decomposition algorithms are inapplicable of capturing this model as they often require an angle /incoherence condition between any pair of the factors in all modes (Anandkumar et al., 2014).

Our model (1) is also closely related to the Tucker low-rank model. When constraining all factors $\boldsymbol{V}_1, \ldots, \boldsymbol{V}_K$ and $\boldsymbol{W}_1, \ldots, \boldsymbol{W}_K$ in (1) to be mutually orthogonal, our model is a special case of the Tucker model with shared factors on the third mode across modalities. However, the mutual orthogonality constraints in Tucker models can be especially restrictive for network factors, again limiting the expressivity of the model. The Tucker core also complicates the one-to-one correspondence between the sample factor $\boldsymbol{u}_k$ and the network factors $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$, which sacrifices interpretability.

Further, our work builds upon the extensive data integration literature. Most existing data integration methods focus on tabular data that can be arranged as matrices, including the JIVE (Lock et al., 2013), the iPCA (Tang and Allen, 2021), the multi-block PCA family (Abdi et al., 2013; Westerhuis et al., 1998), and many others. Some other works (Acar et al., 2011, 2014; Wu et al., 2018; Schenker et al., 2020) consider joint factorizations for tensors and tensor-matrix integration, but are based on the CP decomposition and are less appropriate for joint network analysis as discussed earlier. Finally, to the best of our knowledge, our work also provides the first theoretical guarantees for integrative tensor PCA.

## 2 JisstPCA Algorithms

In this section, we introduce our JisstPCA algorithm for finding the tensor factors in (1) and some extensions of it. We begin with the introduction of some notations.

### 2.1 Notation and Preliminary Tensor Algebra

We first introduce notation that will be used frequently in this paper; we closely follow Kolda and Bader (2009)'s notation for tensor algebra and refer the reader here for further details. We denote tensors as $\mathcal{X}$, matrices as $\mathbf{X}$, vectors as $\mathbf{x}$, and scalars as $x$. Matricization of $\mathcal{X}$ along the $k^{th}$ mode is denoted as $\mathcal{M}_k(\mathcal{X})$; multiplication of $\mathcal{X}$ with a matrix along the first mode is denoted as $\times_1$ with $\times_2$ and $\times_3$ defined similarly; $\langle \mathcal{X}, \mathcal{Y} \rangle$ denotes the inner (trace) product; $\circ$ denotes the outer product; for $\mathcal{X} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$, $\|\mathcal{X}\|_F$ is the tensor Frobenius norm and $\|\mathcal{X}\|_{\text{op}} = \sup_{u_1 \in \mathbb{S}^{p_1-1}, \ldots, u_d \in \mathbb{S}^{p_d-1}} \mathcal{X} \times_1 u_1 \times_2 \cdots \times_d u_d$ is the tensor operator norm. We similarly use $\|\mathbf{X}\|_{\text{op}} = \sup_{u:\|u\|_2=1} \|\mathbf{X}u\|_2$ to denote the matrix operator norm. For a tensor $\mathcal{X} \in \mathbb{R}^{p \times p \times N}$, we say it is semi-symmetric if its $N$ slices are all $p \times p$ symmetric matrices; the trace product of $\mathcal{X}$ and a matrix $\boldsymbol{V} \in \mathbb{R}^{p \times r}$ is denoted by $[\mathcal{X}; \boldsymbol{V}] \in \mathbb{R}^N$, whose $k^{th}$ element is $\langle \mathcal{X}_{:,:,k}, \boldsymbol{V}\boldsymbol{V}' \rangle = \text{Tr}(\boldsymbol{V}'\mathcal{X}_{:,:,k}\boldsymbol{V})$. Similarly, for a diagonal matrix $\boldsymbol{D} \in \mathbb{R}^{r \times r}$, we let $[\mathcal{X}; \boldsymbol{V}, \mathbf{D}] \in \mathbb{R}^N$ denote a vector whose $k^{th}$ element is $\langle \mathcal{X}_{..k}, \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}' \rangle$. Additionally, we let $\sin\theta(\mathbf{u}_*, \mathbf{u}) = \sin(\arccos \mathbf{u}_*'\mathbf{u})$ quantify the distance between two unit vectors and $\sin\Theta(\boldsymbol{V}_*, \boldsymbol{V}) = \text{diag}\{\sin(\arccos\sigma_1), \ldots, \sin(\arccos\sigma_r)\} \in \mathbb{R}^r$ quantify the distance between $\boldsymbol{V}_*, \boldsymbol{V} \in \mathbb{R}^{p \times r}$ and $\sigma_1 \geq \ldots \geq \sigma_r \geq 0$ are the singular values of $\boldsymbol{V}_*'\boldsymbol{V}$. Finally, we let $\mathbb{S}^{N-1} = \{\boldsymbol{u} \in \mathbb{R}^N : \|\boldsymbol{u}\|_2 = 1\}$, $\mathcal{O}_{p,r} = \{\boldsymbol{V} \in \mathbb{R}^{p \times r} : \boldsymbol{V}'\boldsymbol{V} = \boldsymbol{I}_r\}$, and $\mathcal{D}_r$ the set of diagonal matrices of size $r \times r$. More detailed notations can be found in the Appendix.

### 2.2 JisstPCA Algorithm

The goal of JisstPCA is to estimate both the population factor $\boldsymbol{u}_k^*$ and network factors $\boldsymbol{V}_k^*, \boldsymbol{W}_k^*$ in (1) simultaneously. Inspired by the multi-block PCA literature (Westerhuis et al., 1998; Abdi et al., 2013) where different data tables are appropriately scaled before a joint PCA, one may consider the following weighted objective:

$$\begin{aligned} \underset{\boldsymbol{u}_k, \boldsymbol{V}_k, \boldsymbol{W}_k, d_{x,k}, d_{y,k}}{\arg\min} \; & \lambda \left\| \mathcal{X} - \sum_{k=1}^{K} d_{x,k} \cdot \boldsymbol{V}_k \boldsymbol{V}_k' \circ \boldsymbol{u}_k \right\|_F^2 + (1-\lambda) \left\| \mathcal{Y} - \sum_{k=1}^{K} d_{y,k} \cdot \boldsymbol{W}_k \boldsymbol{W}_k' \circ \boldsymbol{u}_k \right\|_F^2 \\ & \text{s.t. } \boldsymbol{V}_k \in \mathcal{O}_{p,r_{x,k}}, \; \boldsymbol{W}_k \in \mathcal{O}_{q,r_{y,k}}, \; \boldsymbol{u}_k \in \mathbb{S}^{N-1}. \end{aligned} \tag{2}$$

Here, the weight parameter $\lambda \in [0, 1]$ is analogous to the scaling parameter in multiblock PCA; and it can reflect our knowledge about which data set is more important or more trustworthy. If we let $\lambda = 0$ or 1, the problem degenerates to non-integrated semi-symmetric tensor factorization studied in Weylandt and Michailidis (2022). We discuss more about the selection of $\lambda$ in Section A.4.2 of the Appendix.

#### 2.2.1 Single-Factor JisstPCA

We start from the single-factor ($K = 1$) case and will then extend it to $K > 1$. The subscript $k$ will be omitted when we discuss the single-factor model. Inspired by the success of power iteration in the tensor PCA literature (Zhang and Xia, 2018; Weylandt and Michailidis, 2022) and the weighted objective (2), we propose a integrated power iteration algorithm that incorporates the

weight parameter $\lambda$ into the updates. In particular, given the prior update of the joint factor $\boldsymbol{u}$, we update $\boldsymbol{V}$ and $\boldsymbol{W}$ as the top eigenvectors of $\mathcal{X} \times_3 \boldsymbol{u}$ and $\mathcal{Y} \times_3 \boldsymbol{u}$; given $\boldsymbol{V}$, $\boldsymbol{W}$, $\boldsymbol{u}$ can be updated by pooling together the trace products $[\mathcal{X}; \boldsymbol{V}^{(t+1)}]$ and $[\mathcal{Y}, \boldsymbol{W}^{(t+1)}]$ defined in Section 2.1, with the weight parameter $\lambda$. The detailed procedure is summarized in Algorithm 1. The main difference between Algorithm 1 and the SS-TPCA algorithm in Weylandt and Michailidis (2022) lies in the update of $\boldsymbol{u}$ that utilizes weighted power iterates from both tensors. Note that we do not claim to solve the optimization problem (2), but instead, we will show the statistical convergence properties of our factor updates with high probability when the data is sampled from model (1) in Section 3. For the initialization $\boldsymbol{u}^{(0)}$, Weylandt and Michailidis (2022) suggests using the warm initialization

**Algorithm 1:** Single-factor JisstPCA

- Input: $\mathcal{X}$, $\mathcal{Y}$, $r_x$, $r_y$, $\lambda$, and maximum iteration $t_{\max}$.
- Initialization: Let $t = 0$, and $\boldsymbol{u}^{(0)} =$ Leading singular vector of $[\lambda \mathcal{M}_3(\mathcal{X}), (1-\lambda)\mathcal{M}_3(\mathcal{Y})]$.
- **repeat** until $t = t_{\max}$ or convergence:

$$\begin{aligned}
&\boldsymbol{V}^{(t+1)} = \text{Leading } r_x \text{ singular vectors of } \mathcal{X} \times_3 \boldsymbol{u}^{(t)} \\
&\boldsymbol{W}^{(t+1)} = \text{Leading } r_y \text{ singular vectors of } \mathcal{Y} \times_3 \boldsymbol{u}^{(t)} \\
&\boldsymbol{u}^{(t+1)} = \frac{\lambda[\mathcal{X}; \boldsymbol{V}^{(t+1)}] + (1-\lambda)[\mathcal{Y}; \boldsymbol{W}^{(t+1)}]}{\left\| \lambda[\mathcal{X}; \boldsymbol{V}^{(t+1)}] + (1-\lambda)[\mathcal{Y}; \boldsymbol{W}^{(t+1)}] \right\|_2} \\
&t = t + 1
\end{aligned}$$

- **return** $\hat{\boldsymbol{u}} = \boldsymbol{u}^{(t)}$, $\hat{\boldsymbol{V}} = \boldsymbol{V}^{(t)}$, $\hat{\boldsymbol{W}} = \boldsymbol{W}^{(t)}$; $\hat{d}_x = \langle \mathcal{X}, \hat{\boldsymbol{V}}\hat{\boldsymbol{V}}' \circ \hat{\boldsymbol{u}} \rangle / r_x$, $\hat{d}_y = \langle \mathcal{Y}, \hat{\boldsymbol{W}}\hat{\boldsymbol{W}}' \circ \hat{\boldsymbol{u}} \rangle / r_y$; $\hat{\mathcal{X}} = \hat{d}_x \cdot \hat{\boldsymbol{V}}\hat{\boldsymbol{V}}' \circ \hat{\boldsymbol{u}}$, $\hat{\mathcal{Y}} = \hat{d}_y \cdot \hat{\boldsymbol{W}}\hat{\boldsymbol{W}}' \circ \hat{\boldsymbol{u}}$.

$\boldsymbol{u}^{(0)} = \mathbf{1}_N / \sqrt{N}$. We will show in Section 3 that when the networks across the population are not too different from each other, warm initialization may suffice to guarantee convergence; while without such prior knowledge/belief, we suggest using the spectral initialization as in many prior works (Zhang and Xia, 2018):

$$\boldsymbol{u}^{(0)} = \text{leading singular vector of } [\lambda \mathcal{M}_3(\mathcal{X}), (1-\lambda)\mathcal{M}_3(\mathcal{Y})]. \tag{3}$$

Here we still use $\lambda$ to weight the two tensors as in Algorithm 1.

### 2.2.2 Multi-Factor JisstPCA

We now consider the general and more challenging case with $K > 1$. Inspired by the success of deflation methods in various matrix and tensor PCA approaches (Mackey, 2008; Allen, 2012a; Weylandt and Michailidis, 2022; Ge et al., 2021), we adopt a scheme that successively applies single-factor JisstPCA followed by deflating the tensor based on previously estimated factors. Specifically, we apply Algorithm 1 to $\mathcal{X}^k$ and $\mathcal{Y}^k$, deflate each of these tensors to get $\mathcal{X}^{k+1}$ and $\mathcal{Y}^{k+1}$, and repeat

until we have extracted all $K$ factors; note that we set $\mathcal{X}^1 = \mathcal{X}$ and $\mathcal{Y}^1 = \mathcal{Y}$. There are several possible deflation schemes (Mackey, 2008), but perhaps subtraction deflation is the simplest and imposes the fewest assumptions:

$$\mathcal{X}^{k+1} = \mathcal{X}^k - \hat{d}_{x,k} \cdot \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k' \circ \hat{\boldsymbol{u}}_k, \quad \mathcal{Y}^{k+1} = \mathcal{Y}^k - \hat{d}_{y,k} \cdot \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k' \circ \hat{\boldsymbol{u}}_k. \tag{4}$$

In many scenarios, however, one may wish to impose orthogonality constraints on all the tensor factors. To achieve this, one could employ projection deflation for tensors as in (Mackey, 2008; Allen, 2012a). We note, however, that mutual orthogonality of the network factors across the layers may be quite restrictive for real data and limit the expressivity and interpretability of the model. Instead, one might want to impose orthogonality of the population factors, $\mathbf{u}_k$, to enable similar interpretation of these as sample PCs. To achieve this, we propose partial projection deflation:

$$\begin{aligned} \mathcal{X}^{k+1} &= \left(\mathcal{X}^k - \hat{d}_{x,k} \cdot \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k' \circ \hat{\boldsymbol{u}}_k\right) \times_3 \left(\boldsymbol{I}_N - \hat{\boldsymbol{u}}_k \hat{\boldsymbol{u}}_k'\right) \\ \mathcal{Y}^{k+1} &= \left(\mathcal{Y}^k - \hat{d}_{y,k} \cdot \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k' \circ \hat{\boldsymbol{u}}_k\right) \times_3 \left(\boldsymbol{I}_N - \hat{\boldsymbol{u}}_k \hat{\boldsymbol{u}}_k'\right). \end{aligned} \tag{5}$$

More details on this, other possibilities for deflation schemes, and the selection of number of factors $K$ are provided in the Appendix.

### 2.3 Extensions and Practical Considerations

**Generalized JisstPCA** In this section, we consider an important extension of the JisstPCA model (1) and algorithm to incorporate broader application scenarios. In particular, model (1) assumes one single eigenvalue for each network factor, which may not be reasonable when one network factor consists of multiple components with different connection strengths. Therefore, we extend (1) to allow *multiple eigenvalues* for each network factor:

$$\mathcal{X} = \sum_{k=1}^{K} \boldsymbol{V}_k^* \boldsymbol{D}_{x,k}^* \boldsymbol{V}_k^{*\prime} \circ \boldsymbol{u}_k^* + \mathcal{E}_x, \quad \mathcal{Y} = \sum_{k=1}^{K} \boldsymbol{W}_k^* \boldsymbol{D}_{y,k}^* \boldsymbol{W}_k^{*\prime} \circ \boldsymbol{u}_k^* + \mathcal{E}_y. \tag{6}$$

where $\boldsymbol{D}_{x,k}^* \in \mathbb{R}^{r_{x,k} \times r_{x,k}}$ and $\boldsymbol{D}_{y,k}^* \in \mathbb{R}^{r_{y,k} \times r_{y,k}}$ are diagonal matrices, replacing the scalar eigenvalues $d_{x,k}^*$, $d_{y,k}^*$ in (1). To estimate factors in this generalozed JisstPCA model, we still propose a power iteration algorithm for the single-factor case and apply a successive deflation scheme for multi-factor models. The main change we make to the single-factor JisstPCA algorithm is that we update the diagonal matrix $\boldsymbol{D}_x$ and $\boldsymbol{D}_y$ within each iteration due to its increased importance, and we use them weight the columns of $\boldsymbol{V}$, $\boldsymbol{W}$ when updating $\boldsymbol{u}$. We term this new algorithm the generalized JisstPCA (G-JisstPCA) algorithm, whose detailed procedures are summarized in Section A.4 of the Appendix.

**Other extensions and hyperparameter tuning** Another important extension of our JisstPCA framework is to jointly analyze network data together with vector-valued covariates, organized as a matrix. This is especially useful for integrating feature vectors in neuroimaging studies, such as genomics, demographic, or behavioral traits for each subject. The detailed matrix-tensor JisstPCA

model and its associated algorithm are included in Section A.4 of the Appendix. We also include a detailed discussion on how to select the hyperparameters in practice in Section A.4.2 of the Appendix, including the number of factors $K$, the ranks of each factor $r_{x,k}$, $r_{y,k}$, $1 \le k \le K$, and the integrative scaling parameter $\lambda$.

# 3 Theoretical Guarantees

In this section, we provide theoretical properties of single-factor JisstPCA (Algorithm 1) and Generalized JisstPCA (Algorithm 5) in terms of the estimation errors for population and network factors. We first show a deterministic one-step convergence result that does not assume the noise distribution, under a deterministic SNR condition and an initialization condition. In Section 3.2, we focus on the special case of sub-Gaussian noise, showing that under suitable SNR assumptions, spectral initialization is good enough to ensure one-step statistical convergence.

## 3.1 Deterministic Convergence Guarantee

To begin with, we state two assumptions on the SNR and initialization, $\boldsymbol{u}^{(0)}$.

**Assumption 1** (SNR condition)**.** $d_x^* \ge 5\|\mathcal{E}_x\|_{\mathrm{op}}$, $d_y^* \ge 5\|\mathcal{E}_y\|_{\mathrm{op}}$.

Assumption 1 enforces that the signal strength $d_x^*$ and $d_y^*$ for tensors $\mathcal{X}$ and $\mathcal{Y}$ are at least constant times larger than the noise level $\|\mathcal{E}_x\|_{\mathrm{op}}$ and $\|\mathcal{E}_y\|_{\mathrm{op}}$.

**Assumption 2** (Initialization condition)**.** *The initialization* $\boldsymbol{u}^{(0)}$ *satisfies* $|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(0)})|^2 \le 1 - 8\left(\frac{\|\mathcal{E}_x\|_{\mathrm{op}}^2}{d_x^{*2}} \vee \frac{\|\mathcal{E}_y\|_{\mathrm{op}}^2}{d_y^{*2}}\right)$.

Assumption 2 is concerned with the initialization error $|\sin\theta(\boldsymbol{u}, \boldsymbol{u}^{(0)})| \in [0, 1]$. We note that when Assumption 1 holds, Assumption 2 can be implied by $|\sin\theta(\boldsymbol{u}, \boldsymbol{u}^{(0)})| \le \frac{4}{5}$; when the *SNR continues to grow, Assumption 2 becomes weaker and easier to satisfy.* We will discuss this Assumption in more detail after Theorem 1.

Before presenting our deterministic one-step convergence guarantee for JisstPCA, we also define a notation representing the integrated noise level: $\|\lambda\mathcal{E}_x; (1-\lambda)\mathcal{E}_y\|_{r_x, r_y, \mathrm{op}} = \sup_{\boldsymbol{V}\in\mathcal{O}_{p\times r_x}, \boldsymbol{W}\in\mathcal{O}_{q\times r_y}} \|\lambda[\mathcal{E}_x; \boldsymbol{V}] + (1-\lambda)[\mathcal{E}_y; \boldsymbol{W}]\|_2$.

**Theorem 1.** *Suppose* $\mathcal{X}, \mathcal{Y}$ *satisfy* (1) *with* $K = 1$*, and Assumptions 1 and 2 hold. Then the output of Algorithm 1 satisfies the following: for* $k \ge 1$*,*

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})| \le \frac{4\|\lambda\mathcal{E}_x; (1-\lambda)\mathcal{E}_y\|_{r_x, r_y, \mathrm{op}}}{\lambda r_x d_x^* + (1-\lambda) r_y d_y^*}, \tag{7}$$

$$\left\|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\right\|_{\mathrm{op}} \le \frac{4\|\mathcal{E}_x\|_{\mathrm{op}}}{d_x^*}, \quad \left\|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k+1)})\right\|_{\mathrm{op}} \le \frac{4\|\mathcal{E}_y\|_{\mathrm{op}}}{d_y^*}. \tag{8}$$

For the single-factor model, Theorem 1 suggests that as long as the signal is not masked by the noise (Assumption 1), and the initialization is reasonably good, then both the population factor $\boldsymbol{u}$

and network factors $\boldsymbol{V}$, $\boldsymbol{W}$ can be well estimated after updating each factor only once. The one-step convergence may be inherited from the nice properties of the power iteration. The estimation error for the joint factor $\boldsymbol{u}$ is the ratio between the spectral norm of integrated noise and the integrated signal; the errors for the network factors $\boldsymbol{V}$ and $\boldsymbol{W}$ are the corresponding SNR for each tensor. This resembles the matrix perturbation bounds (Davis-Kahan's Theorem) and recent tensor perturbation theory under the CP and Tucker models (Luo et al., 2021; Anandkumar et al., 2014). (7) shows that by integrating two tensors, the estimation accuracy of $\boldsymbol{u}^*$ achieves a balance between the SNRs of $\mathcal{X}$ and $\mathcal{Y}$.

**Proof sketch**: Recall that Algorithm 1 takes a similar form to a power iteration, where we alternatively update the estimates of $\boldsymbol{V}$, $\boldsymbol{W}$ and $\boldsymbol{u}$ by taking the top singular vectors of a matrix computed from prior updates. In our proof, we apply the Davis-Kahan's Theorem to show that the perturbation bound for each update mainly depends on the SNR, inflated by a factor depending on the error of prior updates. Whenever the initialization and SNR conditions hold, the errors of $\boldsymbol{V}^{(1)}$, $\boldsymbol{W}^{(1)}$, $\boldsymbol{u}^{(1)}$ are all bounded by constants, leading to the sufficiently small statistical error for the next updates $\boldsymbol{u}^{(1)}$, $\boldsymbol{V}^{(2)}$, $\boldsymbol{W}^{(2)}$.

**Remark 1** (Initialization condition)**.** *As we will show later in Section 3.2, spectral initialization will satisfy Assumption 2 when the noise is sub-Gaussian and under SNR conditions comparable to those in the literature. Furthermore, in the special case where the population variation is not too large ($\boldsymbol{u}_i^*$'s are not too different), a warm initialization with $\boldsymbol{u}^{(0)} = (\frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}})'$ may also work without imposing any distributional assumption on the noise. We formalize this intuition in Corollary 1.*

**Corollary 1** (Warm initialization)**.** *Let $\widehat{\mathrm{Var}}(u^*) = \frac{1}{N}\sum_i(\boldsymbol{u}_i^* - \frac{1}{N}\sum_j \boldsymbol{u}_j^*)^2$, $\widehat{\mathbb{E}}(u^*) = \frac{1}{N}\sum_i \boldsymbol{u}_i^*$. Then as long as Assumption 1 holds and $\frac{\widehat{\mathrm{Var}}(u^*)}{(\widehat{\mathbb{E}}(u^*))^2} \le \left(\frac{d_x^{*2}}{8\|\mathcal{E}_x\|_{\mathrm{op}}^2} \wedge \frac{d_y^{*2}}{8\|\mathcal{E}_y\|_{\mathrm{op}}^2}\right) - 1$, the output of Algorithm 1 with warm initialization $\boldsymbol{u}^{(0)} = (\frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}})'$ satisfies the estimation error bounds* (7) *and* (8) *for $k \ge 1$.*

Corollary 1 suggests that as long as the variation amongst sample factors does not dominate its mean, then our JisstPCA algorithm with warm initialization enjoys one-step convergence.

### 3.2 Special Case: Sub-Gaussian Noise and Spectral Initialization

In this subsection, we present the convergence property of JisstPCA when the noise tensors are entrywise sub-Gaussian, and when the spectral initialization is applied.

**Assumption 3** (Sub-Gaussian noise)**.** *Suppose that for $1 \le k \le N$, $(\mathcal{E}_x)_{:,:,k}$ and $(\mathcal{E}_y)_{:,:,k}$ have independent, zero-mean, sub-Gaussian-$\sigma$ entries subject to symmetry constraints. Let $\sigma_{i,j,k}^2(\mathcal{E}_x) = \mathrm{Var}((\mathcal{E}_x)_{i,j,k})$, $\sigma_{i,j,k}^2(\mathcal{E}_y) = \mathrm{Var}((\mathcal{E}_y)_{i,j,k})$ be entrywise variances, which satisfy $\sum_{i,j}\sigma_{i,j,k}^2(\mathcal{E}_x) = \sum_{i,j}\sigma_{i,j,k'}^2(\mathcal{E}_x)$, and $\sum_{i,j}\sigma_{i,j,k}^2(\mathcal{E}_y) = \sum_{i,j}\sigma_{i,j,k'}^2(\mathcal{E}_y)$ for all $1 \le k, k' \le N$.*

Assumption 3 is weaker than entrywise i.i.d. noise, as we allow $\sigma_{i,j,k}(\mathcal{E}_x)$ ($\sigma_{i,j,k}(\mathcal{E}_y)$) to be different across different $i, j \in [p]$ ($i, j \in [q]$): variance can be location-varying within the network. We

also allow difference noise variances between the two tensors $\mathcal{E}_x$ and $\mathcal{E}_y$. The only homogeneous requirement is on the third mode (population mode) for a given tensor, meaning that all samples have the same noise level for a given network modality.

**Assumption 4** (SNR condition under sub-Gaussian noise)**.** *Let* $d_\lambda = \sqrt{\lambda r_x d_x^{*2} + (1-\lambda) r_y d_y^{*2}}$ *be the integrated ground truth signal, and suppose that* $d_\lambda \geq C\sigma\left(N^{1/4}\left(\sqrt{p}+\sqrt{q}\right)+\sqrt{N}\right)\sqrt{\log N}$ *for some constant* $C > 0$*. In addition, the signal of each network tensor satisfies* $d_x^* \geq C\sigma(\sqrt{N}+\sqrt{p})$*,* $d_y^* \geq C\sigma(\sqrt{N}+\sqrt{q})$*.*

The individual SNR conditions ($d_x^* \geq C\sigma(\sqrt{N}+\sqrt{p})$, $d_y^* \geq C\sigma(\sqrt{N}+\sqrt{q})$) in Assumption 4 are equivalent to Assumption 1 under the sub-Gaussian noise. The additional integrated SNR condition on $d_\lambda$ ensures a reasonably good spectral initialization (see Proposition 1). In the special case where $d_x^* = d_y^* = d$, $r_x = r_y = r$, $p = q$, Assumption 4 can be implied by $d/\sigma \geq C\max\{r^{-\frac{1}{2}}(\sqrt{p}N^{1/4} + \sqrt{N})\sqrt{\log N}, \sqrt{p}+\sqrt{N}\}$. This is weaker than the SNR condition ($d/\sigma \geq Cr(\sqrt{pN}+\sqrt{N\log N})$) in the prior work on single semi-symmetric tensor PCA (Weylandt and Michailidis, 2022). In addition, when $p = N$, prior work on Tucker low-rank tensor PCA (Zhang and Xia, 2018) requires $d/\sigma \geq Cp^{3/4}$, comparable to our SNR condition $d/\sigma \geq C\max\{r^{-\frac{1}{2}}p^{3/4}\sqrt{\log p}, \sqrt{p}\}$ [†].

**Proposition 1.** *Suppose* $\mathcal{X}, \mathcal{Y}$ *are generated from* (1) *with* $K = 1$*, and Assumptions 3 and 4 hold. Then, for the spectral initialization for* $\boldsymbol{u}^{(0)}$ *as defined in* (3)*, we have*

$$\left|\sin\theta(\boldsymbol{u}, \boldsymbol{u}^{(0)})\right| \leq \frac{C\sigma\left(\sqrt{N}+(N(p^2+q^2))^{\frac{1}{4}}\right)\sqrt{\log N}}{d_\lambda} \leq \sqrt{1 - 8\left(\frac{\|\mathcal{E}_x\|_{\text{op}}^2}{d_x^{*2}} \vee \frac{\|\mathcal{E}_y\|_{\text{op}}^2}{d_y^{*2}}\right)}, \tag{9}$$

*with probability at least* $1 - C\exp(-cN)$ *for some constant* $C, c > 0$*.*

Proposition 1 suggests that under Assumptions 3-4, spectral initialization satisfies Assumption 2 with high probability. We are now in position to state our main statistical error bounds for JisstPCA with sub-Gaussian noise.

**Theorem 2.** *Suppose* $\mathcal{X}, \mathcal{Y}$ *satisfy* (1) *with* $K = 1$*, and Assumptions 3 and 4 hold. Then Algorithm 1 with spectral initialization* (3) *satisfies the following: for* $k \geq 1$*,*

$$\left|\sin\theta(\boldsymbol{u}, \boldsymbol{u}^{(k)})\right| \leq \frac{C\sigma(\lambda r_x\sqrt{p+N} + (1-\lambda)r_y\sqrt{q+N})}{\lambda r_x d_x^* + (1-\lambda) r_y d_y^*}, \tag{10}$$

$$\left\|\sin\Theta(\boldsymbol{V}, \boldsymbol{V}^{(k+1)})\right\|_{\text{op}} \leq \frac{C\sigma(\sqrt{p}+\sqrt{N})}{d_x^*}, \quad \left\|\sin\Theta(\boldsymbol{W}, \boldsymbol{W}^{(k+1)})\right\|_{\text{op}} \leq \frac{C\sigma(\sqrt{q}+\sqrt{N})}{d_y^*}, \tag{11}$$

*with probability at least* $1 - C\exp(-cN)$*, where* $C, c > 0$ *are universal constants.*

Theorem 2 shows that under sub-Gaussian noise and an SNR condition comparable to the prior literature, all factors converge to their statistical errors after being updated at least once. As far as we are aware, this is the first statistical guarantee for integrative tensor analysis. When $r_x = r_y = r$,

[†]We only have an additional log-factor $\sqrt{\log p}$ since we allow location-varying noise variances.

$d_x^* = d_y^* = d$, $p = q$, our statistical error for $\boldsymbol{u}$, $\boldsymbol{V}$, $\boldsymbol{W}$ all scale as $\frac{\sigma(\sqrt{p}+\sqrt{N})}{d}$. which improves upon the prior result (Weylandt and Michailidis, 2022) on non-integrated data ($\frac{\sigma r\sqrt{pN}}{d}$ for $\boldsymbol{u}$ and $\frac{\sigma r^{3/2}\sqrt{pN}}{d}$ for $\boldsymbol{V}$). In addition, our statistical error bounds are satisfied by one-step iterates with provable initialization, also demonstrating extremely fast convergence. Furthermore, our statistical error bound is comparable to prior results on HOOI for Tucker low-rank tensor PCA Zhang and Xia (2018); Luo et al. (2021), where the error of each factor scales as $\frac{\sigma\sqrt{p}}{d}$ when each mode's dimension scales as $p$.

### 3.3 Extension: Convergence of Generalized JisstPCA

To accommodate more general scenarios, we also study the theoretical properties of the Generalized JisstPCA (Algorithm 5) under the model (6). In this setting, we have different signal strengths for estimating the joint factor and individual factors, as reflected by the two separate SNR conditions as follows.

**Assumption 5** (SNR for joint factor)**.** *The integrated signal satisfies:* $\lambda\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)\|\boldsymbol{D}_y^*\|_F^2 \geq C\sigma^2(\sqrt{N(p^2+q^2)}+N)\log N$*, and the individual signals satisfy* $\|\boldsymbol{D}_x^*\|_F^2 \geq C\sigma^2 r_x(p+N)$, $\|\boldsymbol{D}_y^*\|_F^2 \geq C\sigma^2 r_y(q+N)$.

Assumption 5 ensures a good estimate for the joint factor $\boldsymbol{u}^*$. We note that the signal strength depends on the Frobenious norms of $\boldsymbol{D}_x^*$ and $\boldsymbol{D}_y^*$, since they are the singular values of the matricization along the third mode.

