Journal of Machine Learning Research 23 (2022) 1-46 Submitted 2/21; Revised 9/22; Published 10/22

ALMA: Alternating Minimization Algorithm for Clustering Mixture Multilayer Network

Xing Fan FANXING@KNIGHTS.UCF.EDU
Department of Mathematics
University of Central Florida
Orlando, FL 32816, USA

Marianna Pensky MARIANNA.PENSKY@UCF.EDU
Department of Mathematics
University of Central Florida
Orlando, FL 32816, USA

Feng Yu YFENG@KNIGHTS.UCF.EDU
Department of Mathematics
University of Central Florida
Orlando, FL 32816, USA

Teng Zhang * TENG.ZHANG@UCF.EDU
Department of Mathematics
University of Central Florida
Orlando, FL 32816, USA

Editor: Prateek Jain

Abstract

The paper considers a Mixture Multilayer Stochastic Block Model (MMLSBM), where layers can be partitioned into groups of similar networks, and networks in each group are equipped with a distinct Stochastic Block Model. The goal is to partition the multilayer network into clusters of similar layers, and to identify communities in those layers. Jing *et al.* (2020) introduced the MMLSBM and developed a clustering methodology, TWIST, based on regularized tensor decomposition.

The present paper proposes a different technique, an alternating minimization algorithm (ALMA), that aims at simultaneous recovery of the layer partition, together with estimation of the matrices of connection probabilities of the distinct layers. Compared to TWIST, ALMA achieves higher accuracy, both theoretically and numerically.

Keywords: Stochastic Block Model, Multilayer Network, Alternating Minimization, Clustering

1. Introduction

Stochastic networks arise in many areas of research and applications and are used, for example, to study brain connectivity or gene regulatory mechanisms, to monitor cyber and homeland security, and to evaluate and predict social relationships within groups or between groups, such as countries. While in the early years of the field of stochastic networks, research mainly focused on studying a single network, in recent years the frontier moved to

the investigation of collection of networks, the so called *multilayer network*, which allows to model relationships between nodes with respect to various modalities (e.g., relationships between species based on food or space), or consists of network data collected from different individuals (e.g., brain networks).

Although there are many different ways of modeling a multilayer network (see, e.g., an excellent review article of Kivela et al. (2014)), in this paper we consider the case where all layers have the same set of nodes, and all edges between nodes are drawn within layers, i.e., there are no edges connecting the nodes in different layers. MacDonald et al. (2021) called this type of networks the *multiplex* networks and argued that they appear in a variety of applications. Indeed, consider brain networks of several individuals that are drawn on the basis of some imaging modality. The nodes in the networks are associated with brain regions, and the brain regions are considered to be connected if the signals in those regions exhibit some kind of similarity. In this setting, the nodes are the same for each individual network, and there is no connection between brain regions of different individuals. For this reason, one can consider a multiplex network constituted by brain networks of several individuals, with common nodes but possibly different community structures in different layers (individuals). It is known that brain disorders are associated with changes in brain network organizations (see, e.g., Buckner and DiNicola (2019)), and that alterations in the community structure of the brain have been observed in several neuropsychiatric conditions, including Alzheimer disease (see, e.g., Chen et al. (2016)), schizophrenia (see, e.g., Stam (2014)) and epilepsy disease (see, e.g., Munsell et al. (2015)). Hence, assessment of the brain modular organization may provide a key to understanding the relation between aberrant connectivity and brain disease.

The multiplex networks have been studied by many authors who work in a variety of research fields. (see, e.g., Durante et al. (2017), Han and Dunson (2018), Aleta and Moreno (2019), Kao and Porter (2017) among others). In this paper, we consider a multilayer network where all layers are equipped with the Stochastic Block Models (SBM). In this case, the problems of interest include finding groups of layers that are similar in some sense, finding the communities in those groups of layers and estimation of the tensor of connection probabilities. While the scientific community attacked all three of those problems, often in a somewhat ad-hoc manner (see e.g., Brodka et al. (2018), Kao and Porter (2017), Mercado et al. (2018) among others), the theoretically inclined papers in the field of statistics mainly been investigated the case where communities persist throughout all layers of the network. This includes studying the so called "checker board model" in Chi et al. (2020), where the matrices of block probabilities take only a finite number of values, and communities persist in all layers. The tensor block models of Wang and Zeng (2019) and Han et al. (2021) belong to the same category. In recent years, statistics publications extended this type of research to the case, where community structure persists but the matrix of probabilities of connections can take arbitrary values (see, e.g., Bhattacharyya and Chatterjee (2020), Paul and Chen (2020), Lei et al. (2019), Lei and Lin (2020), Paul and Chen (2016) and references therein). The authors studied precision of community detection and provided a comparison between various techniques that can be employed in this case.

In many practical situations, however, the assumption of common community structures in all layers of the network may not be justified. Indeed, as we have stated above, some psychiatric or neurological conditions may be due to the alteration in the brain networks community structures rather than modifications in the strength of connections. For this reason, it is of interest to study a multiplex network with distinct community structures in groups of layers. Recently, Jing et al. (2021) investigated the so called "**M**ixture **M**ulti**L**ayer **S**tochastic **B**lock **M**odel" (MMLSBM), where there are L layers can be partitioned into M different types, with M being a small number. In MMLSBM, each class m of layers is equipped with its own community structure and a distinct matrix of connection probabilities $\boldsymbol{B}_m$, $m = 1, ..., M$. The methodology of Jing et al. (2021) is based on a regularized tensor decomposition, where all tensor dimensions are treated in the same way. The theory is developed under the assumption that the number of layers does not exceed the number of nodes. Note that the latter may not be true, for example, for brain networks, where the number of nodes is in hundreds (and is fixed) while the number of individuals, whose brain images are available, can grow indefinitely.

In this paper, we also consider the MMLSBM and suggest a new algorithm for layer partition and local communities recovery. While the methodology of Jing et al. (2021), called TWIST, is based on a regularized tensor decomposition, our technique is centered around finding the groups of layers. Indeed, the "naive" approach to the problem would be to vectorize all adjacency matrices and cluster them using the k-means procedure. The major difference between our paper and Jing et al. (2021) is that we recognize that it is advantageous to treat within-layer and between-layer dimensions of the adjacency tensor differently. Specifically, we propose a novel **AL**ternating **M**inimization **A**lgorithm (ALMA) which utilizes the fact that, for each layer of the network, the matrix of probabilities of connections can be approximated by a low-rank matrix. As a result, for the MMLSBM, our algorithm consistently recovers the layer labels and the memberships of nodes.

The present paper makes several contributions. First, it introduces the idea that the key to the inference in the MMLSBM is the identification of the groups of layers: as soon as networks in each of M layers are discovered, the communities can be found by the spectral algorithm of Lei and Rinaldo (2015), applied to the averages of the adjacency matrices. In addition, it uses the information that all layers are approximately low-rank. In comparison, the TWIST algorithm of Jing et al. (2021) only uses the information that the underlying tensor is approximately low-rank, which ignores the low-rankness within each layer. Due to this idea, as it follows from our theoretical analysis, ALMA achieves higher accuracy in the between-layer clustering. In addition, unlike the technique in Jing et al. (2021), ALMA does not require the assumption that the number of layers in the network is smaller than the number of nodes. More specifically, if L is the number of layers, n is the number of nodes in each layer and $p_{\max}$ is the sparsity level of the network, the between-layer clustering errors (proportion of misclassified layers) of ALMA and TWIST are

$$R_{BL}^{(ALMA)} = O\left(\frac{\log^4(n+L)}{n^2\, p_{\max}} + \frac{\log^4(n+L)}{n^2[\min(n,L)\, p_{\max}]^2}\right), \quad R_{BL}^{(TWIST)} = O\left(\frac{\log n}{L\, n\, p_{\max}}\right),$$

where the error rate of TWIST holds only if $L \leq n$. Note that the error rates of ALMA hold when $L > n$, and, for $L \leq n$, one has $R_{BL}^{(ALMA)} = o(R_{BL}^{(TWIST)})$ since $L\, n\, p_{\max} \to \infty$. Also, our numerical studies show that ALMA leads to smaller between-layer and within-layer clustering errors than TWIST.

In this paper, we are not interested in the case of $M = 1$, where communities are the same in all layers. Indeed, if one knows that $M = 1$, then, under the assumption that there are only $M = 1$ types of matrices of connection probabilities, one can just find communities by spectral clustering after averaging. For this reason, one should not apply ALMA to the "checker board" or tensor block model, and ALMA should not be compared with techniques designed for this type of models.

Also, we assume that both the number of distinct layers M and the number of communities in each group of layers are fixed and known in advance. While this is usually not true in practice, this is a very common assumption for theoretical investigations. When the algorithm is used in a real data setting, one needs to obtain solutions for several different values of M and then choose the one that agrees with data. Since the probability tensor of the MMLSBM has sets of identical layers, we can borrow the idea from the problem of determining the number of clusters in a data set, when the K-means algorithm is used. One of the most popular heuristic methods is the so called "elbow method". In our setting, we can run the algorithm with an increasing number of clusters M, and plot an error measure of the model as a function of M. This function would decrease as M increases since models with larger M explain more variations. Then, the elbow methods evaluate the curve of the function and find the "elbow of the curve", i.e., the point where the function is no longer decreasing rapidly, as the number of distinct layers M grow (see, e.g., Tibshirani et al. (2001), Zhang et al. (2012); Le and Levina (2015)). Other methods of choosing M include cross-validation (Wang, 2010) and information criterion (Hu and Xu, 2003). After the number of groups of layers has been determined by one of the above mentioned techniques and the between-layers clustering has been implemented, one can identify the number of communities within each group of layers using common techniques employed in the Stochastic Block Models (SBMs) (Zhang et al., 2012; Le and Levina, 2015; Pensky and Zhang, 2019).

Note that dynamic network models can be viewed as a particular case of the multilayer network model where there are no edges connecting the nodes in different layers. The difference between those models and the multilayer network is that, in a dynamic network, the layers are ordered according to time instances, while in a multilayer network the enumeration of layers is completely arbitrary. That is why, although there is a multitude of papers that study the change point detection in the dynamic SBMs (see, e.g., Bhattacharjee et al. (2018), Gangrade et al. (2018) and Wang et al. (2017) among others), the techniques and error bounds in those papers are not applicable in the situation of the MMLSBM.

The rest of the paper is organized as follows. Section 2 describes the MMLSBM and presents the necessary concepts and notations. Section 3 introduces the Alternating Minimization Algorithm (ALMA). Section 4 provides theoretical guarantees for between-layer and within-layer clustering errors. Specifically, the section starts with Section 4.1 that

investigates the situation where ALMA is applied to the true probability tensor. Based on the results of this analysis, Section 4.2 provides assumptions which, as it is confirmed in Section 4.3, guarantee convergence of our algorithm. Finally, Section 4.4 produces upper bounds for between-layer and within-layer clustering errors. Section 4 is concluded by a discussion of various aspects of ALMA in Section 4.5. Section 5 brings up theoretical and numerical comparisons between the ALMA and the TWIST algorithms. In particular, Section 5.1 describes TWIST algorithm while Section 5.2 provides theoretical comparisons between ALMA and TWIST. Section 5.3 produces numerical comparisons between ALMA and TWIST via simulations, and also compares both of them with a simple baseline algorithm. Finally, Section 5.4 extends these comparisons to real data examples. The proofs of all statements in the paper are deferred to Section 6, Appendix.

2. Model framework

This paper considers an L-layer network on the same set of n vertices $\mathcal{V} = \{1, \cdots, n\}$. For any $1 \leq l \leq L$, the observed data is the adjacency matrix $\mathbf{A}_l \in \mathbb{R}^{n\times n}$ of the l-th network, where $\mathbf{A}_l(i,j) = \mathbf{A}_l(j,i) = 1$ if a connection between nodes i and j is observed at the l-th network, and $\mathbf{A}_l(i,j) = \mathbf{A}_l(j,i) = 0$ otherwise. Assume that for all $1 \leq i < j \leq n$ and $1 \leq l \leq L$, $\mathbf{A}_l(i,j)$ are the Bernoulli random variables with $\Pr(\mathbf{A}_l(i,j) = 1) = \mathbf{P}_{*l}(i,j)$, and they are independent with each other. The probability matrices $\{\mathbf{P}_{*l}\}_{l=1}^{L}$ take M different values ($M < L$), that is, there exists a partition of $[L] = \{1, \cdots, L\} = \cup_{m=1}^{M}\mathcal{S}_m$ such that $\mathbf{P}_{*l} = \tilde{\mathbf{Q}}_{*m}$ for all $l \in \mathcal{S}_m$. This means that there exists a clustering function $z : [L] \to [M]$ such that $z(l) = m$ if the l-th network is of the type m, or, equivalently, $l \in \mathcal{S}_m$. Consider a set $\mathcal{F}_{L,M}$ of the *clustering* matrices

$$\mathcal{F}_{L,M} = \left\{\mathbf{Z} \in \{0,1\}^{L\times M}, \quad \mathbf{Z}\mathbf{1} = \mathbf{1}, \quad \mathbf{Z}^T\mathbf{1} \neq \mathbf{0}\right\},$$

and $\mathbf{Z} \in \mathcal{F}_{L,M}$ such that $\mathbf{Z}(l,m) = 1$ if $l \in \mathcal{S}_m$ and $\mathbf{Z}(l,m) = 0$ otherwise, and matrix $\mathbf{Z}$ does not have zero columns. It is easy to see that matrix $\mathbf{Z}^T\mathbf{Z}$ is diagonal, and $\mathbf{W}_* = \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1/2}$ satisfies $\mathbf{W}_*^T\mathbf{W}_* = \mathbf{I}$. Here, $\mathbf{W}_\star(l,m) = L_m^{-1/2}$ if $l \in \mathcal{S}_m$ and $\mathbf{Z}(l,m) = 0$ otherwise, where L_m is the number of networks in the layer of type m, $m = 1, ..., M$.

Furthermore, we assume that each network can be described by a Stochastic Block Model (SBM). Specifically, we assume that, for each m and any $l \in \mathcal{S}_m$, $\mathbf{P}_{*l} = \tilde{\mathbf{Q}}_{*m}$ where $\tilde{\mathbf{Q}}_{*m}$ is generated as follows: the nodes $\mathcal{V}$ are grouped into K_m classes $G_{m,1}, \cdots, G_{m,K_m}$, and the probability of a connection $\mathbf{P}_{*l}(i,j)$ is entirely determined by the groups to which the nodes i and j belong at l. In particular, if $i \in G_{m,k}$ and $j \in G_{m,k'}$, then $\mathbf{P}_{*l}(i,j) = \mathbf{B}_m(k,k')$, where $\mathbf{B}_m \in \mathbb{R}^{K_m\times K_m}$ is the *connectivity matrix* with $\mathbf{B}_m(k,k') = \mathbf{B}_m(k',k)$. In this case, one has

$$\mathbf{P}_{*l} = \mathbf{\Theta}_m\mathbf{B}_m\mathbf{\Theta}_m^T, \quad m = z(l), \quad \mathbf{\Theta}_m \in \mathcal{F}_{n,K_m}, \tag{1}$$

where $\mathbf{\Theta}_m(i,k) = 1$ if and only if node i belongs to the class $G_{m,k}$ and is zero otherwise.

Denote $\mathbf{Q}_{*m} = \sqrt{|\mathcal{S}_m|}\tilde{\mathbf{Q}}_{*m}$. Denote the three-way tensors with the l-th layer $\mathbf{A}_l$ and $\mathbf{P}_{*l}$ by, respectively, $\boldsymbol{A}, \boldsymbol{P}_* \in \mathbb{R}^{L\times n\times n}$, and the three-way tensor with the m-th layer $\mathbf{Q}_{*m}$ by $\boldsymbol{Q}_* \in \mathbb{R}^{M\times n\times n}$. We refer to the description above as the Mixture Multilayer Stochastic Block Model (MMLSBM). Table 1 contains a detailed list of its main parameters.

The objective of this work is to partition the multilayer network $\mathcal{A}$ into M similar layers (between-layer clustering) and furthermore, for each of these sets of layers, to recover

L	Number of layers in the multi-layer network
n	Number of vertices in each layer
M	Number of types of layers
$\cup_{m=1}^{M}\mathcal{S}_m$	Partition of L layers into M types
$L_m = \lvert\mathcal{S}_m\rvert$	The number of layers of type m
$\mathbf{Z} \in \{0,1\}^{L\times M}$	Clustering matrix of L layers
$\mathbf{W}_* = \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1/2}$	Orthonormal basis of the layer clustering matrix
K_m	Number of communities in the layers of the m-th type
$\dot{K} = \sum_{m=1}^{M} K_m$	Total number of communities
$G_{m,1}, \cdots G_{m,K_m}$	Communities in the layers of the m-th type
$\mathbf{B}_m \in \mathbb{R}^{K_m\times K_m}$	Connectivity matrix of the layers of the m-th type
$\boldsymbol{P}_* \in \mathbb{R}^{L\times n\times n}$	Connectivity tensor of the network
$\boldsymbol{A} \in \{0,1\}^{L\times n\times n}$	Tensor of the network, consisting of adjacency matrices of L layers

Table 1: The parameters of the Mixture Multilayer Stochastic Block Model (MMLSBM).

communities $G_{m,1}, \cdots, G_{m,K_m}$, $m = 1, \ldots, M$ (within layer clustering). Specifically, we focus on the setting where M and $\{K_m\}_{m=1}^{M}$ are fixed or grow slowly, while n and L tend to infinity, since usually networks are large but have relatively few similar groups of layers, and the number of communities is also usually small compared to the number of nodes. In order to make the paper more readable, througout the paper we shall illustrate the theoretical developments on a simple example with the two-groups of layers structure.

2.1 Notations

For any matrix $\mathbf{X} \in \mathbb{R}^{n_1\times n_2}$, denote the Frobenius and the operator norm of any matrix $\mathbf{X}$ by $\|\mathbf{X}\|_F$ and $\|\boldsymbol{X}\|$, respectively, and its r-th largest singular value by $\sigma_r(\mathbf{X})$. Let $\mathrm{vec}(\mathbf{X}) \in \mathbb{R}^{n_1 n_2}$ be vectorization of matrix $\mathbf{X}$ obtained by sequentially stacking columns of matrix $\mathbf{X}$. Denote the projection operator onto the nearest orthogonal matrix by Π_o:

$$\Pi_o(\mathbf{X}) = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1/2}. \tag{2}$$

If $n_1 \geq n_2$, then $\Pi_o(\mathbf{X})$ is an orthogonal matrix that has the same column space as $\mathbf{X}$. Specifically, if the singular value decomposition of $\mathbf{X}$ is $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, where $\mathbf{U} \in \mathbb{R}^{n_1\times n_2}$ and $\Sigma, \mathbf{V} \in \mathbb{R}^{n_2\times n_2}$, then $\Pi_o(\mathbf{X}) = \mathbf{U}\mathbf{V}^T$.

For any tensor $\boldsymbol{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, its mode 1 matricization $\mathcal{M}_1(\boldsymbol{X}) \in \mathbb{R}^{n_1\times n_2 n_3}$ is a matrix such that $[\mathcal{M}_1(\boldsymbol{X})](l,:) = \mathrm{vec}(\boldsymbol{X}(l,:,:))$. For any tensor $\boldsymbol{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$ and a matrix $\mathbf{A} \in \mathbb{R}^{m\times n_1}$, their mode-1 product $\boldsymbol{X} \times_1 \mathbf{A}$ is a tensor in $\mathbb{R}^{m\times n_2\times n_3}$ defined by

$$[\boldsymbol{X} \times_1 \mathbf{A}](j, i_2, i_3) = \sum_{i_1=1}^{n_1} \boldsymbol{X}(i_1, i_2, i_3)\mathbf{A}(i_1, j), \quad j = 1, ..., m.$$

In this product, every mode-1 fiber of tensor $\boldsymbol{X}$ is multiplied by matrix $\mathbf{A}$:

$$\mathbf{Y} = \mathbf{X} \times_1 \mathbf{A} \Longleftrightarrow \widehat{\mathbf{Y}} = \mathbf{A}\widehat{\mathbf{X}}, \quad \widehat{\mathbf{Y}} = \mathcal{M}_1(\mathbf{Y}),\ \widehat{\mathbf{X}} = \mathcal{M}_1(\mathbf{X}) \tag{3}$$

If $\boldsymbol{X} \in \mathbb{R}^{n\times n_2\times n_3}$ and $\boldsymbol{Y} \in \mathbb{R}^{m\times n_2\times n_3}$ are two tensors, their mode-(2,3) product denoted by $\boldsymbol{X} \times_{2,3} \boldsymbol{Y}$, is a matrix in $\mathbb{R}^{n\times m}$ with elements $i_1 = 1, ..., n$, $i_2 = 1, ..., m$

$$[\boldsymbol{X} \times_{2,3} \boldsymbol{Y}](i_1, i_2) = \sum_{j_2=1}^{n_2}\sum_{j_3=1}^{n_3} \boldsymbol{X}(i_1, j_2, j_3)\boldsymbol{Y}(i_2, j_2, j_3) = \mathrm{Tr}[X(i_1, :, :)Y(i_2, :, :)^T]$$

The Frobenius norm $\|\boldsymbol{X}\|_F$ and the largest singular value $\sigma_1(\boldsymbol{X})$ of a tensor $\boldsymbol{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$ are defined by

$$\|\boldsymbol{X}\|_F = \sqrt{\sum_{i_1,i_2,i_3=1}^{n_1,n_2,n_3} \boldsymbol{X}(i_1, i_2, i_3)^2},$$
$$\sigma_1(\boldsymbol{X}) = \|\boldsymbol{X}\| = \max_{\mathbf{u}_i\in\mathbb{R}^{n_i}, \|\mathbf{u}_i\|=1, 1\le i\le 3} \boldsymbol{X} \times_1 \mathbf{u}_1 \times_2 \mathbf{u}_2 \times_3 \mathbf{u}_3.$$

Operations above obey the following properties:
1. For $\mathbf{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, $\mathbf{A} \in \mathbb{R}^{m\times n_1}$, $\mathbf{Y} \in \mathbb{R}^{n\times n_2\times n_3}$ one has $(\mathbf{X}\times_1\mathbf{A})\times_{2,3}\mathbf{Y} = \mathbf{A}^T(\mathbf{X}\times_{2,3}\mathbf{Y})$
2. If $\mathbf{W}$ is such that $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, then $\|\mathbf{Q} \times_1 \mathbf{W}^T\|_F^2 = \|\mathbf{Q}\|_F^2$

For a more comprehensive tutorial for tensor algebra, please, see the review article of Kolda and Bader (2009).