**Assumption 6** (SNR for individual factors)**.** $\sigma_{r_x}^2(\boldsymbol{D}_x^*) \geq C\sigma^2(p+N)$, $\sigma_{r_y}^2(\boldsymbol{D}_y^*) \geq C\sigma^2(q+N)$.

Assumption 6 ensures good estimates for the individual factors $\boldsymbol{V}^*$, $\boldsymbol{W}^*$. Different from Assumption 5, the estimation of individual factors depends on the $\sigma_{r_x}(\boldsymbol{D}_x^*)$ and $\sigma_{r_y}(\boldsymbol{D}_y^*)$, since they are the minimum nonzero singular values of the matricizations along first/second modes. Note that we allow entries of $\boldsymbol{D}_x^*$ and $\boldsymbol{D}_y^*$ to have arbitrary signs, different from the original JisstPCA model (1) (Assumption 4). In the special case when $\boldsymbol{D}_x^*$ and $\boldsymbol{D}_y^*$ are diagonal matrices with $d_x^*$, $d_y^*$, Assumptions 5-6 are both implied by Assumption 4.

**Theorem 3.** *Suppose that* $\mathcal{X}$ *and* $\mathcal{Y}$ *are generated from* (6)*, and we apply Algorithm 5 on them with the spectral initialization* (3)*. Then as long as Assumptions 3 and 5 hold, with probability at least* $1 - CN^{-c}$*, for any iteration number* $k \geq 1$*, we have*

$$
\begin{aligned}
|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})| \leq & \frac{C\sigma\left(\lambda\sqrt{r_x(p+N)}\|\boldsymbol{D}_x^*\|_F + (1-\lambda)\sqrt{r_y(q+N)}\|\boldsymbol{D}_y^*\|_F\right)}{\lambda\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)\|\boldsymbol{D}_y^*\|_F^2} \\
\leq & \frac{C\sigma\sqrt{r_x(p+N)}}{\|\boldsymbol{D}_x^*\|_F} \vee \frac{C\sigma\sqrt{r_y(q+N)}}{\|\boldsymbol{D}_y^*\|_F}.
\end{aligned}
$$

*If Assumption 6 also holds, the following holds for any* $k \geq 2$ *with the same probability:*

$$
\|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k)})\|_{\mathrm{op}} \leq \frac{C\sigma\sqrt{p+N}}{\sigma_{r_x}(\boldsymbol{D}_x^*)}, \|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k)})\|_{\mathrm{op}} \leq \frac{C\sigma\sqrt{q+N}}{\sigma_{r_y}(\boldsymbol{D}_y^*)}.
$$

Theorem 3 extends the one-step convergence guarantee in Theorem 2 to the general model and the Generalized JisstPCA algorithm, where the only change lies in the characterization of the SNR. As discussed earlier, the SNR for the joint factor depends on $\|\boldsymbol{D}_x^*\|_F$ and $\|\boldsymbol{D}_y^*\|_F$, while the SNR for the individual factors depend on $\sigma_{r_x}(\boldsymbol{D}_x^*)$ and $\sigma_{r_y}(\boldsymbol{D}_y^*)$. As a comparison, the SNR in prior theoretical results for HOOI and Tucker low-rank tensor PCA (Zhang and Xia, 2018) is characterized by the minimum singular value along all modes, which translates to $\sigma_{r_x}(\boldsymbol{D}_x^*)$ or $\sigma_{r_y}(\boldsymbol{D}_y^*)$ in our setting. Different from prior results, we are able to better separate the estimation guarantees of joint and individual factors since our G-JisstPCA algorithm makes use of a more compact decomposition (6) than the Tucker decomposition.

**Remark 2** (Rank misspecification)**.** *Since no minimum eigenvalue condition on $\boldsymbol{D}_x^*$ or $\boldsymbol{D}_y^*$ is needed to accurately estimate the joint factor $\boldsymbol{u}^*$, using larger ranks $r_x$, $r_y$ in Algorithm 5 does not affect the estimation of $\boldsymbol{u}^*$.*

# 4 Simulation Studies

In this section, we empirically study our JisstPCA algorithms by (i) validating our theoretical guarantees and (ii) comparing the performance of our methods with baseline methods. The code necessary for reproducing our empirical results is available at https://github.com/JmL130169/JisstPCA.

## 4.1 Validation of Theoretical Guarantees

We first validate our main theory (Theorem 2 and 3): one-step convergence and statistical error rates, for the JisstPCA model (1) and the generalized JisstPCA model (6), under the single-factor ($K = 1$) case. For both models, we consider tensor dimensions $p = q = \frac{1}{2}N \in \{60, 90, 120\}$, ranks $r_x = 3$, $r_y = 2$. The ground truth tensor factors $\boldsymbol{V}^*$, $\boldsymbol{W}^*$, and $\boldsymbol{u}^*$ are randomly generated with independent standard Gaussian entries and then orthogonalized. Each slice of the noise tensors $((\mathcal{E}_x)_{:,:,k}, (\mathcal{E}_y)_{:,:,k}, 1 \le k \le N)$ is symmetric mean-zero Gaussian with off-diagonal entries of variance 1 and diagonal entries of variance 2. For JisstPCA model, we let $d_y^* = 1.2d_x^*$, and the values of $d_x^*$ are set such that SNR $= \frac{d_x^*}{\sqrt{p}+\sqrt{N}}$ take values from 1.5 to 24. For the G-JisstPCA model, we consider the same set of $d_x^*$, $d_y^*$ but let $\boldsymbol{D}_x^* = d_x^*\mathrm{diag}(1.5, 1, 0.8)$, $\boldsymbol{D}_y^* = d_y^*\mathrm{diag}(1, 0.8)$. Given the noisy observations $\mathcal{X}$, $\mathcal{Y}$, we run the JisstPCA and Generalized JisstPCA (G-JisstPCA) algorithms with $K = 1$, $\lambda = 0.5$, oracle ranks, and the spectral initialization (3). Figure 1 reports the spectral norm $\sin\Theta$ errors of the three estimated factors ($|\sin\theta(\boldsymbol{u}^{(k)}, \boldsymbol{u}^*)|$, $\|\sin\Theta(\boldsymbol{V}^{(k)}, \boldsymbol{V}^*)\|_{\mathrm{op}}$, $\|\sin\Theta(\boldsymbol{W}^{(k)}, \boldsymbol{W}^*)\|_{\mathrm{op}}$). The first and third panels plot the estimation errors versus iteration number when the SNR is 1.5, and we can see all three factors converge to the statistical errors after one step update; The second and fourth panels plot the estimation errors of the first update versus its theoretical scaling (1/SNR), validating our theory as well.

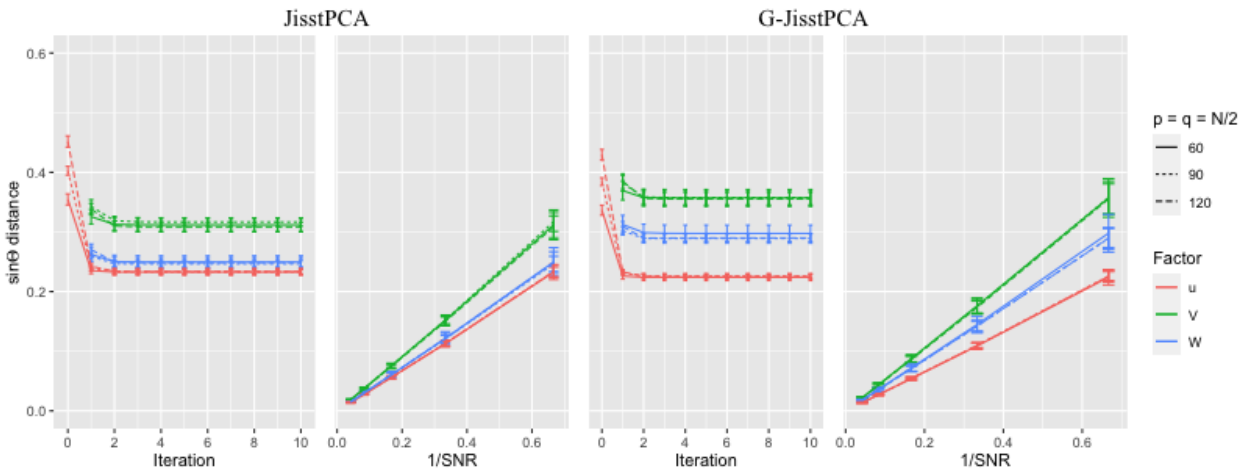


Figure 1: Empirical validation of theoretical results for JisstPCA (left) and G-JisstPCA (right). The first and third figures confirm that the $\sin\Theta$ distances of $\boldsymbol{u}^{(k)}$, $\boldsymbol{V}^{(k)}$, and $\boldsymbol{W}^{(k)}$ converge to the statistical error after one or two step iterations, even under a relatively small SNR = 2.25. The second and fourth figures demonstrate linear dependence of the statistical error on 1/SNR, validating our theory. 20 independent replicates are run, and we plot the mean errors with error bars representing 95% approximate normal confidence interval.

### 4.2 Comparative Study

We seek to study the comparative advantages of our JisstPCA and G-JisstPCA algorithms across various settings. However, notice that there are no existing methods for integrative semi-symmetric tensor PCA that are natural baselines for comparison. Hence, we develop simple integrative extensions of the classical higher-order SVD (HOSVD) and higher-order orthogonal iteration (HOOI) algorithms as our comparison baselines; we call these "iHOSVD" and "iHOOI", respectively. We choose these baselines as the Tucker tensor decomposition model naturally leads to symmetric network factors; existing integrative CP-based approaches (Acar et al., 2011) do not offer symmetric factors and are hence incomparable to our approach. We provide detailed iHOSVD and iHOOI algorithms in the Appendix. In all our simulations to avoid rank selection for the iHOSVD and iHOOI methods, we employ these approaches with oracle ranks. Our JisstPCA and G-JisstPCA algorithms, on the other hand, are employed with both oracle ranks and data-driven selection of ranks via our BIC approach described in Section A.4.2 and A.5 of the Appendix.

We first consider data generated from our own models, (1) and (6). We focus on $K = 2$ factors, ranks $\boldsymbol{r}_x = \boldsymbol{r}_y = (3, 2)'$, $p = 150$, $q = 50$, and $N = 50$. (The Appendix contains additional simulations varying the dimensionality $p$, $q$, and $N$.) Noise tensors are generated as previously described. To simulate different scenarios, the ground truth factors are set as unstructured (randomly generated entries as previously described) or structured (factors of block and star networks as shown in the top panel of Figure 3; more details included in Section A.5 of the Appendix); we also leave these factors as non-orthogonal across $k = 1, 2$ or enforce mutual orthogonality along each mode. For non-orthogonal settings, we set $\boldsymbol{d}_x^* = \text{SNR} * (\sqrt{p} + \sqrt{N})(1, 0.5)'$, $\boldsymbol{d}_y^* = \text{SNR} * (\sqrt{q} + \sqrt{N})(1, 0.5)'$.

In the orthogonal setting, we study the effect of singular gap between different factors by letting $\boldsymbol{d}_x^* = \text{SNR} * (\sqrt{p} + \sqrt{N})(1,\, 1)'$, $\boldsymbol{d}_y^* = \text{SNR} * (\sqrt{q} + \sqrt{N})(1,\, 0.9)'$. While under the generalized model (6), we let $\boldsymbol{D}_{x,1}^* = d_{x,1}^* \text{diag}(2,\, 1.5,\, 1.2)$, $\boldsymbol{D}_{x,2}^* = d_{x,2}^* \text{diag}(2,\, 1.6)$, and $\boldsymbol{D}_{y,1}^*$, $\boldsymbol{D}_{y,2}^*$ similarly defined, where $\boldsymbol{d}_x^*$, $\boldsymbol{d}_y^*$ are set as previously described.

Figure 2 summarizes the $\sin\Theta$ distance errors of the estimated factors by our JisstPCA, G-JisstPCA algorithms (with BIC selected ranks and subtraction deflation), and the baselines iHOSVD and iHOOI (with oracle ranks) under different SNR values. Both iHOSVD and iHOOI enforce orthogonality and hence suffer from biases for non-orthogonal factors. In the orthogonal case, iHOSVD/iHOOI can fail due to the lack of singular gap, while our approach relies on the Frobenious norm gap between different factors and continues to perform well. (For fair comparison, we also include results with larger singular gaps in Figure 12 in the Appendix, where JisstPCA/G-JisstPCA has similar performance as iHOOI.) For the generalized model, JisstPCA has the same performance as G-JisstPCA, showing some robustness against a slight model misspecification. Of course, when the differences between eigenvalues within the same factor are larger, JisstPCA can fail and be worse than G-JisstPCA (as shown in Figure 14 in the Appendix). In summary, our methods always outperform the baselines, except for a few settings with small SNR due to incorrect rank selections. When all methods use oracle ranks, our methods always give the lowest errors (as shown in Figures 7 in the Appendix). Furthermore, Figure 3 visualizes the estimated factors of JisstPCA and iHOOI in the structured simulation along with the true factors, showing that iHOOI tends to mix the two graph components because of the rotations possible with the Tucker tensor core as well as the forced orthogonality. Additional simulation details, more empirical results including JisstPCA with projection deflation, and further discussion is available in the Appendix.

Finally, we seek to test the appropriateness of both our model as well as our algorithms on multi-modal population network data (binary adjacency tensors); this also serves to test the robustness of our approaches. Specifically, we divide the $N$ samples randomly into two clusters ($K = 2$), with probability 0.75 and 0.25, respectively. Each cluster is associated with a pair of stochastic block models (SBMs): $(\mathcal{M}_{k,x},\, \mathcal{M}_{k,y})$ for $k = 1,\, 2$. For any sample $i$ from cluster $k$, its associated networks $\mathcal{X}_{:,:,i}$ and $\mathcal{Y}_{:,:,i}$ are adjacency matrices sampled from $\mathcal{M}_{k,x}$ and $\mathcal{M}_{k,y}$, respectively. We set $\mathcal{M}_{1,x}$ and $\mathcal{M}_{1,y}$ as three-block SBMs and $\mathcal{M}_{2,x}$ and $\mathcal{M}_{2,y}$ as two-block SBMs, with the within-block edge probabilities ranging in $(0.5, 0.8)$ and out-of-block probability 0.3. Thus, these are truly low-rank networks with the rank the number of communities. Given the population of multi-modal adjacency matrices, we test how well our JisstPCA algorithms can extract the population factors and the underlying pairs of network components, which can be further used to detect the population clusters and community structures in each network component by applying k-means. Note that with populations of adjacency matrices, however, the top singular spaces all share the all one's vector; hence the top singular vectors across different network factors are not linearly independent. Thus, we project the rows and columns of each adjacency matrix onto the complement of the all one's vector before applying all methods; we suggest to follow this procedure in practice for populations of networks represented as adjacency matrices. Noting that the population membership factors are mutually orthogonal while the network factors are not, we apply our JisstPCA algorithms with

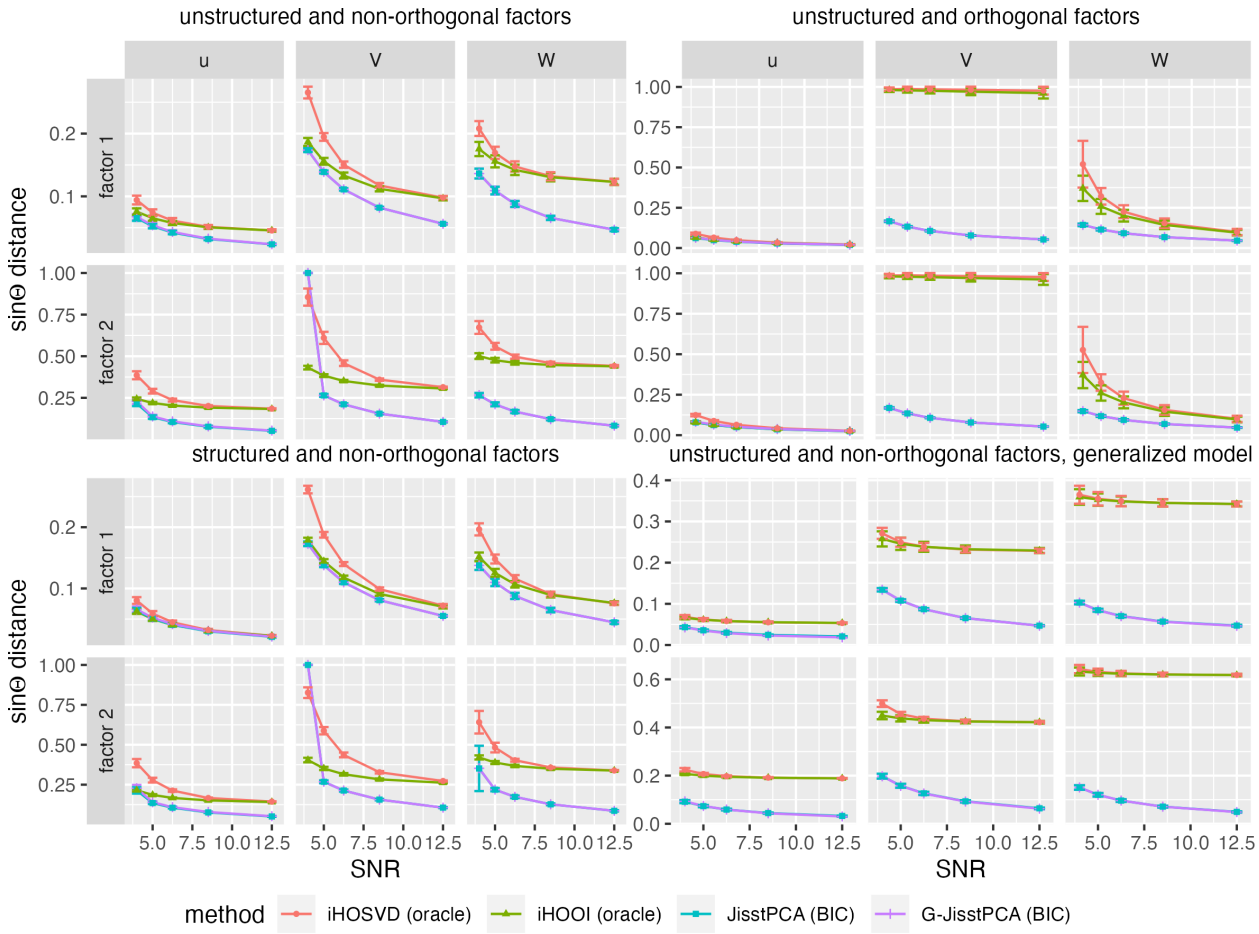


Figure 2: Estimation errors ($\sin\Theta$ distances in spectral norm) of all factors using JisstPCA, Generalized JisstPCA, iHOOI, and iHOSVD. The mean errors of 10 replicates are plotted with error bars. Our methods strongly outperform the baseline methods, even with data-driven rank tuning (BIC).

partial projection deflation on the population mode. We also select the ranks for JisstPCA using BIC-deflation, while applying iHOSVD, and iHOOI with the oracle ranks (the number of stochastic blocks minus one). We then apply k-means on the estimated population factors to cluster the samples, as well as on each network factor to cluster the nodes in the network. Table 1 shows the clustering accuracy (Adjusted Rand Index) based on the extracted factors from all methods with different sample sizes, when $p = 80$, $q = 50$. We also report the $\sin\Theta$ distance errors of each factor; we also perform the same projection for the edge probability tensors as for the adjacency tensors and then extract the ground truth factors. Here, we use relatively smaller dimensions $p$, $q$, $N$ since larger dimensions increase the SNR in SBMs and make the task too easy. Our methods strongly outperform the Tucker model based approaches in all scenarios, hence highlighting the advantanges of our modeling framework and algorithms for analyzing and detecting clusters in real network data. Additional results with the subtraction deflation and different network sizes, more details on the model set-up, and visualizations of the true and estimated network factors can be found in the

Appendix.


Figure 3: Heatmaps and scatterplots for the structured network factors and population factors, respectively. The truth is shown in the top panel, results of JisstPCA (with BIC rank selection) in the middle panel, and iHOOI (with oracle rank) in the bottom panel.

Table 1: Population clustering and network community detection for multi-modal populations of networks, based on k-means on the estimated factors from JisstPCA, G-JisstPCA, iHOSVD, and iHOOI. The presented Adjusted Rand Index (ARI) values demonstrate the accuracy of sample clustering and node clustering of two network factors for each modality, when network sizes $p = 80$, $q = 50$. The $\sin\Theta$ estimation errors of each factor is also presented. The average ARI and $\sin\Theta$ distances of 20 independent repeats are presented, with standard deviation inside the parenthesis. The largest average ARI and lowest estimation error for each setting are marked in bold.

| Clustering ARI | $N = 20$ | | | | $N = 40$ | | | |
|---|---|---|---|---|---|---|---|---|
| | **JisstPCA (BIC)** | **G-JisstPCA (BIC)** | **iHOSVD (oracle)** | **iHOOI (oracle)** | **JisstPCA (BIC)** | **G-JisstPCA (BIC)** | **iHOSVD (oracle)** | **iHOOI (oracle)** |
| Sample | 0.947(0.238) | **1**(0) | **1**(0) | 0.954(0.205) | **1**(0) | **1**(0) | **1**(0) | **1**(0) |
| Network 1 of $\mathcal{X}$ | 0.971(0.108) | **1**(0) | 0.912(0.195) | 0.739(0.251) | **1**(0) | **1**(0) | 0.805(0.241) | 0.663(0.224) |
| Network 2 of $\mathcal{X}$ | 0.995(0.022) | **0.997**(0.011) | 0.146(0.057) | 0.139(0.043) | **1**(0) | **1**(0) | 0.156(0.015) | 0.153(0) |
| Network 1 of $\mathcal{Y}$ | 0.974(0.114) | **1**(0) | 0.99(0.027) | 0.94(0.155) | **1**(0) | **1**(0) | 0.973(0.108) | **1**(0) |
| Network 2 of $\mathcal{Y}$ | **0.87**(0.31) | 0.867(0.309) | 0.014(0.046) | 0.121(0.16) | **1**(0) | **1**(0) | 0.116(0.101) | 0.246(0.153) |
| $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}_1^*)$ | 0.087(0.03) | 0.089(0.031) | 0.138(0.051) | **0.08**(0.026) | 0.092(0.016) | 0.093(0.017) | 0.142(0.028) | **0.083**(0.016) |
| $\sin\theta(\hat{\boldsymbol{u}}_2, \boldsymbol{u}_2^*)$ | 0.167(0.077) | **0.163**(0.066) | 0.214(0.04) | 0.188(0.162) | 0.152(0.014) | 0.152(0.014) | 0.197(0.017) | **0.145**(0.015) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_1, \boldsymbol{V}_1^*)\|_{\text{op}}$ | **0.084**(0.007) | **0.084**(0.007) | 0.163(0.06) | 0.163(0.059) | **0.062**(0.006) | **0.062**(0.007) | 0.154(0.046) | 0.156(0.047) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_2, \boldsymbol{V}_2^*)\|_{\text{op}}$ | **0.21**(0.074) | 0.211(0.072) | 0.817(0.065) | 0.799(0.07) | **0.154**(0.024) | 0.155(0.023) | 0.776(0.022) | 0.768(0.028) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_1, \boldsymbol{W}_1^*)\|_{\text{op}}$ | **0.158**(0.018) | **0.158**(0.018) | 0.253(0.056) | 0.196(0.058) | **0.118**(0.014) | **0.118**(0.014) | 0.202(0.026) | 0.173(0.04) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_2, \boldsymbol{W}_2^*)\|_{\text{op}}$ | **0.416**(0.204) | 0.417(0.203) | 0.969(0.044) | 0.903(0.098) | **0.272**(0.045) | 0.273(0.045) | 0.901(0.062) | 0.771(0.079) |

# 5 Case Study: Multi-Modal Population Brain Connectivity

We apply our proposed Generalized JisstPCA method to understand multi-modal and multi-subject brain connectivity patterns estimated from neuroimaging data. We analyze data from the Human Connectome Project (HCP), which can be easily accessed through the ConnectomeDB website and contains various traits, structural MRI (sMRI), functional MRI (fMRI), and diffusion MRI (dMRI) data for 1058 subjects (Glasser et al., 2013). Our objective is to understand major joint patterns in functional connectivity (FC) and structural connectivity (SC) as well as to see how these patterns vary and are related to other traits across the population. We process the data by applying the population-based connectome (PSC) extraction pipeline (Zhang et al., 2018) to construct the SC. Using the Desikan-Killiany atlas, we identify 68 cortical regions of interest (ROIs) and 19 subcortical ROIs, totaling 87 ROI nodes in our networks. The number of fiber curves between a pair of regions, measured at the logarithmic scale, is used to quantify the connection strength, or edge in our networks. Thus, the final dimension of our population SC tensor is $87 \times 87 \times 1058$, (ROIs by ROIs by subjects). We use the same atlas to extract FC, the computation of which is straightforward since the HCP provides preprocessed fMRI (Glasser et al., 2013). Specifically, we compute the mean fMRI time series for each ROI and calculate the Pearson correlation between different ROIs to generate a full FC matrix for each subject; the final population FC tensor has the same dimension as the SC tensor. Figure 17 in the Appendix shows the mean SC and FC weighted adjacency matrices, where the first 19 rows are the subcortical ROIs and the next 68 are the cortical ROIs; Supplement II is an Excel spreadsheet showing the ROI names. In addition, we also seek to understand how brain connectivity patterns relate to 45 cognitive traits which measure aspects such as fluid intelligence, delay discounting, and language/vocabulary comprehension.

Due to the complexity of human brain networks, we employ Generalized JisstPCA to jointly analyze the SC and FC tensors; we also compare our approach to PCA methods in Section C of the Appendix. All hyperparameters are chosen as discussed in Section A.4.2 of the Appendix, with the number of factors, $K$ chosen via the proportion of variance explained. This yields $K = 2$ factors which explain 85.5% of the variance in the FC data and 77.9% in the SC data; increasing $K$ further does not significantly increase the variance explained. Further, the BIC deflation strategy selects SC and FC network loadings of rank 5 for both sets of factors. The results from G-JisstPCA are visualized in Figure 4 (a), which shows a scatter plot of the estimated population components, $\hat{\boldsymbol{u}}_1$ and $\hat{\boldsymbol{u}}_2$. Figure 5 shows circle plots of the top connections from the estimated SC and FC network loadings, $\hat{\boldsymbol{V}}_1\hat{\boldsymbol{D}}_{SC,1}\hat{\boldsymbol{V}}_1^{'}$, $\hat{\boldsymbol{V}}_2\hat{\boldsymbol{D}}_{SC,2}\hat{\boldsymbol{V}}_2^{'}$ and $\hat{\boldsymbol{W}}_1\hat{\boldsymbol{D}}_{FC,1}\hat{\boldsymbol{W}}_1^{'}$, $\hat{\boldsymbol{W}}_2\hat{\boldsymbol{D}}_{FC,2}\hat{\boldsymbol{W}}_2^{'}$; estimated weighted adjacency matrices are shown in Figure 20 of the Appendix.

From Figure 4 (a), we identify a clear outlier (marked in the red circle). Upon checking the intermediate outputs of PSC, we find that this subject has a much sparser SC due to misalignment between sMRI and dMRI. This demonstrates a potential application of G-JisstPCA - outlier brain network identification, an important problem in brain network analysis (Dey et al., 2022). Next, we explore the relationship between the joint factors obtained by JisstPCA and cognitive traits. In (b) and (c) of Figure 4, we plot $\boldsymbol{u}^1$ and $\boldsymbol{u}^2$ for 200 subjects and color them according to their

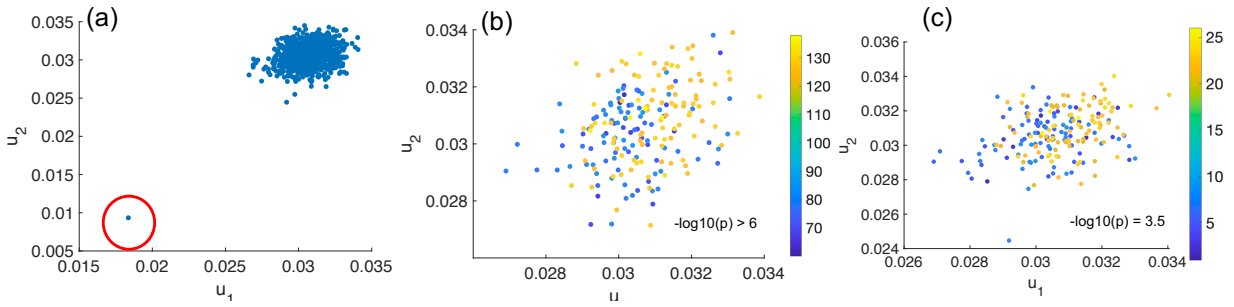


Figure 4: Results of Generalized JisstPCA applied to the HCP connectome data analyzing multi-subject Functional Connectivity (FC) and Structural Connectivity (SC). Panel (a) shows the scatter plot of the first two estimated population components, $\hat{\boldsymbol{u}}_1$ and $\hat{\boldsymbol{u}}_2$, exhibiting a major outlier. Panels (b) and (c) show scatter plots of 200 subjects colored according to their measures on the English Reading and Penn Line Orientation tests, respectively, showing a clear statistically significant association.

measures on the English Reading and Penn Line Orientation tests, respectively. For panel (b), the 200 subjects are selected based on their English Reading scores; the first 100 subjects have the highest scores and the second 100 have the lowest scores. The p-value testing the distribution difference between these two groups is displayed in the lower right corner. Similarly, 200 subjects are selected based on the Penn Line Orientation test for panel (c). The small p-values indicate that both SC and FC are significantly associated with the two traits under consideration.

In Figure 18 of the Appendix, we correlate $\boldsymbol{u}_1^*$ and $\boldsymbol{u}_2^*$ with the 45 cognitive traits. From this result, we observe that 1) $\boldsymbol{u}_1^*$ correlates better with behavioral traits than does $\boldsymbol{u}_2^*$, and 2) most behavioral traits show a decent amount of correlation with the joint factors. We also examine how well we can predict the traits using both $\boldsymbol{u}_1^*$ and $\boldsymbol{u}_2^*$, and compared the prediction with principal components analysis (PCA). Figure 20 of the Appendix shows the results, where we can see that the joint factors obtained by G-JisstPCA give much better prediction results, indicating that the joint components from SC and FC are more closely related to cognitive behavior traits.

## 6 Discussion

In this paper, we establish the first dimension reduction framework for the joint analysis of multi-modal populations of networks. Specifically, we proposed a novel joint integrative semi-symmetric tensor PCA (JisstPCA) model and associated algorithms to extract both the shared population factors across different modalities as well as low-rank network factors for each modality. We prove the convergence and statistical error bounds of our algorithms under the single-factor model, which improves or is comparable to prior results for single tensor PCA. Finally, a series of simulation studies validate the efficacy of our JisstPCA algorithm and its extensions; it also reveals intriguing structures in the human brain when applied to a real neuroimaging data example.

The joint network tensor PCA is a new problem with many potential fruitful future directions, a few of which we list as follows. First, it is challenging but interesting to extend our current theo-

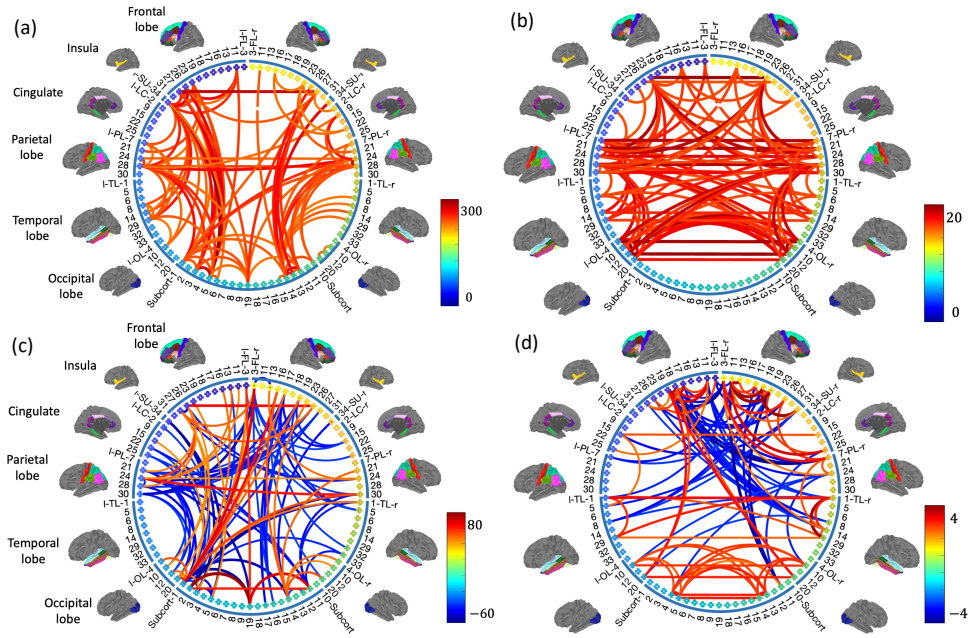


Figure 5: Visualizations of the estimated brain network loadings for Generalized JisstPCA applied to the HCP data. Panels (a) and (b) show the first loadings while panels (c) and (d) show the second loads for Structural Connectivity (SC), panels (a) and (c), and Functional Connectivity (FC), panels (b) and (d). These reveal many expected inter- and intra-hemisphere connections as well as major connectome variations that have associations with cognitive traits.

retical results to the more general multi-factor model. Second, it is of interest to address common real challenges like missing data and heteroskedastic noise, by leveraging the recent advances in PCA. Third, when applied to neuroimaging data, our methods may be extended to incorporate brain connectome data subject to different parcellation. Lastly, there is great potential for applying our methods to integrate genomics data with brain connectomes, possibly revealing genetic effects on brain development. In summary, our work pioneers the analysis of multi-modal populations of networks that paves the way for many future advances.

## Acknowledgments

JL, LZ, and GIA acknowledge support from NSF NeuroNex-1707400, NIH 1R01GM140468, and NSF DMS-2210837. ZZ acknowledges support from NIH award R25DA058940.

# A Extensions, Additional Details, and Results

## A.1 Detailed Literature Review

Both tensor algebra (Kolda and Bader, 2009) and topics about tensors, or multiway arrays, have been studied comprehensively during the recent years. Many efficient tensor PCA algorithms under different tensor low-rank structures have been proposed in the literature with strong theoretical guarantees (Han and Zhang, 2022; Zhang and Xia, 2018; Luo et al., 2021; Zhou et al., 2022), some even exploiting sparsity (Allen, 2012a; Zhang and Han, 2019). Beyond tensor PCA or tensor SVD, there is also rich literature on tensor completion (Yuan and Zhang, 2016; Cai et al., 2019; Xia et al., 2021) and tensor regression (Raskutti et al., 2019; Hao et al., 2020; Zhang et al., 2020a).

In terms of semi-symmetric tensor PCA, aside from Weylandt and Michailidis (2022), some other prior works (Zhang et al., 2019; Winter et al., 2020; Wang et al., 2014) also studied this topic for analyzing brain connectomes and magnetic resonance spectroscopy data. In particular, Zhang et al. (2019) first proposes to study brain connectomes using a semi-symmetric tensor PCA approach, which is further extended by Winter et al. (2020) to jointly analyze multi-scale graphs from different brain parcellations. Wang et al. (2014) considers a semi-symmetric and semi-nonnegative decomposition approach to perform Independent Component Analysis for magnetic resonance spectroscopy data. A comparison between existing modeling approaches and ours is included in Section 1.2. Jing et al. (2021) considers a semi-symmetric Tucker low-rank model of multilayer networks, with a focus on community detection. We note that the phrase "semi-symmetric tensor" is also used by Deng et al. (2023) to denote fourth-order tensors with two pairs of symmetric modes, different from the third-order tensors we are considering.

There also exist an extensive literature on data integration, which aims to find joint patterns across multiple sources of data. Most existing data integration methods focus on tabular data that can be arranged into matrices, including the JIVE (Lock et al., 2013) that decompose multiple data sets into sum of joint and individual principal components, the iPCA (Tang and Allen, 2021) built upon the matrix-variate normal model, the multi-block PCA family (Abdi et al., 2013; Westerhuis et al., 1998) that applies regular PCA on concatenated data sets after normalization, and many others. Extended from matrix integration problem, Acar et al. (2011, 2014); Wu et al. (2018); Schenker et al. (2020) consider joint factorization for tensors and tensor-matrix integration, based on the CP decomposition structure, as we mentioned in Section 1.2. In particular, Acar et al. (2011) solves coupled matrix and tensor factorization (CMTF) problem based on gradient methods, and Acar et al. (2014) extends this to a more general version that incorporate additional linear or nonlinear constraints in CMTF problem; also see some other extensions in Wu et al. (2018); Schenker et al. (2020). As for tensor-tensor type of integration, Genicot et al. (2016) proposes a joint tensor factorization method RCTF that can extract shared and unshared factors as well as robust components between integrated tensors. And Farias et al. (2016) uses Bayesian framework to define flexible coupling models and uncovers joint factors in terms of joint MAP estimators. In addition, some other algorithms, such as CIF-OPT and HOPM, have also been proposed to deal with joint tensor factorization problem or tensor canonical correlation analysis in different scenarios

(Lu et al., 2020; Chen et al., 2021b).

Lastly, there also exist many prior works devoted to collectively or integratively analyzing multiple networks and extracting interpretable knowledge from them. In particular, one line of literature focuses on the analysis of multiplex networks (Mucha et al., 2010; Paul and Chen, 2020b; MacDonald et al., 2022) where one has the access to a collection of networks associated with the same set of nodes, such as networks measured across a population (Wang et al., 2019; Paul and Chen, 2020a; Pavlović et al., 2020) or different time points (Mucha et al., 2010; Kim et al., 2018). Many existing works approach this problem by assuming a latent space model with a joint component and individual components that capture the heterogeneity across networks; under each specific model, estimation methods and statistical guarantees for identifying the latent structures are provided (Paul and Chen, 2020b; MacDonald et al., 2022; Zhang et al., 2020b). On the other hand, another line of work is concerned with the integrative analysis of multi-modal networks, such as the functional and structural brain connectivity networks based off the fMRI and sMRI data (Sui et al., 2012; Yao et al., 2015; Cole et al., 2021). Different from these prior works, we aim to propose a novel statistical method to jointly analyze populations of multimodal networks, extracting meaningful insights both across the population and linking the functional and structural connectivity of the brain.

### A.2 Relationship between the JisstPCA Model and the Tucker and CP Models

Now we give a detailed correspondence between our model and the Tucker/CP low-rank models. For simplicity, we will focus on the single tensor case, while it is straightforward to extend the model connections from a single tensor to joint factorization of multiple tensors.

Recall our multi-factor semi-symmetric tensor decomposition $\mathcal{X} = \sum_{k=1}^{K} d_{x,k}^* \cdot \boldsymbol{V}_k^* \boldsymbol{V}_k^{*\prime} \circ \boldsymbol{u}_k^* \in \mathbb{R}^{p\times p\times N} + \mathcal{E}_x$. When the factors are mutually orthogonal: $\boldsymbol{V}_i^* \perp \boldsymbol{V}_j^*$, $\boldsymbol{u}_i^* \perp \boldsymbol{u}_j^*$ for $1 \le i \ne j \le K$, we can also write $\mathcal{X}$ as the following low-rank Tucker decomposition plus noise: $\mathcal{X} = \mathcal{S}^* \times_1 \boldsymbol{U}_1^* \times_2 \boldsymbol{U}_2^* \times_3 \boldsymbol{U}_3^* + \mathcal{E}_x$. Here,

$$\boldsymbol{U}_1^* = \boldsymbol{U}_2^* = [\boldsymbol{V}_1^*, \cdots, \boldsymbol{V}_K^*] \in \mathbb{R}^{p\times r}, \quad \boldsymbol{U}_3^* = [\boldsymbol{u}_1^*, \cdots, \boldsymbol{u}_k^*] \in \mathbb{R}^{N\times K},$$

where $r = \sum r_k$ is the sum of ranks of each $\boldsymbol{V}_k^*$, $k = 1, \cdots, K$. The core tensor $\mathcal{S}\mathbb{R}^{r\times r\times K}$ satisfies

$$\mathcal{S}_{ijk}^* = d_{x,k}^* \cdot \mathbb{1}\left\{\sum_{l=1}^{k-1} r_l + 1 \le i = j \le \sum_{l=1}^{k} r_l\right\}.$$

That is, the $k$ slice of the core tensor $\mathcal{S}^*$ is a diagonal matrix with only $r_k$ non-zero diagonal entries.

In addition, we can also write $\mathcal{X}$ under the CP decomposition model $\mathcal{X} = \sum_{i=1}^{r} \lambda_i \cdot \boldsymbol{a}_i \circ \boldsymbol{b}_i \circ \boldsymbol{c}_i + \mathcal{E}_x$ with $r = \sum_{k=1}^{K} r_k$. If we further denote $\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_r)$, $\boldsymbol{A} = [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_r], \boldsymbol{B} = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_r]$ and

$\boldsymbol{C} = [\boldsymbol{c}_1, \cdots, \boldsymbol{c}_r]$, they would satisfy the following:

$$\begin{aligned}
&\boldsymbol{\lambda} = (\underbrace{d_{x,1}, \cdots, d_{x,1}}_{r_1}, \underbrace{d_{x,2}, \cdots, d_{x,2}}_{r_2}, \cdots, \underbrace{d_{x,k}, \cdots, d_{x,k}}_{r_k}) \in \mathbb{R}^r, \\
&\boldsymbol{A} = \boldsymbol{B} = [\boldsymbol{V}_1, \cdots, \boldsymbol{V}_K] \in \mathbb{R}^{p \times r}, \\
&\boldsymbol{U} = [\underbrace{u_1, \cdots, u_1}_{r_1}, \underbrace{u_2, \cdots, u_2}_{r_2}, \cdots, \underbrace{u_k, \cdots, u_k}_{r_k}] \in \mathbb{R}^{N \times r}.
\end{aligned}$$

In summary, both the CP and Tucker low-rank models are closely related to our model; however, as discussed in Section 1.2, they both add additional, undesirable constraints (orthogonality or incoherence) to the factors and hence fall short for our network modeling purposes.

### A.3 Additional Notations

Here, we provide additional details of some notations briefly introduced in Section 2.1. Suppose $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ is a general third-order tensor, for a matrix $\boldsymbol{U} \in \mathbb{R}^{p_1 \times r_1}$, the (marginal) multiplication $\times_1$ of tensor and matrix is $\mathcal{X} \times_1 \boldsymbol{U} \in \mathbb{R}^{r_1 \times p_2 \times p_3}$ satisfying:

$$(\mathcal{X} \times_1 \boldsymbol{U})_{i,j,k} = \sum_{l=1}^{p_1} \mathcal{X}_{ljk} \boldsymbol{U}_{il}.$$

And $\times_2$, $\times_3$ can be defined similarly. The matricization of $\mathcal{X}$ by the $k$th-mode $\mathcal{M}_k(\mathcal{X})$ is a $p_k \times p_{-k}$ matrix, where $p_{-k} = \prod_{i \neq k} p_i$. Elementwisely, $\mathcal{M}_k(\mathcal{X})$ can be written as

$$[\mathcal{M}_1(\mathcal{X})]_{i,(k-1)p_2+j} = \mathcal{X}_{ijk},\ [\mathcal{M}_2(\mathcal{X})]_{j,(i-1)p_3+k} = \mathcal{X}_{ijk},\ [\mathcal{M}_3(\mathcal{X})]_{k,(j-1)p_1+i} = \mathcal{X}_{ijk},$$

where $1 \leq i \leq p_1$, $1 \leq j \leq p_2$, $1 \leq k \leq p_3$. For two tensors of the same order and dimension, $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$, the inner product of tensors is $\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \sum_{k=1}^{p_3} \mathcal{X}_{ijk} \mathcal{Y}_{ijk}$. And tensor Frobenius norm is the square root of the inner product with itself, i.e. $\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle} = \sqrt{\sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \sum_{k=1}^{p_3} \mathcal{X}_{ijk}^2}$. We follow the definition of sub-Gaussian random variables in (Wainwright, 2019) and say that $X$ is sub-Gaussian-$\sigma$ if and only if $\mathbb{E}[e^{\lambda(X - \mathbb{E}X)}] \leq e^{\lambda^2 \sigma^2 / 2}$ for all $\lambda \in \mathbb{R}$.

### A.4 Additional Algorithms & Details

In this section, we provide the detailed additional algorithms mentioned in the main paper, including the multi-factor JisstPCA (with and without BIC-based rank selection), different deflation schemes one can apply, matrix-tensor JisstPCA, selection strategies for the number of factors $K$ and weight parameter $\lambda \in (0, 1)$, and our comparison baselines iHOOI and iHOSVD (integrated versions of HOOI and HOSVD).

**Algorithm 2:** Multi-factor JisstPCA with Subtraction Deflation and Prespecified Ranks

- Input: $\mathcal{X}, \mathcal{Y}$, number of factors $K$, $\boldsymbol{r}_x$, $\boldsymbol{r}_y \in \mathbb{R}^K$, and maximum iteartion $t_{\max}$
- Initialization: Let $k = 1$, $\mathcal{X}^1 = \mathcal{X}$, and $\mathcal{Y}^1 = \mathcal{Y}$.
- While $k \leq K$:
  - Let $\lambda = \frac{\|\mathcal{X}^{(k)}\|_F}{\|\mathcal{X}^{(k)}\|_F + \|\mathcal{Y}^{(k)}\|_F}$.
  - Apply Single-Factor JisstPCA (Algorithm 1) on $\mathcal{X}^{(k)}$, $\mathcal{Y}^{(k)}$, with ranks $r_{x,k}$, $r_{y,k}$, $\lambda$, and maximum iteration $t_{\max}$ to obtain $\hat{d}_{x,k}, \hat{d}_{y,k}, \hat{\boldsymbol{V}}_k, \hat{\boldsymbol{W}}_k, \hat{\boldsymbol{u}}_k$.
  - Apply subtract deflation to obtain $\mathcal{X}^{k+1}, \mathcal{Y}^{k+1}$ as

$$\begin{aligned}\mathcal{X}^{k+1} &= \mathcal{X}^k - \hat{d}_k^x \cdot \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k' \circ \hat{\boldsymbol{u}}_k \\ \mathcal{Y}^{k+1} &= \mathcal{Y}^k - \hat{d}_k^y \cdot \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k' \circ \hat{\boldsymbol{u}}_k.\end{aligned}$$

  - $k = k + 1$.
- **return** $\{\hat{\boldsymbol{u}}_k, \hat{\boldsymbol{V}}_k, \hat{\boldsymbol{W}}_k, \hat{d}_{x,k}, \hat{d}_{y,k}\}_{k=1}^K$.

### A.4.1 Multi-factor PCA

The subtraction deflation (4) is computationally efficient and imposes no extra constraint on different factors, such as orthogonality. However, in some cases, orthogonality on certain modes might be desirable due to prior belief or for interpretation purposes. To accommodate for this, one might consider the projection deflation (Mackey, 2008) where the tensor data is projected onto the orthogonal complement of the space spanned by previously estimated factors. In particular, when orthogonality is required for both the joint factor ($\boldsymbol{u}_i \perp \boldsymbol{u}_j$ for $i \neq j$) and individual factors ($\boldsymbol{V}_i \perp \boldsymbol{V}_j$, $\boldsymbol{W}_i \perp \boldsymbol{W}_j$, for $i \neq j$), we let

$$\begin{aligned}\mathcal{X}^{k+1} &= \mathcal{X}^k \times_1 \left(\boldsymbol{I}_p - \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k'\right) \times_2 \left(\boldsymbol{I}_p - \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k'\right) \times_3 \left(\boldsymbol{I}_N - \hat{\boldsymbol{u}}_k \hat{\boldsymbol{u}}_k'\right) \\ \mathcal{Y}^{k+1} &= \mathcal{Y}^k \times_1 \left(\boldsymbol{I}_q - \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k'\right) \times_2 \left(\boldsymbol{I}_q - \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k'\right) \times_3 \left(\boldsymbol{I}_N - \hat{\boldsymbol{u}}_k \hat{\boldsymbol{u}}_k'\right).\end{aligned} \tag{12}$$

Furthermore, one can also enforce pairwise orthogonality only for the joint factors or for the individual factors through a "partial projection deflation" scheme.

$$\begin{aligned}\mathcal{X}^{k+1} &= \left(\mathcal{X}^k - \hat{d}_{x,k} \cdot \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k' \circ \hat{\boldsymbol{u}}_k\right) \times_3 \left(\boldsymbol{I}_N - \hat{\boldsymbol{u}}_k \hat{\boldsymbol{u}}_k'\right) \\ \mathcal{Y}^{k+1} &= \left(\mathcal{Y}^k - \hat{d}_{y,k} \cdot \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k' \circ \hat{\boldsymbol{u}}_k\right) \times_3 \left(\boldsymbol{I}_N - \hat{\boldsymbol{u}}_k \hat{\boldsymbol{u}}_k'\right);\end{aligned} \tag{13}$$

Or, when only the individual factors need to be mutually orthogonal, we can similarly let

$$\begin{aligned}\mathcal{X}^{k+1} &= \left(\mathcal{X}^k - \hat{d}_{x,k} \cdot \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k' \circ \hat{\boldsymbol{u}}_k\right) \times_1 \left(\boldsymbol{I}_p - \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k'\right) \times_2 \left(\boldsymbol{I}_p - \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k'\right) \\ \mathcal{Y}^{k+1} &= \left(\mathcal{Y}^k - \hat{d}_{y,k} \cdot \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k' \circ \hat{\boldsymbol{u}}_k\right) \times_1 \left(\boldsymbol{I}_q - \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k'\right) \times_2 \left(\boldsymbol{I}_q - \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k'\right).\end{aligned} \tag{14}$$

**Remark 3.** *Applying multi-factor JisstPCA with projection deflation* (12) *yield mutually orthogonal factors:* $\hat{\boldsymbol{u}}_i \perp \hat{\boldsymbol{u}}_j$ *for* $i \neq j$*; when using the partial projection deflation* (5) *or* (14)*, the factors with projection would be mutually orthogonal.*

### A.4.2 Selection of Hyperparameters

Our model includes a number of hyperparameters that must be specified or tuned in practice: the number of factors $K$, the ranks of each factor $r_{x,k}$, $r_{y,k}$, $1 \leq k \leq K$, and the integrative scaling parameter $\lambda$. First, there are many widely employed approaches to estimate the number of PCA factors, $K$ (Wold, 1978; Jolliffe and Cadima, 2016; Dobriban and Owen, 2019; Donoho et al., 2023); these methods are also applicable for JisstPCA. Noting that $\mathbb{E}\mathcal{M}_3(\mathcal{X}) = \mathcal{M}_3(\mathcal{X}^*)$ and $\text{rank}(\mathcal{M}_3(\mathcal{X}^*)) = K$, one possible way to estimate $K$ is to apply usual PCA-type methods on $\mathcal{M}_3(\mathcal{X})$. Another possible approach to estimate $K$ is based on the cumulative proportion of variance explained by the factors. Note that the proportion of variance explained by each single factor is given by $d_{x,k}^2/\|\mathcal{X}\|_F^2$ for $\mathcal{X}$ and $d_{y,k}^2/\|\mathcal{Y}\|_F^2$ for $\mathcal{Y}$. But as the components are not orthogonal across factors (unless projection deflation is employed), we cannot simply add the variance explained by each factor to get the cumulative variance explained (Allen, 2012b). Instead, we must calculate this by projecting out the effect of all components from the $k$ factors:

**Remark 4.** *Let* $\mathbf{P}_k^{(\mathbf{U})} = \mathbf{U}_k(\mathbf{U}_k'\mathbf{U}_k)^{-1}\mathbf{U}_k'$*, with* $\mathbf{U}_k = [\mathbf{u}_1, \ldots, \mathbf{u}_k]$*, and let* $\mathbf{P}_k^{(\mathbf{V})} = \tilde{\mathbf{V}}_k(\tilde{\mathbf{V}}_k'\tilde{\mathbf{V}}_k)^{-1}\tilde{\mathbf{V}}_k'$*, with* $\tilde{\mathbf{V}}_k = [\mathbf{V}_1, \ldots, \mathbf{V}_k]$*; define* $\mathbf{P}_k^{(\mathbf{W})}$ *and* $\tilde{\mathbf{W}}_k$ *analogously. Then, the cumulative proportion of variance explained by the first k JisstPCA factors in* $\mathcal{X}$ *and* $\mathcal{Y}$ *is given by* $\frac{\|\mathcal{X}\times_1\mathbf{P}_k^{(\mathbf{V})}\times_2\mathbf{P}_k^{(\mathbf{V})}\times_3\mathbf{P}_k^{(\mathbf{U})}\|_F^2}{\|\mathcal{X}\|_F^2}$ *and* $\frac{\|\mathcal{Y}\times_1\mathbf{P}_k^{(\mathbf{W})}\times_2\mathbf{P}_k^{(\mathbf{W})}\times_3\mathbf{P}_k^{(\mathbf{U})}\|_F^2}{\|\mathcal{Y}\|_F^2}$ *respectively.*

Second, $\lambda$ is important when $\mathcal{X}$ and $\mathcal{Y}$ have different scales. To study the effect of $\lambda$, we generate data from two-factor JisstPCA models with different SNR levels and apply JisstPCA with a range of $\lambda$. The detailed set-up is the same as the unstructured simulation in Section 4 in the main paper, where the SNR of both tensors $\mathcal{X}$ and $\mathcal{Y}$ are the same. The factor estimation errors are summarized in Figure 6, suggesting that when both modalities have similar SNR levels, $\lambda = \dfrac{\|\mathcal{X}^*\|_F}{\|\mathcal{X}^*\|_F + \|\mathcal{Y}^*\|_F}$ is likely the optimal choice (red line). Without the knowledge of the true tensor norms, we suggest using $\lambda = \dfrac{\|\mathcal{X}\|_F}{\|\mathcal{X}\|_F + \|\mathcal{Y}\|_F}$ as a surrogate (blue line); this is also our default selection of $\lambda$ throughout the empirical studies in this paper. We can also see from Figure 6 that JisstPCA is not sensitive to the choice of $\lambda$ as long as we do not use extreme value. when both modalities have similar noise levels, we suggest using $\lambda = \dfrac{\|\mathcal{X}\|_F}{\|\mathcal{X}\|_F + \|\mathcal{Y}\|_F}$; we employ this scheme in all our empirical studies.

Finally, selecting the ranks of each factor is more complicated and this choice has more influence on the results. In the multi-factor ($K > 1$) case, we need to select $2K$ total ranks. Many prior works approach this problem by finding the minimizer of the Bayesian information criterion (BIC) (Allen, 2012a; Zhou et al., 2022; Hu et al., 2022). However, directly minimizing the BIC for all $2K$ ranks simultaneously is a combinatoral problem and computationally infeasible for large $K$. Instead, and inspired by (Allen, 2012a), we propose a "single-factor BIC + deflation" scheme, called

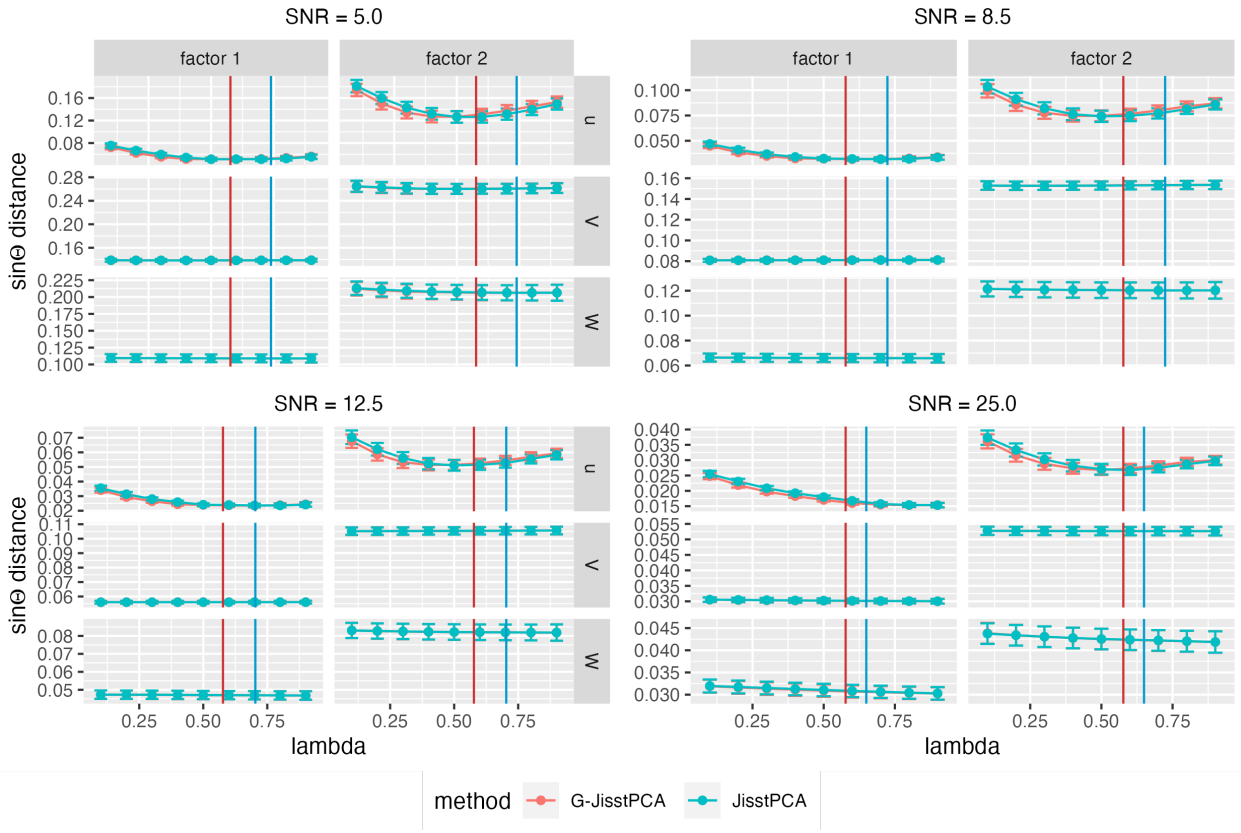


Figure 6: The effect of $\lambda$ values on the estimation errors ($\sin\Theta$ distances in spectral norm) of all factors using JisstPCA and G-JisstPCA. The red vertical line is the conjectured oracle $\lambda = \frac{\|\mathcal{X}^*\|_F}{\|\mathcal{X}^*\|_F+\|\mathcal{Y}^*\|_F}$ when the noise level is the same, and the blue vertical line is its estimate: $\frac{\|\mathcal{X}\|_F}{\|\mathcal{X}\|_F+\|\mathcal{Y}\|_F}$. The estimated $\lambda$ is closer to the oracle $\lambda$ when SNR increases; in addition, the performance of (G-)JisstPCA seems robust when changing $\lambda$.

"BIC deflation". This greedy approach successively applies single-factor BIC to select the rank for each factor, and then single-factor JisstPCA with the selected rank and deflation to get residual data for extracting the next factor.

In particular, given data $\mathcal{X}$, $\mathcal{Y}$ and potential single-factor ranks $r_x$, $r_y$, we first apply single-factor JisstPCA to obtain tensor estimates $\hat{\mathcal{X}}(r_x)$, $\hat{\mathcal{Y}}(r_y)$, and then compute the BIC for this rank combination as

$$\begin{aligned}\mathrm{BIC}(r_x, r_y) = p^2 N \log \|\mathcal{X} - \hat{\mathcal{X}}(r_x)\|_F^2 + q^2 N \log \|\mathcal{Y} - \hat{\mathcal{Y}}(r_y)\|_F^2 \\ + (pr_x + qr_y) \log((p^2+q^2)N) + C(p,q,N).\end{aligned} \tag{15}$$

For a grid of potential rank combinations, we select $(r_x, r_y)$ that minimizes $\mathrm{BIC}(r_x, r_y)$ for the current factor. The detailed "BIC deflation" scheme is summarized in Algorithm 3.

**Algorithm 3:** Multi-factor JisstPCA with Subtraction Deflation and BIC-selected Ranks

- Input: $\mathcal{X}, \mathcal{Y}$, and number of factors $K$, maximum ranks $r_{x,\max} \le p$, $r_{y,\max} \le q$, and maximum iteration $t_{\max}$.
- Initialization: Let $k = 1$, $\mathcal{X}^1 = \mathcal{X}$, and $\mathcal{Y}^1 = \mathcal{Y}$.
- While $k \le K$:
  - Let $\lambda = \frac{\|\mathcal{X}^{(k)}\|_F}{\|\mathcal{X}^{(k)}\|_F + \|\mathcal{Y}^{(k)}\|_F}$.
  - Select rank for factor $k$ via BIC:
    For $i = 1, \cdots, r_{x,\max}$ and $j = 1, \cdots, r_{y,\max}$:
    - Apply single-factor JisstPCA (Algorithm 1) with ranks $(i, j)$, $\lambda$, and maximum iteration $t_{\max}$.
    - Calculate $\mathrm{BIC}(i, j)$ in equation (15) for $\mathcal{X}^k, \mathcal{Y}^k$.

    Let $(\hat{r}_{x,k}, \hat{r}_{y,k}) = \underset{i \in \{1,\cdots,r_{x,\max}\}, j \in \{1,\cdots,r_{y,\max}\}}{\arg\min} \mathrm{BIC}(i,j)$, and save the output of single-factor JisstPCA with ranks $(\hat{r}_{x,k}, \hat{r}_{y,k})$ as $\hat{d}_{x,k}, \hat{d}_{y,k}, \hat{\boldsymbol{V}}_k, \hat{\boldsymbol{W}}_k, \hat{\boldsymbol{u}}_k$.
  - Apply subtract deflation to obtain $\mathcal{X}^{k+1}, \mathcal{Y}^{k+1}$ as

$$\begin{aligned}\mathcal{X}^{k+1} &= \mathcal{X}^k - \hat{d}_k^x \cdot \hat{\boldsymbol{V}}_k \hat{\boldsymbol{V}}_k' \circ \hat{\boldsymbol{u}}_k \\ \mathcal{Y}^{k+1} &= \mathcal{Y}^k - \hat{d}_k^y \cdot \hat{\boldsymbol{W}}_k \hat{\boldsymbol{W}}_k' \circ \hat{\boldsymbol{u}}_k.\end{aligned}$$

  - $k = k + 1$.
- **return** $\hat{\boldsymbol{r}}^x = (\hat{r}_{x,1}, \cdots, \hat{r}_{x,K})$, $\hat{\boldsymbol{r}}_y = (\hat{r}_{y,1}, \cdots, \hat{r}_{y,K})$, $\{\hat{\boldsymbol{u}}_k, \hat{\boldsymbol{V}}_k, \hat{\boldsymbol{W}}_k, \hat{d}_{x,k}, \hat{d}_{y,k}\}_{k=1}^K$.