Next, we introduce some notations that will be used later in the paper. Let $\mathbf{K} = (K_1, \ldots, K_M)$, and denote

$$K_{\max} = \max_{m=1,\cdots,M} K_m, \quad K_{\min} = \min_{m=1,\cdots,M} K_m, \quad \dot{K} = \sum_{m=1}^{M} K_m. \tag{4}$$

Denote the size of the smallest cluster in all networks by $g_{\min}$, i.e., $g_{\min} = \min\limits_{\substack{1\le m\le M\\ 1\le k\le K_m}} |G_{m,k}|$.

Consider the SVDs of matrices $\boldsymbol{Q}_*(m, :, :)$ and the matrices $\Pi_{\mathbf{U}_m^\perp}$ orthogonal to the linear spaces of their eigenvectors:

$$\boldsymbol{Q}_*(m, :, :) = \mathbf{U}_m\boldsymbol{\Lambda}_m\mathbf{U}_m^T, \quad \Pi_{\mathbf{U}_m^\perp} = \mathbf{I} - \mathbf{U}_m\mathbf{U}_m^T \tag{5}$$

Since $\mathrm{rank}(\boldsymbol{Q}_*(m, :, :)) = K_m$, $\mathbf{U}_m \in \mathbb{R}^{n\times K_m}$ is an orthogonal matrix that has the same column space as $\boldsymbol{Q}_*(m, :, :)$. Note that we somewhat abuse notations here: $\Pi_{\mathbf{U}_m^\perp}$ is a matrix and also an operator, so that, for any matrix $\mathbf{X}$, $\Pi_{\mathbf{U}_m^\perp}(\mathbf{X})$ is a projection of the matrix $\mathbf{X}$ on the linear space orthogonal to the column space of matrix $\mathbf{U}_m$.

Now, we introduce operators that will be used later in the paper. For any tensor $\boldsymbol{X} \in \mathbb{R}^{M\times n\times n}$, define a projector $\Pi_{\mathbf{K}} : \mathbb{R}^{M\times n\times n} \to \mathbb{R}^{M\times n\times n}$ by

$$[\Pi_{\mathbf{K}}(\boldsymbol{X})](m, :, :) = \Pi_{K_m}\big(\boldsymbol{X}(m, :, :)\big), \ m = 1, \ldots, M, \tag{6}$$

where $\Pi_{K_m} : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ is the projection onto the nearest rank K_m matrix. Consider operator $\Pi_{T,\mathbf{K}} : \mathbb{R}^{M\times n\times n} \to \mathbb{R}^{M\times n\times n}$ defined as

$$[\Pi_{T,\mathbf{K}}(\boldsymbol{X})](m, :, :) = \boldsymbol{X}(m, :, :) - \Pi_{\mathbf{U}_m^\perp}\boldsymbol{X}(m, :, :)\Pi_{\mathbf{U}_m^\perp}, \tag{7}$$

where $\Pi_{\mathbf{U}_m^\perp}$ is defined in (5). In addition, let $\Pi_{T,K_m} : \mathbb{R}^{L\times n\times n} \to \mathbb{R}^{L\times n\times n}$ be the projection onto the subspace spanned by $\mathbf{U}_m$ for each "slice" of the tensor, i.e.,

$$[\Pi_{T,K_m}(\boldsymbol{X})](m', :, :) = \boldsymbol{X}(m', :, :) - \Pi_{\mathbf{U}_m^\perp}\boldsymbol{X}(m', :, :)\Pi_{\mathbf{U}_m^\perp}, \quad m' = 1, \ldots, M,$$

3. Alternating Minimization Algorithm (ALMA)

As we observe the multi-layer networks $\{\boldsymbol{A}_l\}_{l=1}^L$, our objectives are

- **Between-layer clustering:** recover the network classes $\mathcal{S}_1, \cdots, \mathcal{S}_M$ such that $[L] = \cup_{m=1}^M \mathcal{S}_m$.
- **Within-layer clustering:** recover the community structures for each network class, i.e., for any $m \in [M]$, find a partition of the vertices $G_{m,1}, \cdots, G_{m,K_m}$.

To achieve these goals, we start with the estimation of $\boldsymbol{Q}_* \in \mathbb{R}^{M\times n\times n}$ and $\mathbf{W}_* \in \mathbb{R}^{L\times M}$ based on $\boldsymbol{A}$. Then, the between-layer clustering can be carried out by applying K-means algorithm to the rows of the estimator $\widehat{\mathbf{W}}$ of matrix $\mathbf{W}_*$. Subsequently, the within-layer clustering of the m-th group of networks can be obtained by analyzing estimators $\hat{\boldsymbol{Q}}_*(m,:,:)$ of $\boldsymbol{Q}_*(m,:,:)$ for every $m \in [M]$.

In order to estimate $\boldsymbol{Q}_*$ and $\mathbf{W}_*$, note that the tensor $\boldsymbol{A}$ can be considered as a noisy observation of $\boldsymbol{P}_*$, since $\mathbb{E}(\boldsymbol{A}) = \boldsymbol{P}_*$, where $\boldsymbol{P}_* = \boldsymbol{Q}_* \times_1 \mathbf{W}_*^T$, and $\mathbf{W}_*$ is an orthogonal matrix by definition. For this reason, we propose to find $\boldsymbol{Q}_*$ and $\mathbf{W}_*$ by solving the following optimization problem

$$\begin{aligned} &\underset{\boldsymbol{Q},\mathbf{W}}{\operatorname{argmin}} \ \|\boldsymbol{A} - \boldsymbol{Q} \times_1 \mathbf{W}^T\|_F \qquad (8)\\ &\text{s.t. } \boldsymbol{Q} \in \mathbb{R}^{M\times n\times n}, \mathbf{W} \in \mathbb{R}^{L\times M}, \ \mathbf{W}^T\mathbf{W} = \mathbf{I}, \ \operatorname{rank}(\boldsymbol{Q}(m,:,:)) \le K_m \text{ for all } 1 \le m \le M. \end{aligned}$$

We solve (8) by alternatively minimizing the objective function in (8) over $\mathbf{Q}$ and $\mathbf{W}$.

When $\mathbf{W}$ is fixed, the best approximation to $\boldsymbol{Q}$ is given by $\boldsymbol{Q} = \Pi_{\mathbf{K}}(\boldsymbol{A}\times_1\mathbf{W})$. Indeed, by equation (3), one has $\|\boldsymbol{A}-\boldsymbol{Q}\times_1\mathbf{W}\|_F = \|\mathcal{M}_1(\boldsymbol{A})-\mathbf{W}\mathcal{M}_1(\boldsymbol{Q})\|$. Hence, minimization of the last expression over $\boldsymbol{Q}$ yields $\mathcal{M}_1(\boldsymbol{Q}) = \mathbf{W}^T\mathcal{M}_1(\boldsymbol{A})$ which, by (3), leads to $\boldsymbol{Q} = \boldsymbol{A} \times_1 \mathbf{W}$. The latter, due to the rank restrictions, is approximated by the closest rank projection $\Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W})$.

When $\boldsymbol{Q}$ is fixed, the problem of minimizing of $\|\boldsymbol{A} - \boldsymbol{Q} \times_1 \mathbf{W}^T\|_F$ over $\mathbf{W}$ under the assumption that $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, is called the orthogonal Procrustes problem (see, e.g., Gower and Dijksterhuis (2004)), and it has an explicit solution $\mathbf{W} = \Pi_o(\boldsymbol{A} \times_{2,3} \boldsymbol{Q})$, where $\Pi_o(\mathbf{X})$ is defined in (2). Combining the two steps, we summarize this alternating minimization procedure in Algorithm 1.

After obtaining $\widehat{\mathbf{W}}$ and $\widehat{\boldsymbol{Q}}$, we recover the groups of similar networks $\mathcal{S}_1, \cdots, \mathcal{S}_M$ by clustering the rows of $\widehat{\mathbf{W}}$ into M groups using the $(1+\epsilon)$ approximate K-means algorithm. Finally, for clustering the nodes in each type of networks, we apply spectral clustering with $\widehat{\boldsymbol{Q}}(m,:,:)$ being treated as the affinity matrix. Specifically, we first find the orthogonal matrix of size $n \times K_m$ whose columns are the top K_m eigenvectors of $\widehat{\boldsymbol{Q}}(m,:,:)$, and then cluster its rows into K_m groups using the $(1+\epsilon)$ approximate K-means. There exist efficient algorithms for solving the $(1+\epsilon)$ approximate K-means problem, see, e.g., Kumar et al. (2004a).

Initialization We remark that Algorithm 1 requires an initialization clustering matrix $\mathbf{W}^{(1)}$. For this reason, we present an initialization procedure, which is based on an initial estimator of the between-layer clustering. This procedure is summarized in Algorithm 2,

Algorithm 1 Alternating Minimization Algorithm (ALMA)

Input: Adjacency tensor $\boldsymbol{A} \in \mathbb{R}^{L\times n\times n}$; number of different types of networks M; $\{K_m\}_{m=1}^M$; Initialization clustering matrix $\mathbf{W}^{(1)} \in \mathbb{R}^{L\times M}$ such that $(\mathbf{W}^{(1)})^T\mathbf{W}^{(1)} = \mathbf{I}$
Output: A clustering matrix $\widehat{\mathbf{W}} \in \mathbb{R}^{L\times M}$ such that $\widehat{\mathbf{W}}^T\widehat{\mathbf{W}} = \mathbf{I}$, and a tensor $\widehat{\boldsymbol{Q}} \in \mathbb{R}^{M\times n\times n}$.
Steps:
1: Set iter $= 1$.
2: Let $\boldsymbol{Q}^{(\text{iter}+1)} = \Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\text{iter})})$, where $\Pi_{\mathbf{K}}$ is a projector defined in (6).
3: Let $\mathbf{W}^{(\text{iter}+1)} = \Pi_o(\boldsymbol{A} \times_{2,3} \boldsymbol{Q}^{(\text{iter}+1)})$ where $\Pi_o(\mathbf{X})$ is defined in (2).
4: Set iter $=$ iter $+ 1$.
5: Repeat steps 2-4 until $\|\mathbf{W}^{(\text{iter})} - \mathbf{W}^{(\text{iter}-1)}\|_F \le \epsilon_{n,L}$ where $\epsilon_{n,L}$ is a pre-specified threshold, or the number of iterations exceeds the upper limit: iter $> N^{(iter)}$.
6: Set $\widehat{\mathbf{W}} = \mathbf{W}^{(\text{iter}-1)}$, $\widehat{\boldsymbol{Q}} = \boldsymbol{Q}^{(\text{iter}-1)}$.

Algorithm 2 Initialization of Algorithm 1

Input: Adjacency matrices $\mathbf{A}_l \in \mathbb{R}^{n\times n}$, $1 \le l \le L$; number of different layers M; $\dot{K}$.
Output: $\mathbf{W}^{(1)}$.
Steps:
1: Set $\mathbf{W}^{(0)} \in \mathbb{R}^{L\times M}$ to be composed of the top M left singular vectors of $\mathcal{M}_3(\boldsymbol{A})$.
2: Apply the $(1+\epsilon)$ approximate K-means algorithm to the rows of $\mathbf{W}^{(0)}$ to obtain the initial between layer clustering $[L] = \cup_{m=1}^M \tilde{\mathcal{S}}_m$.
3: Apply (9) to obtain $\mathbf{W}^{(1)}$ from $[L] = \cup_{m=1}^M \tilde{\mathcal{S}}_m$.

In particular, we first obtain an initial between-layer clustering matrix $\mathbf{Z}^{(1)} \in \{0,1\}^{L\times M}$ using, e.g., spectral clustering algorithm on vectorized layer matrices $\{\mathbf{A}_l\}_{l=1}^L$, and then set

$$\mathbf{W}^{(1)} = \mathbf{Z}^{(1)}((\mathbf{Z}^{(1)})^T\mathbf{Z}^{(1)})^{-1/2}. \tag{9}$$

One can also use alternative methods to generate $\mathbf{W}^{(0)}$, and subsequently follow Steps 2 and 3 in Algorithm 2. For example, one can follow initialization strategy in (Jing et al., 2021, Section 5.5) to obtain $\mathbf{W}^{(0)}$. This, however, will require additional assumptions to comply with the theory in Jing et al. (2021).

For the initial clustering, we use the $(1+\epsilon)$ approximate solution of the K-means algorithm, that is, the solution whose objective value is within $(1+\epsilon)$ of the optimal value. While the K-means clustering is NP-hard, efficient procedures exist for finding such an approximate solution (see, e.g., Kumar et al. (2004b)).

4. Theoretical guarantees

4.1 Convergence of the iterative algorithm for the true probability tensor

The purpose of this section is to explain how Algorithm 1 works. Indeed, in order this algorithm delivers acceptable solutions when it is applied the adjacency tensor $\boldsymbol{A}$, it should guarantee convergence when $\boldsymbol{A}$ is replaced by $\boldsymbol{P}_*$, and one starts from an arbitrary matrix $\mathbf{W}^{(1)}$ close to $\mathbf{W}_*$. In this case, the associated optimization problem becomes

$$\begin{aligned} &\underset{\boldsymbol{Q},\mathbf{W}}{\operatorname{argmin}} \|\boldsymbol{P}_* - \boldsymbol{Q} \times_1 \mathbf{W}^T\|_F \\ &\text{s.t. } \boldsymbol{Q} \in \mathbb{R}^{M\times n\times n}, \mathbf{W} \in \mathbb{R}^{L\times M},\ \mathbf{W}^T\mathbf{W} = \mathbf{I},\ \operatorname{rank}(\boldsymbol{Q}(m,:,:)) \le K_m, \text{ for all } 1 \le m \le M. \end{aligned} \tag{10}$$

Then, Algorithm 1 yields

$$\boldsymbol{Q}^{(\text{iter})} = \Pi_{\mathbf{K}}(\boldsymbol{P}_* \times_1 \mathbf{W}^{(\text{iter}-1)}), \quad \mathbf{W}^{(\text{iter})} = \mathbf{W}_*\Pi_o(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}^{(\text{iter})}), \tag{11}$$

where the latter formula is obtained using $\boldsymbol{P}_* = \boldsymbol{Q}_* \times_1 \mathbf{W}_*^T$. Hence,

$$\mathbf{W}^{(\text{iter})} = \Pi_o(\boldsymbol{P}_* \times_{2,3} \boldsymbol{Q}^{(\text{iter})}) = \Pi_o(\mathbf{W}_*(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}^{(\text{iter})})) = \mathbf{W}_*\Pi_o(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}^{(\text{iter})}).$$

As a result, we can reformulate problem (10) by writing $\mathbf{W} = \mathbf{W}_*\mathbf{V}$ for some $\mathbf{V} \in \mathbb{R}^{M\times M}$. Then (10) is simplified to

$$\begin{aligned} &\underset{\boldsymbol{Q},\mathbf{V}}{\operatorname{argmin}} \|\boldsymbol{Q}_* - \boldsymbol{Q} \times_1 \mathbf{V}^T\|_F \\ &\text{s.t. } \boldsymbol{Q} \in \mathbb{R}^{M\times n\times n}, \mathbf{V} \in \mathbb{R}^{M\times M},\ \mathbf{V}^T\mathbf{V} = \mathbf{I},\ \operatorname{rank}(\boldsymbol{Q}(m,:,:)) \le K_m \text{ for all } 1 \le m \le M, \end{aligned} \tag{12}$$

and the iterative relations (11) become

$$\boldsymbol{Q}^{(\text{iter})} = \Pi_{\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{V}^{(\text{iter}-1)}), \quad \mathbf{V}^{(\text{iter})} = \Pi_o(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}^{(\text{iter})}), \tag{13}$$

where the first equation follows from the fact that

$$\boldsymbol{P}_* \times_1 \mathbf{W}_*\mathbf{V}^{(\text{iter}-1)} = (\boldsymbol{P}_* \times_1 \mathbf{W}_*) \times_1 \mathbf{V}^{(\text{iter}-1)} = \boldsymbol{Q}_* \times_1 \mathbf{V}^{(\text{iter}-1)}.$$

The latter implies that, for the sets $\mathcal{S}_1$ and $\mathcal{S}_2$ in $\mathbb{R}^{M\times n\times n}$ defined by

$$\begin{aligned} \mathcal{S}_1 &= \{\boldsymbol{Q} : \operatorname{rank}(\boldsymbol{Q}(m,:,:)) \le K_m \text{ for all } 1 \le m \le M\}, \\ \mathcal{S}_2 &= \{\boldsymbol{Q} = \boldsymbol{Q}_* \times_1 \mathbf{V} : \mathbf{V} \in \mathbb{R}^{M\times M}, \mathbf{V}^T\mathbf{V} = \mathbf{I}\}, \end{aligned}$$

$\boldsymbol{Q}^{(\text{iter})}$ is the nearest point on $\mathcal{S}_1$ to $\boldsymbol{Q}_* \times_1 \mathbf{V}^{(\text{iter}-1)}$, and $\boldsymbol{Q}_* \times_1 \mathbf{V}^{(\text{iter})}$ is the nearest point on $\mathcal{S}_2$ to $\boldsymbol{Q}^{(\text{iter})}$. Hence, the update formula (13) can be viewed as an alternating projection between $\mathcal{S}_1$ and $\mathcal{S}_2$.

Denote the tangent planes to the sets $\mathcal{S}_1$ and $\mathcal{S}_2$ at $\boldsymbol{Q}_*$ by L_1 and L_2, respectively. Then, the explicit formulas for L_1 and L_2 are given by

$$\begin{aligned} L_1 &= \{\boldsymbol{Q} \in \mathbb{R}^{M\times n\times n} : \Pi_{\mathbf{U}_m^\perp}\boldsymbol{Q}(m,:,:)\Pi_{\mathbf{U}_m^\perp} = \mathbf{0}, \quad m = 1,\dots,M,\} \\ L_2 &= \{\boldsymbol{Q} = \boldsymbol{Q}_* \times_1 \mathbf{X} : \mathbf{X} \in \text{Skew}_M\}, \end{aligned} \tag{14}$$

where Skew_M represents the set of skew-symmetric matrices of size $M \times M$. The intuition, the formal definition of tangent space, and the derivations of L_1 and L_2 are deferred to Section 6.1.

Hence, the "alternating projection" viewpoint of (13) reveals that it is approximately an alternating projection procedure between the subspaces L_1 and L_2. Since the projections onto L_1 and L_2 are linear operators, the convergence rate of this alternating projection method can be described by the operator norm of the composite operator:

$$\kappa_H = \max_{\boldsymbol{X} \in \mathbb{R}^{M \times n \times n}} \frac{\|P_{L_2} P_{L_1} \boldsymbol{X}\|_F}{\|\boldsymbol{X}\|_F}. \tag{15}$$

Since P_{L_1} and P_{L_2} are projection operators, one has $\|P_{L_1}(\boldsymbol{X})\|_F \leq \|\boldsymbol{X}\|_F$, $\|P_{L_2}(\boldsymbol{X})\|_F \leq \|\boldsymbol{X}\|_F$ and, therefore, $\kappa_H \leq 1$.