### A.4.3 Extensions of JisstPCA

Here, we provide more details on the matrix-tensor JisstPCA model, the generalized JisstPCA model, as well as their associated algorithms. We focus on the single-factor algorithms, while the multi-factor extensions of them can be similarly defined as our deflation-procedure in Algorithm 2.

**Matrix-Tensor JisstPCA** One important extension of our JisstPCA framework is to jointly analyze vector-valued covariates, organized as a matrix, together with network data organized as tensors. In neuroimaging studies, for example, we may additionally observe feature vectors such as genomics, demographic, or behavioral traits for each subject. In this case, we would like to find joint population factors shared between the network data and vector-valued covariates, as well as individual low-rank factors for networks and for features separately. Specifically, suppose we observe a semi-symmetric tensor $\mathcal{X} \in \mathbb{R}^{p\times p\times N}$ representing a population of networks, and a matrix $\boldsymbol{Y} \in \mathbb{R}^{q\times N}$ representing features measured for the same population. We consider the following joint matrix-tensor PCA model:

$$\mathcal{X} = \sum_{k=1}^{K} d_{x,k} \cdot \boldsymbol{V}_k^* \boldsymbol{V}_k^{*\prime} \circ \boldsymbol{u}_k^* + \mathcal{E}_x, \quad \boldsymbol{Y} = \sum_{k=1}^{K} d_{y,k} \cdot \boldsymbol{w}_k^* \boldsymbol{u}_k^{*'} + \boldsymbol{E}_y, \tag{16}$$

where $\mathcal{E}_x \in \mathbb{R}^{p\times p\times N}$, $\boldsymbol{E}_y \in \mathbb{R}^{p\times N}$ are observational noise, $\boldsymbol{u}_k$ is the population factor, and $\boldsymbol{w}_k^* \in \mathbb{S}^{q-1}$ is the $k$th feature factor associated with the network factor $\boldsymbol{V}_k^*$. By contrasting $\boldsymbol{w}_k^*$ with $\boldsymbol{V}_k^*$, one can establish a connection between the features and the network, which in the neuroimaging example corresponds to how certain gene expression or human behavior is associated with the brain connectome. As in the JisstPCA model, we do not impose orthogonality constraints on $\boldsymbol{w}_k^*$'s. Although orthogonality is required for single matrix decomposition to ensure identifiability, it is not necessary for (16), since matrix $\boldsymbol{Y}$ shares factors $\boldsymbol{u}_k^*$ with tensor $\mathcal{X}$.

As discussed earlier in Section 1.2, although there are many prior methods that jointly factorize matrices and tensors (Acar et al., 2011, 2014; Fu et al., 2015; Wu et al., 2018; Schenker et al., 2020), they are often based on the CP decomposition, which could be less appropriate for our study of tensor networks. Under model (16), in order to estimate factors $\boldsymbol{V}_k^*$, $\boldsymbol{w}_k^*$, and $\boldsymbol{u}_k^*$ from noisy observations $\mathcal{X}$, $\mathcal{Y}$, we consider a similar strategy to the JisstPCA algorithm, which consists of successive deflation and single-factor PCA based on joint power iteration. The detailed single-factor matrix-tensor JisstPCA algorithm is summarized in Algorithm 4.

**Generalized JisstPCA** In addition, we consider a generalized JisstPCA model to accommodate for potential different eigenvalues within each network factor. Specifically, we extend (1) to the following:

$$\mathcal{X} = \sum_{k=1}^{K} \boldsymbol{V}_k^* \boldsymbol{D}_{x,k}^* \boldsymbol{V}_k^{*\prime} \circ \boldsymbol{u}_k^* + \mathcal{E}_x, \quad \mathcal{Y} = \sum_{k=1}^{K} \boldsymbol{W}_k^* \boldsymbol{D}_{y,k}^* \boldsymbol{W}_k^{*\prime} \circ \boldsymbol{u}_k^* + \mathcal{E}_y. \tag{17}$$

where $\boldsymbol{D}_{x,k}^* \in \mathbb{R}^{r_{x,k}\times r_{x,k}}$ and $\boldsymbol{D}_{y,k}^* \in \mathbb{R}^{r_{y,k}\times r_{y,k}}$ are diagonal matrices, replacing the scalar eigenvalues $d_{x,k}^*$, $d_{y,k}^*$ in (1). To estimate factors in this general JisstPCA model, we still propose a power

**Algorithm 4:** Single-factor Matrix-Tensor JisstPCA

- Input: $\mathcal{X}$, $\boldsymbol{Y}$, $r_x$, $\lambda$, and maximum iteration $t_{\max}$.
- Initialization: Let $t = 0$, and $\boldsymbol{u}^{(0)} =$ Leading left singular vector of $[\lambda \mathcal{M}_3(\mathcal{X}), (1-\lambda)\boldsymbol{Y}']$.
- **repeat** until $t = t_{\max}$ or convergence:

$$\begin{aligned}
\boldsymbol{V}^{(t+1)} &= \text{Leading } r_x \text{ left singular vectors of } \mathcal{X} \times_3 \boldsymbol{u}^{(t)} \\
\boldsymbol{w}^{(t+1)} &= \frac{\boldsymbol{Y}\boldsymbol{u}^{(t)}}{\|\boldsymbol{Y}\boldsymbol{u}^{(t)}\|_2} \\
\boldsymbol{u}^{(t+1)} &= \frac{\lambda[\mathcal{X}; \boldsymbol{V}^{(t+1)}] + (1-\lambda)\boldsymbol{Y}'\boldsymbol{w}^{(t+1)}}{\left\|\lambda[\mathcal{X}; \boldsymbol{V}^{(t+1)}] + (1-\lambda)\boldsymbol{Y}'\boldsymbol{w}^{(t+1)}\right\|_2} \\
t &= t+1
\end{aligned}$$

- **return** $\hat{\boldsymbol{u}} = \boldsymbol{u}^{(t)}$, $\hat{\boldsymbol{V}} = \boldsymbol{V}^{(t)}$, $\hat{\boldsymbol{w}} = \boldsymbol{w}^{(t)}$; $\hat{d}_x = \langle \mathcal{X}, \hat{\boldsymbol{V}}\hat{\boldsymbol{V}}' \circ \hat{\boldsymbol{u}} \rangle / r_x$, $\hat{d}_y = \hat{\boldsymbol{w}}'\boldsymbol{Y}\hat{\boldsymbol{u}}$; $\hat{\mathcal{X}} = \hat{d}_x \cdot \hat{\boldsymbol{V}}\hat{\boldsymbol{V}}' \circ \hat{\boldsymbol{u}}$, $\hat{\boldsymbol{Y}} = \hat{d}_y \cdot \hat{\boldsymbol{w}} \circ \hat{\boldsymbol{u}}$.

iteration algorithm for the single-factor case and apply a successive deflation scheme for multi-factor models. The main change we make to the single-factor JisstPCA algorithm is that we update the diagonal matrix $\boldsymbol{D}_x$ and $\boldsymbol{D}_y$ within each iteration due to its increased importance, and we use them weight the columns of $\boldsymbol{V}$, $\boldsymbol{W}$ when updating $\boldsymbol{u}$. We term this new algorithm the generalized JisstPCA (G-JisstPCA) algorithm, whose detailed procedures are summarized in Algorithm 5.

### A.4.4 Comparison Baselines: iHOSVD and iHOOI

To construct comparison baselines for our methods, we consider two straightforward extensions of HOSVD (De Lathauwer et al., 2000a) and HOOI (De Lathauwer et al., 2000b), power method for Tucker model, when we have integrated data. We refer to them as iHOSVD and iHOOI, which simply concatenate the matricized data along the third mode at each iteration. The detailed

**Algorithm 5:** Generalized Single-Factor JisstPCA

- Input: $\mathcal{X}$, $\mathcal{Y}$, $r_x$, $r_y$, $\lambda$, and maximum iteration $t_{\max}$.
- Initialization: Let $t = 0$, and
  $\boldsymbol{u}^{(0)} =$ Leading left singular vector of $[\lambda \mathcal{M}_3(\mathcal{X}), (1-\lambda)\mathcal{M}_3(\mathcal{Y})]$.
- **repeat** until $t = t_{\max}$ or convergence:

$$\begin{aligned}
&\boldsymbol{V}^{(t+1)} = \text{Leading } r_x \text{ left singular vectors of } \mathcal{X} \times_3 \boldsymbol{u}^{(t)} \\
&\boldsymbol{W}^{(t+1)} = \text{Leading } r_y \text{ left singular vectors of } \mathcal{Y} \times_3 \boldsymbol{u}^{(t)} \\
&\boldsymbol{D}_x^{(t+1)} = (\boldsymbol{V}^{(t+1)})'(\mathcal{X} \times_3 \boldsymbol{u}^{(t)})\boldsymbol{V}^{(t+1)} \\
&\boldsymbol{D}_y^{(t+1)} = (\boldsymbol{W}^{(t+1)})'(\mathcal{Y} \times_3 \boldsymbol{u}^{(t)})\boldsymbol{W}^{(t+1)} \\
&\boldsymbol{u}^{(t+1)} = \frac{\lambda[\mathcal{X}; \boldsymbol{V}^{(t+1)}, \boldsymbol{D}_x^{(t+1)}] + (1-\lambda)[\mathcal{Y}; \boldsymbol{W}^{(t+1)}, \boldsymbol{D}_y^{(t+1)}]}{\left\| \lambda[\mathcal{X}; \boldsymbol{V}^{(t+1)}, \boldsymbol{D}_x^{(t+1)}] + (1-\lambda)[\mathcal{Y}; \boldsymbol{W}^{(t+1)}, \boldsymbol{D}_y^{(t+1)}] \right\|_2} \\
&t = t + 1
\end{aligned}$$

- **return** $\hat{\boldsymbol{u}} = \boldsymbol{u}^{(t)}$, $\hat{\boldsymbol{V}} = \boldsymbol{V}^{(t)}$, $\hat{\boldsymbol{W}} = \boldsymbol{W}^{(t)}$; $\hat{\boldsymbol{D}}_x = \boldsymbol{D}_x^{(t)}$, $\hat{\boldsymbol{D}}_y = \boldsymbol{D}_y^{(t)}$; $\hat{\mathcal{X}} = \hat{\boldsymbol{V}}\hat{\boldsymbol{D}}_x\hat{\boldsymbol{V}}' \circ \hat{\boldsymbol{u}}$, $\hat{\mathcal{Y}} = \hat{\boldsymbol{W}}\hat{\boldsymbol{D}}_y\hat{\boldsymbol{W}}' \circ \hat{\boldsymbol{u}}$.

procedures of iHOSVD and iHOOI are summarized in Algorithms 6 and 7.

**Algorithm 6:** Integrated High-Order SVD (iHOSVD)

- Input: $\mathcal{X} \in \mathbb{R}^{p \times p \times N}$, $\mathcal{Y} \in \mathbb{R}^{q \times q \times N}$, $(r_{x_1}, \cdots, r_{x_K})$, $(r_{y_1}, \cdots, r_{y_K})$.
- $\hat{\boldsymbol{V}} =$ The leading $\sum_{k=1}^{K} r_k^x$ singular vectors of $\mathcal{M}_1(\mathcal{X})$.
- $\hat{\boldsymbol{W}} =$ The leading $\sum_{k=1}^{K} r_k^y$ singular vectors of $\mathcal{M}_1(\mathcal{Y})$.
- $\hat{\boldsymbol{U}} =$ The leading $K$ singular vectors of the matrix $(\mathcal{M}_3(\mathcal{X}), \mathcal{M}_3(\mathcal{Y}))$.
- The estimation of Tucker core tensors are

$$\begin{aligned}
\hat{\mathcal{S}}_x &= \mathcal{X} \times_1 \hat{\boldsymbol{V}}' \times_2 \hat{\boldsymbol{V}}' \times_3 \hat{\boldsymbol{U}}' \\
\hat{\mathcal{S}}_y &= \mathcal{Y} \times_1 \hat{\boldsymbol{W}}' \times_2 \hat{\boldsymbol{W}}' \times_3 \hat{\boldsymbol{U}}'.
\end{aligned}$$

  And then the estimation of true parameter tensors are

$$\begin{aligned}
\hat{\mathcal{X}} &= \hat{\mathcal{S}}_x \times_1 \hat{\boldsymbol{V}} \times_2 \hat{\boldsymbol{V}} \times_3 \hat{\boldsymbol{U}} \\
\hat{\mathcal{Y}} &= \hat{\mathcal{S}}_y \times_1 \hat{\boldsymbol{W}} \times_2 \hat{\boldsymbol{W}} \times_3 \hat{\boldsymbol{U}}.
\end{aligned}$$

- **return** $\hat{\boldsymbol{u}}_i = \hat{\boldsymbol{U}}_{\cdot,i}$, $\hat{\boldsymbol{V}}_i = \hat{\boldsymbol{V}}_{\cdot, 1+\sum_{k=1}^{i-1} r_k^x : \sum_{k=1}^{i} r_k^x}$, $\hat{\boldsymbol{W}}_i = \hat{\boldsymbol{W}}_{\cdot, 1+\sum_{k=1}^{i-1} r_k^y : \sum_{k=1}^{i} r_k^y}$ for $i = 1, \cdots, K$ as the corresponding estimated factors of multi-factor semi-symmetric tensor. And $\hat{\mathcal{X}}$, $\hat{\mathcal{Y}}$.



**Algorithm 7:** Integrated High-Order Orthogonal Iteration (iHOOI)

## A.5 Additional Empirical Details and Results

We first provide some additional details on the simulation setup of our comparative studies.

1. Setup of the structured simulation presented in the bottom left panel of Figure 2: $K = 2$, $\boldsymbol{r}_x = \boldsymbol{r}_y = (3, 2)'$. The ground truth factors are generated as follows.

   (a) To generate the true population factors $\boldsymbol{u}_1^*$ and $\boldsymbol{u}_2^*$, we consider a Gaussian mixture model with three components, where the mean of three components $\mu_k$, $k = 1, 2, 3$, are generated from $U[0, 1]$. We first randomly assign each sample $i$ into one of the three clusters, say $k$, and then generate $((\boldsymbol{u}_1^*)_i, (\boldsymbol{u}_2^*)_i)$ from $\mathcal{N}(\mu_k, 0.05)$. We then normalize both $\boldsymbol{u}_1^*$ and $\boldsymbol{u}_2^*$ to unit vectors.

   (b) To generate the true network factors in the first layer, $\boldsymbol{V}_1^* \in \mathbb{R}^{p\times 3}$ and $\boldsymbol{W}_1^* \in \mathbb{R}^{q\times 3}$, we consider a three-block graph structure. In particular, each row $i$ of $\boldsymbol{V}_1^*$ ($\boldsymbol{W}_1^*$) is randomly assigned to one of the three blocks, say $k$, and then we generate $(\boldsymbol{V}_1^*)_{i,k}$ from $\mathcal{N}(3, 1)$ and let $(\boldsymbol{V}_1^*)_{i,\setminus k} = 0$. Both $\boldsymbol{V}_1^*$ and $\boldsymbol{W}_1^*$ are then orthogonalized individually.

   (c) To generate the true network factors in the second layer, $\boldsymbol{V}_2^* \in \mathbb{R}^{p\times 2}$ and $\boldsymbol{W}_2^* \in \mathbb{R}^{q\times 2}$, we consider two-star graphs of $p$ and $q$ nodes, respectively. We randomly assign each of the $p$ or $q$ nodes into two components, each being a star graph, and then compute the top two singular vectors of the Laplacian matrices for the two-star graphs. These singular vectors yield are $\boldsymbol{V}_2^*$ and $\boldsymbol{W}_2^*$, respectively.

   (d) The signal strength $\boldsymbol{d}_x^*$, $\boldsymbol{d}_y^*$ are set as described in Section 4, which are the same across structured and unstructured simulations.

2. Setup of the structured simulation example in Figure 3 is slightly different from Figure 2. In particular, for clearer visualization, we set $p = q = 50$ and $N = 200$, but the network factors are still generated from three-block graphs, two-star graphs, and the population factors are generated from a Gaussian mixture with three components.

3. Setup of the generalized models: we consider two settings of the diagonal matrices $\boldsymbol{D}_{x,k}^*$, $\boldsymbol{D}_{y,k}^*$.

   (a) For the results presented in Figure 2 and the setting 1 in Figure 13 and 14, we set $\boldsymbol{D}_{x,1}^* = \text{SNR} * (\sqrt{p} + \sqrt{N})\text{diag}([2, 1.5, 1.2])$, $\boldsymbol{D}_{x,2}^* = \text{SNR} * (\sqrt{p} + \sqrt{N})\text{diag}([1, 0.8])$; $\boldsymbol{D}_{y,1}^* = \text{SNR} * (\sqrt{q} + \sqrt{N})\text{diag}([2, 1.5, 1.2])$, $\boldsymbol{D}_{y,2}^* = \text{SNR} * (\sqrt{q} + \sqrt{N})\text{diag}([1, 0.8])$. The eigenvalues are moderately different from each other within each factor.

   (b) We also consider a more difficult setting where $\boldsymbol{D}_{x,1}^* = \text{SNR}*(\sqrt{p}+\sqrt{N})\text{diag}([3.2, 2, 1.2])$, $\boldsymbol{D}_{x,2}^* = \text{SNR} * (\sqrt{p} + \sqrt{N})\text{diag}([1, 0.8])$; $\boldsymbol{D}_{y,1}^* = \text{SNR} * (\sqrt{q} + \sqrt{N})\text{diag}([3.2, 2, 1.2])$, $\boldsymbol{D}_{y,2}^* = \text{SNR} * (\sqrt{q} + \sqrt{N})\text{diag}([1, 0.8])$.. The results are presented as the setting 2 in Figure 13 and 14.

4. Setup of the network simulations: we construct two pairs of stochastic block models (SBMs) for modeling the multi-modal networks, and each sample $i$ falls within each of the two pairs

with probability 0.75 and 0.25. For the first pair, the two SBMs are of size $p$, $q$ with three blocks of sizes $0.4p$, $0.3p$, $0.3p$ and $0.4q$, $0.4q$, $0.2q$, respectively. For the second pair, the two SBMs have two blocks with sizes $0.5p$, $0.5p$ and $0.6q$, $0.4q$ respectively. The top panel of Figure 16 shows an example of the block structures of the two pairs of SBMs when $p = q = 80$. The within-block connection probabilities range from 0.5 to 0.8, and the out-of-block probabilities are set as 0.3. Formally, the two pairs of SBM probability matrices can be denoted as $(\boldsymbol{P}_{x,1}, \boldsymbol{P}_{y,1})$, $(\boldsymbol{P}_{x,2}, \boldsymbol{P}_{y,2})$, where $\boldsymbol{P}_{x,k} = \boldsymbol{\Theta}_{x,k}\boldsymbol{B}_{x,k}\boldsymbol{\Theta}'_{x,k}$, $\boldsymbol{P}_{y,k} = \boldsymbol{\Theta}_{y,k}\boldsymbol{B}_{y,k}\boldsymbol{\Theta}'_{y,k}$. Specifically, $\boldsymbol{\Theta}_{x,1} = \begin{pmatrix} 1_{0.4p} & 0 & 0 \\ 0 & 1_{0.3p} & 0 \\ 0 & 0 & 1_{0.3p} \end{pmatrix}$, $\boldsymbol{B}_{x,1} = \begin{pmatrix} 0.8 & 0.3 & 0.3 \\ 0.3 & 0.8 & 0.3 \\ 0.3 & 0.3 & 0.8 \end{pmatrix}$; $\boldsymbol{\Theta}_{x,2} = \begin{pmatrix} 1_{0.3p} & 0 \\ 0 & 1_{0.2p} \\ 1_{0.2p} & 0 \\ 0 & 1_{0.3p} \end{pmatrix}$, $\boldsymbol{B}_{x,2} = \begin{pmatrix} 0.6 & 0.3 \\ 0.3 & 0.6 \end{pmatrix}$; $\boldsymbol{\Theta}_{y,1} = \begin{pmatrix} 1_{0.4q} & 0 & 0 \\ 0 & 1_{0.4q} & 0 \\ 0 & 0 & 1_{0.2q} \end{pmatrix}$, $\boldsymbol{B}_{y,1} = \begin{pmatrix} 0.7 & 0.3 & 0.3 \\ 0.3 & 0.7 & 0.3 \\ 0.3 & 0.3 & 0.7 \end{pmatrix}$; $\boldsymbol{\Theta}_{y,2} = \begin{pmatrix} 1_{0.3p} & 0 \\ 0 & 1_{0.4p} \\ 1_{0.3p} & 0 \end{pmatrix}$, $\boldsymbol{B}_{y,2} = \begin{pmatrix} 0.5 & 0.3 \\ 0.3 & 0.5 \end{pmatrix}$.

Next, we give more implementation details of our methods in the simulation studies. For the hyperparameter selection of our JisstPCA and G-JisstPCA methods, we choose $K$ as the true number of factors, $\lambda = \frac{\|\mathcal{X}\|_F}{\|\mathcal{X}\|_F + \|\mathcal{Y}\|_F}$, and we select ranks using the BIC-deflation scheme described in Algorithm 3, with input rank range from 1 to 5. For the comparative studies with JisstPCA or G-JisstPCA models, subtraction deflation is applied for all the results presented in the main paper, while the results of projection deflation are also included in the Figures 7 and 8. Partial projection deflation (projection on the population mode) and subtraction deflation are used for the network simulation. For the clustering experiments in network simulations, we apply the k-means function in Matlab, which uses the squared Euclidean distance metric and the k-means++ algorithm for cluster center initialization.

Finally, we present and discuss some additional empirical results briefly mentioned in the main paper.

1. Figures 7 and 8 present more detailed results for the settings in Figure 2. In particular, both subtraction deflation and projection deflation versions of (Generalized) JisstPCA are presented; Subtraction deflation turns out to be the best across different settings. When using BIC tuned ranks (Figure 8), (Generalized) JisstPCA sometimes give larger errors with low SNR. When using oracle ranks as iHOOI and iHOSVD, our methods are always the best.

2. Figure 9 and 10 focus on the non-orthogonal, unstructured factor setting. Baseline methods include iHOSVD and iHOOI with oracle ranks; our methods include JisstPCA and Generalized JisstPCA with oracle ranks and BIC-selected ranks, when the deflation strategy is subtraction or projection. The results suggest that

   - When the factors are non-orthogonal, our JisstPCA and G-JisstPCA algorithms with

subtraction deflation works the best. iHOOI, iHOSVD and JisstPCA with orthogonal deflation makes wrong assumptions so they all have larger errors, but for some reasons, JisstPCA is still slightly better.

- When SNR is small, sometimes the BIC selected ranks are inaccurate, leading to $\sin\Theta$ distance 1. But when all methods use oracle ranks, JisstPCA and G-JisstPCA with subtraction deflation work the best.

3. Figure 11 and 12 focus on a setting where factors are mutually orthogonal, unstructured, and has sufficient singular gap. In particular, we set $\boldsymbol{d}_x^* = \text{SNR} * (\sqrt{p} + \sqrt{N})(1, 0.5)'$, $\boldsymbol{d}_y^* = \text{SNR} * (\sqrt{q} + \sqrt{N})(1, 0.5)'$ as in the non-orthogonal simulations. The ground truth tensors are Tucker low-rank tensors and iHOOI and iHOSVD are designed to perform well in this setting. We can see that our methods behave comparably to iHOOI and iHOSVD. Also note that our methods with subtraction deflation has the same performance as projection deflation, suggesting it to be a safe scheme to use when it is unclear if the factors should be orthogonal or not.

4. Effects of dimensionality: Figure 9-12 study the effect of dimenisonalities on the estimation accuracy of different factors. Four cases of different dimensions are considered: $p = 50$, $q = 50$, $N = 200$ (case 1); $p = 150$, $q = 150$, $N = 50$ (case 2); $p = 150$, $q = 50$, $N = 200$ (case 3); $p = 150$, $q = 50$, $N = 50$ (case 4). The results suggest that

   - when $N$ is larger, $u$ and $V$ tend to be estimated better;
   - when $p > q$, $V$ tends to be estimated better.

5. Figure 13 and 14 consider two different generalized model settings with (i) different but similar eigenvalues within each layer, and (ii) more different eigenvalues within each layer, as described earlier in the simulation setup. The factors are not orthogonal across factors. We have the following observations:

   - When eigenvalues within each layer are not that different, both JisstPCA and G-JisstPCA with subtraction deflation work very well.
   - When eigenvalues within each layer are very different, JisstPCA with oracle ranks and subtraction deflation fails miserably on the second factor.
   - Interestingly, JisstPCA with projection deflation performs better in this case, probably since it left less residuals from the first factor. G-JisstPCA with subtraction deflation works the best since it assumes the correct model for the generated data.

6. Network visualization: We visualize the true factors and the estimated factors using different methods in Figure 16. The ground truth population factor is the normalized cluster membership vector. To obtain the ground truth network factors, we take the edge probability matrix of each network component, project both its rows and columns onto the orthogonal complement of the all-one's vector, and then take the top singular vectors.

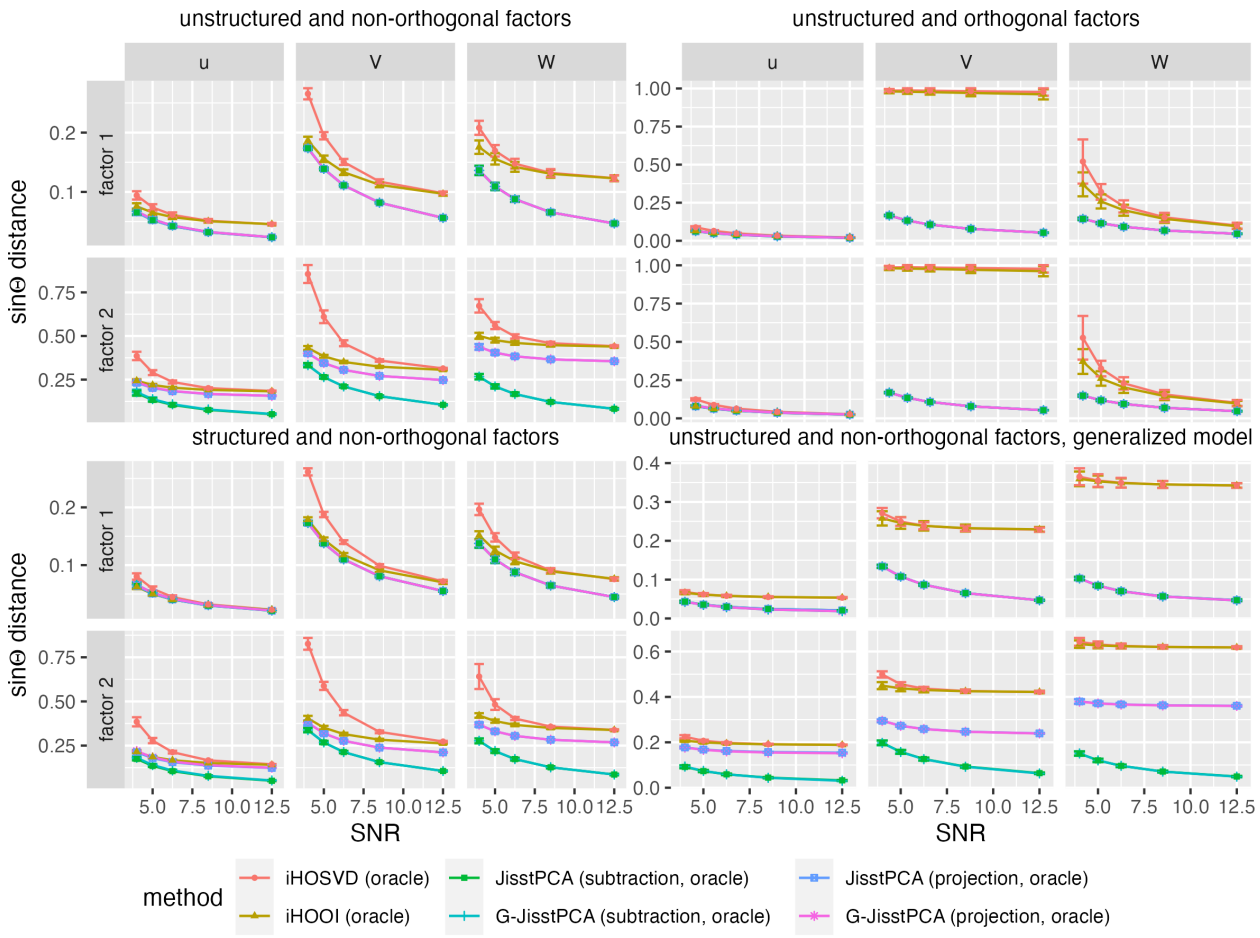


Figure 7: Detailed results of Figure 2: Estimation errors ($\sin\Theta$ distances in spectral norm) of all factors using iHOSVD, iHOOI, JisstPCA, and Generalized JisstPCA with subtraction and projected deflation, where all methods use the **true ranks**. This figure presents the same four scenarios as those included in the main paper. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

# B Proof of Main Results

## B.1 Additional Notations

Throughout this paper, we use the following notations repeatedly. The calligraphic letters represent tensors (e.g. $\mathcal{X}$), and the boldface uppercase letters represent matrices (e.g. $\boldsymbol{V}$). Also, we use boldface lowercase letters to denote vectors (e.g. $\boldsymbol{u}$) and lowercase letters to denote the real numbers (e.g. $a$). For $p, q \in \mathbb{R}$, let $p \vee q = \max\{p, q\}$ and $p \wedge q = \min\{p, q\}$. Suppose $\{a_n : n \in \mathbb{N}\}$ and $\{b_n : n \in \mathbb{N}\}$ are two sequences of real numbers, we claim that $a_n = O(b_n)$ if and only if there exists some $N \in \mathbb{N}$ and some positive constant $C > 0$, such that $a_n \leq Cb_n$ for all $n \geq N$. And we claim $a_n = o(b_n)$ if for $\forall \epsilon > 0, \exists N \in \mathbb{N}$ such that $a_n \leq \epsilon b_n$ for all $n \geq N$. If there are two positive constants $c > 0, C > 0$ such that $ca_n \leq b_n \leq Ca_n$ for $\forall n$, we say $a_n \asymp b_n$.

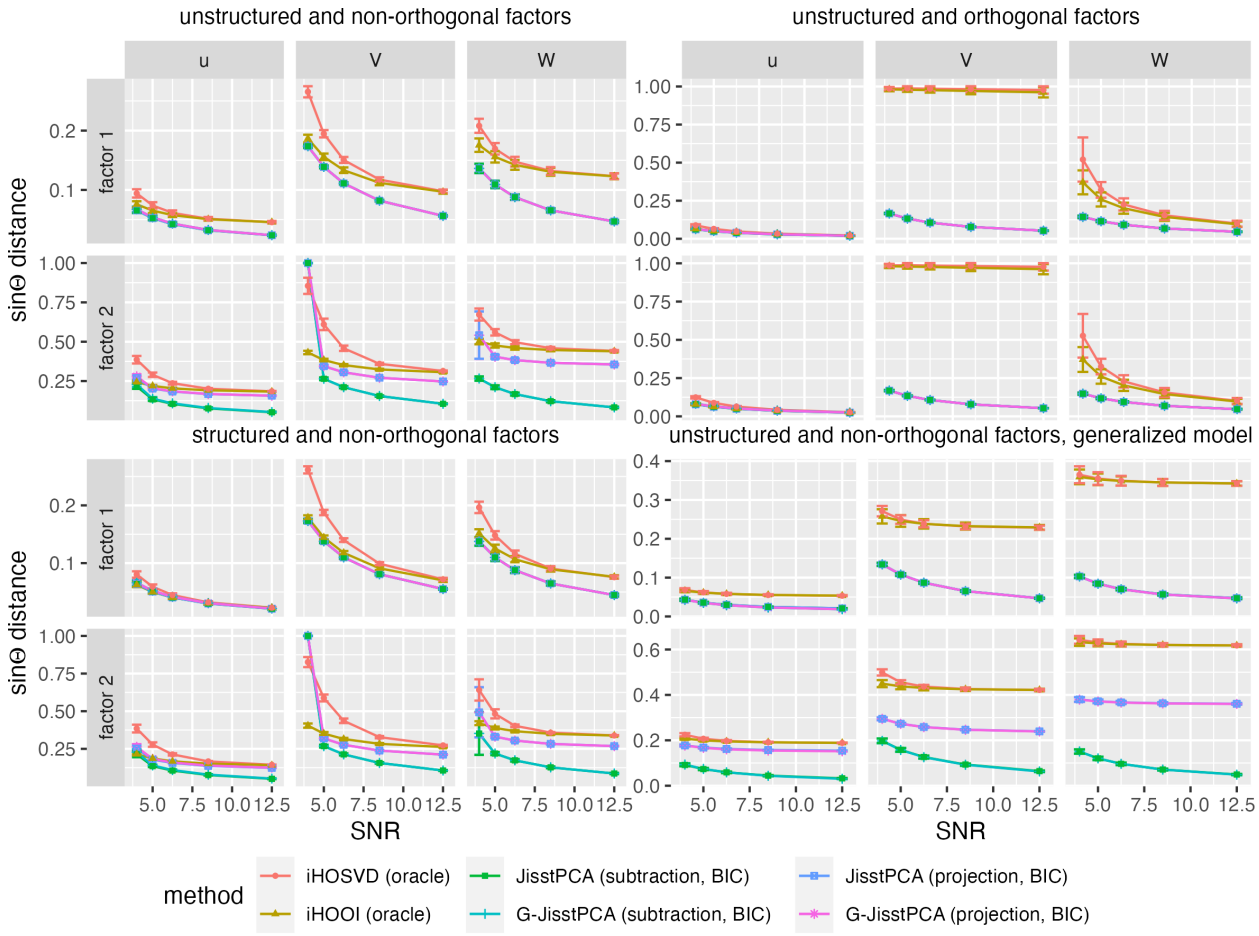


Figure 8: Detailed results of Figure 2: Estimation errors ($\sin\Theta$ distances in spectral norm) of all factors using iHOSVD, iHOOI, JisstPCA, and Generalized JisstPCA with **both subtraction and projected deflation**, where our methods (JisstPCA and G-JisstPCA) use the BIC selected ranks. The underlying factors are unstructured and non-orthogonal. This figure presents the same four scenarios as those included in the main paper. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

We let $\mathbb{S}^{N-1}$ be the Euclidean unit sphere in $\mathbb{R}^N$, i.e. $\mathbb{S}^{N-1} = \{\boldsymbol{u} \in \mathbb{R}^N : \|\boldsymbol{u}\|_2 = 1\}$, and $\mathcal{O}_{p,r}$ is the set of $p \times r$ orthogonal matrices, i.e. $\mathcal{O}_{p,r} = \{\boldsymbol{V} \in \mathbb{R}^{p\times r} : \boldsymbol{V}'\boldsymbol{V} = \boldsymbol{I}_r\}$.

## B.2 Proof of Main Theoretical Results for the JisstPCA

*Proof of Theorem 1.* We use an induction proof for Theorem 1. Specifically, we will show the following claim holds true, as long as Assumptions 1 and 2 hold.

**Claim 1.** *For $l \geq 0$,*

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(l)})|^2 \leq 1 - 8\left(\frac{\|\mathcal{E}_x\|_{\text{op}}^2}{d_x^{*2}} \vee \frac{\|\mathcal{E}_y\|_{\text{op}}^2}{d_y^{*2}}\right). \tag{18}$$

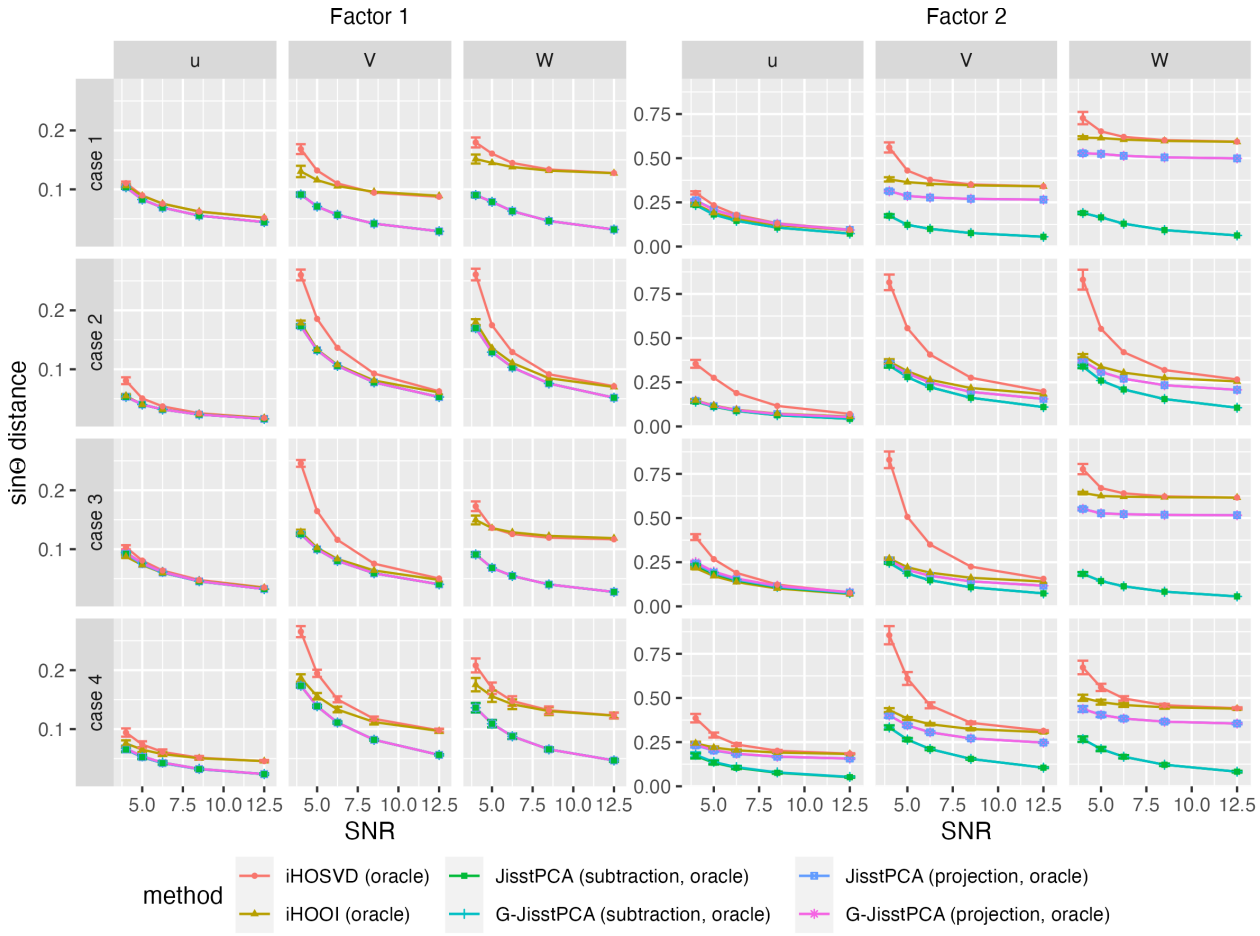


Figure 9: Dimensionality study: Four cases of different dimensions are considered: $p = 50$, $q = 50$, $N = 200$ (case 1); $p = 150$, $q = 150$, $N = 50$ (case 2); $p = 150$, $q = 50$, $N = 200$ (case 3); $p = 150$, $q = 50$, $N = 50$ (case 4). All methods use the true ranks. The underlying factors are unstructured and non-orthogonal. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

*For* $l \geq 1$,

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(l)})| \leq \frac{4\|\lambda\mathcal{E}_x; (1-\lambda)\mathcal{E}_y\|_{r_x, r_y, \mathrm{op}}}{\lambda r_x d_x^* + (1-\lambda) r_y d_y^*}, \tag{19}$$

$$|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(l+1)})| \leq \frac{2\|\mathcal{E}_x\|_{\mathrm{op}}}{d_x^* \sqrt{1 - \sin^2\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(l)})}}, \quad |\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(l)})| \leq \frac{2\|\mathcal{E}_y\|_{\mathrm{op}}}{d_y^* \sqrt{1 - \sin^2\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(l)})}}. \tag{20}$$

We first note that (18) is directly implied by Assumption 2 when $l = 0$. In the following, we will show that when (18) holds for $l = k$, then (18)-(20) would all hold for $l = k+1$. For notational simplicity, throughout the rest of the proof, we denote $|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})|$, $\|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k)})\|_{\mathrm{op}}$, and $\|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k)})\|_{\mathrm{op}}$ by $\varepsilon_{u,k}$, $\varepsilon_{v,k}$, and $\varepsilon_{w,k}$, respectively.

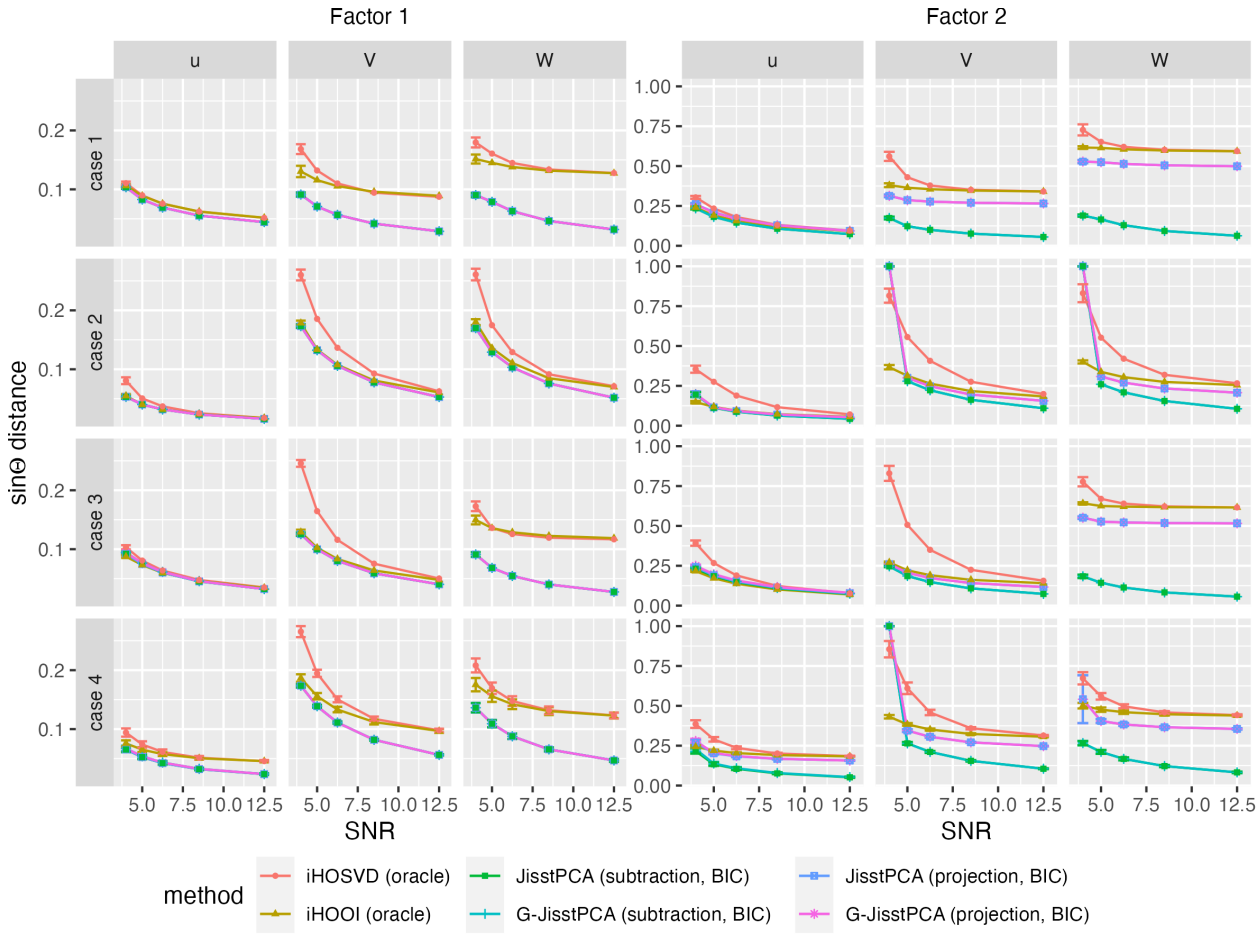


Figure 10: Dimensionality study: Four cases of different dimensions are considered: $p = 50$, $q = 50$, $N = 200$ (case 1); $p = 150$, $q = 150$, $N = 50$ (case 2); $p = 150$, $q = 50$, $N = 200$ (case 3); $p = 150$, $q = 50$, $N = 50$ (case 4). Our methods (JisstPCA and G-JisstPCA) use the BIC selected ranks. The underlying factors are unstructured and non-orthogonal. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

**Analysis of $\boldsymbol{V}^{(k+1)}$ and $\boldsymbol{W}^{(k+1)}$.** We start with the update of network factors $\boldsymbol{V}^{(k+1)}$ and $\boldsymbol{W}^{(k+1)}$. Based on the Algorithm 1, we know in the $(k+1)$th iteration, the update of $\boldsymbol{V}^{(k+1)}$ is leading $r^x$ singular vectors of $\mathcal{X} \times_3 \boldsymbol{u}^{(k)}$. Since $\mathcal{X} = d_x^* \cdot \boldsymbol{V}^* \boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^* + \mathcal{E}_x$, by some basic tensor algebra, we have

$$\begin{aligned}
\mathcal{X} \times_3 \boldsymbol{u}^{(k)} &= (d_x^* \cdot \boldsymbol{V}^* \boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^* + \mathcal{E}_x) \times_3 \boldsymbol{u}^{(k)} \\
&= (d_x^* \cdot \boldsymbol{V}^* \boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^*) \times_3 \boldsymbol{u}^{(k)} + \mathcal{E}_x \times_3 \boldsymbol{u}^{(k)} \\
&= d_x^* \cdot \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)} \rangle \cdot \boldsymbol{V}^* \boldsymbol{V}^{*\prime} + \mathcal{E}_x \times_3 \boldsymbol{u}^{(k)},
\end{aligned}$$

where $\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)} \rangle = \sqrt{1 - \sin^2 \theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})} > 2 \left( \frac{\|\mathcal{E}_x\|_{\text{op}}}{d_x^*} \vee \frac{\|\mathcal{E}_y\|_{\text{op}}}{d_y^*} \right)$, where the last inequality arises from our induction assumption that (18) holds for $l = k$. By Weyl's inequality, we know that $\lambda_{r+1}(\mathcal{X} \times_3 \boldsymbol{u}^{(k+1)}) \leq \|\mathcal{E}_x \times_3 \boldsymbol{u}^{(k)}\|_{\text{op}} \leq \|\mathcal{E}_x\|_{\text{op}} < \frac{d_x}{2}$, where the third inequality is due to the

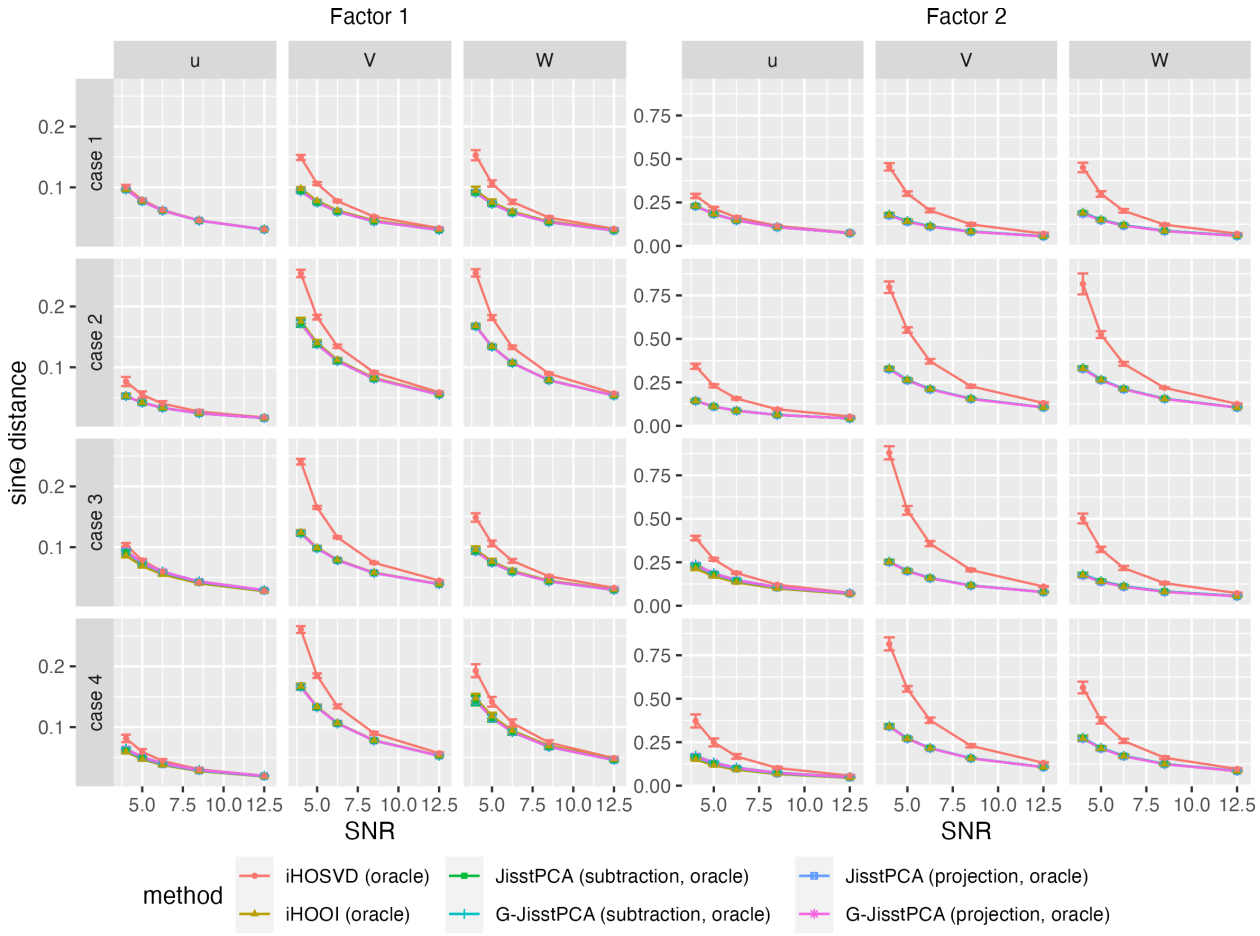


Figure 11: Orthogonal factors with significant singular gaps: we have comparative performance with iHOOI. All methods use the true ranks. Four cases of different dimensions are considered as in Figures 9-10 are considered. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

definition of the tensor operator norm: $\|\mathcal{E}_x\|_{\text{op}} = \sup_{\boldsymbol{u}\in\mathbb{R}^N,\boldsymbol{v}\in\mathbb{R}^p,\boldsymbol{w}\in\mathbb{R}^p} \mathcal{E}_x \times_1 \boldsymbol{v} \times_2 \boldsymbol{w} \times_3 \boldsymbol{u}$, and the last inquality is due to the lower bound for $\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)} \rangle$ we just derived. Therefore, by Davis-Kahan's theorem (see, e.g., Theorem 2.7 in Chen et al., 2021a), we know that

$$\varepsilon_{v,k} = \|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\|_{\text{op}} \le \frac{\|\mathcal{E}_x\|_{\text{op}}}{d_x^* \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)} \rangle - \lambda_{r+1}(\mathcal{X} \times_3 \boldsymbol{u}^{(k+1)})} \le \frac{2\|\mathcal{E}_x\|_{\text{op}}}{d_x^* \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)} \rangle} = \frac{2\|\mathcal{E}_x\|_{\text{op}}}{d_x^* \sqrt{1-\varepsilon_{u,k}^2}}. \tag{21}$$

Furthermore, since we have assumed $\varepsilon_{u,k} = |\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})| \le \sqrt{1 - 8\left(\frac{\|\mathcal{E}_x\|_{\text{op}}^2}{d_x^{*2}} \vee \frac{\|\mathcal{E}_y\|_{\text{op}}^2}{d_y^{*2}}\right)}$, one can immediately show that

$$\varepsilon_{v,k}^2 \le \frac{4\|\mathcal{E}_x\|_{\text{op}}^2}{d_x^{*2}(1-\varepsilon_{u,k}^2)} \le \frac{1}{2}. \tag{22}$$

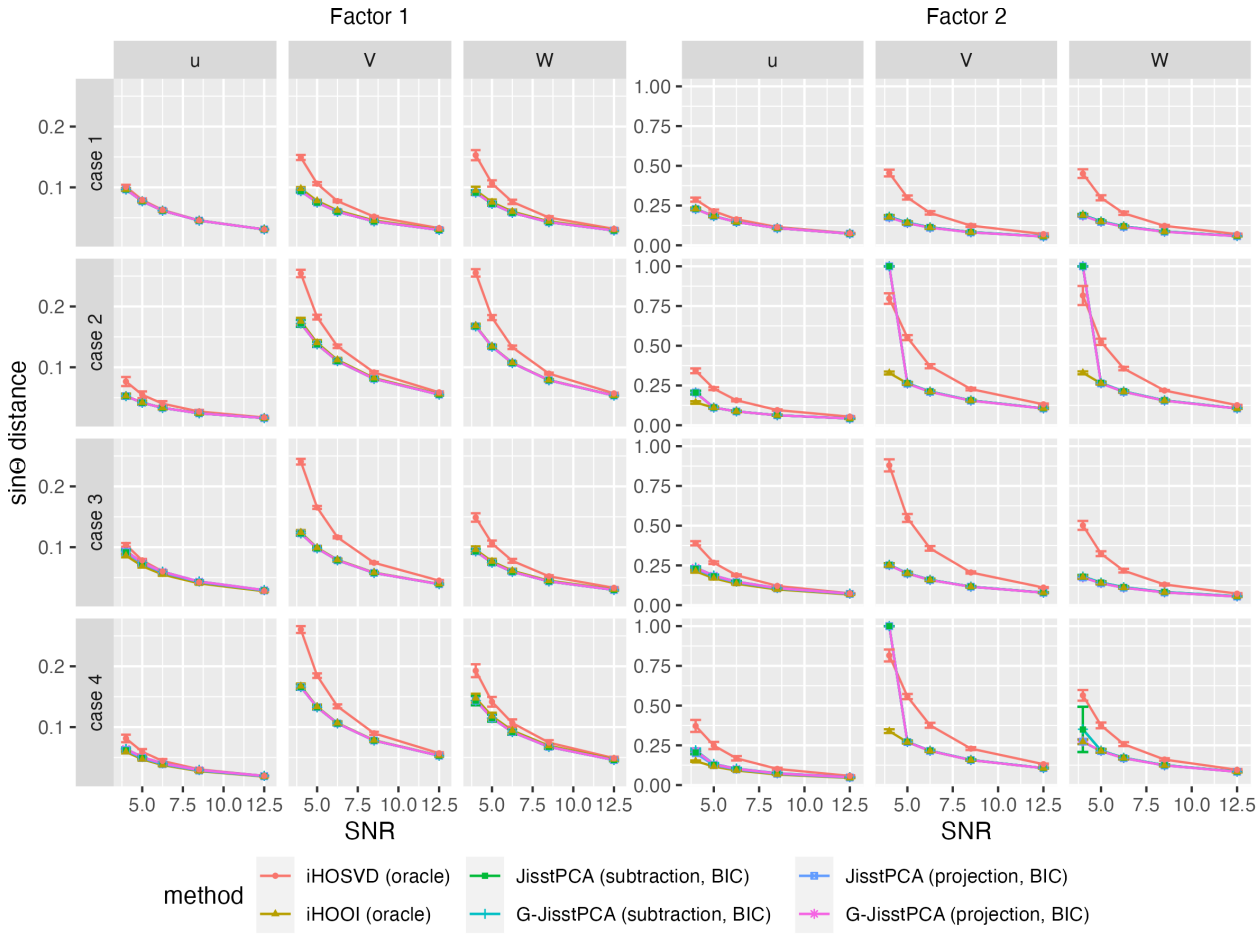


Figure 12: Orthogonal factors with significant singular gaps. Our methods (JisstPCA and G-JisstPCA) use the BIC selected ranks. We have comparative performance with iHOOI except for some small SNR scenario where ranks are selected wrong. Four cases of different dimensions are considered as in Figures 9-10 are considered. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

Following the same argument, we can also bound $\|\sin\Theta(\boldsymbol{W}, \boldsymbol{W}^{(k+1)})\|_{\text{op}}$ as follows:

$$\varepsilon_{w,k} = \|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k+1)})\|_{\text{op}} \leq \frac{2\|\mathcal{E}_y\|_{\text{op}}}{d_y^* \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)} \rangle} \leq \frac{2\|\mathcal{E}_y\|_{\text{op}}}{d_y^* \sqrt{1-\varepsilon_{u,k}^2}} \leq \frac{\sqrt{2}}{2}. \tag{23}$$

Now we have verified that (20) holds for $l = k+1$ when (18) holds for $l = k$.

**Analysis of $\boldsymbol{u}^{(k+1)}$.** Recall that we update $\boldsymbol{u}$ by $\boldsymbol{u}^{(k+1)} = \text{Norm}\left(\lambda \left[\mathcal{X}; \boldsymbol{V}^{(k+1)}\right] + (1-\lambda)\left[\mathcal{Y}; \boldsymbol{W}^{(k+1)}\right]\right)$, where $\text{Norm}(\cdot)$ is a normalization function that outputs a unit vector, and $[\mathcal{X}; \boldsymbol{V}^{(k+1)}] \in \mathbb{R}^N$ denotes

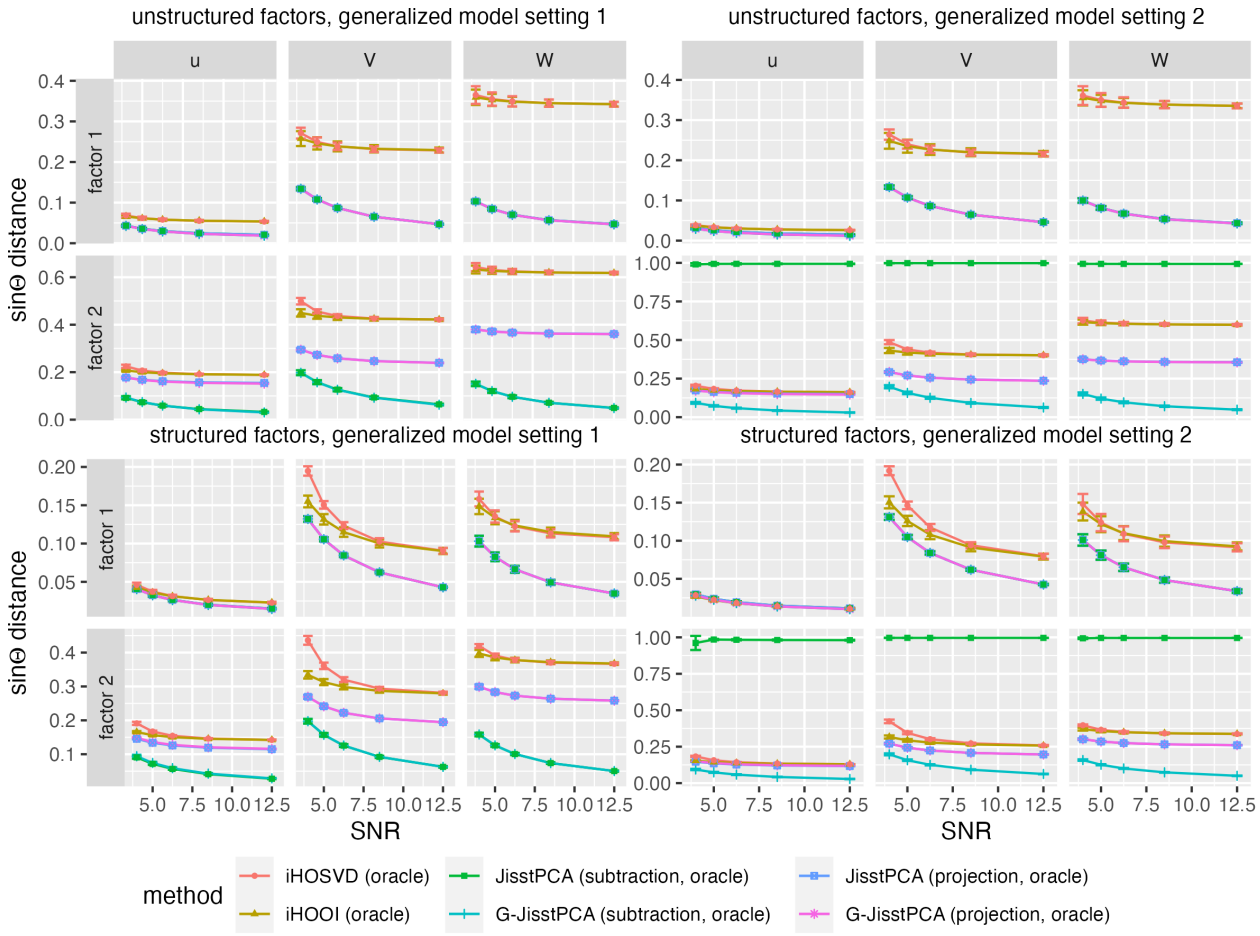


Figure 13: General models with different eigenvalues within each factor: setting 1 has more similar eigenvalues in the first factor while setting 2 is more different. All methods use the true ranks. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

the trace product defined in Section 2.1. Due to the definition of trace product, one has

$$\begin{aligned}
\lambda\left[\mathcal{X}; \boldsymbol{V}^{(k+1)}\right] &= \lambda\left[d_x^* \cdot \boldsymbol{V}^*\boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^* + \mathcal{E}_x; \boldsymbol{V}^{(k+1)}\right] \\
&= \lambda\left[d_x^* \cdot \boldsymbol{V}^*\boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^*; \boldsymbol{V}^{(k+1)}\right] + \lambda\left[\mathcal{E}_x; \boldsymbol{V}^{(k+1)}\right] \\
&= \lambda d_x^* \mathrm{Tr}\left((\boldsymbol{V}^{(k+1)})^\prime \boldsymbol{V}^*\boldsymbol{V}^{*\prime}\boldsymbol{V}^{(k+1)}\right) \cdot \boldsymbol{u}^* + \lambda \cdot \sum_{i=1}^{r} \mathcal{E}_x \times_1 \boldsymbol{v}_i^{(k+1)} \times_2 \boldsymbol{v}_i^{(k+1)} \\
&= \lambda d_x^* \left\|\boldsymbol{V}^{*\prime}\boldsymbol{V}^{(k+1)}\right\|_F^2 \cdot \boldsymbol{u}^* + \lambda \cdot \sum_{i=1}^{r} \mathcal{E}_x \times_1 \boldsymbol{v}_i^{(k+1)} \times_2 \boldsymbol{v}_i^{(k+1)}
\end{aligned}$$

where in the third equality $\boldsymbol{v}_i$ denotes $i$th column of $\boldsymbol{V}$. By the same argument, we also have

$$(1-\lambda)\left[\mathcal{Y}; \boldsymbol{W}^{(k+1)}\right] = (1-\lambda) d_y^* \left\|\boldsymbol{W}^{*\prime}\boldsymbol{W}^{(k+1)}\right\|_F^2 \cdot \boldsymbol{u} + (1-\lambda) \cdot \sum_{i=1}^{r} \mathcal{E}_y \times_1 \boldsymbol{w}_i^{(k+1)} \times_2 \boldsymbol{w}_i^{(k+1)},$$