Note that κ_H can be expressed via the smallest principal angle θ between planes L_1 and L_2: $\kappa_H = \cos(\theta)$. In particular, $\kappa_H = 0$ if L_1 and L_2 are perpendicular to each other, and $\kappa_H = 1$ if the intersection $L_1 \cap L_2$ is nontrivial. Since the algorithm in (13) is approximately an alternating projection procedure between the subspaces L_1 and L_2, it converges faster for smaller values of κ_H. Note that $\kappa_H = 1$ if and only if $P_{L_2} P_{L_1} \boldsymbol{X} = \boldsymbol{X}$ for some $\boldsymbol{X}$, i.e., when there is a nontrivial intersection between planes L_1 and L_2.

Illustrative example. Consider an example, of a multilayer network with L layers and n nodes, where layers can be partitioned into $M = 2$ groups and each of the groups of layers has $K_1 = K_2 = 2$ communities. For simplicity, we assume that layers of the network are perfectly balanced, i.e., $L_1 = L_2 = L/2$, and each of those layers contains two communities of sizes exactly $n/2$. Furthermore, we assume that all layers have the same matrix of block probabilities $\mathbf{B}_1 = \mathbf{B}_2 = \mathbf{B}$ where $\mathbf{B}_{i,i} = p$ and $\mathbf{B}_{i,j} = \alpha p$, $i, j = 1, 2$, $i \neq j$, $0 < \alpha < 1$. The latter means that a pair of vertices in different communities are α times less likely to be connected by an edge than two vertices in the same community.

In this example, all differences between the layers come from the community structures, so we assume that βn, $0 < \beta < 1$, nodes switch their community memberships between groups of layers. For simplicity we assume that $\beta n/2$ nodes move from community 1 to community 2 and visa versa, so that $|G_{1,i} \cap G_{2,i}| = (1 - \beta)n/2$, $i = 1, 2$. Since $M = 2$, the skew-symmetric matrix of size 2 must be a scalar multiple of $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$. Hence, one has $\dim(L_2) = 1$, and κ_H can be explicitly written as

$$\kappa_H = \sqrt{\frac{\|\boldsymbol{Q}_*(1,:,:) - \Pi_{\mathbf{U}_2^\perp} \boldsymbol{Q}_*(1,:,:) \Pi_{\mathbf{U}_2^\perp}\|_F^2 + \|\boldsymbol{Q}_*(2,:,:) - \Pi_{\mathbf{U}_1^\perp} \boldsymbol{Q}_*(2,:,:) \Pi_{\mathbf{U}_1^\perp}\|_F^2}{\|\boldsymbol{Q}_*(1,:,:)\|_F^2 + \|\boldsymbol{Q}_*(2,:,:)\|_F^2}}$$

and $\kappa_H = 0$ only if $\beta = 0$. In particular, under the settings above, $\|\boldsymbol{Q}_*(i,:,:)\|_F^2 = Ln^2(1 + \alpha^2)p^2/4$ and $\|\boldsymbol{Q}_*(i,:,:) - \Pi_{\mathbf{U}_j^\perp} \boldsymbol{Q}_*(i,:,:) \Pi_{\mathbf{U}_j^\perp}\|_F^2 = Ln^2\beta(1 - \beta)(1 - \alpha^2)p^2$, $i, j = 1, 2$, $i \neq j$, so that $\kappa_H = 4\beta(1 - \beta)(1 - \alpha^2)/(1 + \alpha^2)$. We remark that in this example, κ_H does not depend on n or L.

4.2 Assumptions

In order to guarantee linear convergence of Algorithm 1 when it is applied to the true probability tensor $\mathbf{P}_*$, we make the following assumption:

(A1). The subspaces L_1 and L_2, defined in (14), have only trivial intersection at the origin.

While Assumption **(A1)** is somewhat complicated, it is actually not very restrictive. Specifically, the statement below provides two very simple sufficient conditions that guarantee Assumption **(A1)**. In particular, Assumption **(A1(b))** holds if each clustering pattern is not obtained by mixing other clustering patterns via combining or intersecting the clusters. For example, Assumption **(A1(b))** holds with high probability when the M clustering patterns are drawn uniformly at random.

Lemma 1. *Let at least one of the following conditions hold:*

(A1(a)). *For every $1 \le m \le M$, the sets of $(M-1)$ matrices*

$$\left\{\Pi_{\mathbf{U}_m^\perp} \boldsymbol{Q}_*(m',:,:)\Pi_{\mathbf{U}_m^\perp}, m' \ne m, 1 \le m' \le M\right\}$$

are linearly independent. That is, the vectorized versions of those matrices, a matrix of size $n^2 \times M$ with the m-th column given by $vec(\Pi_{\mathbf{U}_m^\perp} \boldsymbol{Q}_*(m',:,:)\Pi_{\mathbf{U}_m^\perp})$, has rank $(M-1)$.

(A1(b)). *For all $1 \le m \le M$, one has*

$$Span(\boldsymbol{\Theta}_m) \notin \bigoplus_{1 \le m' \le M, m' \ne m} Span(\boldsymbol{\Theta}_{m'}),$$

where $\boldsymbol{\Theta}_m \in \mathbb{R}^{n \times K_m}$ is the membership matrix for the m-th network as defined in (1) *and $\oplus$ stands for the direct sum of subspaces.*

The proof that these conditions are sufficient is presented in Section 6.6. While conditions in Lemma 1 are sufficient, they are not necessary. A more detailed discussion of assumption **(A1)** is deferred to Section 4.5.

In addition, we impose few other natural assumptions as follows:

(A2). There exist absolute constants $c_0 > 0$ such that $K_{\max} M \le c_0 \dot{K}$, where $\dot{K}$ is defined in (4).

(A3). The layers in the network, as well as local communities in each network are balanced, i.e., there exist absolute constants c_1, c_2, c_3, c_4 such that

$$c_1 \frac{L}{M} \le L_m \le c_2 \frac{L}{M}, \quad c_3 \frac{n}{K_m} \le |G_{m,k}| \le c_4 \frac{n}{K_m} \quad \text{for any } \ 1 \le m \le M, \ 1 \le k \le K_m, \tag{16}$$

where $|G_{m,k}|$ is the size of the k-th community in cluster m.

(A4). There exist matrices $\mathbf{B}_m^0 \in \mathbb{R}^{K_m \times K_m}$, $m = 1, \ldots, M$, such that $\mathbf{B}_m = p_{\max}\mathbf{B}_m^0$, where $p_{\max} \in (0, 1]$ controls the overall network sparsity and the matrices $\mathbf{B}_m^0$ are such that, for all $m = 1, ..., M$, there exists some absolute constants $b_1 > 0$ and $b_2 > 0$ such that

$$\sigma_{K_m}(\mathbf{B}_m^0) \geq b_1, \quad \|\mathbf{B}_m^0\|_F \geq b_2\, K_m, \quad \|\mathbf{B}_m^0\|_\infty \leq 1. \tag{17}$$

We remark that the first two inequalities of (17) imply that the magnitude of $\mathbf{B}_m^0$ is bounded from below, while the last inequality of (17) ensures that the magnitude of $\mathbf{B}_m^0$ is bounded from above. In addition, $\|\mathbf{B}_m^0\|_\infty \geq K_m^{-1}\|\mathbf{B}_m^0\|_F$, and (17) guarantees that $\|\mathbf{B}_m^0\|_\infty$ is bounded from both below and above: $b_2 \leq \|\mathbf{B}_m^0\|_\infty \leq 1$.

Illustrative example (continuation) . In our example, Assumption **(A1)** holds if the communities in the layers of type 1 are different from the communities in the layers of type 2, which is true if if $\beta > 0$. Assumption **(A2)** holds with $c_0 = 1$. Assumption **(A3)** holds with $c_1 = c_2 = c_3 = c_4 = 1$. Assumption **(A4)** holds with $p_{\max} = p$, $b_1 = 1 - \alpha$ and $b_2 = \sqrt{(1+\alpha^2)/2}$.

4.3 Convergence of ALMA

Denote by κ_0 the condition number of matrix $\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* \in \mathbb{R}^{M \times M}$:

$$\kappa_0 = \frac{\sigma_1(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)}{\sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)}.$$

Let

$$\beta_{n,L} = \log^2(n+L)\sqrt{M^3\,\kappa_0^5}\left(\sqrt{\frac{1}{p_{\max}n^2}} + \frac{\dot{K}^2}{p_{\max}n\min(n,L)}\right) \tag{18}$$

Then, the following proposition shows that, with a good initialization that satisfies (20), as the number of iterations tends to infinity, Algorithm 1 converges to a fixed point that is close to $\boldsymbol{Q}_*$ and $\mathbf{W}_*$.

Proposition 1 requires that κ_H is bounded away from 1. We remark that by definition, $\kappa_H \leq 1$. In many models such as the illustrative example at the end of Section 4.1, κ_H is bounded away from 1 and it does not depend on n or L. However, Proposition 1 still implicitly requires a lower bound of n and L through (19) since $p_{\max}$ can not be larger than 1.

Proposition 1. *[Theoretical guarantees of Algorithm 1.] Let Assumptions **(A1)-(A4)** hold and κ_H be bounded away from one, i.e., there exists a constant $c_{\kappa_H} < 1$ such that $\kappa_H \leq c_{\kappa_H}$. Let, for some positive absolute constant C_1*

$$p_{\max} \geq C_1 \max\left(\frac{\kappa_0^{12}\log^6(n+L)M^3\dot{K}^3}{(1-\kappa_H)^4\, n\, \min(n,L)}, \frac{\log(n+L)}{n+L}\right) \tag{19}$$

Let the initialization $\mathbf{W}^{(1)}$ have an estimation error bounded above by some positive function h of $M, \dot{K}$ and κ_0:

$$\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F \leq h(M, \dot{K}, \kappa_0). \tag{20}$$

Then, for some absolute constant $C_2 > 0$ *and* $\text{iter} \geq 1$*, with probability* $1-o(1)$ *as* $n, L \to \infty$*, one has*

$$\|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*\|_F \leq \frac{1+\kappa_H}{2}\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_F + C_2\,\beta_{n,L} \tag{21}$$

and, therefore,

$$\|\widehat{\mathbf{W}} - \mathbf{W}_*\|_F \leq \frac{2\epsilon_{n,L}}{1-\kappa_H} + \frac{2C_2}{1-\kappa_H}\,\beta_{n,L}. \tag{22}$$

If $\epsilon_{n,L} \geq 6\,C_2\,(1-\kappa_H)^{-1}\,\beta_{n,L}$*, then, under our stopping criterion, Algorithm 1 terminates within at most* $T_{n,L}$ *iterations, where*

$$T_{n,L} = \log\left(\frac{6\,\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F}{\epsilon_{n,L}}\right) \Big/ \log\left(\frac{2}{1+\kappa_H}\right),$$

In addition, for some absolute constant $C_3 > 0$ *and any* $m = 1, \ldots, M$*, with probability* $1 - o(1)$ *as* $n, L \to \infty$*, one has*

$$\left\|[\boldsymbol{Q}^{(\text{iter})} - \boldsymbol{Q}_*](m,:,:)\right\| \leq 2p_{\max}\,n\,\sqrt{L}\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_F + C_3\,\sqrt{p_{\max}(n+L)}\,\log(n+L), \tag{23}$$

and (23) *holds when* $(\boldsymbol{Q}^{(\text{iter})}, \mathbf{W}^{(\text{iter})}$ *are replaced with* $\widehat{\boldsymbol{Q}}, \widehat{\mathbf{W}}$*.*

Sketch of the proof of Proposition 1. The proof of Proposition 1 is deferred to Section 6.2. Below we provide some insight into how this theorem can be proved. The proof of the main inequality (21) in Proposition 1 can be divided into four steps.

The first step establishes a deterministic bound on $\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|$ for any given fixed $\boldsymbol{A}$. The second and third steps establish probabilistic bounds for a random tensor $\boldsymbol{A}$ under the probabilistic model in Section 2. Finally, the fourth step simplifies this probabilistic bound using Assumptions **(A2)-(A4)**.

As for the proof of (23), it is based on the following chain of inequalities

$$\begin{aligned}\|[\hat{\boldsymbol{Q}} - \boldsymbol{Q}_*](m,:,:)\| &= \|\Pi_{K_m}[\boldsymbol{A} \times_1 \hat{\mathbf{W}}] - \boldsymbol{Q}_*(m,:,:)\| \leq 2\|\boldsymbol{A} \times_1 \hat{\mathbf{W}} - \boldsymbol{Q}_*(m,:,:)\| \\ &\leq 2\|\boldsymbol{P}_* \times_1 (\hat{\mathbf{W}} - \mathbf{W}_*)\| + 2\|\boldsymbol{\Delta} \times_1 \hat{\mathbf{W}}(:,m)\| \leq 2\|\boldsymbol{Q}_*\|\|\hat{\mathbf{W}} - \mathbf{W}_*\| + 2\|\boldsymbol{\Delta}\| \\ &\leq 2\|\boldsymbol{Q}_*\|\|\hat{\mathbf{W}} - \mathbf{W}_*\|_F + 2\|\boldsymbol{\Delta}\|, \end{aligned} \tag{24}$$

where $\boldsymbol{\Delta} = \boldsymbol{A} - \boldsymbol{P}_*$, and the factor 2 in the first inequality follows from

$$\|\boldsymbol{A} \times_1 \hat{\mathbf{W}} - \Pi_{K_m}[\boldsymbol{A} \times_1 \hat{\mathbf{W}}]\| \leq \|\boldsymbol{A} \times_1 \hat{\mathbf{W}} - \boldsymbol{Q}_*(m,:,:)\|.$$

Inequality (23) is then obtained by combining (24) with the upper bounds on $\|\boldsymbol{\Delta}\|$ in Lemma 3 (see Appendix) and the fact that $\|\boldsymbol{Q}_*\| \leq \|\boldsymbol{Q}_*\|_F \leq p_{\max} n\sqrt{L}$.

Remark 1. Permutations. We remark that instead of (20), $\mathbf{W}^{(1)}$ can be close to $\mathbf{W}_*$ up to a permutation of columns, and this permutation has no impact on the between-layer and within-layer clustering results, as the outputs of Algorithm 1, $\widehat{\mathbf{W}}$ and $\widehat{\boldsymbol{Q}}$, would also be close to $\mathbf{W}_*$ and $\boldsymbol{Q}_*$ up to permutations of columns and layers, respectively.

Remark 2. Initialization. Proposition 1 requires initialization $\mathbf{W}^{(1)}$ that satisfies condition (20). If M, $\dot{K}$ and κ_0 are uniformly bounded above for any values of n and L, then (20) is satisfied by any matrix $\mathbf{W}^{(1)}$ since for any $L \times M$ matrix $\mathbf{W}$ such that $\mathbf{W}^T\mathbf{W} = I_M$ one has $\|\mathbf{W}\|_F^2 = M$, i.e. condition (20) holds with $h(M, \dot{K}, \kappa_0) = 2\sqrt{M}$.

The theoretical guarantees on the initialization in Algorithm 2 are given by the statement below. The proof of Proposition 2 is deferred to Section 6.3.

Proposition 2. *[Theoretical guarantees of Algorithm 2.] Assume that* $p_{\max} \geq \frac{\log(n^2+L)}{\min(n^2,L)}$. *Then for any* $r > 0$, *there exists a constant* $C_5 > 0$ *depending only on* r, ϵ, *such that, with probability at least* $1 - \min(n^2, L)^{-r}$,

$$\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F \leq C_5 \sqrt{\frac{c_2 M(n^2+L)}{c_1 \kappa_0 p_{\max} n^2 L}}. \tag{25}$$

Let us discuss the assumptions in Proposition 2. The condition $p_{\max} \geq \frac{\log(n^2+L)}{\min(n^2,L)}$ holds if (19) is true. In addition, under the assumptions that $M, \dot{K}, \kappa_0$ are uniformly bounded above, as $n, L \to \infty$, Proposition 2 implies that $\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F = O([p_{\max} \min(n^2, L)]^{-1/2})$, while (20) requires that $\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F = O(1)$. As a result, the output of Algorithm 2 satisfies the initialization condition (20) in Proposition 1. Combining Proposition 1 and 2, we obtain the following result that guarantees success of Algorithms 1 and 2

Theorem 1. *Let Assumptions* ***(A1)-(A4)*** *hold and* κ_H *be uniformly bounded away from one for any* n *and* L *large enough. Let, for some positive absolute constant* C_1

$$p_{\max} \geq C_1 \max\left(\frac{\kappa_0^{12} \log^6(n+L) M^3 \dot{K}^3}{(1-\kappa_H)^4 \, n \, \min(n, L)}, \frac{\log(n+L)}{n+L}, \frac{C_\epsilon c_0 M(n^2+L)}{h^2(M, \dot{K}, \kappa_0) \kappa_0 n^2 L} \right). \tag{26}$$

Then, with probability $1 - o(1)$ *as* $n, L \to \infty$, *for some absolute positive constants* C_2 *and* C_3, *and any* $m = 1, \ldots, M$, *one has*

$$\|\widehat{\mathbf{W}} - \mathbf{W}_*\|_F \leq \frac{2C_2}{1-\kappa_H} \, \beta_{n,L} \tag{27}$$

$$\left\| [\widehat{\boldsymbol{Q}} - \boldsymbol{Q}_*](m, :, :) \right\| \leq 2 p_{\max} \, n \sqrt{L} \|\widehat{\mathbf{W}} - \mathbf{W}_*\|_F + C_3 \sqrt{p_{\max}(n+L)} \, \log(n+L). \tag{28}$$

Illustrative example (continuation) . In our example, one has

$$(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*) = \frac{(1+\alpha^2)(np)^2}{2} \begin{pmatrix} 1 & 1-h \\ 1-h & 1 \end{pmatrix}$$

where $h = 2\,(1+\alpha^2)^{-1}\,(1+\alpha)\beta(1-\beta)$. It is easy to check that $\kappa_0 = h/(2-h)$, so it is independent of n and L. Since

$$\beta_{n,L} = 2\sqrt{2}\,\kappa_0^{5/2} \, \log^2(n+L) \left(\sqrt{\frac{1}{p_{\max} n^2}} + \frac{4}{p_{\max} n \min(n, L)} \right),$$

Theorem 1 implies that, whenever

$$p_{\max} \geq C \max\left(\frac{\log^6(n+L)}{n\min(n,L)}, \frac{\log(n+L)}{n+L}\right),$$

one has $\|\widehat{\mathbf{W}} - \mathbf{W}_*\|_F \leq C\beta_{n,L} \to 0$ as $n, L \to \infty$.

4.4 Consistency of between-layer and within-layer clustering

This section studies misclassification error rates of network clustering and local community detection. The misclassification error rates are measured by the Hamming distance between clustering partitions. Since the clustering is unique only up to a permutation of clusters, denote the set of permutation functions of $[r] = \{1, \cdots, r\}$ by $\aleph(r)$.

Given the true partition of network layers $[L] = \{1, \cdots, L\} = \cup_{m=1}^{M} \mathcal{S}_m$ and the estimated partition $[L] = \cup_{m=1}^{M} \hat{\mathcal{S}}_m$, the misclassification error rate of between-layer clustering is given by

$$R_{BL} = L^{-1} \min_{\tau \in \aleph(M)} \sum_{m=1}^{M} |\mathcal{S}_m \setminus \hat{\mathcal{S}}_{\tau(m)}|. \tag{29}$$

The misclassification rate of within-layer clustering is defined similarly: given the true partition of vertices $[n] = \{1, \cdots, n\} = \cup_{k=1}^{K_m} G_{m,k}$ and the estimated partition $\cup_{k=1}^{K_m} \hat{G}_{m,k}$, the misclassification error rate of within-layer clustering for the m-th group of layers is given by

$$R_{WL}(m) = n^{-1} \min_{\tau \in \aleph(K_m)} \sum_{k=1}^{K_m} |G_{m,k} \setminus \hat{G}_{m,\tau(k)}|. \tag{30}$$

The derivations of both misclassification rates are based on the upper bound (23) and Lemma C.1 of Lei and Lin (2020). In addition, the analysis of within-layer clustering also applies the Davis-Kahan theorem. Our results on the misclassification errors are as follows, with the proof deferred to Section 6.4.

Theorem 2. *(a) [Between-layer clustering error] For an $(1+\epsilon)$ approximate solution of the K-means problem, with probability $1 - o(1)$ as $n, L \to \infty$, the between-layer clustering error is bounded by*

$$R_{BL} \leq C_\epsilon \frac{\beta_{n,L}^2}{M^2(1-\kappa_H)^2}. \tag{31}$$

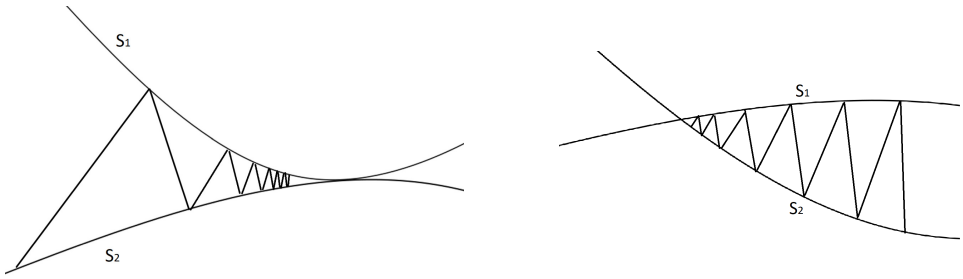
for some constant C_ϵ depending on ϵ.