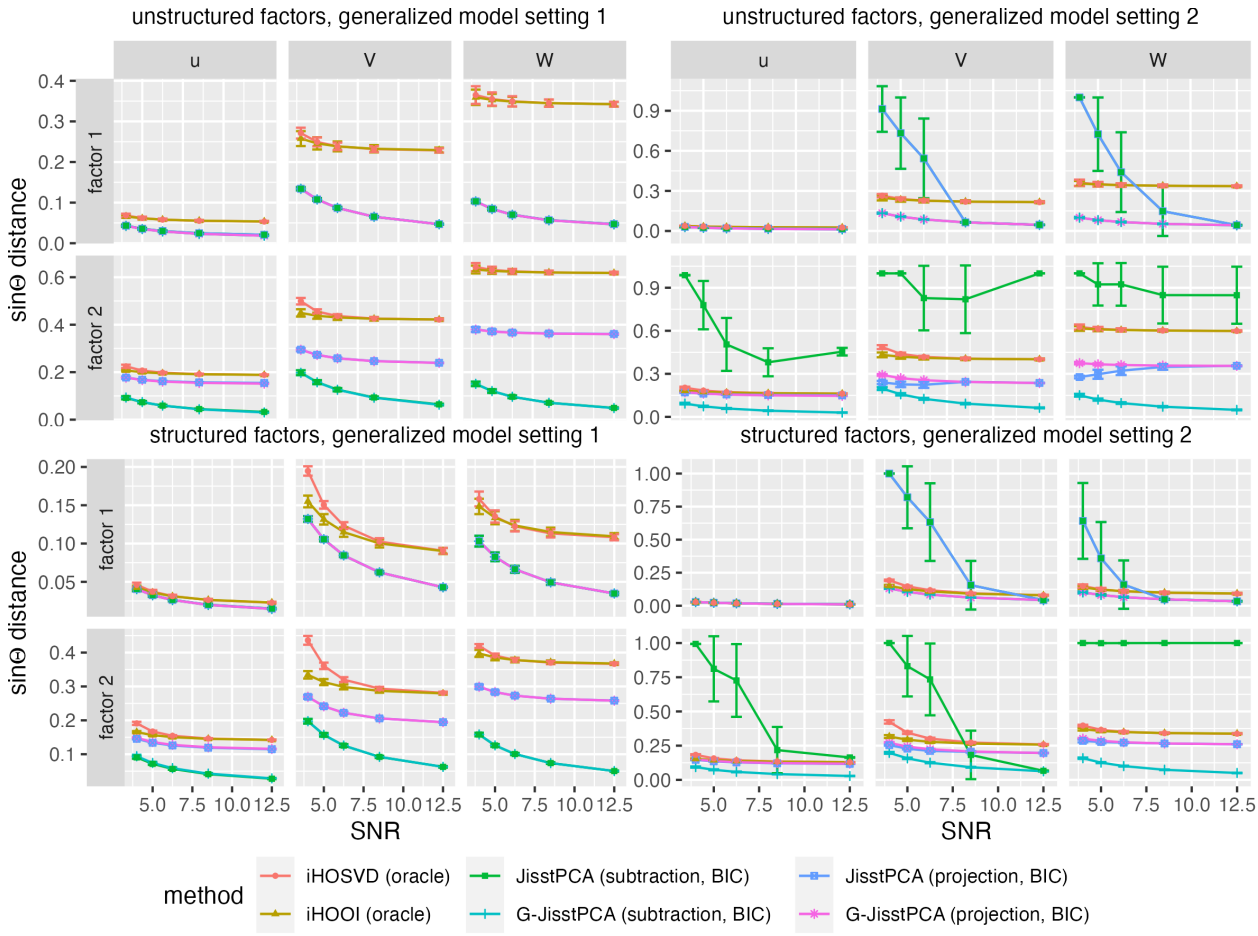


Figure 14: General models with different eigenvalues within each factor: setting 1 has more similar eigenvalues in the first factor while setting 2 is more different. Our methods (JisstPCA and G-JisstPCA) use the BIC selected ranks. The mean errors of 10 replicates are plotted, where the error bars represent 95% confidence intervals.

with $\boldsymbol{w}_i$ being the $i$th column of $\boldsymbol{W}$. Let

$$\begin{aligned}
\alpha_\lambda^{(k+1)} =& \lambda d_x^* \|\boldsymbol{V}^{*\prime}\boldsymbol{V}^{(k+1)}\|_F^2 + (1-\lambda) d_y^* \|\boldsymbol{W}^{*\prime}\boldsymbol{W}^{(k+1)}\|_F^2, \\
\boldsymbol{e}_\lambda^{(k+1)} =& \lambda [\mathcal{E}_x; \boldsymbol{V}^{(k+1)}] + (1-\lambda)[\mathcal{E}_y; \boldsymbol{W}^{(k+1)}] \\
=& \lambda \sum_{i=1}^{r} \mathcal{E}_x \times_1 \boldsymbol{v}_i^{(k+1)} \times_2 \boldsymbol{v}_i^{(k+1)} + (1-\lambda)\mathcal{E}_y \times_1 \boldsymbol{w}_i^{(k+1)} \times_2 \boldsymbol{w}_i^{(k+1)}.
\end{aligned}$$

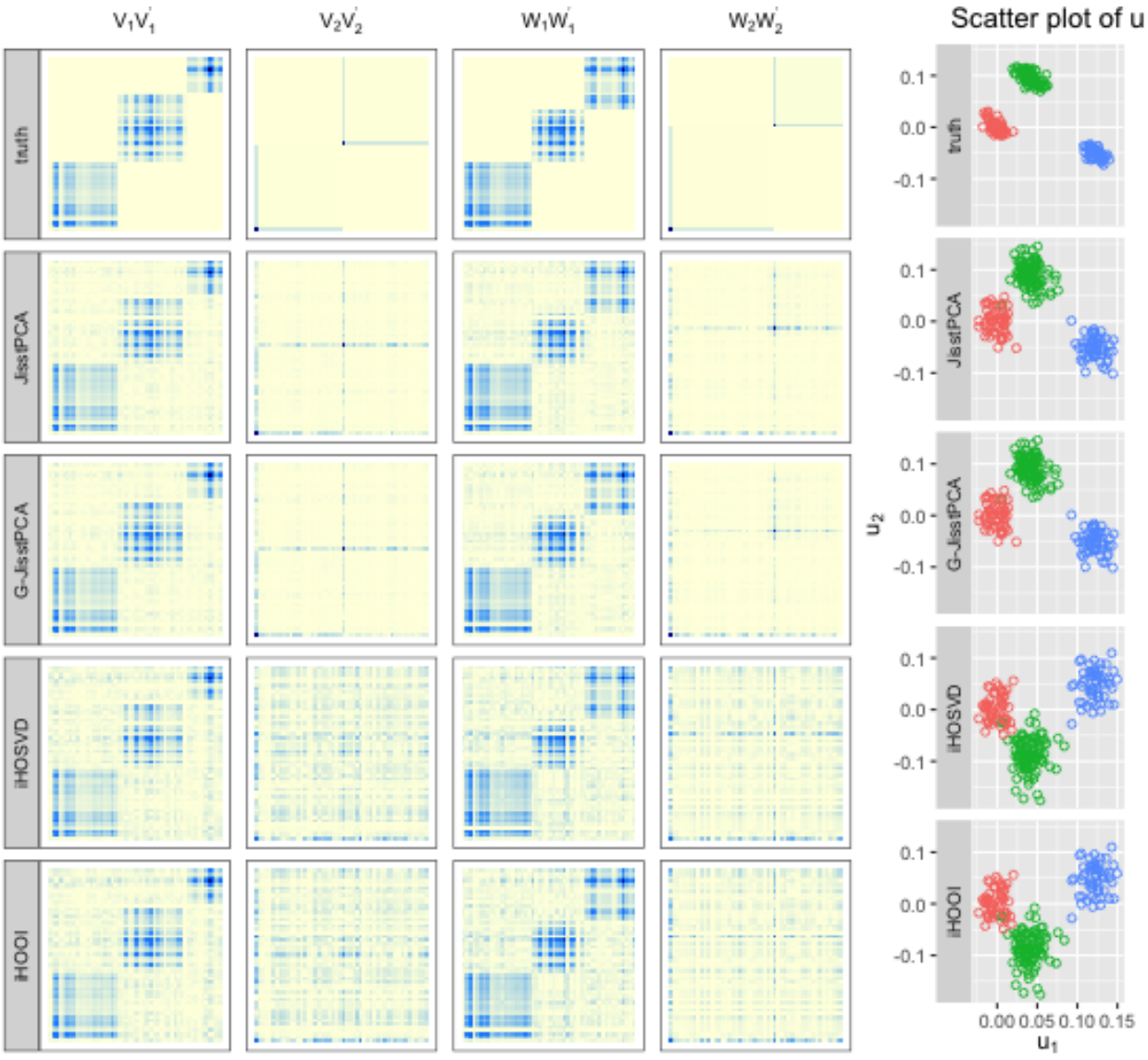


Figure 15: Heatmaps and scatterplots for structured network and population tensor factors reconstruction, by JisstPCA, G-JisstPCA, iHOSVD, iHOOI, together with the true factors.

Table 2: Population clustering and network community detection for multi-modal populations of networks, based on k-means on the estimated factors from JisstPCA, G-JisstPCA, iHOSVD, and iHOOI. The presented Adjusted Rand Index (ARI) values demonstrate the accuracy of sample clustering and node clustering of two network factors for each modality, when network sizes $p = 80$, $q = 50$, $N = 20$. The $\sin\Theta$ estimation errors of each factor is also presented. The average ARI and $\sin\Theta$ distances of 20 independent repeats are presented, with standard deviation inside the parenthesis. The largest average ARI and lowest estimation error for each setting are marked in bold. Both partial projection deflation and subtraction deflation are considered for JisstPCA and G-JisstPCA.

| Clustering ARI | **JisstPCA partial projection (BIC)** | **G-JisstPCA partial projection (BIC)** | **JisstPCA subtraction (BIC)** | **G-JisstPCA subtraction (BIC)** | **iHOSVD (oracle)** | **iHOOI (oracle)** |
|---|---|---|---|---|---|---|
| Sample | 0.947(0.238) | **1**(0) | 0.801(0.41) | 0.842(0.386) | **1**(0) | 0.954(0.205) |
| Network 1 of $\mathcal{X}$ | 0.971(0.108) | **1**(0) | 0.942(0.162) | **1**(0) | 0.912(0.195) | 0.739(0.251) |
| Network 2 of $\mathcal{X}$ | 0.995(0.022) | **0.997**(0.011) | 0.699(0.471) | 0.749(0.446) | 0.146(0.057) | 0.139(0.043) |
| Network 1 of $\mathcal{Y}$ | 0.974(0.114) | **1**(0) | **1**(0) | 0.974(0.115) | 0.99(0.027) | 0.94(0.155) |
| Network 2 of $\mathcal{Y}$ | **0.87**(0.31) | 0.867(0.309) | 0.685(0.468) | 0.732(0.442) | 0.014(0.046) | 0.121(0.16) |
| $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}_1^*)$ | 0.087(0.03) | 0.089(0.031) | 0.087(0.03) | 0.089(0.031) | 0.138(0.051) | 0.08(0.026) |
| $\sin\theta(\hat{\boldsymbol{u}}_2, \boldsymbol{u}_2^*)$ | 0.167(0.077) | 0.163(0.066) | 0.431(0.349) | 0.38(0.343) | 0.214(0.04) | 0.188(0.162) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_1, \boldsymbol{V}_1^*)\|_{\text{op}}$ | 0.084(0.007) | 0.084(0.007) | 0.084(0.007) | 0.084(0.007) | 0.163(0.06) | 0.163(0.059) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_2, \boldsymbol{V}_2^*)\|_{\text{op}}$ | 0.21(0.074) | 0.211(0.072) | 0.427(0.384) | 0.428(0.383) | 0.817(0.065) | 0.799(0.07) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_1, \boldsymbol{W}_1^*)\|_{\text{op}}$ | 0.158(0.018) | 0.158(0.018) | 0.158(0.018) | 0.158(0.018) | 0.253(0.056) | 0.196(0.058) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_2, \boldsymbol{W}_2^*)\|_{\text{op}}$ | 0.416(0.204) | 0.417(0.203) | 0.536(0.311) | 0.504(0.293) | 0.969(0.044) | 0.903(0.098) |

Table 3: Clustering ARI and factor estimation error for multi-modal populations of networks. The set-up is the same as in Table 2, but with $N = 40$.

| Clustering ARI | **JisstPCA partial projection (BIC)** | **G-JisstPCA partial projection (BIC)** | **JisstPCA subtraction (BIC)** | **G-JisstPCA subtraction (BIC)** | **iHOSVD (oracle)** | **iHOOI (oracle)** |
|---|---|---|---|---|---|---|
| Sample | 1(0) | **1**(0) | 0.863(0.334) | 0.851(0.363) | **1**(0) | **1**(0) |
| Network 1 of $\mathcal{X}$ | **1**(0) | **1**(0) | **1**(0) | **1**(0) | 0.805(0.241) | 0.663(0.224) |
| Network 2 of $\mathcal{X}$ | **1**(0) | **1**(0) | 0.748(0.448) | 0.85(0.365) | 0.156(0.015) | 0.153(0) |
| Network 1 of $\mathcal{Y}$ | **1**(0) | **1**(0) | **1**(0) | **1**(0) | 0.973(0.108) | **1**(0) |
| Network 2 of $\mathcal{Y}$ | **1**(0) | **1**(0) | 0.747(0.45) | 0.847(0.373) | 0.116(0.101) | 0.246(0.153) |
| $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}_1^*)$ | 0.092(0.016) | 0.093(0.017) | 0.092(0.016) | 0.093(0.017) | 0.142(0.028) | **0.083**(0.016) |
| $\sin\theta(\hat{\boldsymbol{u}}_2, \boldsymbol{u}_2^*)$ | 0.152(0.014) | 0.152(0.014) | 0.387(0.324) | 0.3(0.276) | 0.197(0.017) | **0.145**(0.015) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_1, \boldsymbol{V}_1^*)\|_{\text{op}}$ | **0.062**(0.006) | **0.062**(0.007) | **0.062**(0.006) | **0.062**(0.007) | 0.154(0.046) | 0.156(0.047) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_2, \boldsymbol{V}_2^*)\|_{\text{op}}$ | **0.154**(0.024) | 0.155(0.023) | 0.36(0.379) | 0.319(0.349) | 0.776(0.022) | 0.768(0.028) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_1, \boldsymbol{W}_1^*)\|_{\text{op}}$ | **0.118**(0.014) | **0.118**(0.014) | **0.118**(0.014) | **0.118**(0.014) | 0.202(0.026) | 0.173(0.04) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_2, \boldsymbol{W}_2^*)\|_{\text{op}}$ | **0.272**(0.045) | 0.273(0.045) | 0.448(0.327) | 0.375(0.271) | 0.901(0.062) | 0.771(0.079) |

Then we can also write the distance between $\boldsymbol{u}^{(k+1)}$ and $\boldsymbol{u}^*$ as follows:

$$
\begin{aligned}
|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k+1)})| =& \sqrt{1 - \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k+1)} \rangle^2} \\
=& \sqrt{1 - \frac{\langle \boldsymbol{u}^*, \alpha_\lambda^{(k+1)} \boldsymbol{u}^* + \boldsymbol{e}_\lambda^{(k+1)} \rangle^2}{\|\alpha_\lambda^{(k+1)} \boldsymbol{u}^* + \boldsymbol{e}_\lambda^{(k+1)}\|^2}} \\
=& \sqrt{\frac{\|\boldsymbol{e}_\lambda^{(k+1)}\|_2^2 - \langle \boldsymbol{e}_\lambda^{(k+1)}, \boldsymbol{u}^* \rangle^2}{\alpha_\lambda^{(k+1)2} + \|\boldsymbol{e}_\lambda^{(k+1)}\|_2^2 + 2\alpha_\lambda^{(k+1)} \langle \boldsymbol{e}_\lambda^{(k+1)}, \boldsymbol{u}^* \rangle}} \\
\leq& \frac{\|\boldsymbol{e}_\lambda^{(k+1)}\|_2}{\alpha_\lambda^{(k+1)} - \|\boldsymbol{e}_\lambda^{(k+1)}\|_2},
\end{aligned}
\tag{24}
$$

Table 4: Clustering ARI and factor estimation error for multi-modal populations of networks. The set-up is the same as in Table 2, but with $q = 80$, $N = 20$.

| Clustering ARI | **JisstPCA partial projection (BIC)** | **G-JisstPCA partial projection (BIC)** | **JisstPCA subtraction (BIC)** | **G-JisstPCA subtraction (BIC)** | **iHOSVD (oracle)** | **iHOOI (oracle)** |
|---|---|---|---|---|---|---|
| Sample | **1**(0) | **1**(0) | 0.809(0.35) | 0.905(0.294) | **1**(0) | **1**(0) |
| Network 1 of $\mathcal{X}$ | 0.992(0.026) | **1**(0) | 0.994(0.019) | **1**(0) | 0.792(0.248) | 0.686(0.237) |
| Network 2 of $\mathcal{X}$**0.995**(0.022) | **0.995**(0.022) | 0.497(0.516) | 0.698(0.473) | 0.142(0.042) | 0.16(0.032) | |
| Network 1 of $\mathcal{Y}$ | **0.977**(0.105) | 0.975(0.113) | **1**(0) | **1**(0) | 0.946(0.143) | **1**(0) |
| Network 2 of $\mathcal{Y}$ | **0.948**(0.135) | 0.929(0.227) | 0.499(0.509) | 0.699(0.467) | 0.135(0.093) | 0.298(0.15) |
| $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}_1^*)$ | 0.081(0.022) | 0.08(0.023) | 0.081(0.022) | 0.08(0.023) | 0.13(0.043) | **0.072**(0.022) |
| $\sin\theta(\hat{\boldsymbol{u}}_2, \boldsymbol{u}_2^*)$ | 0.148(0.058) | 0.145(0.058) | 0.55(0.389) | 0.398(0.365) | 0.206(0.048) | **0.139**(0.055) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_1, \boldsymbol{V}_1^*)\|_{\text{op}}$ | **0.082**(0.011) | **0.082**(0.011) | **0.082**(0.011) | **0.082**(0.011) | 0.158(0.061) | 0.159(0.062) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_2, \boldsymbol{V}_2^*)\|_{\text{op}}$ | **0.207**(0.069) | **0.207**(0.069) | 0.582(0.428) | 0.423(0.387) | 0.815(0.059) | 0.777(0.03) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_1, \boldsymbol{W}_1^*)\|_{\text{op}}$ | 0.135(0.015) | **0.134**(0.015) | 0.135(0.015) | **0.134**(0.015) | 0.203(0.055) | 0.192(0.059) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_2, \boldsymbol{W}_2^*)\|_{\text{op}}$ | **0.323**(0.115) | 0.339(0.175) | 0.624(0.377) | 0.488(0.342) | 0.87(0.067) | 0.782(0.075) |

Table 5: Clustering ARI and factor estimation error for multi-modal populations of networks. The set-up is the same as in Table 2, but with $q = 80$, $N = 40$.

| Clustering ARI | **JisstPCA partial projection (BIC)** | **G-JisstPCA partial projection (BIC)** | **JisstPCA subtraction (BIC)** | **G-JisstPCA subtraction (BIC)** | **iHOSVD (oracle)** | **iHOOI (oracle)** |
|---|---|---|---|---|---|---|
| Sample | **1**(0) | **1**(0) | **1**(0) | 0.949(0.229) | **1**(0) | **1**(0) |
| Network 1 of $\mathcal{X}$ | **1**(0) | **1**(0) | **1**(0) | **1**(0) | 0.714(0.262) | 0.689(0.236) |
| Network 2 of $\mathcal{X}$ | **1**(0) | **1**(0) | 0.647(0.494) | 0.848(0.371) | 0.156(0.015) | 0.153(0) |
| Network 1 of $\mathcal{Y}$ | **1**(0) | **1**(0) | **1**(0) | **1**(0) | 0.975(0.11) | 0.977(0.102) |
| Network 2 of $\mathcal{Y}$ | **0.997**(0.011) | **0.997**(0.011) | 0.656(0.48) | 0.849(0.37) | 0.281(0.109) | 0.338(0.079) |
| $\sin\theta(\hat{\boldsymbol{u}}_1, \boldsymbol{u}_1^*)$ | 0.084(0.016) | 0.084(0.016) | 0.084(0.016) | 0.084(0.016) | 0.131(0.027) | **0.077**(0.015) |
| $\sin\theta(\hat{\boldsymbol{u}}_2, \boldsymbol{u}_2^*)$ | 0.14(0.015) | 0.136(0.015) | 0.457(0.379) | 0.291(0.285) | 0.184(0.013) | **0.131**(0.018) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_1, \boldsymbol{V}_1^*)\|_{\text{op}}$ | **0.061**(0.006) | **0.061**(0.006) | **0.061**(0.006) | **0.061**(0.006) | 0.152(0.044) | 0.153(0.045) |
| $\|\sin\Theta(\hat{\boldsymbol{V}}_2, \boldsymbol{V}_2^*)\|_{\text{op}}$ | **0.146**(0.014) | **0.146**(0.014) | 0.441(0.42) | 0.399(0.403) | 0.774(0.019) | 0.764(0.029) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_1, \boldsymbol{W}_1^*)\|_{\text{op}}$ | **0.093**(0.006) | **0.093**(0.006) | **0.093**(0.006) | **0.093**(0.006) | 0.161(0.029) | 0.154(0.033) |
| $\|\sin\Theta(\hat{\boldsymbol{W}}_2, \boldsymbol{W}_2^*)\|_{\text{op}}$ | 0.**218**(0.028) | **0.218**(0.028) | 0.468(0.365) | 0.33(0.289) | 0.787(0.054) | 0.718(0.018) |

where the third line is due to direct calculations, while the last line is due to Cauchy's inequality. Recall that we have shown in (22) and (23) that $\|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\|_{\text{op}} \leq \frac{\sqrt{2}}{2}$, $\|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k+1)})\|_{\text{op}} \leq \frac{\sqrt{2}}{2}$. Hence one can also show that

$$\|\boldsymbol{V}^{*\prime}\boldsymbol{V}^{(k+1)}\|_F^2 = r_x - \|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\|_F^2 \geq r_x\left(1 - \|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\|_{\text{op}}^2\right) \geq \frac{r_x}{2},$$

suggesting that $\alpha_\lambda^{(k+1)} \geq \frac{1}{2}(\lambda r_x d_x^* + (1-\lambda) r_y d_y^*)$. On the other hand, the $\ell_2$ norm of the integrated noise term $\boldsymbol{e}_\lambda^{(k+1)}$ can be bounded as follows:

$$\|\boldsymbol{e}_\lambda^{(k+1)}\|_2 \leq \lambda r_x \|\mathcal{E}_x\|_{\text{op}} + (1-\lambda) r_y \|\mathcal{E}_y\|_{\text{op}} \leq \frac{1}{2}\alpha_\lambda^{(k+1)},$$

where we have applied the SNR condition that $d_x^* \geq 5\|\mathcal{E}_x\|_{\text{op}}$, $d_y^* \geq 5\|\mathcal{E}_y\|_{\text{op}}$(Assumption 1). Therefore, combining the results above with (24) leads us to

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k+1)})| \leq \frac{2\|\boldsymbol{e}_\lambda^{(k+1)}\|_2}{\alpha_\lambda^{(k+1)}} \leq \frac{4\|\lambda\mathcal{E}_x; (1-\lambda)\mathcal{E}_y\|_{r_x, r_y, \text{op}}}{\lambda r_x d_x^* + (1-\lambda) r_y d_y^*}.$$

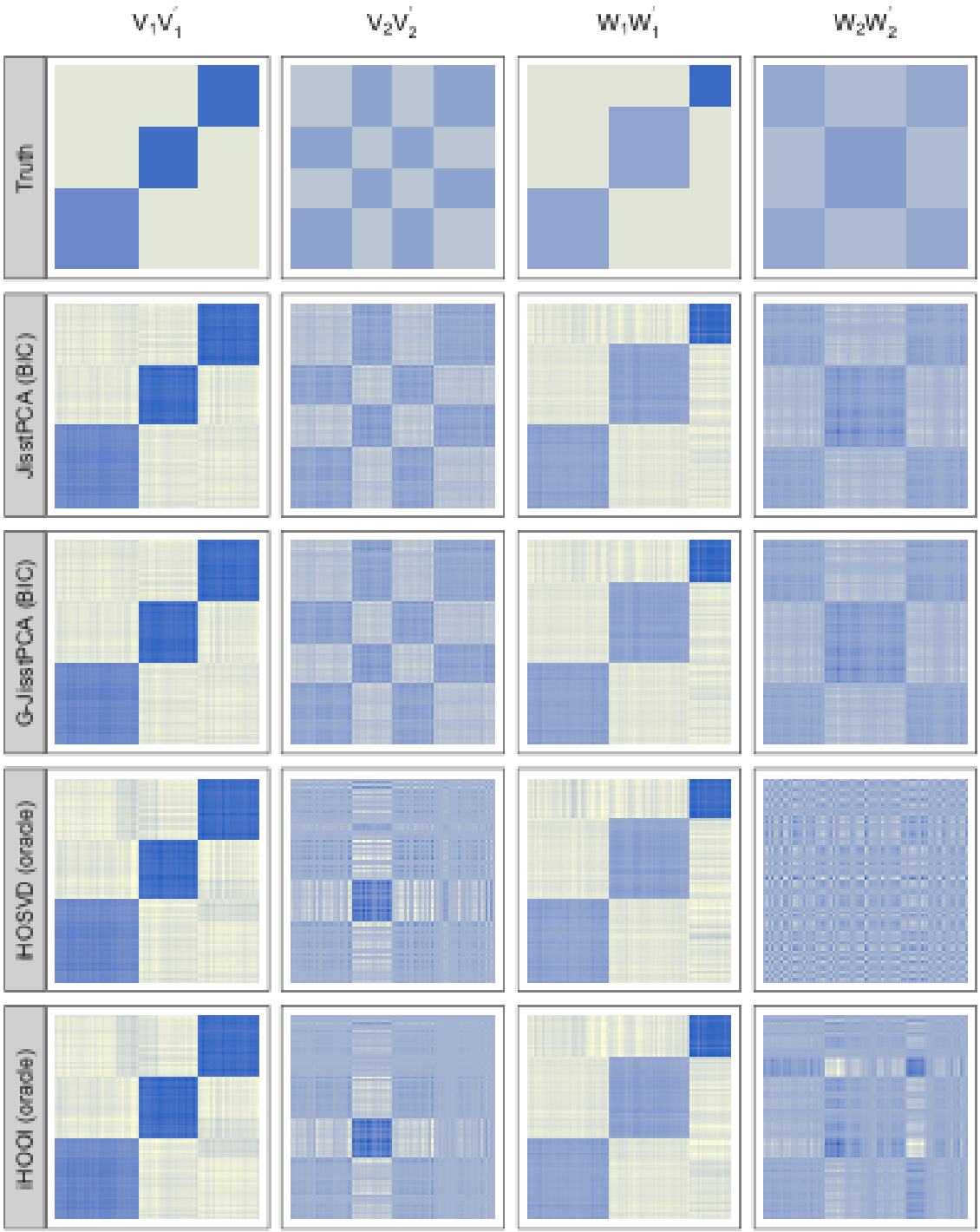


Figure 16: Heatmaps for the network factors reconstruction from network data, by JisstPCA, G-JisstPCA, iHOSVD, and iHOOI, together with the true factors.