(b) [Within-layer clustering] With probability $1 - o(1)$ as $n, L \to \infty$, the within-layer clustering error of the m-th type of network is bounded by

$$R_{WL}(m) \leq C_\epsilon K_{\max}\left(\frac{\beta_{n,L}^2}{(1-\kappa_H)^2} + \frac{\log^2(n+L)}{n\,L\,p_{\max}}\right), \quad m = 1, \ldots, M. \tag{32}$$

Here, κ_H and $\beta_{n,L}$ are defined in, respectively, (15) *and* (18).

Figure 1: Examples of convergence rates of the alternating projection algorithm. Left: $\mathcal{S}_1$ and $\mathcal{S}_2$ are "tangent" to each other and their tangent planes have nontrivial intersections. Right: the tangent planes of $\mathcal{S}_1$ and $\mathcal{S}_2$ only have a trivial intersection.



Illustrative example (continuation) . In our example, Theorem 2 can be simplified to

$$R_{BL} \leq O\left(\frac{\log^4(n+L)}{n^2\, p_{\max}} + \frac{\log^4(n+L)}{n^2[\min(n,L)\, p_{\max}]^2}\right), \quad R_{WL} = O\left(\frac{\log^4(n+L)}{n\, \min(n,L)\, p_{\max}}\right).$$

It shows that clustering is consistent as long as $p_{\max} \geq C\,[n \min(n,L)]^{-1}\ \log^4(n+L)$.

4.5 Discussion of theoretical guarantees

This section shows that Assumption **(A1)** is not restrictive, and is usually satisfied in practice. In Lemma 1, we have already provided sufficient conditions that guarantee validity of Assumption **(A1)**. Below, we continue the discussion of this assumption.

The fact that Assumption **(A1)** is not restrictive can also be inferred from counting the dimensions of L_1 and L_2. Since a symmetric matrix $\boldsymbol{Q}(m,:,:)$ has $n(n-1)/2$ degrees of freedom, the set

$$\{\boldsymbol{Q}(m,:,:) : \Pi_{\mathbf{U}_m^\perp} \boldsymbol{Q}_(m,:,:)\Pi_{\mathbf{U}_m^\perp} = \mathbf{0}\}$$

has $n(n-1)/2 - (n-K_m)(n-K_m-1)/2 = K_m n - K_m(K_m+1)/2$ degrees of freedom. Summing those up for $1 \leq m \leq M$, obtain that the dimension of L_1 is $\sum_{m=1}^{M}(K_m n - K_m(K_m+1)/2)$. Since the set Skew_M has $M(M-1)/2$ degrees of freedom, L_2 has a dimension of $M(M-1)/2$. Since random subspaces of dimensions d_1 and d_2 in $\mathbb{R}^D$ do not intersect if $d_1 + d_2 \leq D$, that is, if

$$\frac{M(M-1)}{2} + \sum_{m=1}^{M}(K_m n - \frac{K_m(K_m+1)}{2}) \leq M\, n^2,$$

(which holds when n is large), then **(A1(b))** should usually hold.

We also remark that Assumption **(A1)** is slightly more restrictive than the local uniqueness of the solution to the problem (8) in the noiseless scenario, which only requires that $\mathcal{S}_1$ and $\mathcal{S}_2$ intersect only at $\boldsymbol{Q}_*$. However, our goal is to prove the linear convergence of Algorithm 1, and, as it is shown in Figure 1 below, the convergence rate of the alternating

method in Algorithm 1 would be slow and nonlinear if $\mathcal{S}_1$ and $\mathcal{S}_2$ are "tangent" to each other and their tangent planes have nontrivial intersections. On the other hand, the convergence rate is linear if the tangent planes to $\mathcal{S}_1$ and $\mathcal{S}_2$ have only a trivial intersection.

We should mention that Assumption **(A1)** fails in the case of the "checker board" model of Chi et al. (2020), where all networks have the same community structures. As we have indicated, we are not interested in carrying out the inference in this case. However, we remark that the ALMA still succeeds empirically in the checker board model, even though the Assumption **(A1)** is violated.

Remark 3. Uniqueness of the solution. Similar to many clustering problems, the solution of the optimization problem (10) is unique only up to permutations of clusters. The non-uniqueness due to permutation of clusters, however, does not cause difficulty for Algorithm 1. Hence, Proposition 1 still applies: if the initialization $\mathbf{W}^{(1)}$ is reasonably close to $\mathbf{W}_*$, then Algorithm 1 converges to the true solution.

5. Comparison with existing results

To the best of our knowledge, the only paper that studied the model considered in this paper is Jing et al. (2021), where the authors introduced algorithm TWIST, based on regularized tensor decomposition. In this section, we provide theoretical and numerical comparisons with their results.

5.1 Description of TWIST

While Jing et al. (2021) consider the model described in this paper, their methodology and their assumptions are somewhat different. Specifically, TWIST iterates Tucker decomposition with regularization step on the observation tensor $\boldsymbol{A}$ to obtain a low-rank approximation of $\boldsymbol{A}$, where the intention of the regularization is to dampen the stochastic errors. The Tucker structure of the approximation is used to cluster the nodes and the layers. Jing et al. (2021) start with compiling a collection of all clustering matrices $\boldsymbol{\Theta}_m$, $m = 1, \dots, M$, in (1) into one matrix $\boldsymbol{\Theta} \in \mathbb{R}^{n \times \dot{K}}$ defined as $\boldsymbol{\Theta} = [\boldsymbol{\Theta}_1, \cdots, \boldsymbol{\Theta}_M]$. With this notation, they obtain the Tucker decomposition of the true probability tensor $\mathbf{P}_*$ as

$$\mathbf{P}_* = \boldsymbol{B} \times_2 \boldsymbol{\Theta} \times_3 \boldsymbol{\Theta} \times_1 \mathbf{Z}, \quad \mathbf{C} \in \mathbb{R}^{\dot{K} \times \dot{K} \times M}, \tag{33}$$

where $\mathbf{Z} \in \{0,1\}^{L \times M}$ is the clustering matrix of layers such that $\mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1/2} = \mathbf{W}_*$ and $\boldsymbol{B}$ is defined as

$$\boldsymbol{B}(:,:,m) = \mathrm{diag}(0, \cdots, 0, \mathbf{B}_m, 0, \cdots, 0), \quad m = 1, \dots, M. \tag{34}$$

Furthermore, they obtain the SVD $\boldsymbol{\Theta} = \bar{\mathbf{U}}\bar{\mathbf{D}}\bar{\mathbf{R}}$ of $\boldsymbol{\Theta}$ where matrices $\bar{\mathbf{U}} \in \mathbb{R}^{n \times r}$ and $\bar{\mathbf{R}} \in \mathbb{R}^{\dot{K} \times \times r}$ have orthonormal columns, r is the rank of $\boldsymbol{\Theta}$ and $\bar{\mathbf{D}}$ is the diagonal matrix of nonzero singular values. The objective of the technique is to recover matrix $\mathbf{W}_*$ as well as $\bar{\mathbf{U}}$.

The TWIST algorithm is based on iterative updates of matrices $\widehat{\mathbf{U}}^{(iter)}$ and $\widehat{\mathbf{W}}^{(iter)}$. Specifically, given $\widehat{\mathbf{U}}^{(iter)}$ and $\widehat{\mathbf{W}}^{(iter)}$, TWIST sets

$$\tilde{\mathbf{U}}^{(iter)} = \mathcal{P}_{\delta_1,r}(\widehat{\mathbf{U}}^{(iter)}), \quad \tilde{\mathbf{W}}^{(iter)} = \mathcal{P}_{\delta_2,M}(\widehat{\mathbf{W}}^{(iter)}),$$

where, for matrix $\mathbf{V}$, any $\delta > 0$ and positive integer s, one has

$$\mathcal{P}_{\delta,s}(\mathbf{V}) = SVD_s(\mathbf{V}_*) \quad \text{with} \quad \mathbf{V}_*(i,:) = \mathbf{V}(i,:)\,\frac{\min(\delta, \|\mathbf{V}(i,:)\|)}{\|\mathbf{V}(i,:)\|}, \tag{35}$$

and SVD_s returns the top s left singular vectors. Subsequently, the new iterations $\widehat{\mathbf{U}}^{(iter+1)}$ and $\widehat{\mathbf{W}}^{(iter+1)}$ are obtained as, respectively, the top r left singular vectors of $\mathcal{M}_3(\boldsymbol{A} \times_1 (\tilde{\mathbf{U}}^{(iter)})^T \times_2 (\tilde{\mathbf{W}}^{(iter)})^T$, and the top M left singular vectors of $\mathcal{M}_1(\boldsymbol{A} \times_2 (\tilde{\mathbf{U}}^{(iter)})^T \times_3 (\tilde{\mathbf{U}}^{(iter)})^T$. The process is carried out till the number of iterations reaches the pre-specified value $\text{iter}_{\max}$.

5.2 Theoretical comparisons of TWIST and ALMA

Note that TWIST aims at revealing both the global and the local memberships of nodes, together with the memberships of layers. Since Algorithm 1 (ALMA) does not deal with the concept of global communities, in the context of this paper, we use the terms "within–layer" and "between–layer" clustering to stand for the local memberships and memberships of layers, respectively. The global communities, defined in Jing et al. (2021), are related to, but not identical, to the persistence of the local ones in all layers.

Since Jing et al. (2021) apply a different technique, their assumptions, theoretical analysis, and final results differ from ours. We start with the comparison of the assumptions.

Specifically, Jing et al. (2021) impose the following conditions:

1. Denote $\sigma_{\min}(\boldsymbol{B}) = \min\{\sigma_{\min}(\mathcal{M}_1(\boldsymbol{B})), \sigma_{\min}(\mathcal{M}_2(\boldsymbol{B})), \sigma_{\min}(\mathcal{M}_3(\boldsymbol{B}))\}$. Then, for the tensor $\boldsymbol{B}$, defined in (34), one has $\sigma_{\min}(\boldsymbol{B}) \geq \tilde{c}_1 p_{\max}$. Note that

$$\sigma_{\min}(\boldsymbol{B}) \leq \sigma_{\min}(\mathcal{M}_1(\boldsymbol{B})) = \min_{m=1,\cdots,M} \sigma_{K_m}(\mathbf{B}_m) = p_{\max} \min_{m=1,\cdots,M} \sigma_{K_m}(\mathbf{B}_m^0).$$

   Hence, the assumption on $\sigma_{\min}(\mathcal{M}_1(\boldsymbol{B}))$ is equivalent to the first part of Assumption **(A4)** in (17). While the assumptions on $\sigma_{\min}(\mathcal{M}_j(\boldsymbol{B}))$ for $j = 2, 3$, are not directly comparable to the second part of Assumption **(A4)** in (17), both serve a similar purpose that $\boldsymbol{B}$ is not too small.

2. Matrix $\boldsymbol{\Theta}$ in (33), for some $\tilde{\kappa}_0 < \infty$, is assumed to satisfy the condition $\sigma_{\max}(\boldsymbol{\Theta}) \leq \tilde{\kappa}_0 \sigma_r(\boldsymbol{\Theta})$ where $r = \text{rank}(\boldsymbol{\Theta})$. This assumption implicitly implies our assumption **(A2)** and the "balanced local community" assumption in **(A3)**, i.e., the second condition in (16).

3. Layer sizes and community sizes in the layers are assumed to be similar. This is equivalent to the assumption **(A3)**.

4. Network sparsity assumption $L\,n\,p_{\max} \geq C(\dot{K}+\tilde{\kappa}_0^2\,r^2\,\log^2 n)M^{-1}\tilde{\kappa}_0^6\,r^2\,\log^2(r\tilde{\kappa}_0)\,\log^2 n$. In comparison, we have a similar assumption in (19).

5. Theoretical analysis of TWIST is carried out under the condition that $L \leq n$. We do not impose this assumption.

Under these assumptions, the theoretical upper bound for the error rate of the between-layer clustering of TWIST is

$$R_{BL}^{(TWIST)} = O\left(\tilde{\kappa}_0^4 \frac{r^2 \log n}{M\,L\,n\,p_{\max}}\right).$$

Under the additional assumption $\sqrt{L}np_{\max} \geq CM^{-1}\tilde{\kappa}_0^5 r^{5/2}\log(r\tilde{\kappa}_0)\log^{5/2} n$, the theoretical upper bound for the error rate of within-layer clustering for the m-th type of network is

$$R_{WL}^{(TWIST)}(m) = O\left(\frac{\tilde{\kappa}_0^4 K_m^2 \log n}{M\,L\,n\,p_{\max}}\right).$$

We remark that the comparison with κ_0 and $\tilde{\kappa}_0$ is not straightforward, as they are defined very differently (even though both measure how "well-conditioned" the model is). In order to compare the clustering errors of ALMA and TWIST, we assume that M, K_m, κ_0 and κ_H are uniformly bounded by constants independent of n and L, so that, as a result, the same is true for $\dot{K}, r$ and $\tilde{\kappa}_0$. Then the clustering error rates are more comparable since they depend only on n, L, and $p_{\max}$. Specifically, the theoretical upper bound for the error rate of the between-layer clustering are

$$R_{BL}^{(ALMA)} = O\left(\frac{\log^4(n+L)}{n^2\,p_{\max}} + \frac{\log^4(n+L)}{n^2[\min(n,L)\,p_{\max}]^2}\right), \quad R_{BL}^{(TWIST)} = O\left(\frac{\log n}{L\,n\,p_{\max}}\right). \tag{36}$$

The theoretical upper bound for the error rate of the within-layer clustering for the m-th type of network are

$$R_{WL}^{(ALMA)}(m) = O\left(\frac{\log^4(n+L)}{n\,\min(n,L)\,p_{\max}}\right), \quad R_{WL}^{(TWIST)}(m) = O\left(\frac{\log n}{L\,n\,p_{\max}}\right). \tag{37}$$

Recall that, for both the between-layer clustering and the within-layer clustering, the error rates of TWIST are derived under the assumption that $L \leq n$.

In comparison, the error rates of the between-layer clustering of ALMA are better in two aspects. First, they hold in the case of $L > n$. Second, since both methods require that the quantity $n\,L\,p_{\max}$ grows with n and L, it is easy to see that, up to the logarithmic factors,

$$R_{BL}^{(ALMA)} = o\left(R_{BL}^{(TWIST)}\right) \quad \text{if} \quad n \to \infty,\ L/n \to 0.$$

Also, up to the logarithmic factors, the within-layer clustering error rates are equivalent.

However, Jing et al. (2021) do not have anything similar to Assumption **(A1)** imposed in the present paper. This assumption is due to the fact that Proposition 1 attempts to achieve something more than clustering: it aims at recovering $\boldsymbol{Q}_*$ and $\mathbf{W}_*$ directly.

In addition, we have somewhat stronger assumption on sparsity, requiring that $p_{\max} > O\Big(\frac{\log(L+n)}{n+L} + \frac{\log^6(n+L)}{n\min(n,L)}\Big)$, instead of $p_{\max} > O(\frac{\log n}{nL})$ in Jing et al. (2021). We suspect that the difference in the assumptions is due to the technicalities in our analysis, rather than the inherent drawbacks of our algorithm.

To get an idea of how tight the error bounds are, one can compare them with the existing results for SBM, which is the special case of the MMLSBM with $L = M = 1$. In particular, Lei and Rinaldo (2015) show that when $p_{\max} \geq C \log n/n$, then the spectral clustering could achieve a misclassification rate of $O\left(K/(n\, p_{\max})\right)$. On the other hand, it is known (see, e.g., Abbe (2018)) that, when $K = 2$, the threshold for the exact recovery of the membership is $p_{\max} = O(\log n/n)$ and the threshold for the weak recovery of the membership (i.e., better than random guess) is $p_{\max} = O(1/n)$. In this sense, the theoretical guarantees of both TWIST and ALMA recover the case $L = M = 1$ up to some logarithmic factors.

It would also be interesting to compare the computational cost of ALMA and TWIST. In ALMA, the steps $\boldsymbol{Q}^{(\text{iter}+1)} = \Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\text{iter})})$ and $\mathbf{W}^{(\text{iter}+1)} = \Pi_o(\boldsymbol{A} \times_{2,3} \boldsymbol{Q}^{(\text{iter}+1)})$ require, respectively, $O(LMn^2 + \dot{K}n^2)$ and $O(LMn^2 + LM^2)$ operations, so that, each iteration of ALMA requires $O(LMn^2 + LM^2 + \dot{K}n^2)$ operations. When L and n are large, the dominant term is $O(Ln^2M)$. In comparison, each iteration of TWIST has a computational cost of $O(Ln^2(r + M))$, which is larger than that of ALMA, since r is larger than M, due to $[\boldsymbol{\Theta}_1, \cdots, \boldsymbol{\Theta}_M] \in \mathbb{R}^{n \times \dot{K}}$, where $\dot{K} = \sum_{m=1}^{M} K_m > M$.

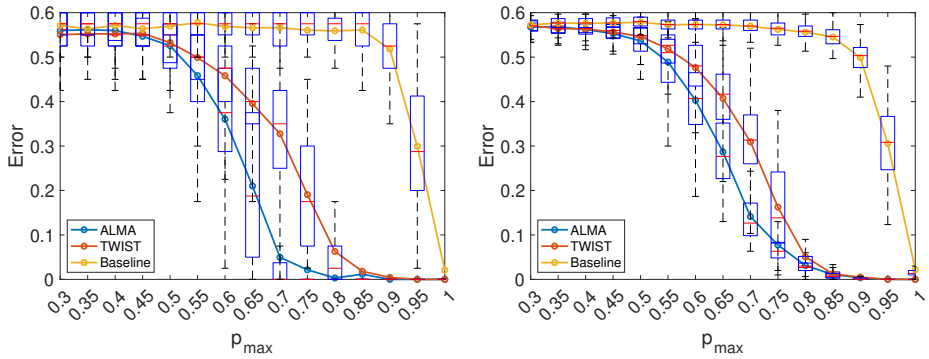
5.3 Numerical comparisons

As it is evident from the previous section, the theoretical comparison between ALMA and TWIST is very difficult due to the differences between assumptions. We also compare both algorithms with a simple baseline algorithm for the between-layer clustering. The baseline algorithm first applies spectral clustering to each of the L layers, obtaining initial within-layer clustering matrices $\widehat{\boldsymbol{\Theta}}^{(l)} \in \{0,1\}^{n \times K}$, $l = 1, ..., L$. After that, it generates estimated connectivity matrices $\widehat{\boldsymbol{\Theta}}^{(l)}(\widehat{\boldsymbol{\Theta}}^{(l)})^T \in \{0,1\}^{n \times n}$, where entries are equal to one if nodes belong to the same community and zero otherwise. Subsequently, it carries out spectral clustering of the vectorized versions of those connectivity matrices to partition the layers.

In order to test the performance of Algorithm 1 (ALMA) and subsequent within-layer clustering, and to provide a fair comparison of the clustering precisions with the TWIST technique of Jing et al. (2021) and the baseline algorithm, we carry out a limited simulation study with various choices of parameters $p_{\max}$, L and n. We use the misclassification rates as the measure of the performance of our algorithm. Specifically, we characterize the between layer clustering precision by (29). For the error of the within-layer clustering, we average the rates in (30) over the M layers and use

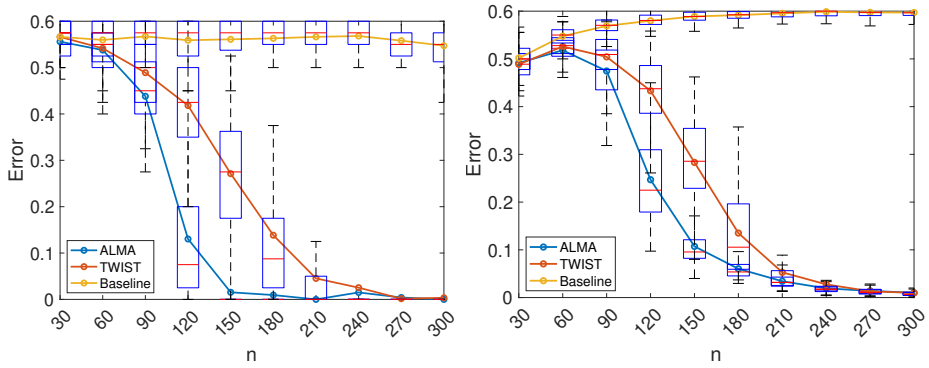
$$R_{WL} = M^{-1} \sum_{m=1}^{M} R_{WL}(m). \tag{38}$$

We choose M and fix $K_1 = \cdots = K_M = K$, so that in each cluster, network follows SBM with K communities. The underlying class for each layer, and the membership for each node in every class of layers are randomly sampled using the multinomial distributions with equal class probabilities $1/M$ for the layers of the networks, and $1/K$ for the nodes in each



(a) Between-layer clustering (b) Within-layer clustering

Figure 2: Simulation Scenario 1: $L = 40, n = 100, M = 3, K = 3, \alpha = 0.9$. The between-layer clustering errors and within-layer clustering errors are plotted versus p_{max}. The solid lines exhibit the average misclassification errors.

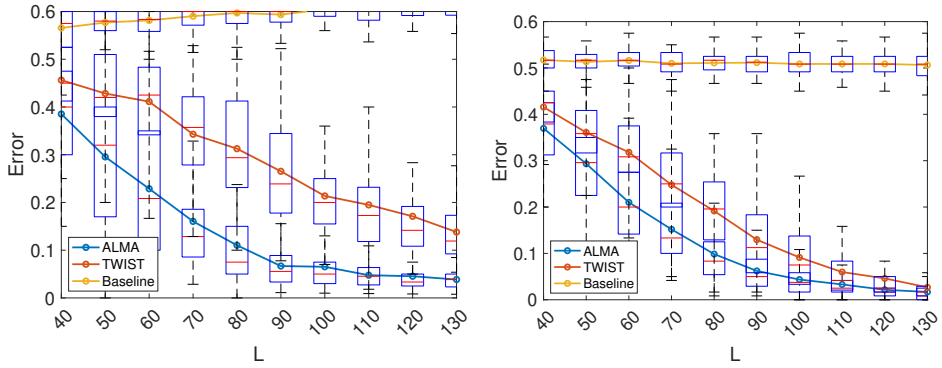


(a) Between-layer clustering (b) Within-layer clustering

Figure 3: Simulation Scenario 2: $L = 40, M = 3, K = 3, p_{max} = 0.6, \alpha = 0.9$. The between-layer clustering errors and within-layer clustering errors are plotted versus the number of vertices n. The solid lines exhibit the average misclassification errors.

of the layer clusters. In each of the layers, we use identical connectivity matrices $\mathbf{B}_m \equiv \mathbf{B}$ where the diagonal values are set to $p = p_{max}$ while the off-diagonal entries are equal to $q = \alpha p_{max}$ with $\alpha < 1$. The constant α controls the ratio of the probability of connection of a node outside its own community versus inside it. Consequently, the within layer clustering is easier when α is small and harder when it is large.

We investigate the performances of ALMA (Algorithm 1) and compare it with the performances of TWIST and the baseline method in four simulation scenarios. In our simulations, we set $M = 3$, $K = 3$ and $r = 7$, since $r = \text{rank}([\mathbf{\Theta}_1, \cdots, \mathbf{\Theta}_M]) \leq \sum_{m=1}^{M} K_m - (M - 1)$, with inequality occurring in degenerate settings. Since our approach does not



(a) Between-layer clustering (b) Within-layer clustering

Figure 4: Simulation Scenario 3: $n = 40, M = 3, K = 3, p_{max} = 0.5, \alpha = 0.8$. The between-layer clustering errors and within-layer clustering errors are plotted versus the number of layers L when $L > n$. The solid lines exhibit the average misclassification errors.

involve the concept of global membership, we only compare ALMA with TWIST in terms of "within–layer" and "between–layer clustering". Furthermore, we choose the stopping criterion $\|\mathbf{W}^{(\text{iter})} - \mathbf{W}^{(\text{iter}-1)}\| \leq 10^{-4}$ for both of ALMA and TWIST to make a fair comparison between the algorithms. Below we describe the simulation schemes.

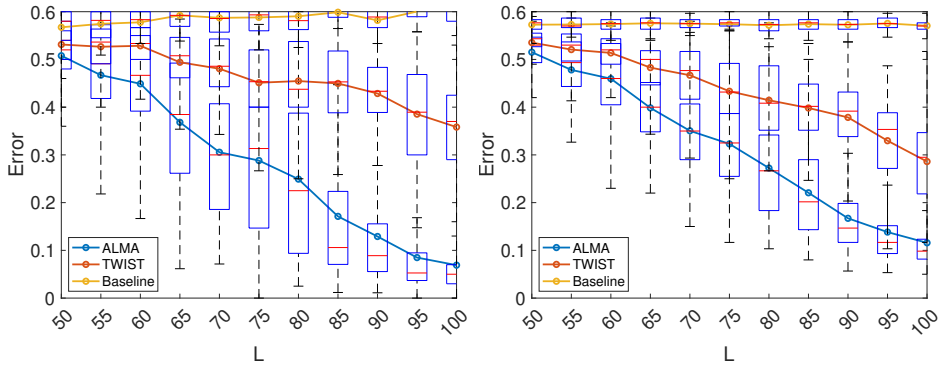
In Simulation 1, we investigate the effect of the network sparsity on the precision of the algorithms. For this purpose, we choose the number of vertices $n = 100$, the number of layers $L = 40$, the number of network clusters $M = 3$, the number of communities in each cluster of layers $K = 3$ and $\alpha = 0.9$. The variable p_{max}, which controls the overall network sparsity, varies from 0.3 to 1. Fig. 2 shows that both between-layer and within-layer clustering errors decrease as p_{max} is increasing.

In Simulation 2, the settings are the same as Simulation 1 except that $p_{max} = 0.6$ is fixed, and the number of vertices varies from 30 to 300. As n increases, the between-layer and within-layer clustering error rates decrease to zero, as predicted by Theorem 2.

In Simulations 3 and 4, we study the effect of the numbers of layers in the network, when $L > n$ and $L < n$, respectively. Specifically, in Simulation 3, we set $n = 40, M = 3, K = 3, \alpha = 0.8, p_{max} = 0.6$ and vary the number of layers L between 40 and 140. The settings in Simulation 4 are the same as Simulation 3, except $n = 100$ is larger, and L varies from 50 to 100.

For all algorithms, in each of the simulation scenarios, we report the between-layer and the within-layer clustering errors (29) and (30), respectively, averaged over 100 independent simulation runs. The results are summarized in Figures 2–5.

As it is evident from Figures 2–5, for all four scenarios, ALMA has smaller both the between-layer and the within-layer clustering errors, and both ALMA and TWIST outperform the baseline method. Note also that ALMA has better precision not only in the case of $L > n$, that violates the assumptions of TWIST, but also in the case of $L \leq n$.



(a) Between-layer clustering (b) Within-layer clustering

Figure 5: Simulation Scenario 4: $n = 100, M = 3, K = 3, p_{max} = 0.5, \alpha = 0.9$. The between-layer clustering errors and within-layer clustering errors are plotted versus the number of layers L when $L < n$. The solid lines exhibit the average misclassification errors.

5.4 Real Data Examples

In this section, we apply ALMA to two real data sets: worldwide food trading networks and airline flight networks. The two datasets have been studied previously in Jing et al. (2021) and Wu et al. (2016), respectively.

The Worldwide food trading networks data have been described in De Domenico et al. (2015), and is available at `https://www.fao.org/faostat/en/#data/TM`. The data contain trading volumes among 245 countries for more than 300 food items. To avoid sparsity and to be consistent with Jing et al. (2021), we consider trading relations between 98 countries/regions on 30 food products (such as tea, sugar, wine, etc.) in the year 2010. These data can be described by a multilayer network, in which layers represent different products, nodes are countries, and edges at each layer represent trading relationships of a specific food product among countries.

Similar to the procedure in Jing et al. (2021), we convert the original directed networks to undirected ones by ignoring the directions, and delete the links with weights less than 8. Thus, we draw an edge in a network if the trading volume in a food product is higher than 8 units. After this pre-processing step, we obtain a 30-layers network with 98 nodes, that can be represented by an adjacency tensor $\boldsymbol{A}$ of dimension $30 \times 98 \times 98$.

We also analyze the airline-airport network data set, available at `https://openflights.org/data.html#route`, which contains information on 67663 flight routes, including the name of the airline and the source and the destination airports. By including only the major airlines, with more than 200 routes, and major airports, with more than 40 routes, we obtain a flight network with 89 layers and 583 nodes, where layers represent different airlines, nodes are airports and edges at each layer represent the routes of a specific airline among the airports. Formally, these data can be represented by a multi-layer adjacency tensor $\boldsymbol{A}$ of dimension $89 \times 583 \times 583$.

We compare the performances of ALMA and TWIST and show that ALMA consistently outperforms TWIST in terms of the "goodness of fit" of the MMLSBM model. In this

paper, for a tensor $\mathcal{A} \in \mathbb{R}^{L\times n\times n}$, we measure the "goodness of fit" of the MMLSBM model, that consists of the between-layer clusters $\mathcal{S}_1 \cup \cdots \mathcal{S}_M$ and within-layer clusters $\{G_{m,k}\}$, $1 \le k \le K_m$, $1 \le m \le M$, by the mean of squared errors (MSE) defined as

$$\mathrm{MSE}(\{\mathcal{S}_m\}, \{G_{m,k}\}) = \frac{1}{n^2} \sum_{m=1}^{M} \sum_{i=1}^{K_m} \sum_{j=1}^{K_m} \|\mathcal{A}_{\mathcal{S}_m, G_{m,i}, G_{m,j}} - \overline{\mathcal{A}}_{\mathcal{S}_m, G_{m,i}, G_{m,j}}\|_F^2 \tag{39}$$

Here, $\mathcal{A}_{\mathcal{S}_m, G_{m,i}, G_{m,j}}$ represents the sub-tensor that consists of the layers of type m and the nodes in the i-th and j-th community of the network of type m, while $\overline{\mathcal{A}}_{\mathcal{S}_m, G_{m,i}, G_{m,j}}$ represents the average of this sub-tensor. As a result, $\mathrm{MSE}(\{\mathcal{S}_m\}, \{G_{m,k}\})$ represents the sum of squared errors, resulting from fitting $\mathcal{A}$ to the MMLSBM model $(\{\mathcal{S}_m\}, \{G_{m,k}\},\ 1 \le m \le M, 1 \le k \le K_m)$. We expect that a more precise between-layer and within-layer clustering gives a smaller MSE.

Table 2: Comparison of the "goodness of fit" of ALMA, TWIST and the baseline method in food trading network and airline-airport network for various choices of M and K. The table reports the values of the MSE in (39).

(M, K)	(2,2)	(2,3)	(2,4)	(2,6)	(2,9)	(3,3)	(3,5)	(4,2)	(4,3)	(4,5)
	Food Trading Network									
ALMA	4.8855	4.4273	4.3242	4.1368	4.0275	4.3805	4.1347	4.8730	4.3461	4.1129
TWIST	4.7574	4.5023	4.3857	4.2076	4.0962	4.4325	4.1389	4.8167	4.3940	4.1535
Baseline	5.0521	4.6449	4.5033	4.3805	4.3086	4.5731	4.3482	4.8844	4.5023	4.3170
	Airline-airport Network									
ALMA	0.1380	0.1251	0.1222	0.1169	0.1138	0.1238	0.1168	0.1377	0.1228	0.1162
TWIST	0.1344	0.1272	0.1239	0.1189	0.1157	0.1252	0.1169	0.1361	0.1242	0.1174
Baseline	0.1428	0.1312	0.1272	0.1238	0.1217	0.1292	0.1229	0.1380	0.1272	0.1220

For both data sets, we assume that all layer networks have the same number of communities, i.e., $K_1 = \cdots = K_M = K$, and apply ALMA and TWIST with various choices of M and K. Subsequently, we record the MSE for ALMA and TWIST for each of the choices of (M, K). Since the K-means algorithm is random and produces different outputs over each run, we record the average MSE over 100 runs. Results are presented in Table 2.

Table 2 establishes that, in most cases, ALMA outperforms TWIST in terms of the MSE defined in (39). It confirms the observation that ALMA utilizes the structure of the MMLSBM better than TWIST, and that both ALMA and TWIST outperform the baseline method. The only exception might be the setting with $K = 2$, where ALMA performs slightly worse or comparable to TWIST. We remark that, in the TWIST algorithm, the layer of the m-th type increases the rank of the estimated tensor by $K_m - 1$, while it requires an additional estimation of a matrix of rank K_m in the ALMA algorithm. This might explain a minor advantage of TWIST when $K = K_m = 2$.

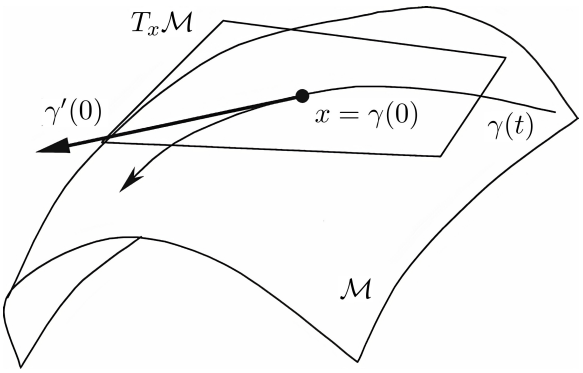


Figure 6: A visualization of the manifold $\mathcal{M}$, the curve γ in $\mathcal{M}$, the tangent vector $\gamma'(0)$, and the tangent space $T_{\mathbf{x}}(\mathcal{M})$.

Acknowledgements

Marianna Pensky was partially supported by National Science Foundation (NSF) grants DMS-1712977 and DMS-2014928. Teng Zhang was partially supported by National Science Foundation (NSF) grant CNS-1818500.

6. Appendix

6.1 Manifold and tangent space

The concepts of tangent vector and tangent space to an abstract manifold can be found in, e.g., Boothby and Boothby (2003) and Absil et al. (2009). When $\mathcal{M}$ is a manifold embedded in the Euclidean space $\mathbb{R}^p$, then a smooth function $\gamma : \mathbb{R} \to \mathcal{M}$ is called a *curve* in $\mathcal{M}$, and $\gamma'(0)$ is a *tangent vector* to the manifold $\mathcal{M}$ at the point $\gamma(0)$. The *tangent space* of $\mathcal{M}$ at x, denoted by $T_x\mathcal{M}$, is the set of all tangent vectors of $\mathcal{M}$ at $\mathbf{x}$, that is, $T_x\mathcal{M} = \{\gamma'(0) : \gamma : \mathbb{R} \to \mathcal{M}$ is a smooth function with $\gamma(0) = x\}$ (Absil and Oseledets, 2015). Intuitively, the tangent plane $T_x\mathcal{M}$ is the subspace that approximates the manifold $\mathcal{M}$ in a local neighborhood around x. For example, if $\mathcal{M}$ is the unit sphere $\{x = (x_1, x_2, x_3) : x_1^2 + x_2^2 + x_3^2 = 1\}$, then the tangent space at point $(\hat{x}_1, \hat{x}_2, \hat{x}_3)$ is given by $\{(x_1, x_2, x_3) : \hat{x}_1 x_1 + \hat{x}_2 x_2 + \hat{x}_3 x_3 = 0\}$. A visualization of the tangent space is given in Figure 6.

It remains to derive the tangent planes to the sets $\mathcal{S}_1$ and $\mathcal{S}_2$ at $\boldsymbol{Q}_*$ in (14). The expression for L_1 follows from the formula for the tangent planes for the manifold of low-rank matrices (Absil and Oseledets, 2015, equation (13)). Specifically, the explicit formula for the tangent plane to the manifold of rank K matrices at $\mathbf{Q}$ is given by the equation $\Pi_{\mathbf{U}^\perp} \boldsymbol{Q} \Pi_{\mathbf{U}^\perp} = \mathbf{0}$, where $\mathbf{U}$ is an orthogonal matrix that has the same column space as $\mathbf{Q}$. Now, the first formula in (14) is due to the fact that $\mathcal{S}_1$ is the product of M manifolds of low-rank matrices: $\mathcal{S}_1 = \otimes_{m=1}^{M} \mathcal{M}_m$, where

$$\mathcal{M}_m = \{\mathbf{X} \in \mathbb{R}^{n \times n} : \mathrm{rank}(\mathbf{X}) \leq K_m\}.$$

In order to obtain the second equation in (14), note that the tangent plane to the set of orthogonal matrices $\mathcal{M}_0$ at $\mathbf{I}$ is the set of skew-symmetric matrices: $T_{\mathbf{I}}\mathcal{M}_0 = \mathrm{Skew}_M$

(Edelman et al., 1998, Section 2.2.1). Now, the explicit formula for L_2 follows from the facts that $\mathcal{S}_2$ is obtained by multiplying $\boldsymbol{Q}_*$ with each element from $\mathcal{M}_0$, where $\mathcal{M}_0$ is the set of orthogonal matrices of size $M \times M$: $\mathcal{S}_2 = \{\boldsymbol{Q} = \boldsymbol{Q}_* \times_1 \mathbf{V} : \mathbf{V} \in \mathcal{M}_0\}$.

6.2 Proof of Proposition 1

The organization of this section follows from the sketch of the proof after Proposition 1 in four steps: the first step establishes a deterministic bound of $\mathbf{W}^{(\text{iter})} - \mathbf{W}_*$, the second and the third steps establish a probabilistic bound, and the fourth step simplifies the probabilistic bound using Assumptions **(A2)**-**(A4)**.

6.2.1 Step 1: Deterministic analysis of Algorithm 1

In this step, we aim to find a metric $\|\cdot\|_d$ on $\mathbb{R}^{L\times M}$ such that $\{\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d\}_{\text{iter}=1}^{\infty}$ should be monotonically decreasing approximately. While it is natural to consider the Frobenius norm, the previous analysis of the noiseless case in algorithm (13) does not support the monotonicity of $\{\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_F\}_{\text{iter}=1}^{\infty}$. Instead, it establishes the "approximate" monotonicity of $\|\boldsymbol{Q}_* \times_1 (\mathbf{V}^{(\text{iter})} - \mathbf{I})\|_F$ of

$$\|\boldsymbol{Q}_* \times_1 (\mathbf{V}^{(\text{iter}+1)} - \mathbf{I})\|_F \lesssim \kappa_H \|\boldsymbol{Q}_* \times_1 (\mathbf{V}^{(\text{iter})} - \mathbf{I})\|_F. \tag{40}$$

Recall that in the noiseless case, $\mathbf{W}^{(\text{iter})} = \mathbf{W}_*\mathbf{V}^{(\text{iter})}$, we expect that $\|\cdot\|_d$ should be defined such that when $\mathbf{V}$ is orthogonal and close to $\mathbf{I}$,

$$\|\mathbf{W}_*(\mathbf{V} - \mathbf{I})\|_d \approx \|\boldsymbol{Q}_* \times_1 (\mathbf{V} - \mathbf{I})\|_F. \tag{41}$$

Since the tangent plane of the set of orthogonal matrices at $\mathbf{I}$ is the set of skew-symmetric matrices (Gallier, 2001, Theorem 14.2.2), the tangent space of $\{\mathbf{W}_*\mathbf{V} : \mathbf{V}^T\mathbf{V} = \mathbf{I}\}$ at $\mathbf{V} = \mathbf{I}$ is $L_0 = \{\mathbf{W}_*\mathbf{Y} : \mathbf{Y} \in \text{Skew}_M\} \subseteq \mathbb{R}^{L\times M}$, (41) implies that for any $\mathbf{W}_*\mathbf{Y} \in L_0$, $\|\cdot\|_d$ should be defined such that $\|\mathbf{W}_*\mathbf{Y}\|_d = \lambda\|\boldsymbol{Q}_* \times_1 \mathbf{Y}\|_F = \lambda\|\boldsymbol{P}_* \times_1 \mathbf{W}_*\mathbf{Y}\|_F$ for some constant $\lambda > 0$. Combining this metric on L_0 and the standard Euclidean/Frobenius metric on the orthogonal subspace $L_0^{\perp}$, we define the metric $\|\cdot\|_d : \mathbb{R}^{L\times M} \to \mathbb{R}$ by

$$\|\mathbf{W}\|_d = \sqrt{(\lambda\|\boldsymbol{P}_* \times_1 P_{L_0}\mathbf{W}\|_F)^2 + \|P_{L_0^{\perp}}\mathbf{W}\|_F^2}, \tag{42}$$

Here, λ balances the weights from the two components, so that

$$\lambda = \left(\min_{\mathbf{Y}\in\text{Skew}_M, \|\mathbf{Y}\|_F=1} \|\boldsymbol{Q}_* \times_1 \mathbf{Y}\|_F\right)^{-1},$$

and the projection operators can be explicitly written as

$$\begin{aligned} P_{L_0}\mathbf{W} &= \mathbf{W}_*(\mathbf{W}_*^T\mathbf{W}^{(\text{iter})} - \mathbf{W}^{(\text{iter})\,T}\mathbf{W}_*)/2, \\ P_{L_0^{\perp}}\mathbf{W} &= \mathbf{W}_*(\mathbf{W}_*^T\mathbf{W}^{(\text{iter})} + \mathbf{W}^{(\text{iter})\,T}\mathbf{W}_*)/2 + (\mathbf{I} - \mathbf{W}_*\mathbf{W}_*^T)\mathbf{W}^{(\text{iter})}. \end{aligned}$$

By the definition of the metric $\|\cdot\|_d$ in (42), we have the following equivalence between $\|\cdot\|_d$ and $\|\cdot\|_F$:

$$\|\mathbf{W}\|_d \geq \sqrt{\|P_{L_0}\mathbf{W}\|_F^2 + \|P_{L_0^{\perp}}\mathbf{W}\|_F^2} = \|\mathbf{W}\|_F$$

and for $C_H = \max_{\mathbf{Y}\in \mathrm{Skew}_M, \|\mathbf{Y}\|_F=1} \|\boldsymbol{Q}_* \times_1 \mathbf{Y}\|_F / \min_{\mathbf{Y}\in \mathrm{Skew}_M, \|\mathbf{Y}\|_F=1} \|\boldsymbol{Q}_* \times_1 \mathbf{Y}\|_F$,

$$\|\mathbf{W}\|_d \le \sqrt{C_H^2 \|P_{L_0}\mathbf{W}\|_F^2 + C_H^2 \|P_{L_0^\perp}\mathbf{W}\|_F^2} = C_H \|\mathbf{W}\|_F.$$

Before stating our main result, we introduce two additional parameters:

$$\kappa_1 = \frac{p_{\max}^2 n^2 L}{\|\boldsymbol{Q}_*\|_F^2}, \quad \kappa_2 = \frac{\sqrt{\dot{K}\, p_{\max} L}\; n}{\min_{m=1,\ldots M} \sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))}.$$

Both parameters are greater than 1, and describe how "well-conditioned" $\boldsymbol{Q}_*$ is, and when $\boldsymbol{Q}_*$ is well-conditioned, then all parameters are close to 1. In particular, $\kappa_1 \ge 1$ because $\|\boldsymbol{Q}_*\|_F = \|\boldsymbol{P}_*\|_F$ and all elements of $\boldsymbol{P}_*$ are bounded by $p_{\max}$, and $\kappa_2 \ge 1$ because

$$\dot{K} \min_{m=1,\cdots,M} \sigma_{K_m}^2(\boldsymbol{Q}_*(m,:,:)) \le \sum_{m=1}^{M}\sum_{k=1}^{K_m} \sigma_k^2(\boldsymbol{Q}_*(m,:,:)) = \|\boldsymbol{Q}_*\|_F^2 \le p_{\max}^2 n^2 L.$$

When $\boldsymbol{Q}_*(m,:,:)$ is "degenerate" in the sense that $\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:)) \approx 0$, then κ_2 is large.