Now we only need to show that (18) holds for $l = k + 1$. Since

$$\begin{aligned}
|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k+1)})| \leq & \frac{4\|\lambda\mathcal{E}_x; (1-\lambda)\mathcal{E}_y\|_{r_x, r_y, \mathrm{op}}}{\lambda r_x d_x^* + (1-\lambda) r_y d_y^*} \\
\leq & \frac{4\lambda r_x \|\mathcal{E}_x\|_{\mathrm{op}} + 4(1-\lambda) r_y \|\mathcal{E}_y\|_{\mathrm{op}}}{\lambda r_x d_x^* + (1-\lambda) r_y d_y^*} \\
\leq & \frac{4\|\mathcal{E}_x\|_{\mathrm{op}}}{d_x^*} \vee \frac{4\|\mathcal{E}_y\|_{\mathrm{op}}}{d_y^*},
\end{aligned}$$

and $d_x^* \geq 5\|\mathcal{E}_x\|_{\mathrm{op}}$, $d_y^* \geq \|\mathcal{E}_y\|_{\mathrm{op}}$ by Assumption 1, we have $|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k+1)})| \leq \frac{4}{5}$. On the other hand, Assumption 1 suggests that the R.H.S. of (18) is lower bounded by $\frac{17}{25} < \frac{4}{5}$, and hence (18) holds for $l = k+1$. The proof of Claim 1 is now complete. Furthermore, combining the fact that $|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k+1)})| \leq \frac{4}{5}$ and (20), we have also validated (8). Our proof for Theorem 1 is now complete. □

*Proof of Theorem 2.* We first note that $\|\mathcal{E}_x\|_{\mathrm{op}}$ and $\|\mathcal{E}_y\|_{\mathrm{op}}$ can both be bounded with high probability based on existing results for spectral norms of sub-Gaussian tensors. To deal with the dependency brought by the semi-symmetric constraint, we decompose them as upper and lower triangular components: $\mathcal{E}_x = \mathcal{E}_{x,1} + \mathcal{E}_{x,2}$, $\mathcal{E}_y = \mathcal{E}_{y,1} + \mathcal{E}_{y,2}$, where $\mathcal{E}_{x,1}, \mathcal{E}_{x,2} \in \mathbb{R}^{p\times p\times N}$, $\mathcal{E}_{y,1}$ satisfy $(\mathcal{E}_{x,1})_{i,j,k} = \begin{cases} (\mathcal{E}_x)_{i,j,k}, & i<j, \\ \frac{1}{2}(\mathcal{E}_x)_{i,j,k}, & i=j, \\ 0, & i>j, \end{cases}$ $(\mathcal{E}_{x,2})_{i,j,k} = \begin{cases} 0, & i<j, \\ \frac{1}{2}(\mathcal{E}_x)_{i,j,k}, & i=j, \\ (\mathcal{E}_x)_{i,j,k}, & i>j, \end{cases}$; $\mathcal{E}_{y,1}, \mathcal{E}_{y,2} \in \mathbb{R}^{q\times q\times N}$ are defined similarly. $\mathcal{E}_{x,1}, \mathcal{E}_{x,2}, \mathcal{E}_{y,1}, \mathcal{E}_{y,2}$ have independent, zero-mean, sub-Gaussian entries with sub-Gaussian parameter bounded by $\sigma$. Therefore, we can apply Theorem 1 and Lemma 1 in Tomioka and Suzuki (2014) on them with $K=3$, $\delta = 2e^{-N}$. Then with probability at least $1-4e^{-N}$, we have $\|\mathcal{E}_x\|_{\mathrm{op}} \leq \|\mathcal{E}_{x,1}\|_{\mathrm{op}} + \|\mathcal{E}_{x,2}\|_{\mathrm{op}} \leq 16\sigma\sqrt{N+p}$, $\|\mathcal{E}_y\|_{\mathrm{op}} \leq \|\mathcal{E}_{y,1}\|_{\mathrm{op}} + \|\mathcal{E}_{y,2}\|_{\mathrm{op}} \leq 16\sigma\sqrt{N+q}$. Therefore, Assumption 4 implies Assumption 1. Combining Theorem 1 and Proposition 1, and also noting that

$$\|\lambda\mathcal{E}_x; (1-\lambda)\mathcal{E}_y\|_{r_x,r_y,\mathrm{op}} \leq \lambda r_x\|\mathcal{E}_x\|_{\mathrm{op}} + (1-\lambda)r_y\|\mathcal{E}_y\|_{\mathrm{op}} \leq 16\sigma(\lambda r_x\sqrt{N+p} + (1-\lambda)r_y\sqrt{N+q}),$$

we have completed the proof of Theorem 2. □

### B.3 Proof of Warm Initialization (Corollary 1) and Spectral Initialization (Proposition 1)

*Proof of Corollary 1.* Given Theorem 1, it suffices to show the warm initialization $\boldsymbol{u}^{(0)} = (\frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}})$ satisfies Assumption 2. Now note that $\widehat{\mathrm{Var}}(\boldsymbol{u}^*) = \frac{1}{N}\sum_i(\boldsymbol{u}_i^* - \frac{1}{N}\sum_j \boldsymbol{u}_j^*)^2 = \frac{1}{N}\sum_i(\boldsymbol{u}_i^*)^2 - (\widehat{\mathbb{E}}(u^*))^2 = \frac{1}{N} - (\widehat{\mathbb{E}}(u^*))^2$, where we have utilized the fact that $\boldsymbol{u}^*$ is a unit vector. Therefore, $\frac{\widehat{\mathrm{Var}}(u^*)}{(\widehat{\mathbb{E}}(u^*))^2} \leq \left(\frac{d_x^{*2}}{8\|\mathcal{E}_x\|_{\mathrm{op}}^2} \wedge \frac{d_y^{*2}}{8\|\mathcal{E}_y\|_{\mathrm{op}}^2}\right) - 1$ implies $(\widehat{\mathbb{E}}(u^*))^2 \geq \frac{8}{N}\left(\frac{\|\mathcal{E}_x\|_{\mathrm{op}}^2}{d_x^{*2}} \vee \frac{\|\mathcal{E}_y\|_{\mathrm{op}}^2}{d_y^{*2}}\right)$, and hence

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(0)})|^2 = 1 - (\boldsymbol{u}^{*\prime}\boldsymbol{u}^{(0)})^2 \leq 1 - N(\widehat{\mathbb{E}}(u^*))^2 \leq 1 - 8\left(\frac{\|\mathcal{E}_x\|_{\mathrm{op}}^2}{d_x^{*2}} \vee \frac{\|\mathcal{E}_y\|_{\mathrm{op}}^2}{d_y^{*2}}\right),$$

which is Assumption 2. The proof is now complete. □

*Proof of Proposition 1.* Recall that $\mathcal{X} = d_x^* \cdot \boldsymbol{V}^*\boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^* + \mathcal{E}_x$ and $\mathcal{Y} = d_y^* \cdot \boldsymbol{W}^*\boldsymbol{W}^{*\prime} \circ \boldsymbol{u}^* + \mathcal{E}_y$. By our construction, $\boldsymbol{u}^{(0)}$ is the leading left singular vector of $[\lambda\mathcal{M}_3(\mathcal{X}), (1-\lambda)\mathcal{M}_3(\mathcal{Y})] \in \mathbb{R}^{N\times(p^2+q^2)}$, which can be written as

$$\begin{aligned}[\lambda\mathcal{M}_3(\mathcal{X}), (1-\lambda)\mathcal{M}_3(\mathcal{Y})] &= \boldsymbol{u}^*\left[\lambda d_x^* \cdot \mathrm{Vec}\left(\boldsymbol{V}^*\boldsymbol{V}^{*\prime}\right)', (1-\lambda)d_y^* \cdot \mathrm{Vec}\left(\boldsymbol{W}^*\boldsymbol{W}^{*\prime}\right)'\right] + [\lambda\mathcal{M}_3(\mathcal{E}_x), \mathcal{M}_3(\mathcal{E}_y)] \\ &= d_\lambda \cdot \boldsymbol{u}^*\boldsymbol{z}' + \boldsymbol{E}_\lambda,\end{aligned}$$

by letting $d_\lambda = \sqrt{\lambda^2 r_x d_x^{*2} + (1-\lambda)^2 r_y d_y^{*2}}$, $\boldsymbol{z} = \mathrm{Norm}(\left[\lambda d_x^* \cdot \mathrm{Vec}\left(\boldsymbol{V}\boldsymbol{V}'\right)', (1-\lambda) d_y^* \cdot \mathrm{Vec}\left(\boldsymbol{W}\boldsymbol{W}'\right)'\right])$, where $\mathrm{Norm}(\cdot)$ is a normalization function that outputs a unit vector, and $\boldsymbol{E}_\lambda = [\lambda \mathcal{M}_3(\mathcal{E}_x), (1-\lambda)\mathcal{M}_3(\mathcal{E}_y)]$. We can also think of $\boldsymbol{u}^{(0)}$ as the leading left singular vector of

$$(d_\lambda \cdot \boldsymbol{u}^* \boldsymbol{z}' + \boldsymbol{E}_\lambda)(d_\lambda \cdot \boldsymbol{u}^* \boldsymbol{z}' + \boldsymbol{E}_\lambda)' = d_\lambda^2 \boldsymbol{u}^* \boldsymbol{u}^{*\prime} + d_\lambda \boldsymbol{u}^* (\boldsymbol{E}_\lambda \boldsymbol{z})' + d_\lambda \boldsymbol{E}_\lambda \boldsymbol{z} \boldsymbol{u}^{*\prime} + \boldsymbol{E}_\lambda \boldsymbol{E}_\lambda'.$$

Recall Assumption 3 on the distributional properties of noise tensors. Since the variances of $\mathcal{E}_x$ and $\mathcal{E}_y$ are homogeneous across the third mode, we know that

$$\mathbb{E}\|(\boldsymbol{E}_\lambda)_{k,:}\|_2^2 = \lambda^2 \sum_{i,j=1}^{p} \mathrm{Var}((\mathcal{E}_x)_{i,j,k}) + (1-\lambda)^2 \sum_{i,j=1}^{q} \mathrm{Var}((\mathcal{E}_y)_{i,j,k})$$

take the same value for all $1 \le k \le N$. We also denote this variance term by $\sigma_\lambda^2$, and thus $\mathbb{E}\boldsymbol{E}_\lambda \boldsymbol{E}_\lambda' = \sigma_\lambda^2 \boldsymbol{I}_{N\times N}$. Therefore, $\boldsymbol{u}^{(0)}$ is also the leading left singular vector of $d_\lambda^2 \boldsymbol{u}^* \boldsymbol{u}^{*\prime} + d_\lambda \boldsymbol{u}^* (\boldsymbol{E}_\lambda \boldsymbol{z})' + d_\lambda \boldsymbol{E}_\lambda \boldsymbol{z} \boldsymbol{u}^{*\prime} + \boldsymbol{E}_\lambda \boldsymbol{E}_\lambda' - \mathbb{E}(\boldsymbol{E}_\lambda \boldsymbol{E}_\lambda')$, with the first term being the rank-1 signal and the rest three terms being random mean-zero perturbations. We would like to invoke the Davis-Kahan's theorem to upper bound $\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(0)})$, but before that, we first show an upper bound of the spectral norm of the noise term $d_\lambda \boldsymbol{u}^* (\boldsymbol{E}_\lambda \boldsymbol{z})' + d_\lambda \boldsymbol{E}_\lambda \boldsymbol{z} \boldsymbol{u}^{*\prime} + \boldsymbol{E}_\lambda \boldsymbol{E}_\lambda' - \mathbb{E}(\boldsymbol{E}_\lambda \boldsymbol{E}_\lambda')$.

To avoid the dependency issue brought by the semi-symmetric constraint, we define a reorganized noise matrix $\widetilde{\boldsymbol{E}}_\lambda \in \mathbb{R}^{N \times [p(p-1)+q(q-1)]}$ that satisfies $(\widetilde{\boldsymbol{E}}_\lambda)_{i,:} = [\lambda \boldsymbol{e}_{x,i}', (1-\lambda)\boldsymbol{e}_{y,i}']$ where $\boldsymbol{e}_{x,i} \in \mathbb{R}^{p(p-1)}$ ($\boldsymbol{e}_{y,i} \in \mathbb{R}^{q(q-1)}$) consists of diagonal and upper triangular entries of $(\mathcal{E}_x)_{:,:,i}$ ($(\mathcal{E}_y)_{:,:,i}$):

$$\boldsymbol{e}_{x,i} = [\mathrm{diag}((\mathcal{E}_x)_{j,j,i})_{1\le j\le p}, \sqrt{2}((\mathcal{E}_x)_{j,k,i})_{1\le j<k\le p}].$$

We also define $\widetilde{\boldsymbol{z}} \in \mathbb{R}^{p(p-1)+q(q-1)}$ based on $\boldsymbol{z}$ similarly:

$$\widetilde{\boldsymbol{z}} = \mathrm{Norm}([\lambda d_x^* [\mathrm{diag}(\boldsymbol{V}\boldsymbol{V}'), \sqrt{2}((\boldsymbol{V}\boldsymbol{V}')_{j,k})_{1\le j<k\le p}], (1-\lambda) d_y^* [\mathrm{diag}(\boldsymbol{W}\boldsymbol{W}'), \sqrt{2}((\boldsymbol{W}\boldsymbol{W}')_{j,k})_{1\le j<k\le q}]).$$

Then we can write $\boldsymbol{E}_\lambda \boldsymbol{z} = \widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}}$, $\boldsymbol{E}_\lambda \boldsymbol{E}_\lambda' = \widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{E}}_\lambda'$, where $\widetilde{\boldsymbol{E}}_\lambda$ has independent, mean-zero, sub-Gaussian-$\sqrt{2}\sigma$ entries.

Now we first bound $\|d_\lambda \widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}} \boldsymbol{u}^{*\prime}\| = \|d_\lambda \boldsymbol{u}^* (\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}})'\| = d_\lambda \|\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}}\|_2$. Since $\{(\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}})_i\}_{i=1}^N$ are independent zero-mean sub-Gaussian random variables with sub-Gaussian parameter $\sqrt{\sum_i 2\sigma^2 \widetilde{\boldsymbol{z}}_i^2} = \sqrt{2}\sigma$, and hence $(\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}})_i^2 - \mathbb{E}(\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}})_i^2$ are independent sub-exponential-$4\sigma^2$ random variables. Therefore, we can apply the Bernstein-type inequality (see e.g., Proposition 5.6 in Vershynin, 2010) for sub-exponential random variables to obtain the following:

$$\|\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}}\|_2^2 \le \mathbb{E}\|\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{z}}\|_2^2 + C\sigma^2 \sqrt{N \log N} \le C\sigma^2 N$$

with probability at least $1 - N^{-c}$, which implies

$$\|d_\lambda \boldsymbol{u}^* (\boldsymbol{E}_\lambda \boldsymbol{z})' + d_\lambda \boldsymbol{E}_\lambda \boldsymbol{z} \boldsymbol{u}^{*\prime}\| \le C\sigma\sqrt{N} d_\lambda.$$

For the last noise term $\boldsymbol{E}_\lambda \boldsymbol{E}_\lambda' - \mathbb{E}(\boldsymbol{E}_\lambda \boldsymbol{E}_\lambda') = \widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{E}}_\lambda' - \mathbb{E}(\widetilde{\boldsymbol{E}}_\lambda \widetilde{\boldsymbol{E}}_\lambda')$, we would like to apply a technical lemma from Zhou and Chen (2023). In particular, since $\widetilde{\boldsymbol{E}}_\lambda$ has independent zero-mean

sub-Gaussian-$\sqrt{2}\sigma$, it satisfies Assumption 3 in Zhou and Chen (2023) with $\omega_{\max} = \sqrt{2}\sigma$, $B = C\sigma\sqrt{\log N}$ and $\varepsilon = N^{-c}$. We apply Lemma 7 in Zhou and Chen (2023) with $\boldsymbol{E} = \widetilde{\boldsymbol{E}}_\lambda$, which gives us

$$\|\widetilde{\boldsymbol{E}}_\lambda\widetilde{\boldsymbol{E}}_\lambda' - \mathrm{diag}(\widetilde{\boldsymbol{E}}_\lambda\widetilde{\boldsymbol{E}}_\lambda')\| \le C\sigma^2(N + \sqrt{N(p^2+q^2)})\log N,$$

with probability at least $1-N^{-c}$. Furthermore, since $\|\mathrm{diag}(\widetilde{\boldsymbol{E}}_\lambda\widetilde{\boldsymbol{E}}_\lambda') - \mathbb{E}(\widetilde{\boldsymbol{E}}_\lambda\widetilde{\boldsymbol{E}}_\lambda')\| = \max_i \|(\widetilde{\boldsymbol{E}}_\lambda)_{i,:}\|_2^2 - \sigma_\lambda^2$ and $\|(\widetilde{\boldsymbol{E}}_\lambda)_{i,:}\|_2^2 - \sigma_\lambda^2$ is the sum of $p(p-1)+q(q-1)$ independent zero-mean sub-exponential-$4\sigma^2$ random variables, we can again apply the Bernstein-type inequality (see e.g., Proposition 5.6 in Vershynin, 2010) to derive the following bounds:

$$\begin{aligned}
&\mathbb{P}(\|\mathrm{diag}(\widetilde{\boldsymbol{E}}_\lambda\widetilde{\boldsymbol{E}}_\lambda') - \mathbb{E}(\widetilde{\boldsymbol{E}}_\lambda\widetilde{\boldsymbol{E}}_\lambda')\| > C\sigma^2\sqrt{p^2+q^2}\sqrt{\log N}) \\
\le&\sum_{i=1}^N \mathbb{P}(|\|(\widetilde{\boldsymbol{E}}_\lambda)_{i,:}\|_2^2 - \sigma_\lambda^2| > C\sigma^2\sqrt{p^2+q^2}\sqrt{\log N}) \\
\le&2N\exp\{-c\log N\} \\
\le&N^{-c}.
\end{aligned}$$

Hence, with probability at least $1 - CN^{-c}$,

$$\begin{aligned}
&\|d_\lambda\boldsymbol{u}^*(\boldsymbol{E}_\lambda\boldsymbol{z})' + d_\lambda\boldsymbol{E}_\lambda\boldsymbol{z}\boldsymbol{u}^{*\prime} + \boldsymbol{E}_\lambda\boldsymbol{E}_\lambda' - \mathbb{E}(\boldsymbol{E}_\lambda\boldsymbol{E}_\lambda')\| \\
\le&Cd_\lambda\sigma\sqrt{N} + C\sigma^2(N+\sqrt{N(p^2+q^2)})\log N \\
\le&\frac{1}{2}d_\lambda^2,
\end{aligned}$$

where the last inequality is due to Assumption 4.

Now we return to the original problem and invoke the Davis-Kahan's theorem (see, e.g., Theorem 2.7 in Chen et al., 2021a):

$$\begin{aligned}
\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(0)}) \le&\frac{2\|d_\lambda\boldsymbol{u}^*(\boldsymbol{E}_\lambda\boldsymbol{z})' + d_\lambda\boldsymbol{E}_\lambda\boldsymbol{z}\boldsymbol{u}^{*\prime} + \boldsymbol{E}_\lambda\boldsymbol{E}_\lambda' - \mathbb{E}(\boldsymbol{E}_\lambda\boldsymbol{E}_\lambda')\|}{d_\lambda^2} \\
\le&\frac{C\sigma\sqrt{N}}{d_\lambda} + \frac{C\sigma^2(N+\sqrt{N(p^2+q^2)})\log N}{d_\lambda^2} \\
\le&\frac{C\sigma\left(\sqrt{N} + (N(p^2+q^2))^{\frac{1}{4}}\right)\sqrt{\log N}}{d_\lambda}
\end{aligned}$$

□

## B.4 Proof of Theoretical Guarantees for the Generalized JisstPCA: Theorem 3

In this section, we will first establish bounds for $\boldsymbol{u}^{(k)}$, $\boldsymbol{V}^{(k)}$, and $\boldsymbol{W}^{(k)}$ under an initialization condition and some deterministic conditions for $\mathcal{E}_x$, $\mathcal{E}_y$; we then prove that under Assumptions 5 and 6, all these conditions hold with high probability.

### B.4.1 Deterministic Bounds for the Joint Factor

In particular, we will first assume the following two conditions hold and will revisit and show them hold with desired probabilities at the end of the proof.

**Condition 1** (Initialization condition for generalized JisstPCA)**.**

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(0)})| \leq \min\left\{\sqrt{1 - \frac{32r_x\|\mathcal{E}_x\|_{\text{op}}^2}{\|\boldsymbol{D}_x^*\|_F^2}}, \sqrt{1 - \frac{32r_y\|\mathcal{E}_y\|_{\text{op}}^2}{\|\boldsymbol{D}_y^*\|_F^2}}\right\}.$$

**Condition 2** (Deterministic SNR condition for generalized JisstPCA)**.**

$$\|\boldsymbol{D}_x^*\|_F \geq 12\sqrt{r_x}\|\mathcal{E}_x\|_{\text{op}}, \quad \|\boldsymbol{D}_x^*\|_F \geq 12\sqrt{r_y}\|\mathcal{E}_y\|_{\text{op}}.$$

We first note that in the generalized JisstPCA algorithm, the update $\boldsymbol{u}^{(1)}$ is computed as $\boldsymbol{u}^{(1)} = \text{Norm}(\lambda[\mathcal{X}; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}] + (1-\lambda)[\mathcal{Y}; \boldsymbol{W}^{(1)}, \boldsymbol{D}_y^{(1)}])$, where $\text{Norm}(\cdot)$ is a normalization function outputting a unit vector, and $[\mathcal{X}; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}]$, $[\mathcal{Y}; \boldsymbol{W}^{(1)}, \boldsymbol{D}_y^{(1)}] \in \mathbb{R}^N$ are the trace products:

$$[\mathcal{X}; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}]_i = \langle \mathcal{X}_{:,:,i}, \boldsymbol{V}^{(1)}\boldsymbol{D}_x^{(1)}\boldsymbol{V}^{(1)\prime}\rangle, \quad [\mathcal{Y}; \boldsymbol{W}^{(1)}, \boldsymbol{D}_y^{(1)}]_i = \langle \mathcal{Y}_{:,:,i}, \boldsymbol{W}^{(1)}\boldsymbol{D}_y^{(1)}\boldsymbol{W}^{(1)\prime}\rangle.$$

Due to the generative model for $\mathcal{X}$ and $\mathcal{Y}$ in (6) with $K = 1$, we have

$$\begin{aligned}[\mathcal{X}; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}] =& \boldsymbol{V}^*\boldsymbol{D}^{x*}\boldsymbol{V}^{*\prime} \circ \boldsymbol{u}^*; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}] + [\mathcal{E}_x; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}] \\ =& \langle \boldsymbol{V}^*\boldsymbol{D}^{x*}\boldsymbol{V}^{*\prime}, \boldsymbol{V}^{(1)}\boldsymbol{D}_x^{(1)}\boldsymbol{V}^{(1)\prime}\rangle\boldsymbol{u}^* + \sum_{i=1}^{r^x}(\boldsymbol{D}_x^{(1)})_{i,i}\mathcal{E}_x \times_1 \boldsymbol{v}_i^{(1)} \times_2 \boldsymbol{v}_i^{(1)},\end{aligned}$$

and

$$[\mathcal{Y}; \boldsymbol{W}^{(1)}, \boldsymbol{D}_y^{(1)}] = \langle \boldsymbol{W}^*\boldsymbol{D}^{y*}\boldsymbol{W}^{*\prime}, \boldsymbol{W}^{(1)}\boldsymbol{D}_y^{(1)}\boldsymbol{W}^{(1)\prime}\rangle\boldsymbol{u}^* + \sum_{i=1}^{r^y}(\boldsymbol{D}_y^{(1)})_{i,i}\mathcal{E}_y \times_1 \boldsymbol{w}_i^{(1)} \times_2 \boldsymbol{w}_i^{(1)},$$

where $\boldsymbol{v}_i^{(1)}$ and $\boldsymbol{w}_i^{(1)}$ are the $i$th columns of $\boldsymbol{V}^{(1)} \in \mathbb{R}^{p\times r^x}$ and $\boldsymbol{W}^{(1)} \in \mathbb{R}^{p\times r^y}$. Let

$$\alpha_\lambda = \lambda\langle \boldsymbol{V}^*\boldsymbol{D}^{x*}\boldsymbol{V}^{*\prime}, \boldsymbol{V}^{(1)}\boldsymbol{D}_x^{(1)}\boldsymbol{V}^{(1)\prime}\rangle + (1-\lambda)\langle \boldsymbol{W}^*\boldsymbol{D}^{y*}\boldsymbol{W}^{*\prime}, \boldsymbol{W}^{(1)}\boldsymbol{D}_y^{(1)}\boldsymbol{W}^{(1)\prime}\rangle$$

be the pooled signal, and let

$$\boldsymbol{e}_\lambda = \lambda\sum_{i=1}^{r_x}(\boldsymbol{D}_x^{(1)})_{i,i}\mathcal{E}_x \times_1 \boldsymbol{v}_i^{(1)} \times_2 \boldsymbol{v}_i^{(1)} + (1-\lambda)\sum_{i=1}^{r^y}(\boldsymbol{D}_y^{(1)})_{i,i}\mathcal{E}_y \times_1 \boldsymbol{w}_i^{(1)} \times_2 \boldsymbol{w}_i^{(1)}$$

be the pooled noise in $\lambda[\mathcal{X}; \boldsymbol{V}^{(1)}, \boldsymbol{D}_x^{(1)}] + (1-\lambda)[\mathcal{Y}; \boldsymbol{W}^{(1)}, \boldsymbol{D}_y^{(1)}]$. Then similar to the proof of Theorem 1, we have

$$|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(1)})| \leq \frac{\|\boldsymbol{e}_\lambda\|_2}{\alpha_\lambda - \|\boldsymbol{e}_\lambda\|_2}. \tag{25}$$

In the following, we provide a lower bound for $\alpha_\lambda$ and an upper bound for $\|\boldsymbol{e}_\lambda\|_2$. To lower bound $\alpha_\lambda$, we first notice that $\boldsymbol{V}^{(1)}\boldsymbol{D}_x^{(1)}\boldsymbol{V}^{(1)\prime}$ is the top rank-$r^x$ SVD of $\mathcal{X} \times \boldsymbol{u}^{(0)}$, meaning that it is also its best rank-$r^x$ approximation in Frobenius norm error. In the meantime, we have the decomposition $\mathcal{X} \times_3 \boldsymbol{u}^{(0)} = \langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle\boldsymbol{V}^*\boldsymbol{D}_x^*\boldsymbol{V}^{*\prime} + \mathcal{E}_x \times_3 \boldsymbol{u}^{(0)}$, which is a rank-$r^x$ matrix plus a perturbation. The following lemma shows that given a perturbed low-rank matrix, one can upper bound the estimation error of the top SVD solution for the true low-rank matrix.

**Lemma 1.** *Suppose that $\boldsymbol{X} = \boldsymbol{X}^* + \boldsymbol{E}$ where $\boldsymbol{X}^*$ is of rank $r$. Let $\boldsymbol{X}_r = \hat{\boldsymbol{U}}\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{V}}'$ be the top-$r$ SVD of $\boldsymbol{X}$, then $\|\boldsymbol{X}_r - \boldsymbol{X}^*\|_F \le 2\sqrt{2r}\|\boldsymbol{E}\|$.*

Applying Lemma 1 with $\boldsymbol{X}^* = \langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \boldsymbol{V}^* \boldsymbol{D}_x^* \boldsymbol{V}^{*\prime}$, $\boldsymbol{E} = \mathcal{E}_x \times_3 \boldsymbol{u}^{(0)}$, we then have

$$\|\boldsymbol{V}^{(1)}\boldsymbol{D}_x^{(1)}\boldsymbol{V}^{(1)\prime} - \langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \boldsymbol{V}^* \boldsymbol{D}_x^* \boldsymbol{V}^{*\prime}\|_F \le 2\sqrt{2r^x}\|\mathcal{E}_x \times_3 \boldsymbol{u}^{(0)}\| \le 2\sqrt{2r^x}\|\mathcal{E}_x\|_{\mathrm{op}}. \tag{26}$$

The Frobenious norm error bound (26) also suggests

$$\begin{aligned}\langle \boldsymbol{V}^{(1)}\boldsymbol{D}_x^{(1)}\boldsymbol{V}^{(1)\prime}, \boldsymbol{V}^*\boldsymbol{D}_x^*\boldsymbol{V}^{*\prime}\rangle \ge& \langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \|\boldsymbol{V}^*\boldsymbol{D}_x^*\boldsymbol{V}^{*\prime}\|_F^2 - 2\sqrt{2r^x}\|\mathcal{E}_x\|_{\mathrm{op}}\|\boldsymbol{V}^*\boldsymbol{D}_x^*\boldsymbol{V}^{*\prime}\|_F \\ \ge& \frac{1}{2}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \|\boldsymbol{D}_x^*\|_F^2,\end{aligned}$$

where the last line is due to Condition 1, which implies

$$2\sqrt{2r^x}\|\mathcal{E}_x\|_{\mathrm{op}} \le \frac{1}{2}\sqrt{1-\sin^2\theta(\boldsymbol{u}, \boldsymbol{u}^{(0)})}\|\boldsymbol{D}_x^*\|_F = \frac{1}{2}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \|\boldsymbol{V}^*\boldsymbol{D}_x^*\boldsymbol{V}^{*\prime}\|_F. \tag{27}$$

Similarly, one can show that $\langle \boldsymbol{W}^{(1)}\boldsymbol{D}_y^{(1)}\boldsymbol{W}^{(1)\prime}, \boldsymbol{W}^*\boldsymbol{D}_y^*\boldsymbol{W}^{*\prime}\rangle \ge \frac{1}{2}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \|\boldsymbol{D}_y^*\|_F^2$, leading to a lower bound for $\alpha_\lambda$:

$$\alpha_\lambda \ge \frac{1}{2}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle \big(\lambda\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)\|\boldsymbol{D}_y^*\|_F^2\big).$$

On the other hand, to upper bound $\|\boldsymbol{e}_\lambda\|_2$, we note that

$$\begin{aligned}\left\|\sum_{i=1}^{r_x}(\boldsymbol{D}_x^{(1)})_{i,i}\mathcal{E}_x \times_1 \boldsymbol{v}_i^{(1)} \times_2 \boldsymbol{v}_i^{(1)}\right\|_2 \le& \sum_{i=1}^{r_x}|(\boldsymbol{D}_x^{(1)})_{i,i}|\left\|\mathcal{E}_x \times_1 \boldsymbol{v}_i^{(1)} \times_2 \boldsymbol{v}_i^{(1)}\right\|_2 \\ \le& \|\boldsymbol{D}_x^{(1)}\|_F\sqrt{\sum_{i=1}^{r_x}\left\|\mathcal{E}_x \times_1 \boldsymbol{v}_i^{(1)} \times_2 \boldsymbol{v}_i^{(1)}\right\|_2^2} \\ \le& \sqrt{r_x}\|\boldsymbol{D}_x^{(1)}\|_F\|\mathcal{E}_x\|_{\mathrm{op}} \\ \le& \sqrt{r_x}(\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle\|\boldsymbol{D}_x^*\|_F + 2\sqrt{2r_x}\|\mathcal{E}_x\|_{\mathrm{op}})\|\mathcal{E}_x\|_{\mathrm{op}} \\ \le& \frac{3}{2}\sqrt{r_x}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle\|\boldsymbol{D}_x^*\|_F\|\mathcal{E}_x\|_{\mathrm{op}}.\end{aligned}$$

where the fourth line is due to (26), and the last line is due to (27). Recall Condition 2, we know that $\left\|\sum_{i=1}^{r_x}(\boldsymbol{D}_x^{(1)})_{i,i}\mathcal{E}_x \times_1 \boldsymbol{v}_i^{(1)} \times_2 \boldsymbol{v}_i^{(1)}\right\|_2 \le \frac{1}{8}\sqrt{r_x}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle\|\boldsymbol{D}_x^*\|_F^2 \le \frac{1}{4}\alpha_\lambda$. Similarly, we can apply the same argument on $\left\|\sum_{i=1}^{r_y}(\boldsymbol{D}_y^{(1)})_{i,i}\mathcal{E}_y \times_1 \boldsymbol{w}_i^{(1)} \times_2 \boldsymbol{w}_i^{(1)}\right\|_2$ and obtain that

$$\left\|\sum_{i=1}^{r_y}(\boldsymbol{D}_y^{(1)})_{i,i}\mathcal{E}_y \times_1 \boldsymbol{w}_i^{(1)} \times_2 \boldsymbol{w}_i^{(1)}\right\|_2 \le \frac{3}{2}\sqrt{r_y}\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle\|\boldsymbol{D}_y^*\|_F \le \frac{1}{4}\alpha_\lambda.$$

Therefore, plugging in these bounds into (25) gives us a deterministic upper bound for $\big|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(1)})\big|$ under Conditions 1 and 2:

$$\begin{aligned}\big|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(1)})\big| \le& \frac{4\|\boldsymbol{e}_\lambda\|_2}{3\alpha_\lambda} \\ \le& \frac{8\|\boldsymbol{e}_\lambda\|_2}{3\langle \boldsymbol{u}^*, \boldsymbol{u}^{(0)}\rangle(\lambda\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)\|\boldsymbol{D}_y^*\|_F^2)} \\ \le& \frac{4\lambda\sqrt{r_x}\|\boldsymbol{D}_x^*\|_F\|\mathcal{E}_x\|_{\mathrm{op}} + 4(1-\lambda)\sqrt{r_y}\|\boldsymbol{D}_y^*\|_F\|\mathcal{E}_y\|_{\mathrm{op}}}{\lambda\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)\|\boldsymbol{D}_y^*\|_F^2}.\end{aligned} \tag{28}$$

Furthermore, (28) and Condition 2 together imply

$$\begin{aligned}\left|\sin\theta(\boldsymbol{u}^*,\boldsymbol{u}^{(1)})\right| \leq & \frac{4\sqrt{r_x}\|\mathcal{E}_x\|_{\mathrm{op}}}{\|\boldsymbol{D}_x^*\|_F} \vee \frac{4\sqrt{r_y}\|\mathcal{E}_y\|_{\mathrm{op}}}{\|\boldsymbol{D}_y^*\|_F} \\ \leq & \frac{1}{3} \\ \leq & \sqrt{1-\frac{32r_x\|\mathcal{E}_x\|_{\mathrm{op}}^2}{\|\boldsymbol{D}_y^*\|_F^2}} \vee \sqrt{1-\frac{32r_y\|\mathcal{E}_y\|_{\mathrm{op}}^2}{\|\boldsymbol{D}_y^*\|_F^2}},\end{aligned}$$

and thus the initialization condition (Condition 1) is also satisfied by $\boldsymbol{u}^{(1)}$. Therefore, we can apply the same arguments with initialization $\boldsymbol{u}^{(k)}$ for $k \geq 0$ to show that (28) holds not only for $\boldsymbol{u}^{(1)}$, but for $\boldsymbol{u}^{(k+1)}$ with any $k \geq 0$.