The main result in this step states that, if the noise $\boldsymbol{\Delta}$ is small and when Algorithm 1 is applied to the observed adjacency tensor $\boldsymbol{A}$ with a good initialization $\mathbf{W}^{(0)}$, the estimations are likely to improve over each iteration, and Algorithm 1 converges to $\mathbf{W}_*$ approximately. The statement is as follows, and its proof is rather complicated and deferred to Section 6.5.

Lemma 2 (Step 1: A deterministic result on Algorithm 1)**.** *For*

$$a_1 = 6\kappa_0 \frac{\sqrt{M}\left(\max_{m=1,\ldots M} \|\Pi_{T,K_m}(\boldsymbol{\Delta}) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\| + 2\max_{m=1,\ldots M} \|\Pi_{T,K_m}(\boldsymbol{Q}_*) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\|\right)}{\sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)}$$
$$+ \frac{192\kappa_0 \dot{K}\|\boldsymbol{\Delta}\|^2(\|\boldsymbol{Q}_*\| + \|\boldsymbol{\Delta}\|)}{\sqrt{M}\left(\min_{m=1,\ldots M} \sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))\right)\, \sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)},$$

$$a_2 = \frac{192\kappa_0 \dot{K}\|\boldsymbol{Q}_*\|^2(\|\boldsymbol{Q}_*\| + \|\boldsymbol{\Delta}\|)}{\sqrt{M}\min_{m=1,\ldots M} \sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))\; \sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)} + 6\kappa_0,$$

if $\|\boldsymbol{\Delta}\| \le \frac{1}{4}\min_{m=1,\ldots M} \sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))$,

$$\frac{2C_H a_1}{1-\kappa_H} \le \min\Big(\frac{1-\kappa_H}{2C_H(a_2 + 80\kappa_0^2 + 32\kappa_0^3)}, \frac{\min_{m=1,\cdots,M} \sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))}{4\|\boldsymbol{Q}_*\|}\Big) \tag{43}$$

and the initialization satisfies

$$\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_d \le \min\left(\frac{1-\kappa_H}{2C_H(a_2 + 80\kappa_0^2 + 32\kappa_0^3)}, \frac{\min_{m=1,\cdots,M} \sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))}{4\|\boldsymbol{Q}_*\|}\right), \tag{44}$$

then for all iter ≥ 1*,*

$$\|\mathbf{W}^{(\mathrm{iter}+1)} - \mathbf{W}_*\|_d \le \frac{1+\kappa_H}{2}\|\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*\|_d + C_H a_1, \tag{45}$$

which implies

$$\lim_{\mathrm{iter}\to\infty} \|\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*\|_d \le \frac{2C_H a_1}{1-\kappa_H}.$$

6.2.2 Step 2: Probabilistic estimation

Since $\boldsymbol{Q}_*$ is deterministic in our model, we only need to estimate the terms that depend on $\boldsymbol{\Delta}$ in Lemma 2. The estimations are summarized as follows, and the proof is deferred to the Appendix.

Lemma 3 (Step 2: Probabilistic estimation)**.** *(a) [Restatement of (Zhou and Zhu, 2019, Theorem 1.2)] If* $p_{\max} \geq \frac{c\log(\min(n,L))}{\min(n,L)}$ *for some constant* $c > 0$*, then for any* $r > 0$*, there exists a constant* $C > 0$ *depending only on* r, c *such that with probability at least* $1 - \min(n,L)^{-r}$*,* $\|\boldsymbol{\Delta}\| = \sup_{\mathbf{u}\in\mathbb{R}^L,\mathbf{v}\in\mathbb{R}^n} \frac{\boldsymbol{\Delta}\times_1\mathbf{u}\times_2\mathbf{v}\times_3\mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|^2}$ *satisfies* $\|\boldsymbol{\Delta}\| \leq C\sqrt{p_{\max}\max(n,L)}\log(\max(n,L))$*. (b) For any* $t > 0$*,*

$$\Pr\left(\left\|\Pi_{T,K_m}(\boldsymbol{Q}_*)\times_{2,3}\Pi_{T,K_m}(\boldsymbol{\Delta})\right\|_F > 3tnL\sqrt{p_{\max}^3}\right) \leq 2\dot{K}L\exp\left(\frac{-\frac{1}{2}t^2}{1+\frac{t}{3\sqrt{p_{\max}ng_{\min}}}}\right). \quad (46)$$

(c) For any $t > 0$*,*

$$\begin{aligned}&\Pr\left(\max_{1\leq m\leq M}\|\Pi_{T,K_m}(\boldsymbol{\Delta})\times_{2,3}\Pi_{T,K_m}(\boldsymbol{\Delta})\| \geq 9K^2t^2p_{\max}\max(n,L)\right) \quad (47)\\ \leq& 2K(L+n)\exp\left(-\frac{t^2/2}{1+\frac{t}{\sqrt{p_{\max}\max(n,L)g_{\min}}}}\right),\end{aligned}$$

where $g_{\min}$ *is the size of the smallest community.*

6.2.3 Step 3: A probabilistic result on Algorithm 1 without Assumptions (A2)-(A4)

From the definition of $\kappa_0 = \frac{\sigma_1(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)}{\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)}$ and the fact $\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_* = \mathcal{M}_1(\boldsymbol{Q}_*)\mathcal{M}_1(\boldsymbol{Q}_*)^T$, we have

$$\kappa_0 = \left(\frac{\sigma_1(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)}{\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)}\right) = \left(\frac{\sigma_1(\mathcal{M}_1(\boldsymbol{Q}_*))}{\sigma_M(\mathcal{M}_1(\boldsymbol{Q}_*))}\right)^2,$$

$\|\boldsymbol{Q}_*\times_1\mathbf{Y}\|_F = \|\mathcal{M}_1(\boldsymbol{Q}_*)\mathbf{Y}\|_F \leq \sigma_1(\mathcal{M}_1(\boldsymbol{Q}_*))\|\mathbf{Y}\|_F$, and $\|\boldsymbol{Q}_*\times_1\mathbf{Y}\|_F = \|\mathcal{M}_1(\boldsymbol{Q}_*)\mathbf{Y}\|_F \geq \sigma_M(\mathcal{M}_1(\boldsymbol{Q}_*))\|\mathbf{Y}\|_F$. As a result, $C_H \leq \sqrt{\kappa_0}$.

Combining $C_H \leq \sqrt{\kappa_0}$ with $\|\boldsymbol{Q}_*\| \leq \|\boldsymbol{Q}_*\|_F \leq p_{\max}n\sqrt{L}$, one obtains

$$\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*) \geq \frac{1}{\kappa_0}\sigma_1(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*) = \frac{1}{\kappa_0}\|\mathcal{M}_1(\boldsymbol{Q}_*)\|^2 \geq \frac{1}{M\kappa_0}\|\mathcal{M}_1(\boldsymbol{Q}_*)\|_F^2 = \frac{1}{M\kappa_0}\|\boldsymbol{Q}_*\|_F^2 \geq \frac{p_{\max}^2n^2L}{M\kappa_0\kappa_1},$$

Then, Lemma 2 and Lemma 3 imply the following statement.

Theorem 3 (Step 3: A generic result on Algorithm 1 without Assumptions **(A2)-(A4)**)**.** *If*

$$p_{\max} \geq \frac{c\log(\max(n,L))}{\max(n,L)} \text{ for some constant } c > 0, \quad (48)$$

then for any $r > 0$, there exists $C > 0$ that depending only on r, c such that for

$$a_1 = C\kappa_0^2\kappa_1\sqrt{M^3}\Big(\frac{t}{n\sqrt{p_{\max}}} + \frac{\dot{K}^2t^2}{p_{\max}n\min(n,L)}\Big) + C\kappa_0^2\kappa_1\kappa_2\sqrt{M\dot{K}}\frac{\log^2(\max(n,L))}{p_{\max}n\min(n,L)}$$
$$a_2 = C\kappa_0^2\kappa_1\kappa_2\sqrt{M\dot{K}}\Big(1 + \frac{\log(\max(n,L))}{\sqrt{p_{\max}n\min(n,L)}}\Big),$$

if

$$C\kappa_2\sqrt{\dot{K}} \le \sqrt{p_{\max}n\min(n,L)},\ \ \sqrt{4a_1\kappa_0(a_2+112\kappa_0^3)} \le 1-\kappa_H,\ \ 2\kappa_0\kappa_2a_1\sqrt{\dot{K}} \le (1-\kappa_H) \tag{49}$$

and the initialization satisfies

$$\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F \le \min\Big(\frac{1-\kappa_H}{2\kappa_0(a_2+112\kappa_0^3)}, \frac{1}{4\kappa_2\sqrt{\kappa_0\dot{K}}}\Big) \tag{50}$$

then with probability at least

$$1 - n^{-r} - 2\dot{K}L\exp\left(\frac{-\frac{1}{2}t^2}{1+\frac{t}{3\sqrt{p_{\max}ng_{\min}}}}\right) - 2K(L+n)\exp\left(-\frac{t^2/2}{1+\frac{t}{\sqrt{p_{\max}\max(n,L)g_{\min}}}}\right),$$

$$\|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*\|_d \le \frac{1+\kappa_H}{2}\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d + a_1\sqrt{\kappa_0}, \tag{51}$$

holds for all iter ≥ 1*, which implies*

$$\lim_{\text{iter}\to\infty}\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_F \le \lim_{\text{iter}\to\infty}\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d \le \frac{2a_1\sqrt{\kappa_0}}{1-\kappa_H}.$$

6.2.4 Step 4: Simplification under Assumptions **(A2)-(A4)**

We will need to estimate the parameters κ_1, κ_2 under Assumptions **(A2)-(A4)**. Since $\boldsymbol{Q}_*(m,:,:) = \sqrt{L_m}\boldsymbol{\Theta}_m\mathbf{B}_m\boldsymbol{\Theta}_m^T$ and $\boldsymbol{\Theta}_m^T\boldsymbol{\Theta}_m = \text{diag}(|G_{m,1}|, |G_{m,2}|, \cdots, |G_{m,K_m}|)$, we have $\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:)) \ge \sqrt{L_m}\sigma_{K_m}(\mathbf{B}_m)\sigma_{K_m}(\boldsymbol{\Theta}_m)^2 \ge c_1\sqrt{c_3}p_{\max}\frac{n\sqrt{L}}{K_m\sqrt{M}}\min_{m=1,\cdots,M}\sigma_{K_m}(\mathbf{B}_m^0)$, which suggests

$$\kappa_2 \le \frac{K_{\max}\sqrt{M}}{c_1\sqrt{c_3}b_1}.$$

Similarly, we have the estimation

$$\kappa_1 \le \frac{1}{c_1c_3^2b_2}.$$

Let $p^* = p_{\max} n \min(n, L)$ and $t = \log(n + L)$, then we have the estimation that

$$\begin{aligned}
a_1 =& \frac{C\kappa_0^2\sqrt{M^3}}{c_1 c_3^2 b_2}\Big(\frac{t}{n\sqrt{p_{\max}}} + \frac{\dot{K}^2 t^2}{p_{\max} n \min(n, L)}\Big) + C\kappa_0^2 \frac{\sqrt{\dot{K}} K_{\max} M}{c_1^{1.5} c_3^3 b_1 b_2} \frac{\log^2(\max(n, L))}{p_{\max} n \min(n, L)} \\
\leq& C\kappa_0^2 \log^2(n + L) \left(\frac{1}{\sqrt{p_{\max} n^2}} \frac{\sqrt{M^3}}{c_1 c_3^2 b_2} + \frac{1}{p^*}\Big(\frac{\dot{K}^2\sqrt{M^3}}{c_1 c_3^2 b_2} + \frac{\sqrt{\dot{K}} K_{\max} M}{c_1^{1.5} c_3^3 b_1 b_2}\Big)\right) \\
\leq& C\kappa_0^2 \log^2(n + L)\sqrt{M^3} \left(\sqrt{\frac{1}{p_{\max} n^2}} + \frac{\dot{K}^2}{p^*} \right) \\
a_2 =& C\kappa_0^2 \frac{\sqrt{\dot{K}} K_{\max} M}{c_1^{1.5} c_3^3 b_1 b_2}\Big(1 + \frac{\log(\max(n, L))}{\sqrt{p_{\max} n \min(n, L)}}\Big) \leq C\kappa_0^3 \log(n + L) \frac{1}{\sqrt{p^*}} \frac{\sqrt{\dot{K}} K_{\max} M}{c_1^{1.5} c_3^3 b_1 b_2} \\
\leq& C\kappa_0^2 \log(n + L) \sqrt{\frac{\dot{K}^3}{p^*}},
\end{aligned}$$

By calculation, a sufficient condition for the requirement in (49) becomes (19).

With (19), we have $\frac{p^*(1-\kappa_H)}{\kappa_0^2 \log(n+L)\dot{K}^3} \geq \frac{\sqrt{M}}{\dot{K}^2}$. Combining it with $a_2 \geq C\kappa_0^2\kappa_1\kappa_2\sqrt{M\dot{K}}\Big(1 + \frac{\log(\max(n,L))}{\sqrt{p^*}}\Big)$, the assumption on the initialization (50) can be guaranteed by (20). In addition, (48) and (49) follow from (19). Then (21) is proved by applying Theorem 3. The stopping criterion (1) and (22) follows from (21). The proof of (23) is presented in (24) at the end of Section 4.3.

6.3 Proof of Proposition 2

Proof of Proposition 2. Applying (Zhou and Zhu, 2019, Theorem 1.2), for any $r > 0$ there exists C depending on r such that

$$\Pr\Big(\|\mathcal{M}_1(\boldsymbol{\Delta})\| \leq C_{r,c}\sqrt{p_{\max}(n^2 + L)}\Big) \geq 1 - \min(n^2, L)^{-r}.$$

On the other hand,

$$\sigma_M(\mathcal{M}_1(\boldsymbol{P}_*)) = \sigma_M(\mathcal{M}_1(\boldsymbol{Q}_*)) \leq \sqrt{\kappa_0}\|\mathcal{M}_1(\boldsymbol{Q}_*)\|_F \leq \sqrt{\kappa_0} p_{\max} n\sqrt{L}.$$

Then Wedin's $\sin\theta$-theorem (Wedin, 1972) and the fact that $\mathbf{W}_*$ and $\mathbf{W}^{(0)}$ are the top left singular vectors of $\mathcal{M}_1(\boldsymbol{P}_*)$ and $\mathcal{M}_1(\boldsymbol{P}_* + \boldsymbol{\Delta})$ imply that

$$\|\mathbf{W}^{(0)}\mathbf{W}^{(0)\,T} - \mathbf{W}_*\mathbf{W}_*^T\| \leq \frac{\|\mathcal{M}_1(\boldsymbol{\Delta})\|}{\sigma_M(\mathcal{M}_1(\boldsymbol{P}_*))} \leq C_{r,c}\frac{\sqrt{n^2 + L}}{n\sqrt{\kappa_0 p_{\max} L}}.$$

Following from the theory of principal angles (Lerman and Zhang, 2014, Section 3.2.1) and (Golub and Van Loan, 2013, Section 6.4.3), up to a orthogonal transformation of size $M \times M$, the distance between $\mathbf{W}^{(0)}$ and $\mathbf{W}_*$ can be bounded above as

$$\min_{\mathbf{U}\in\mathbb{R}^{M\times M}:\mathbf{U}\mathbf{U}^T=\mathbf{I}} \|\mathbf{W}^{(0)}\mathbf{U} - \mathbf{W}_*\|_F \leq C_{r,c}\sqrt{M}\frac{\sqrt{n^2 + L}}{n\sqrt{p_{\max} L}}.$$

Combining it with (Lei and Lin, 2020, Lemma 9) and note that γ in (Lei and Lin, 2020, Lemma 9) can be set as $1/\sqrt{L_{\max}}$, where $L_{\max} = \max_{1\le m\le M} L_m$, one has at most

$$C_\epsilon \min_{\mathbf{U}\in\mathbb{R}^{M\times M}:\mathbf{U}\mathbf{U}^T=\mathbf{I}} \|\mathbf{W}^{(0)}\mathbf{U} - \mathbf{W}_*\|_F^2 L_{\max}$$

misclassified layers in the $(1+\epsilon)$ K-means step 2.

For any two sets of indices, $\mathcal{S}_1, \mathcal{S}_2 \in [1, \cdots, L]$, let $\mathbf{u}_{\mathcal{S}_1}$ and $\mathbf{u}_{\mathcal{S}_2}$ be their normalized indicator vectors and let $\mathcal{S}_3$ be their difference $\mathcal{S}_3 = (\mathcal{S}_1^c \cap \mathcal{S}_2) \cup (\mathcal{S}_2^c \cap \mathcal{S}_1)$. Then, if $n|\mathcal{S}_3|/|\mathcal{S}_1| \le 1/2$, one obtains

$$\begin{aligned}
\|\mathbf{u}_{\mathcal{S}_1} - \mathbf{u}_{\mathcal{S}_2}\|^2 = 2 - 2\langle \mathbf{u}_{\mathcal{S}_1}, \mathbf{u}_{\mathcal{S}_2}\rangle = 2 - 2\frac{|\mathcal{S}_1 \cap \mathcal{S}_2|}{\sqrt{|\mathcal{S}_1||\mathcal{S}_2|}} \le 2 - 2\frac{|\mathcal{S}_1| - |\mathcal{S}_3|}{\sqrt{|\mathcal{S}_1|(|\mathcal{S}_1| - |\mathcal{S}_3|)}} \\
\le \frac{2}{\sqrt{|\mathcal{S}_1|(|\mathcal{S}_1| - |\mathcal{S}_3|)}} \frac{|\mathcal{S}_1|(|\mathcal{S}_1| - |\mathcal{S}_3|) - (|\mathcal{S}_1| - |\mathcal{S}_3|)^2}{\sqrt{|\mathcal{S}_1|(|\mathcal{S}_1| - |\mathcal{S}_3|)} + |\mathcal{S}_1| - |\mathcal{S}_3|} \\
= \frac{2}{\sqrt{|\mathcal{S}_1|(|\mathcal{S}_1| - |\mathcal{S}_3|)}} \frac{|\mathcal{S}_3|(|\mathcal{S}_1| - |\mathcal{S}_3|)}{\sqrt{|\mathcal{S}_1|(|\mathcal{S}_1| - |\mathcal{S}_3|)} + |\mathcal{S}_1| - |\mathcal{S}_3|} \le 8\frac{|\mathcal{S}_3|}{|\mathcal{S}_1|}
\end{aligned}$$

Note that, when $|\mathcal{S}_3|/|\mathcal{S}_1| \ge 1/2$, one also has

$$\|\mathbf{u}_{\mathcal{S}_1} - \mathbf{u}_{\mathcal{S}_2}\|^2 \le 4 \le 8\frac{|\mathcal{S}_3|}{|\mathcal{S}_1|}.$$

The above inequalities then imply that, in step 3, for some constant C one has

$$\|\mathbf{W}^{(1)} - \mathbf{W}_*\|_F^2 \le 8C_\epsilon \min_{\mathbf{U}\in\mathbb{R}^{M\times M}:\mathbf{U}\mathbf{U}^T=\mathbf{I}} \|\mathbf{W}^{(0)}\mathbf{U} - \mathbf{W}_*\|_F^2 \frac{L_{\max}}{L_{\min}}.$$

Combining the upper bounds above and noting that Assumption **(A3)** implies that $L_{\max}/L_{\min} \le c_2/c_1$, completes the proof of Proposition 2. □

6.4 Proof of Theorem 2

Proof of Theorem 2. (a) For completeness, we will first write down the statement from (Lei and Lin, 2020, Lemma C.1):

Let $\mathbf{U}$ *be an* $n \times d$ *matrix with* K *distinct rows with minimum pairwise Euclidean norm separation* γ*. Let* $\hat{\mathbf{U}}$ *be another* $n \times d$ *matrix and* $(\hat{\mathbf{\Theta}}, \hat{\mathbf{X}})$ *be an* $(1+\epsilon)$*-approximate solution to* K*-means problem with input* $\hat{\mathbf{U}}$*, then the number of errors in* $\hat{\mathbf{\Theta}}$ *as an estimate of the row clusters of* $\mathbf{U}$ *is no larger than* $C_\epsilon \|\mathbf{U} - \hat{\mathbf{U}}\|_F^2 \gamma^{-2}$ *for some constant* C_ϵ *depending only on* ϵ*.*

Note that $\mathbf{W}_*$ is an $L \times M$ matrix with M distinct rows with minimum pairwise Euclidean norm separation larger than $2/\max_{m=1,\cdots,M}\sqrt{L_m}$, the misclassification rate is not larger than

$$\frac{\max_{m=1,\cdots,M} L_m}{4L} C_\epsilon \|\mathbf{W}_* - \hat{\mathbf{W}}\|_F^2.$$

Combining it with the estimation of $\|\mathbf{W}_* - \hat{\mathbf{W}}\|_F^2$ and assumption **(A3)** on $\frac{\max_{m=1,\cdots,M} L_m}{L}$, part (a) is proved.