### B.4.2 Deterministic Bounds for Network Factors

Now we turn to the estimation error of $\boldsymbol{V}^{(k+1)}$ and $\boldsymbol{W}^{(k+1)}$ for $k \geq 1$. To achieve this, we require the following SNR condition.

**Condition 3** (Deterministic SNR condition for network factors in generalized Jisst PCA)**.**

$$\sigma_{r_x}(\boldsymbol{D}_x^*) \geq \frac{3\sqrt{2}}{2}\|\mathcal{E}_x\|_{\mathrm{op}}, \quad \sigma_{r_y}(\boldsymbol{D}_y^*) \geq \frac{3\sqrt{2}}{2}\|\mathcal{E}_y\|_{\mathrm{op}}.$$

Recall our updating rules for $\boldsymbol{V}^{(k+1)}$ and $\boldsymbol{W}^{(k+1)}$ in Algorithm 5: $\boldsymbol{V}^{(k+1)}$ and $\boldsymbol{W}^{(k+1)}$ are the leading $r_x$ and $r_y$ singular vectors of $\mathcal{X} \times_3 \boldsymbol{u}^{(k)}$ and $\mathcal{Y} \times_3 \boldsymbol{u}^{(k)}$, respectively. We can also write

$$\begin{aligned}\mathcal{X} \times_3 \boldsymbol{u}^{(k)} =& \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \boldsymbol{V}^* \boldsymbol{D}_x^* \boldsymbol{V}^{*\prime} + \mathcal{E}_x \times_3 \boldsymbol{u}^{(k)}, \\ \mathcal{Y} \times_3 \boldsymbol{u}^{(k)} =& \langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \boldsymbol{W}^* \boldsymbol{D}_y^* \boldsymbol{W}^{*\prime} + \mathcal{E}_y \times_3 \boldsymbol{u}^{(k)},\end{aligned}$$

with signal matrices $\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \boldsymbol{V}^* \boldsymbol{D}_x^* \boldsymbol{V}^{*\prime}$ and $\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \boldsymbol{W}^* \boldsymbol{D}_y^* \boldsymbol{W}^{*\prime}$ and noise matrices $\mathcal{E}_x \times_3 \boldsymbol{u}^{(k)}$ and $\mathcal{E}_y \times_3 \boldsymbol{u}^{(k)}$. As we have shown earlier, the joint factor $\boldsymbol{u}^{(k)}$ satisfies $|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})| \leq \frac{1}{3}$ for $k \geq 1$, which implies $\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \geq \frac{2\sqrt{2}}{3}$. Furthermore, similar to the proof of Theorem 1, it is not hard to show that $\|\mathcal{E}_x \times_3 \boldsymbol{u}^{(k)}\| \leq \|\mathcal{E}_x\|_{\mathrm{op}}$ and $\|\mathcal{E}_y \times_3 \boldsymbol{u}^{(k)}\| \leq \|\mathcal{E}_y\|_{\mathrm{op}}$. Combining these two results with Condition 3, we have $\|\mathcal{E}_x \times_3 \boldsymbol{u}^{(k)}\| \leq \frac{1}{2}\sigma_{r_x}(\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \boldsymbol{V}^* \boldsymbol{D}_x^* \boldsymbol{V}^{*\prime})$, $\|\mathcal{E}_y \times_3 \boldsymbol{u}^{(k)}\| \leq \frac{1}{2}\sigma_{r_y}(\langle \boldsymbol{u}^*, \boldsymbol{u}^{(k)}\rangle \boldsymbol{W}^* \boldsymbol{D}_y^* \boldsymbol{W}^{*\prime})$. Now we can invoke the Davis-Kahan's theorem (see, e.g., Theorem 2.7 in Chen et al., 2021a) to obtain the following:

$$\|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\| \leq \frac{3\sqrt{2}\|\mathcal{E}_x\|_{\mathrm{op}}}{2\sigma_{r_x}(\boldsymbol{D}_x^*)\rangle}, \quad \|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k+1)})\| \leq \frac{3\sqrt{2}\|\mathcal{E}_y\|_{\mathrm{op}}}{2\sigma_{r_y}(\boldsymbol{D}_y^*)\rangle} \tag{29}$$

hold for any $k \geq 2$ as long as Conditions 1-3 are satisfied.

### B.4.3 Probabilistic Bounds with Sub-Gaussian Noise

Now we would like to show that under Assumption 5, Conditions 1 and 2 are satisfied with high probability; If Assumption 6 also holds, then Conditions 1-3 are all satisfied. As has been shown

in the proof of Theorem 2, with probability at least $1-4\exp\{-N\}$, $\|\mathcal{E}_x\|_{\text{op}} \le C\sigma(\sqrt{N+p})$, $\|\mathcal{E}_y\|_{\text{op}} \le C\sigma(\sqrt{N+q})$. As long as the constant $C>0$ in Assumption 5 is chosen sufficiently large, Condition 2 holds. In addition, Condition 2 also implies $\sqrt{1-\frac{32r_x\|\mathcal{E}_x\|_{\text{op}}^2}{\|\boldsymbol{D}_x^*\|_F^2}}, \sqrt{1-\frac{32r_y\|\mathcal{E}_y\|_{\text{op}}^2}{\|\boldsymbol{D}_y^*\|_F^2}} \ge \sqrt{1-\frac{2}{9}} = \frac{\sqrt{7}}{3}$. To show that Condition 1 holds, we can apply the same arguments as in the proof of Proposition 1. Note that similar to the vanilla setting with scalar $d_x^*$ and $d_y^*$, we can also write $\boldsymbol{u}^{(0)}$ as the top left singular vector of $d_\lambda \boldsymbol{u}^*\boldsymbol{z}' + \boldsymbol{E}_\lambda$, where $d_\lambda = \sqrt{\lambda^2\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)^2\|\boldsymbol{D}_y^*\|_F^2}$, $\boldsymbol{z} = \text{Norm}(\Big[\lambda\text{Vec}\left(\boldsymbol{V}\boldsymbol{D}_x^*\boldsymbol{V}'\right)', (1-\lambda)\text{Vec}\left(\boldsymbol{W}\boldsymbol{D}_y^*\boldsymbol{W}'\right)'\Big]$, and $\boldsymbol{E}_\lambda = [\lambda\mathcal{M}_3(\mathcal{E}_x), (1-\lambda)\mathcal{M}_3(\mathcal{E}_y)]$. By Assumption 5 which is analogous to Assumption 1, we have

$$\|d_\lambda \boldsymbol{u}^*(\boldsymbol{E}_\lambda\boldsymbol{z})' + d_\lambda \boldsymbol{E}_\lambda\boldsymbol{z}\boldsymbol{u}^{*\prime} + \boldsymbol{E}_\lambda\boldsymbol{E}_\lambda' - \mathbb{E}(\boldsymbol{E}_\lambda\boldsymbol{E}_\lambda')\| \le \frac{1}{2}d_\lambda^2,$$

and hence applying the Davis-Kahan's theorem gives us the following bound with probability at least $1-CN^{-c}$:

$$\begin{aligned}
\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(0)}) \le& \frac{C\sigma\left(\sqrt{N} + (N(p^2+q^2))^{\frac{1}{4}}\right)\sqrt{\log N}}{d_\lambda} \\
\le& \frac{\sqrt{7}}{3} \\
\le& \min\left\{\sqrt{1-\frac{32r_x\|\mathcal{E}_x\|_{\text{op}}^2}{\|\boldsymbol{D}_x^*\|_F^2}}, \sqrt{1-\frac{32r_y\|\mathcal{E}_y\|_{\text{op}}^2}{\|\boldsymbol{D}_y^*\|_F^2}}\right\},
\end{aligned}$$

as long as the constant $C>0$ in Assumption 5 is sufficiently large. Therefore, under Assumption 5, with probability at least $1-CN^{-c}$, the following holds for $k \ge 1$:

$$\begin{aligned}
|\sin\theta(\boldsymbol{u}^*, \boldsymbol{u}^{(k)})| \le& \frac{C\sigma\left(\lambda\sqrt{r_x(p+N)}\|\boldsymbol{D}_x^*\|_F + (1-\lambda)\sqrt{r_y(q+N)}\|\boldsymbol{D}_y^*\|_F\right)}{\lambda\|\boldsymbol{D}_x^*\|_F^2 + (1-\lambda)\|\boldsymbol{D}_y^*\|_F^2} \\
\le& \frac{C\sigma\sqrt{r_x(p+N)}}{\|\boldsymbol{D}_x^*\|_F} \vee \frac{C\sigma\sqrt{r_y(q+N)}}{\|\boldsymbol{D}_y^*\|_F}.
\end{aligned}$$

Finally, when Assumption 6 also holds, we can apply the spectral norm bounds of $\mathcal{E}_x$, $\mathcal{E}_y$ again to show that Condition 3 holds with probability at least $1-4\exp\{-N\}$. Then with probability at least $1-CN^{-c}$, the following holds for $k \ge 1$:

$$\|\sin\Theta(\boldsymbol{V}^*, \boldsymbol{V}^{(k+1)})\| \le \frac{C\sigma\sqrt{p+N}}{\sigma_{r_x}(\boldsymbol{D}_x^*)}, \quad \|\sin\Theta(\boldsymbol{W}^*, \boldsymbol{W}^{(k+1)})\| \le \frac{C\sigma\sqrt{q+N}}{\sigma_{r_y}(\boldsymbol{D}_y^*)}.$$

The proof of Theorem 3 is now complete. In the end of this section, we present the proof of the technical lemma used earlier.

*Proof of Lemma 1.* Since $\boldsymbol{X}_r - \boldsymbol{X}^*$ is of rank at most $2r$, we have

$$\|\boldsymbol{X}_r - \boldsymbol{X}^*\|_F \le \sqrt{2r}\|\boldsymbol{X}_r - \boldsymbol{X}^*\| \le \sqrt{2r}(\|\boldsymbol{X}_r - \boldsymbol{X}\| + \|\boldsymbol{X}^* - \boldsymbol{X}\|).$$

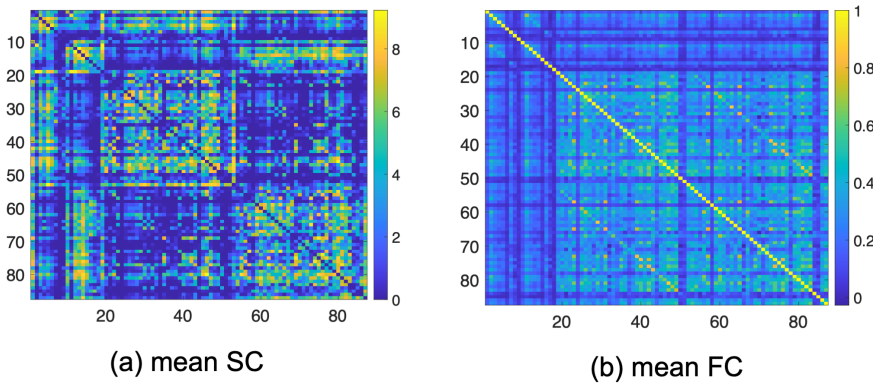


Figure 17: Mean SC (left) and FC (right) from the 1058 HCP subjects.

Meanwhile, since $\boldsymbol{X}_r$ is also the best rank-$r$ approximation of $\boldsymbol{X}$ in terms of spectral norm error, we can write

$$\|\boldsymbol{X}_r - \boldsymbol{X}^*\|_F \leq \sqrt{2r}(\|\boldsymbol{X}_r - \boldsymbol{X}\| + \|\boldsymbol{X}^* - \boldsymbol{X}\|) \leq 2\sqrt{2r}\|\boldsymbol{X}^* - \boldsymbol{X}\| = 2\sqrt{2r}\|\boldsymbol{E}\|.$$

□

# C Additional Results for Real Data Analysis

Figure 17 shows the mean SC and FC from the 1058 subjects.

In Figure 18 panels (a) and (b), we correlate $\boldsymbol{u}_1^*$ and $\boldsymbol{u}_2^*$ with the 45 cognitive traits. In these plots, traits encoded in the opposite direction (e.g., higher values indicate worse cognitive ability) are colored in pink. We observe that 1) $\boldsymbol{u}_1^*$ correlates better with behavioral traits than does $\boldsymbol{u}_2^*$, and 2) most behavioral traits show a decent amount of correlation with the joint factors. Finally, we examine how well we can predict these behavioral traits using both $\boldsymbol{u}_1^*$ and $\boldsymbol{u}_2^*$. Panel (c) displays the correlations between the predicted and measured behavioral traits in test datasets, based on a simple linear regression model. These results are computed as the average over 50 runs of 80-20 random splitting of training and testing datasets.

Figure 19 compares the prediction power of PCA scores with G-JisstPCA scores. We first apply the principal components analysis (PCA) to vectorized SC and FC matrices separately to extract the first two PC scores. We then treat the two PC scores as predictors and use simple linear regression to predict the 45 behavior traits. Panel (a) shows results based on the PCA with the FC data, panel (b) shows results based on the PCA with the SC data, and panel (c) shows results based on JisstPCA with both the SC and FC data.

Figure 20 shows the adjacency matrices for the network loadings for SC ($\{\boldsymbol{V}_1^*\boldsymbol{D}_{SC,1}^*\boldsymbol{V}_1^{*\prime}, \boldsymbol{V}_2^*\boldsymbol{D}_{SC,2}^*\boldsymbol{V}_2^{*\prime}\}$) and FC ($\{\boldsymbol{W}_1^*\boldsymbol{D}_{FC,1}^*\boldsymbol{W}_1^{*\prime}, \boldsymbol{W}_2^*\boldsymbol{D}_{FC,2}^*\boldsymbol{W}_2^{*\prime}\}$). The first 19 ROIs (rows of the adjacency matrix) are subcortical regions and the next 68 ROIs are cortical regions from the Desikan-Killiany atlas. Figure 5 in the main paper shows circular plots with top 200 connections.

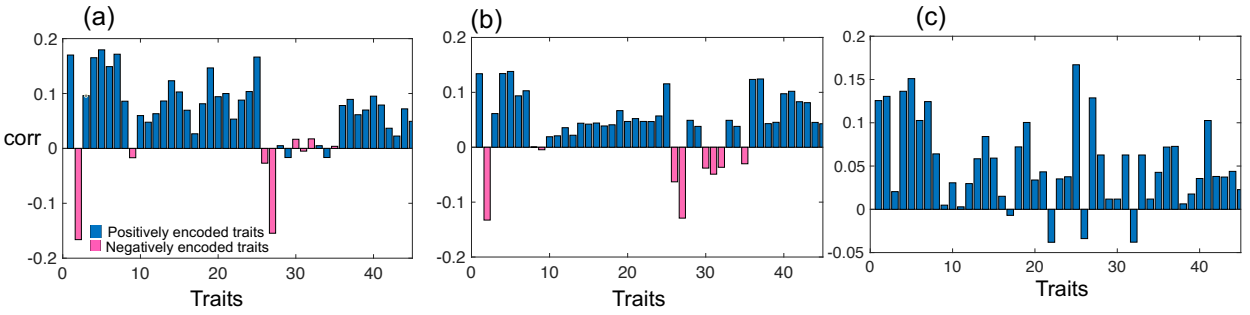


Figure 18: We correlate $\boldsymbol{u}_1^*$ (in panel a) and $\boldsymbol{u}_2^*$ (in panel b) with the 45 cognitive traits. Traits encoded in the opposite direction (e.g., higher values indicate worse cognitive ability) are colored in pink. Panel (c) displays the correlations between the predicted and measured behavioral traits in test datasets, based on a simple linear regression model. These results are computed as the average over 50 runs of 80-20 random splitting of training and testing datasets.

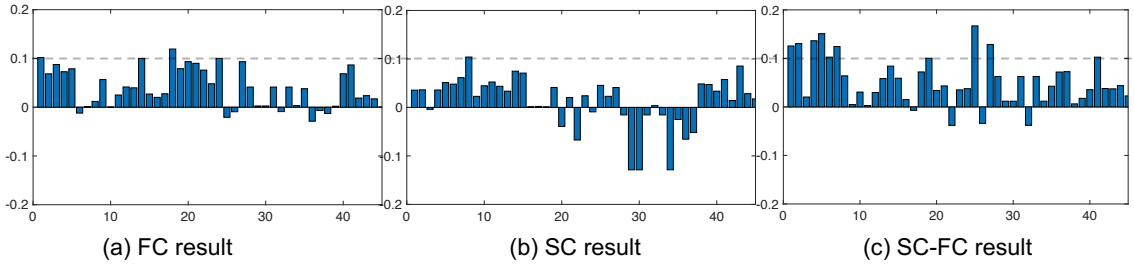


Figure 19: Prediction results using the first two PC scores extracted with different methods. Panel (a) shows results based on the PCA with the FC data, panel (b) shows results based on the PCA with the SC data, and panel (c) shows results based on JisstPCA with both the SC and FC data.

From the correlation plots in panel (a) of Figure 18, it is evident that traits of fluid intelligence assessment, English reading and vocabulary comprehension, and line orientation test have high ($r > 0.15$) and positive correlations with $\boldsymbol{u}_1^*$. The corresponding loadings in the circular plots highlight major subcortical to cortical SC pathways, such as those from the putamen to the frontal lobe and insula, as well as from the thalamus to the parietal lobe. Additionally, several significant cross-hemisphere FC pathways are noted, including those from the left parietal lobe to the right parietal lobe. The positive nature of these pathways in the loadings indicates that higher SC and FC connections correlate with enhanced cognitive abilities. Similarly, the second JisstPCA score shows high and positive correlations with cognitive traits. Although some negative connections are observed in the loadings, positive connections are predominant.

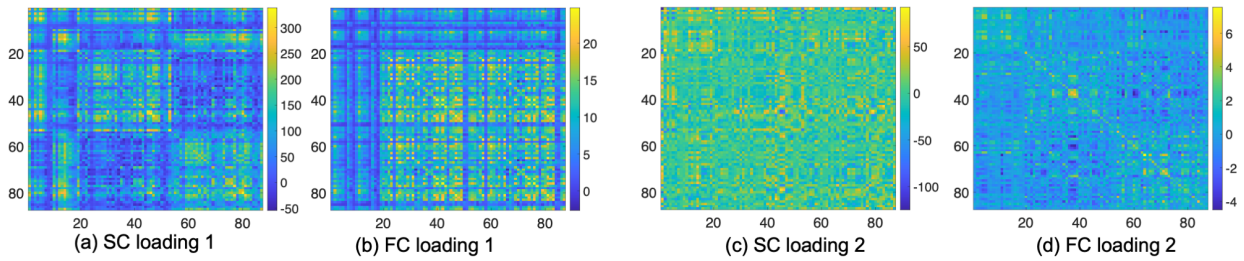


Figure 20: The first two JisstPCA loadings. Panel (a) shows $\boldsymbol{V}_1\boldsymbol{D}_{SC,1}\boldsymbol{V}_1'$, panel (b) shows $\boldsymbol{W}_1\boldsymbol{D}_{FC,1}\boldsymbol{W}_1'$, panel (c) shows $\boldsymbol{V}_2\boldsymbol{D}_{SC,2}\boldsymbol{V}_2'$, panel (b) shows $\boldsymbol{W}_2\boldsymbol{D}_{FC,2}\boldsymbol{W}_2'$

## References


Abdi, H., Williams, L. J., and Valentin, D. (2013). Multiple factor analysis: principal component analysis for multitable and multiblock data sets. *Wiley Interdisciplinary reviews: computational statistics*, 5(2):149–179.

Acar, E., Kolda, T. G., and Dunlavy, D. M. (2011). All-at-once optimization for coupled matrix and tensor factorizations. *arXiv preprint arXiv:1105.3422*.

Acar, E., Nilsson, M., and Saunders, M. (2014). A flexible modeling framework for coupled matrix and tensor factorizations. In *2014 22nd European Signal Processing Conference (EUSIPCO)*, pages 111–115. IEEE.

Allen, G. (2012a). Sparse higher-order principal components analysis. In *Artificial Intelligence and Statistics*, pages 27–36. PMLR.

Allen, G. I. (2012b). Regularized tensor factorizations and higher-order principal components analysis. *arXiv preprint arXiv:1202.2476*.

Anandkumar, A., Ge, R., and Janzamin, M. (2014). Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates. *arXiv preprint arXiv:1402.5180*.

Bullmore, E. and Sporns, O. (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature reviews neuroscience*, 10(3):186–198.

Cai, C., Li, G., Poor, H. V., and Chen, Y. (2019). Nonconvex low-rank tensor completion from noisy data. *Advances in neural information processing systems*, 32.

Carroll, J. D. and Chang, J.-J. (1970). Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35(3):283–319.

Chen, Y., Chi, Y., Fan, J., Ma, C., et al. (2021a). Spectral methods for data science: A statistical perspective. *Foundations and Trends® in Machine Learning*, 14(5):566–806.

Chen, Y.-L., Kolar, M., and Tsay, R. S. (2021b). Tensor canonical correlation analysis with convergence and statistical guarantees. *Journal of Computational and Graphical Statistics*, 30(3):728–744.

Cole, M., Murray, K., St-Onge, E., Risk, B., Zhong, J., Schifitto, G., Descoteaux, M., and Zhang, Z. (2021). Surface-based connectivity integration: An atlas-free approach to jointly study functional and structural connectivity. *Human Brain Mapping*, 42(11):3481–3499.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000a). A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000b). On the best rank-1 and rank-(r 1, r 2,..., rn) approximation of higher-order tensors. *SIAM journal on Matrix Analysis and Applications*, 21(4):1324–1342.

Deng, Y., Tang, X., and Qu, A. (2023). Correlation tensor decomposition and its application in spatial imaging data. *Journal of the American Statistical Association*, 118(541):440–456.

Dey, P., Zhang, Z., and Dunson, D. B. (2022). Outlier detection for multi-network data. *Bioinformatics*, 38(16):4011–4018.

Dobriban, E. and Owen, A. B. (2019). Deterministic parallel analysis: an improved method for selecting factors and principal components. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 81(1):163–183.

Donoho, D., Gavish, M., and Romanov, E. (2023). Screenot: Exact mse-optimal singular value thresholding in correlated noise. *The Annals of Statistics*, 51(1):122–148.

D'Angelo, S., Murphy, T. B., and Alfò, M. (2019). Latent space modelling of multidimensional networks with application to the exchange of votes in eurovision song contest. *The Annals of Applied Statistics*, 13(2):900–930.

Farias, R. C., Cohen, J. E., and Comon, P. (2016). Exploring multimodal data fusion through joint decompositions with flexible couplings. *IEEE Transactions on Signal Processing*, 64(18):4830–4844.

Fu, X., Huang, K., Ma, W.-K., Sidiropoulos, N. D., and Bro, R. (2015). Joint tensor factorization and outlying slab suppression with applications. *IEEE Transactions on Signal Processing*, 63(23):6315–6328.

Gao, L. L., Witten, D., and Bien, J. (2022). Testing for association in multiview network data. *Biometrics*, 78(3):1018–1030.

Ge, R., Ren, Y., Wang, X., and Zhou, M. (2021). Understanding deflation process in over-parametrized tensor decomposition. *Advances in Neural Information Processing Systems*, 34:1299–1311.

Genicot, M., Absil, P.-A., Lambiotte, R., and Sami, S. (2016). Coupled tensor decomposition: a step towards robust components. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 1308–1312. IEEE.

Glasser, M. F., Sotiropoulos, S. N., Wilson, J. A., Coalson, T. S., Fischl, B., Andersson, J. L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J. R., et al. (2013). The minimal preprocessing pipelines for the human connectome project. *Neuroimage*, 80:105–124.

Han, Q., Xu, K., and Airoldi, E. (2015). Consistent estimation of dynamic and multi-layer block models. In *International Conference on Machine Learning*, pages 1511–1520. PMLR.

Han, Y. and Zhang, C.-H. (2022). Tensor principal component analysis in high dimensional cp models. *IEEE Transactions on Information Theory*, 69(2):1147–1167.

Hao, B., Zhang, A. R., and Cheng, G. (2020). Sparse and low-rank tensor estimation via cubic sketchings. In *International Conference on Artificial Intelligence and Statistics*, pages 1319–1330. PMLR.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098.

Hu, J., Lee, C., and Wang, M. (2022). Generalized tensor decomposition with features on multiple modes. *Journal of Computational and Graphical Statistics*, 31(1):204–218.

Jing, B.-Y., Li, T., Lyu, Z., and Xia, D. (2021). Community detection on mixture multilayer networks via regularized tensor decomposition. *The Annals of Statistics*, 49(6):3181–3205.

Jolliffe, I. T. and Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202.

Kim, B., Lee, K. H., Xue, L., and Niu, X. (2018). A review of dynamic network models with latent variables. *Statistics surveys*, 12:105.

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

Li, W., Liu, C.-C., Zhang, T., Li, H., Waterman, M. S., and Zhou, X. J. (2011). Integrative analysis of many weighted co-expression networks using tensor computation. *PLoS computational biology*, 7(6):e1001106.

Lock, E. F., Hoadley, K. A., Marron, J. S., and Nobel, A. B. (2013). Joint and individual variation explained (jive) for integrated analysis of multiple data types. *The annals of applied statistics*, 7(1):523.

Lu, L., Ren, X., Yeh, K.-H., Tan, Z., and Chanussot, J. (2020). Exploring coupled images fusion based on joint tensor decomposition. *Human-centric Computing and Information Sciences*, 10(1):1–26.

Luo, Y., Raskutti, G., Yuan, M., and Zhang, A. R. (2021). A sharp blockwise tensor perturbation bound for orthogonal iteration. *The Journal of Machine Learning Research*, 22(1):8106–8153.

MacDonald, P. W., Levina, E., and Zhu, J. (2022). Latent space models for multiplex networks with shared structure. *Biometrika*, 109(3):683–706.

Mackey, L. (2008). Deflation methods for sparse pca. *Advances in neural information processing systems*, 21.

Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., and Onnela, J.-P. (2010). Community structure in time-dependent, multiscale, and multiplex networks. *science*, 328(5980):876–878.

Murden, R. J., Zhang, Z., Guo, Y., and Risk, B. B. (2022). Interpretive jive: Connections with cca and an application to brain connectivity. *Frontiers in Neuroscience*, 16:969510.

Paul, S. and Chen, Y. (2020a). A random effects stochastic block model for joint community detection in multiple networks with applications to neuroimaging. *Annals of Applied Statistics*, 14(2):993–1029.

Paul, S. and Chen, Y. (2020b). Spectral and matrix factorization methods for consistent community detection in multi-layer networks. *The Annals of Statistics*, 48(1).

Pavlović, D. M., Guillaume, B. R., Towlson, E. K., Kuek, N. M., Afyouni, S., Vértes, P. E., Yeo, B. T., Bullmore, E. T., and Nichols, T. E. (2020). Multi-subject stochastic blockmodels for adaptive analysis of individual differences in human brain network cluster structure. *NeuroImage*, 220:116611.

Raskutti, G., Yuan, M., and Chen, H. (2019). Convex regularization for high-dimensional multiresponse tensor regression. *The Annals of Statistics*, 47(3):1554–1584.

Rubinov, M. and Sporns, O. (2010). Complex network measures of brain connectivity: uses and interpretations. *Neuroimage*, 52(3):1059–1069.

Schenker, C., Cohen, J. E., and Acar, E. (2020). A flexible optimization framework for regularized matrix-tensor factorizations with linear couplings. *IEEE Journal of Selected Topics in Signal Processing*, 15(3):506–521.

Sui, J., Adali, T., Yu, Q., Chen, J., and Calhoun, V. D. (2012). A review of multivariate methods for multimodal fusion of brain imaging data. *Journal of neuroscience methods*, 204(1):68–81.

Tang, T. M. and Allen, G. I. (2021). Integrated principal components analysis. *J. Mach. Learn. Res.*, 22:198–1.

Tomioka, R. and Suzuki, T. (2014). Spectral norm of random tensors. *arXiv preprint arXiv:1407.1870*.

Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E., Yacoub, E., Ugurbil, K., Consortium, W.-M. H., et al. (2013). The wu-minn human connectome project: an overview. *Neuroimage*, 80:62–79.

Vershynin, R. (2010). Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press.

Wang, L., Albera, L., Kachenoura, A., Shu, H., and Senhadji, L. (2014). Canonical polyadic decomposition of third-order semi-nonnegative semi-symmetric tensors using lu and qr matrix factorizations. *EURASIP Journal on Advances in Signal Processing*, 2014:1–23.

Wang, L., Zhang, Z., and Dunson, D. (2019). Common and individual structure of brain networks. *The Annals of Applied Statistics*, 13(1):85–112.

Westerhuis, J. A., Kourti, T., and MacGregor, J. F. (1998). Analysis of multiblock and hierarchical pca and pls models. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 12(5):301–321.

Weylandt, M. and Michailidis, G. (2022). Multivariate analysis for multiple network data via semi-symmetric tensor pca. *arXiv preprint arXiv:2202.04719*.

Winter, S., Zhang, Z., and Dunson, D. (2020). Multi-scale graph principal component analysis for connectomics. *arXiv e-prints*, pages arXiv–2010.

Wold, S. (1978). Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*, 20(4):397–405.

Wu, M., He, S., Zhang, Y., Chen, J., Sun, Y., Liu, Y.-Y., Zhang, J., and Poor, H. V. (2019). A tensor-based framework for studying eigenvector multicentrality in multilayer networks. *Proceedings of the National Academy of Sciences*, 116(31):15407–15413.

Wu, Q., Li, X., Do, Q., Fan, J., Ge, R., and Wang, J. (2018). Ctf-psf: Coupled tensor factorization with partially shared factors. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Xia, D., Yuan, M., and Zhang, C.-H. (2021). Statistically optimal and computationally efficient low rank tensor completion from noisy entries. *The Annals of Statistics*, 49(1).

Yao, Z., Hu, B., Xie, Y., Moore, P., and Zheng, J. (2015). A review of structural and functional brain networks: small world and atlas. *Brain informatics*, 2:45–52.

Yuan, M. and Zhang, C.-H. (2016). On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics*, 16(4):1031–1068.

Zhang, A. and Han, R. (2019). Optimal sparse singular value decomposition for high-dimensional high-order data. *Journal of the American Statistical Association*, 114(528):1708–1725.

Zhang, A. and Xia, D. (2018). Tensor svd: Statistical and computational limits. *IEEE Transactions on Information Theory*, 64(11):7311–7338.

Zhang, A. R., Luo, Y., Raskutti, G., and Yuan, M. (2020a). Islet: Fast and optimal low-rank tensor regression via importance sketching. *SIAM journal on mathematics of data science*, 2(2):444–479.

Zhang, X., Xue, S., and Zhu, J. (2020b). A flexible latent space model for multilayer networks. In *International Conference on Machine Learning*, pages 11288–11297. PMLR.

Zhang, Z., Allen, G. I., Zhu, H., and Dunson, D. (2019). Tensor network factorizations: Relationships between brain structural connectomes and traits. *Neuroimage*, 197:330–343.

Zhang, Z., Descoteaux, M., Zhang, J., Girard, G., Chamberland, M., Dunson, D., Srivastava, A., and Zhu, H. (2018). Mapping population-based structural connectomes. *NeuroImage*, 172:130–145.

Zhou, Y. and Chen, Y. (2023). Deflated heteropca: Overcoming the curse of ill-conditioning in heteroskedastic pca. *arXiv preprint arXiv:2303.06198*.

Zhou, Y., Zhang, A. R., Zheng, L., and Wang, Y. (2022). Optimal high-order tensor svd via tensor-train orthogonal iteration. *IEEE Transactions on Information Theory*, 68(6):3991–4019.