(b) Denote the orthogonal matrix of size $n \times K_m$ whose columns are the top K_m eigenvectors of $\widehat{\boldsymbol{Q}}(m,:,:)$ by $\hat{\mathbf{U}}_m$ and the orthogonal matrix of size $n \times K_m$ whose columns are the top K_m eigenvectors of $\boldsymbol{Q}_*(m,:,:)$ by $\mathbf{U}_m$, then the Davis-Kahan theorem implies that

$$\|\hat{\mathbf{U}}_m - \mathbf{U}_m\|_F \leq \frac{\|\widehat{\boldsymbol{Q}}(m,:,:) - \boldsymbol{Q}_*(m,:,:)\|_F}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))}.$$

In addition, $\mathbf{U}_m$ has K_m distinct rows with minimum pairwise Euclidean norm separation at least $2/\sqrt{g_{m,\max}}$, where $g_{m,\max} = \max_{1\leq k\leq K_m} |G_{m,k}|$. As a result, (23) implies that the misclassification rate is bounded by

$$\begin{aligned}
&\frac{g_{m,\max}}{4n} C_\epsilon \frac{\|\widehat{\boldsymbol{Q}}(m,:,:) - \boldsymbol{Q}_*(m,:,:)\|_F^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))^2} \\
\leq &\frac{1}{4K_m} C_\epsilon \frac{\|\widehat{\boldsymbol{Q}}(m,:,:) - \boldsymbol{Q}_*(m,:,:)\|_F^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))^2} \leq C_\epsilon \frac{\|\widehat{\boldsymbol{Q}}(m,:,:) - \boldsymbol{Q}_*(m,:,:)\|^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))^2} \\
\leq &C_\epsilon \frac{(2\|\boldsymbol{Q}_*\|\|\hat{\mathbf{W}} - \mathbf{W}_*\|_F + 2\|\boldsymbol{\Delta}\|)^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))^2} \leq C_\epsilon \frac{(2p_{\max} n\sqrt{L}\|\hat{\mathbf{W}} - \mathbf{W}_*\|_F + 2\|\boldsymbol{\Delta}\|)^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))^2} \\
\leq &C_\epsilon (\frac{\kappa_2^2}{\dot{K}}) \bigg(\|\hat{\mathbf{W}} - \mathbf{W}_*\|_F^2 + \Big(\frac{\sqrt{p_{\max}\max(n,L)}\log(\max(n,L))}{p_{\max} n\sqrt{L}} \Big)^2 \bigg) \\
\leq &C_\epsilon (\frac{K_{\max}^2 M}{\dot{K}}) \bigg(\|\hat{\mathbf{W}} - \mathbf{W}_*\|_F^2 + \Big(\frac{\sqrt{p_{\max}\max(n,L)}\log(\max(n,L))}{p_{\max} n\sqrt{L}} \Big)^2 \bigg) \\
\leq &C_\epsilon (\frac{K_{\max}^2 M}{\dot{K}}) \bigg(\Big(\log^2(n+L)\sqrt{M^3}\frac{\sqrt{\kappa_0}}{1-\kappa_H} \Big(\sqrt{\frac{1}{p_{\max} n^2}} + \frac{\dot{K}^2}{p_{\max} n \min(n,L)} \Big) \Big)^2 \\
&+ \Big(\frac{\sqrt{p_{\max}\max(n,L)}\log(\max(n,L))}{p_{\max} n\sqrt{L}} \Big)^2 \bigg) \\
\leq &C_\epsilon K_{\max} \bigg(\log^4(n+L) M^3 \frac{\kappa_0}{(1-\kappa_H)^2} \Big(\frac{1}{p_{\max} n^2} + \frac{\dot{K}^4}{p_{\max}^2 n^2 \min(n,L)^2} \Big) + \frac{(n+L)\log(n+L)^2}{n^2 p_{\max} L} \bigg)
\end{aligned}$$

□

6.5 Proof of Lemma 2

Proof of Lemma 2. The main idea of the proof of Lemma 2 is as follows. Assumption **(A1)** implies that, when the observation is noise-free in the sense that $\boldsymbol{A} = \boldsymbol{P}_*$, Algorithm 1 converges linearly. As a result, we only need to show that the output of the algorithm does not change much if we replace $\boldsymbol{A}$ with $\boldsymbol{P}_*$ and Algorithm 1 with its linear approximation $P_{L_2} P_{L_1}$.

Given $\mathbf{W}^{(\text{iter})}$, we construct a skew-symmetric matrix $\mathbf{Y}^{(\text{iter})} \in \mathbb{R}^{M\times M}$ by

$$\mathbf{Y}^{(\text{iter})} = \frac{1}{2}(\mathbf{W}_*^T \mathbf{W}^{(\text{iter})} - \mathbf{W}^{(\text{iter})\,T} \mathbf{W}_*), \tag{52}$$

and then the update formula for the "clean" version of the algorithm is the solution to the equation

$$\boldsymbol{Q}_* \times_1 \hat{\mathbf{Y}}^{(\text{iter}+1)} = P_{L_2} P_{L_1} (\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\text{iter})}). \tag{53}$$

Intuitively, $\hat{\mathbf{Y}}^{(\text{iter}+1)}$ is the algorithmic update when $\boldsymbol{A}_*$ is replaced by $\boldsymbol{P}_*$, and Algorithm 1 is replaced by its linear approximation $P_{L_2}P_{L_1}$. By the definition of κ_H in (15), we have

$$\|\mathbf{W}_*\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_d \le \kappa_H\|\mathbf{W}_*\mathbf{Y}^{(\text{iter})}\|_d. \tag{54}$$

We will bound $\|\mathbf{W}^{(\text{iter}+1)}-\mathbf{W}_*\|_d$ as a function of $\|\mathbf{W}^{(\text{iter})}-\mathbf{W}_*\|_d$ by (54) and the following perturbation bounds in Lemma 4, with its proof deferred to Section 6.5.1.

Lemma 4. *For any* $\mathbf{W}^{\text{iter}}$*, let* $\mathbf{Y}^{\text{iter}}$ *and* $\hat{\mathbf{Y}}^{\text{iter}+1}$ *be defined as in* (52) *and* (53)*, then*

1.

$$\|\mathbf{W}_*\mathbf{Y}^{(\text{iter})}\|_d \le \|\mathbf{W}^{(\text{iter})}-\mathbf{W}_*\|_d. \tag{55}$$

2. For

$$\tilde{\mathbf{W}}^{(\text{iter}+1)} = \Pi_o(\boldsymbol{P}_* \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\text{iter})}))), \tag{56}$$

we have

$$\|\tilde{\mathbf{W}}^{(\text{iter}+1)} - \mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)}+\mathbf{I})\|_F \le \frac{32\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F^2}{2-8\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F - 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F(2+4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)}$$
$$+\Big(1+\frac{2\kappa_0+8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F}{2-8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F-\kappa_0(1+4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F}\Big)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2. \tag{57}$$

3. If

$$\|\mathbf{W}_*-\mathbf{W}^{(\text{iter})}\|_F \le \frac{\min_{m=1,\cdots,M}\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))}{\|\boldsymbol{Q}_*\|}, \tag{58}$$

then,

$$\|\mathbf{W}^{(\text{iter}+1)}-\tilde{\mathbf{W}}^{(\text{iter}+1)}\|_F \le \frac{\beta_1}{\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)-2\|\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*\|\|\mathbf{Y}^{(\text{iter})}\|_F-\beta}, \tag{59}$$

where, for a_1 *and* a_2 *are defined in Lemma 2,*

$$\beta_1 = \frac{\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)}{6}(a_1+a_2\|\mathbf{W}_*-\mathbf{W}^{(\text{iter})}\|_F^2)$$

With the perturbation bounds (57) and (59) in Lemma 4, we have

$$\begin{aligned}
&\|\mathbf{W}^{(\text{iter}+1)}-\mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)}+\mathbf{I})\|_F\\
\le&\|\mathbf{W}^{(\text{iter}+1)}-\tilde{\mathbf{W}}^{(\text{iter}+1)}\|_F + \|\tilde{\mathbf{W}}^{(\text{iter}+1)}-\mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)}+\mathbf{I})\|_F\\
\le&\frac{\beta_1}{\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)-2\|\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*\|\|\mathbf{Y}^{(\text{iter})}\|_F-\beta_1}\\
&+\frac{32\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F^2}{2-8\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F-4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F(2+4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)}\\
&+\Big(1+\frac{2\kappa_0+8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F}{2-8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F-\kappa_0(1+4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F}\Big)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2.
\end{aligned}$$

When

$$a_1 \le 1, \|\mathbf{W}^{(iter)} - \mathbf{W}_*\|_F \le \min(\frac{1}{32\kappa_0^3}, \frac{1}{\sqrt{a_2}}), \tag{60}$$

we have (using $\|\mathbf{Y}^{(iter)}\|_F \le \|\mathbf{W}^{(iter)} - \mathbf{W}_*\|_F$) $4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F \le 1$, $8\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F + 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F(2 + 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F) < 1$, $8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F + \kappa_0(1 + 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F < 1$, which imply $\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\|\|\mathbf{Y}^{(\text{iter})}\|_F \le \sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)/2$ and $\|\mathbf{W}_* - \mathbf{W}^{(\text{iter})}\|_F^2 \le 1$, and

$$\begin{aligned}
&\|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})\|_F \qquad (61)\\
\le&\frac{\beta_1}{\sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)(1 - 1/2 - 1/3)} + 32\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F^2 + (1 + 2\kappa_0 + 8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2\\
=&\frac{6\beta_1}{\sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)} + (3 + 2\kappa_0 + 8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2\\
\le&\frac{6\beta_1}{\sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)} + (5 + 2\kappa_0)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2\\
=&\kappa_0(a_1 + a_2\|\mathbf{W}_*^T\mathbf{W}^{(\text{iter})} - \mathbf{I}\|^2) + (5 + 2\kappa_0)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2.
\end{aligned}$$

Combining (54), (55), and (61), we have

$$\begin{aligned}
&\|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*\|_d \le \|\mathbf{W}_*\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_d + \|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})\|_d\\
&\le \kappa_H\|\mathbf{W}_*\mathbf{Y}^{(\text{iter})}\|_d + \|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})\|_d\\
&\le \kappa_H\|\mathbf{W}_* - \mathbf{W}^{(\text{iter})}\|_d + \|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})\|_d\\
&\le \kappa_H\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d + C_H\Big(a_1 + a_2\|\mathbf{W}_* - \mathbf{W}^{(\text{iter})} - \mathbf{I}\|_F^2 + (5 + 2\kappa_0)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2\Big)\\
&\le \kappa_H\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d + C_H\Big(a_1 + a_2\|\mathbf{W}_* - \mathbf{W}^{(\text{iter})}\|_F^2 + 16(5 + 2\kappa_0)\kappa_0^2\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d^2\Big)\\
&\le \kappa_H\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d + C_H\Big(a_1 + (a_2 + 80\kappa_0^2 + 32\kappa_0^3)\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d^2\Big)
\end{aligned}$$

As a result, if in addition we have

$$C_H(a_2 + 80\kappa_0^2 + 32\kappa_0^3)\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d \le (1 - \kappa_H)/2, \tag{62}$$

then

$$\|\mathbf{W}^{(\text{iter}+1)} - \mathbf{W}_*\|_d \le \frac{1 + \kappa_H}{2}\|\mathbf{W}^{(\text{iter})} - \mathbf{W}_*\|_d + C_H a_1. \tag{63}$$

By the assumptions in (43) and (44), the argument of induction implies that (58), (60), and (62) hold for all iter ≥ 1. Therefore, (63) holds for all iter ≥ 1 and the theorem is proved.

6.5.1 Proof of Lemma 4

The proof of Lemma 4 is based on Lemma 5-8. Among these lemmas, the proofs of Lemmas 5, 7, 8 will be presented in Section 6.6, and Lemma 6 is a restatement of Theorem VII.5.1 in Bhatia (1997). We shall prove the three perturbation bounds (55), (57), and (59) separately.

Lemma 5. *Given any symmetric matrix* $\mathbf{Q} \in \mathbb{R}^{M\times M}$*, if* $\mathbf{QY}-\mathbf{X}=\mathbf{S}$ *holds for a symmetric matrix* $\mathbf{S} \in \mathbb{R}^{M\times M}$ *and a skew symmetric matrix* $\mathbf{Y} \in \mathrm{Skew}_M$*, then we have*

$$\max(\|\mathbf{QY}\|_F, \|\mathbf{S}\|_F) \le 2\|\mathbf{X}\|_F.$$

Lemma 6. *For any square matrices* $\mathbf{A}$ *and* $\mathbf{B}$*,*

$$\|\Pi_o(\mathbf{A}) - \Pi_o(\mathbf{B})\|_F \le 2\frac{\|\mathbf{A}-\mathbf{B}\|_F}{\sigma_{\min}(\mathbf{A})+\sigma_{\min}(\mathbf{B})}.$$

The inequality also holds if the operator norm is replaced with Frobenius norm.

Lemma 7. *For any positive definite matrix* $\mathbf{Q} \in \mathbb{R}^{M\times M}$ *and any skew symmetric matrix* $\mathbf{Y} \in \mathrm{Skew}_M$ *with* $\|\mathbf{Y}\| \le 1$*, we have*

$$\|\Pi_o(\mathbf{Q}(\mathbf{I}+\mathbf{Y}) - (\mathbf{I}+\mathbf{Y})\|_F \le \Big(1+\frac{2\|\mathbf{Q}\|}{2\sigma_{\min}(\mathbf{Q}) - e\|\mathbf{Q}\|\|\mathbf{Y}\|}\Big)(e-2)\|\mathbf{Y}\|_F^2.$$

Lemma 8. *Let* Π_o *be defined in* (2)*. Then*

$$\|\Pi_o(\mathbf{X}+\mathbf{Y}) - \Pi_o(\mathbf{X})\|_F \le (1+\sqrt{2})\frac{\|\mathbf{Y}\|_F}{\sigma_M(\mathbf{X}) - \|\mathbf{Y}\|},$$

where $\sigma_M(\mathbf{X})$ *represents the* M*-th singular value of* $\mathbf{X}$*.*

Lemma 9. *For a symmetric matrix* $\mathbf{X}_0 \in \mathbb{R}^{n\times n}$ *with rank* r*, let* $\Pi_T : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ *be projection onto the tangent space of* $\{\mathbf{X} : \mathrm{rank}(\mathbf{X}) = r\}$ *at* $\mathbf{X}_0$ *and* $\Pi_{T,\perp}$ *be the remainder of the projection, then for any symmetric matrix* $\mathbf{\Delta}$*,*

$$\|\Pi_{T,\perp}(\mathbf{\Delta})\|_F \le \frac{\|\Pi_T(\mathbf{\Delta})\|_F^2}{\sigma_r(\mathbf{X}_0) - \|\mathbf{\Delta}\|}.$$

Proof of bound 1 in (55)

It follows from the observation that $\mathbf{W}_*\mathbf{Y}^{(\mathrm{iter})} = P_{L_0}(\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*)$ and from the definition (42), $\|\mathbf{W}_*\mathbf{Y}^{(\mathrm{iter})}\|_d = \lambda\|\mathbf{P}_* \times \mathbf{W}_*\mathbf{Y}^{(\mathrm{iter})}\|_F$ and

$$\|\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*\|_d = \sqrt{(\lambda\|\mathbf{P}_* \times \mathbf{W}_*\mathbf{Y}^{(\mathrm{iter})}\|_F)^2 + \|P_{L_0^\perp}(\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*)\|_F^2}.$$

Proof of bound 2 in (57)

By the definition of (53), we have that for any $\mathbf{\Delta} \in \mathrm{Skew}_M$,

$$\begin{aligned} 0 &= \langle \boldsymbol{Q}_* \times_1 \hat{\mathbf{Y}}^{(\mathrm{iter}+1)} - P_{L_1}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})}), \boldsymbol{Q}_* \times_1 \mathbf{\Delta}\rangle \\ &= \Big\langle \boldsymbol{Q}_* \times_{2,3} \big(\boldsymbol{Q}_* \times_1 \hat{\mathbf{Y}}^{(\mathrm{iter}+1)} - P_{L_1}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})})\big), \mathbf{\Delta}\Big\rangle. \end{aligned}$$

As a result, $\boldsymbol{Q}_* \times_{2,3} \big(\boldsymbol{Q}_* \times_1 \hat{\mathbf{Y}}^{(\mathrm{iter}+1)} - P_{L_1}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})})\big) = (\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)\hat{\mathbf{Y}}^{(\mathrm{iter}+1)} - \boldsymbol{Q}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})})$ is a symmetric matrix. Denoting it by $\mathbf{S}$, Lemma 5 implies that

$$\begin{aligned} &\max(\|(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)\hat{\mathbf{Y}}^{(\mathrm{iter}+1)}\|_F, \|\mathbf{S}\|_F) \le 2\|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})})\|_F \\ \le& 2\max_{1\le m\le M} \|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,K_m}(\boldsymbol{Q}_*)\|\|\mathbf{Y}^{(\mathrm{iter})}\|_F \le 4\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\|\|\mathbf{Y}^{(\mathrm{iter})}\|_F, \end{aligned} \tag{64}$$

where the last inequality is due to

$$\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* = \|\mathcal{M}_1(\boldsymbol{Q}_*)\|^2, \|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,K_m}(\boldsymbol{Q}_*)\| \le \|\mathcal{M}_1(\boldsymbol{Q}_*)\|\|\mathcal{M}_1(\Pi_{T,K_m}(\boldsymbol{Q}_*))\|,$$

$$\mathcal{M}_1(\Pi_{T,K_m}(\boldsymbol{Q}_*)) = \mathcal{M}_1(\boldsymbol{Q}_*)(\Pi_{\mathbf{U}_m} \otimes \mathbf{I}) + \mathcal{M}_1(\boldsymbol{Q}_*)(\Pi_{\mathbf{U}_m^\perp} \otimes \Pi_{\mathbf{U}_m}),$$

where $\otimes$ represents the Kronecker product, and $\|\mathbf{A} \otimes \mathbf{B}\| \le \|\mathbf{A}\|\|\mathbf{B}\|$.

As a result,

$$\|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F \le 4\kappa_0 \|\mathbf{Y}^{(\text{iter})}\|_F. \tag{65}$$

By the definitions of $\tilde{\mathbf{W}}^{(\text{iter}+1)}$ and $\mathbf{S}$, we have $\mathbf{W}_*^T \tilde{\mathbf{W}}^{(\text{iter}+1)} = \Pi_o(\boldsymbol{Q}_* \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\text{iter})}))) = \Pi_o((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I}) - \mathbf{S})$. Lemma 6, the upper bounds of $\|\mathbf{S}\|_F$ in (64), and $\|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F$ in (65) imply

$$\begin{aligned}
&\|\mathbf{W}_* \tilde{\mathbf{W}}^{(\text{iter}+1)} - \Pi_o((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* - \mathbf{S})(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I}))\|_F \\
=&\|\Pi_o((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I}) - \mathbf{S}) - \Pi_o((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* - \mathbf{S})(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I}))\|_F \\
\le&\frac{2\|\mathbf{S}\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F}{2\sigma_{\min}((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})) - \|\mathbf{S}\|(2 + \|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|)} \\
\le&\frac{2\|\mathbf{S}\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F}{2\sigma_{\min}((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})) - \|\mathbf{S}\|_F(2 + \|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|)} \\
\le&\frac{2\|\mathbf{S}\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F}{2\sigma_{\min}(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*) - 2\|(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*)\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F - \|\mathbf{S}\|_F(2 + \|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|)} \\
\le&\frac{2(4\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\|\|\mathbf{Y}^{\text{iter}}\|_F)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)}{2\sigma_{\min}(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*) - 2\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\|(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F) - (4\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\|\|\mathbf{Y}^{\text{iter}}\|_F)(2 + 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)} \\
\le&\frac{32\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F^2}{2 - 8\kappa_0^2\|\mathbf{Y}^{\text{iter}}\|_F - 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F(2 + 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)}.
\end{aligned}$$

In addition, Lemma 7 implies

$$\begin{aligned}
&\|\Pi_o((\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* - \mathbf{S})(\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})) - (\hat{\mathbf{Y}}^{(\text{iter}+1)} + \mathbf{I})\|_F \\
\le&\Big(1 + \frac{2\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* - \mathbf{S}\|}{2\sigma_{\min}(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* - \mathbf{S}) - e\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_* - \mathbf{S}\|\|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|}\Big)(e-2)\|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F^2 \\
\le&\Big(1 + \frac{2\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\| + 2\|\mathbf{S}\|_F}{2\sigma_{\min}(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*) - 2\|\mathbf{S}\|_F - (\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\| + \|\mathbf{S}\|_F)\|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|}\Big)\|\hat{\mathbf{Y}}^{(\text{iter}+1)}\|_F^2 \\
\le&\Big(1 + \frac{2\kappa_0 + 8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F}{2 - 8\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F - \kappa_0(1 + 4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F}\Big)(4\kappa_0\|\mathbf{Y}^{\text{iter}}\|_F)^2.
\end{aligned}$$

Combining the previous two estimations, part 2 is proved.

Proof of bound 3 in (59)

By the definition of $\tilde{\mathbf{W}}^{(\text{iter}+1)}$ in (56) and $\mathbf{W}^{(\text{iter}+1)} = \Pi_o(\boldsymbol{A} \times_{2,3} \Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\text{iter})}))$, Lemma 8 implies that

$$\|\tilde{\mathbf{W}}^{(\text{iter}+1)} - \mathbf{W}^{(\text{iter}+1)}\|_F \le \frac{\beta_1}{\sigma_M(\boldsymbol{P}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 (\mathbf{I} + \mathbf{Y}^{(\text{iter})}))) - \beta_1} \le \frac{\beta_1}{\sigma_M(\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*) - \beta_2 - \beta_1}.$$

for

$$\beta_1 = \|\boldsymbol{A} \times_{2,3} \Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})}) - \boldsymbol{P}_* \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})}))\|_F$$

and

$$\beta_2 = \|\boldsymbol{P}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 \mathbf{Y}^{(\mathrm{iter})})\|.$$

By the same calculation as in (64), we have

$$\beta_2 \le \max_{1\le m\le M} \|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,K_m}(\boldsymbol{Q}_*)\| \|\mathbf{Y}^{(\mathrm{iter})}\|_F \le 2\|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\| \|\mathbf{Y}^{(\mathrm{iter})}\|_F. \tag{66}$$

To estimate the upper bound of β_1, we note that $\beta_1 \le \beta_3 + \beta_4 + \beta_5$, where

$$\beta_3 = \|\boldsymbol{A} \times_{2,3} \Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})}) - \boldsymbol{A} \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})} - \boldsymbol{Q}_*))\|_F, \tag{67}$$

$$\begin{aligned}\beta_4 =& \|\boldsymbol{A} \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})} - \boldsymbol{Q}_*)) - \boldsymbol{P}_* \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{P}_* \times_1 (\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*)))\|_F \\ =& \|\boldsymbol{\Delta} \times_{2,3} \Pi_{T,\mathbf{K}}[\boldsymbol{P}_* \times_1 \mathbf{W}^{(\mathrm{iter})}] + \boldsymbol{\Delta} \times_{2,3} \Pi_{T,\mathbf{K}}[\boldsymbol{\Delta} \times_1 \mathbf{W}^{(\mathrm{iter})}] + \boldsymbol{P}_* \times_{2,3} \Pi_{T,\mathbf{K}}[\boldsymbol{\Delta} \times_1 \mathbf{W}^{(\mathrm{iter})}]\|_F,\end{aligned} \tag{68}$$

and

$$\begin{aligned}\beta_5 =& \|\boldsymbol{P}_* \times_{2,3} (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{P}_* \times_1 (\mathbf{W}^{(\mathrm{iter})} - \mathbf{W}_*))) - \boldsymbol{P}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 (\mathbf{I} + \mathbf{Y}^{(\mathrm{iter})}))\|_F \\ =& \|\boldsymbol{P}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 (\mathbf{W}_*^T \mathbf{W}^{(\mathrm{iter})} - \mathbf{I} - \mathbf{Y}^{(\mathrm{iter})}))\|_F \\ =& \frac{1}{2}\|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 (\mathbf{W}_*^T \mathbf{W}^{(\mathrm{iter})} + \mathbf{W}^{(\mathrm{iter})\,T} \mathbf{W}_* - 2\mathbf{I}))\|_F \\ =& \frac{1}{2}\|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{Q}_* \times_1 (\mathbf{W}_* - \mathbf{W}^{(\mathrm{iter})})^T (\mathbf{W}_* - \mathbf{W}^{(\mathrm{iter})})\|_F\end{aligned} \tag{69}$$

To obtain an upper bound for β_3 in (67), let $\Pi_{T,\mathbf{K},\perp} \in \mathbb{R}^{M\times n\times n}$ be $\Pi_{\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})}) - (\boldsymbol{Q}_* + \Pi_{T,\mathbf{K}}(\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})} - \boldsymbol{Q}_*))$, then by Lemma 9 and

$$\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})} - \boldsymbol{Q}_* = \boldsymbol{\Delta} \times_1 \mathbf{W}^{(\mathrm{iter})} + \boldsymbol{P}_* \times_1 (\mathbf{W}^{(\mathrm{iter})} - \mathbf{W} - *),$$

we have $\mathrm{rank}(\Pi_{T,\mathbf{K},\perp}(m,:,:)) \le 2K_m$ and

$$\begin{aligned}\|\Pi_{T,\mathbf{K},\perp}(m,:,:)\| \le& \frac{\|\Pi_{T,K_m}[\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})} - \boldsymbol{Q}_*](m,:,:)\|^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:)) - \|\boldsymbol{Q}_*\|\|\mathbf{W}_* - \mathbf{W}^{(iter)}\| - \|\boldsymbol{\Delta}\|} \\ \le& \frac{4\|[\boldsymbol{A} \times_1 \mathbf{W}^{(\mathrm{iter})} - \boldsymbol{Q}_*](m,:,:)\|^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:)) - \|\boldsymbol{Q}_*\|\|\mathbf{W}_* - \mathbf{W}^{(iter)}\| - \|\boldsymbol{\Delta}\|} \\ \le& \frac{4(\|\boldsymbol{Q}_*\|\|[\mathbf{W}_*^T \mathbf{W}^{(\mathrm{iter})} - \mathbf{I}](m,:)\| + \|\boldsymbol{\Delta}\|)^2}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:)) - \|\boldsymbol{Q}_*\|\|\mathbf{W}_* - \mathbf{W}^{(iter)}\| - \|\boldsymbol{\Delta}\|} \le \frac{16(\|\boldsymbol{Q}_*\|^2\|[\mathbf{W}_*^T \mathbf{W}^{(\mathrm{iter})} - \mathbf{I}](m,:)\|^2 + \|\boldsymbol{\Delta}\|^2)}{\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))}.\end{aligned} \tag{70}$$

Since $\mathrm{rank}(\Pi_{T,\mathbf{K},\perp}(m,:,:)) \le 2K_m$, we have

$$\beta_3 \le \|\boldsymbol{A} \times_{2,3} \Pi_{T,\mathbf{K},\perp}(m,:,:)\| \le 2K_m \|\boldsymbol{A}\| \|\Pi_{T,\mathbf{K},\perp}(m,:,:)\|.$$

Summing $1 \le m \le M$, (70) yields

$$\begin{aligned}\beta_3 =&\|\boldsymbol{A} \times_{2,3} \Pi_{T,\mathbf{K},\perp}\|_F \le 16\|\boldsymbol{A}\|\sqrt{\sum_{m=1}^{M} \frac{K_m^2(\|\boldsymbol{Q}_*\|^2\|[\mathbf{W}_*^T\mathbf{W}^{(\text{iter})}-\mathbf{I}](m,:)\|^2+\|\boldsymbol{\Delta}\|^2)^2}{\big(\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))-\|\boldsymbol{Q}_*\|\|\mathbf{W}_*-\mathbf{W}^{(iter)}\|-\|\boldsymbol{\Delta}\|\big)^2}}\\ \le&16\|\boldsymbol{A}\|\sum_{m=1}^{M}\frac{K_m(\|\boldsymbol{Q}_*\|^2\|[\mathbf{W}_*^T\mathbf{W}^{(\text{iter})}-\mathbf{I}](m,:)\|^2+\|\boldsymbol{\Delta}\|^2)}{\sqrt{M}\big(\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))-\|\boldsymbol{Q}_*\|\|\mathbf{W}_*-\mathbf{W}^{(iter)}\|-\|\boldsymbol{\Delta}\|\big)}\\ \le&16\|\boldsymbol{A}\|\frac{\dot{K}(\|\boldsymbol{Q}_*\|^2\|\mathbf{W}_*^T\mathbf{W}^{(\text{iter})}-\mathbf{I}\|^2+\|\boldsymbol{\Delta}\|^2)}{\sqrt{M}\big(\min_{k=1}^{M}\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))-\|\boldsymbol{Q}_*\|\|\mathbf{W}_*-\mathbf{W}^{(iter)}\|-\|\boldsymbol{\Delta}\|\big)}\\ \le&32\frac{\dot{K}(\|\boldsymbol{Q}_*\|+\|\boldsymbol{\Delta}\|)(\|\boldsymbol{Q}_*\|^2\|\mathbf{W}_*^T\mathbf{W}^{(\text{iter})}-\mathbf{I}\|^2+\|\boldsymbol{\Delta}\|^2)}{\sqrt{M}\big(\min_{k=1}^{M}\sigma_{K_m}(\boldsymbol{Q}_*(m,:,:))\big)}\end{aligned}$$

To find an upper bound for β_4 in (68), note that

$$\begin{aligned}&\|[\boldsymbol{\Delta} \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{\Delta} \times_1 \mathbf{W}^{(\text{iter})})](:,m)\| = \|(\boldsymbol{\Delta} \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta}))[\mathbf{W}^{(\text{iter})}](:,m)\|\\ =&\|(\Pi_{T,K_m}(\boldsymbol{\Delta}) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta}))[\mathbf{W}^{(\text{iter})}](:,m)\| \le \|\Pi_{T,K_m}(\boldsymbol{\Delta}) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\|,\end{aligned}$$

which implies

$$\|\boldsymbol{\Delta} \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{\Delta} \times_1 \mathbf{W}^{(\text{iter})})\|_F \le \max_{1\le m\le M}\|\Pi_{T,K_m}(\boldsymbol{\Delta}) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\|\sqrt{M}. \tag{71}$$

Similarly,

$$\begin{aligned}&\|\boldsymbol{P}_* \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{\Delta} \times_1 \mathbf{W}^{(\text{iter})})\|_F \le \max_{1\le m\le M}\|\Pi_{T,K_m}(\boldsymbol{P}_*) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\|\sqrt{M},\\ &\|\boldsymbol{\Delta} \times_{2,3} \Pi_{T,\mathbf{K}}(\boldsymbol{P}_* \times_1 \mathbf{W}^{(\text{iter})})\|_F \le \max_{1\le m\le M}\|\Pi_{T,K_m}(\boldsymbol{P}_*) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\|\sqrt{M}.\end{aligned} \tag{72}$$

As a result,

$$\beta_4 \le \sqrt{M}\Big(\max_{1\le m\le M}\|\Pi_{T,K_m}(\boldsymbol{\Delta}) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\| + 2\max_{1\le m\le M}\|\Pi_{T,K_m}(\boldsymbol{P}_*) \times_{2,3} \Pi_{T,K_m}(\boldsymbol{\Delta})\|\Big).$$

To find an upper bound for β_5 in (69), we use

$$\begin{aligned}\beta_5 =&\frac{1}{2}\|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,\mathbf{K}}[\boldsymbol{Q}_* \times_1 (\mathbf{W}_*-\mathbf{W}^{(\text{iter})})^T(\mathbf{W}_*-\mathbf{W}^{(\text{iter})})]\|_F\\ \le&\frac{1}{2}\max_{1\le m\le M}\|\boldsymbol{Q}_* \times_{2,3} \Pi_{T,K_m}\boldsymbol{Q}_*\|\|(\mathbf{W}_*-\mathbf{W}^{(\text{iter})})^T(\mathbf{W}_*-\mathbf{W}^{(\text{iter})})\|_F \le \|\boldsymbol{Q}_* \times_{2,3} \boldsymbol{Q}_*\|\|\mathbf{W}_*-\mathbf{W}^{(\text{iter})}\|_F^2,\end{aligned}$$

where the inequalities follow the same calculation as in (64) and (66), and $\|(\mathbf{W}_*-\mathbf{W}^{(\text{iter})})^T(\mathbf{W}_*-\mathbf{W}^{(\text{iter})})\|_F=\|\mathbf{W}_*-\mathbf{W}^{(\text{iter})}\|_F^2$.

Combining the estimations of β_2, β_3, β_4, β_5 with $\|\mathbf{W}_*^T\mathbf{W}^{(\text{iter})}-\mathbf{I}\| \le \|\mathbf{W}_*^T\mathbf{W}^{(\text{iter})}-\mathbf{I}\|_F \le \|\mathbf{W}_*-\mathbf{W}^{(\text{iter})}\|_F$, $\beta_1 \le \beta_3+\beta_4+\beta_5$, and $\kappa_0 = \frac{\|\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*\|}{\sigma_M(\boldsymbol{Q}_*\times_{2,3}\boldsymbol{Q}_*)}$, (59) is proved. □

6.6 Proofs of auxiliary Lemmas and Propositions

Proof of Lemma 1. For any $\mathbf{X} \in \mathrm{Skew}_m$ such that $\boldsymbol{Q}_* \times_1 \mathbf{X} \in L_1$, due to

$$[\boldsymbol{Q}_* \times_1 \mathbf{X}](m,:,:) = \sum_{m'=1}^{M} \mathbf{X}(m,m')\boldsymbol{Q}_*(m',:,:) = \sum_{m'=1, m'\neq m}^{M} \mathbf{X}(m,m')\boldsymbol{Q}_*(m',:,:),$$

one has $\sum_{m'=1,m'\neq m}^{M} \mathbf{X}(m,m')\Pi_{\mathbf{U}_m^\perp}\boldsymbol{Q}_*(m',:,:)\Pi_{\mathbf{U}_m^\perp} = 0$. When the first sufficient condition holds, then $\mathbf{X}(m,m') = 0$ for all $1 \le m' \le M$. Combining the analysis for all $1 \le m \le M$, we have $\mathbf{X} = 0$. As a result, $L_1 \cap L_2 = \{0\}$ and **(A1)** holds.

The second sufficient condition follows from the first sufficient condition directly. □

Proof of Lemma 3. We first summarize a special case of (Lei and Lin, 2020, Theorem 2.1) as follows:

Lemma 10. *If $\mathbf{X}_l \in \mathbb{R}^{n\times r}$, $l = 1, \cdots, L$ are independent, elementwise sampled from a centered Bernoulli distribution with parameters not larger than p, then*

$$\Pr\left(\Big\|\sum_{l=1}^{L}\mathbf{X}_l\Big\| \ge t\right) \le 2(r+n)\exp\left(-\frac{t^2/2}{pL\max(n,r)+t}\right)$$

Since $\mathbf{U}_m$ is a matrix of size $n\times K_m$ such that the i-th column is the normalized indicator vector of the set $G_{m,i}$, i.e., the indicator vector with scale $1/\sqrt{|G_{m,i}|}$. As a result,

$$\left\langle \boldsymbol{Q}_*(m_1,:,:), \boldsymbol{\Delta}(l,:,:)\mathbf{U}_m(k,:)\mathbf{U}_m(k,:)^T\right\rangle = \sum_{j_1=1}^{n}\sum_{j_2\in G_{m,i}} \boldsymbol{\Delta}(l,j_1,j_2')\frac{\sum_{j_2'\in G(m,i)}\boldsymbol{Q}_*(m_1,j_1,j_2')}{|G(m,i)|},$$

and, by Bernstein's inequality, since each term is no larger than $\sqrt{L_m}p_{\max}$ and

$$\mathbb{E}\left[\left(\boldsymbol{\Delta}(l,j_1,j_2')\ \frac{\sum_{j_2'\in G(m,k)}\boldsymbol{Q}_*(m_1,j_1,j_2')}{|G(m,k)|}\right)^2\right] \le L_m p_{\max}^3,$$

$$\Pr\left(\left|\left\langle \boldsymbol{Q}_*(m_1,:,:), \boldsymbol{\Delta}(l,:,:)\mathbf{U}_m(k,:)\mathbf{U}_m(k,:)^T\right\rangle\right| > t\sqrt{L_m p_{\max}^3 n|G_{m,k}|}\right) \le 2\exp\left(\frac{-\frac{1}{2}t^2}{1+\frac{t}{3\sqrt{p_{\max}n|G_{m,k}|}}}\right).$$

Summing it over $1 \le i \le K_m$, $1 \le l \le L$, and $1 \le m_1 \le M$, we proved (46).

By definition, $\|\mathcal{M}_1(\Pi_{T,K_m}(\boldsymbol{\Delta}))\| \le 3\sum_{k=1}^{K_m}\|\frac{1}{\sqrt{G_{m,k}}}\sum_{i\in G_{m,k}}\boldsymbol{\Delta}(:,:,i)\|$, and Lemma 10 implies that

$$\Pr\left(\|\sum_{i\in G_{m,k}}\boldsymbol{\Delta}(:,:,i)\| \ge t\sqrt{p_{\max}\max(n,L)|G_{m,k}|}\right) \le 2(L+n)\exp\left(-\frac{t^2/2}{1+\frac{t}{\sqrt{p_{\max}\max(n,L)|G_{m,k}|}}}\right).$$

As a result,

$$\Pr\left(\|\mathcal{M}_1(\Pi_{T,K_m}(\mathbf{\Delta}))\| \le 3K_m t\sqrt{p_{\max}\max(n,L)}\right) \ge 2K_m(L+n)\exp\left(-\frac{t^2/2}{1+\frac{t}{\sqrt{p_{\max}\max(n,L)g_{\min}}}}\right)$$

and (47) is then proved. Therefore, Lemma 3 is proved. □

Proof of Lemma 5. Without loss of generality, assume that $\mathbf{Q}$ is a diagonal matrix with $\mathbf{Q} = \mathrm{diag}(q_1, \cdots, q_M)$. Then for each $1 \le i, j \le M$, $q_i\mathbf{Y}_{i,j} - \mathbf{X}_{i,j} = \mathbf{S}_{i,j}$ and $q_j\mathbf{Y}_{j,i} - \mathbf{X}_{j,i} = \mathbf{S}_{j,i}$. Since $\mathbf{S}_{i,j} = \mathbf{S}_{j,i}$ and $\mathbf{Y}_{i,j} = -\mathbf{Y}_{j,i}$, we have

$$\mathbf{Y}_{i,j} = \frac{\mathbf{X}_{i,j} - \mathbf{X}_{j,i}}{q_i + q_j},\ \ \mathbf{S}_{i,j} = \frac{q_j\mathbf{X}_{i,j} - q_i\mathbf{X}_{j,i}}{q_i + q_j}.$$

Since $\{q_i\}_{i=1}^M$ are positive, the lemma is proved. □

Proof of Lemma 7. Let $\mathbf{U} = \exp(\mathbf{Y})$, then it is an orthogonal matrix. In addition, $\|\mathbf{U}-\mathbf{I}\| = \|\sum_{k=1}^{\infty}\frac{1}{k!}\mathbf{Y}^k\| \le \sum_{k=1}^{\infty}\frac{1}{k!}\|\mathbf{Y}\|^k \le (e-1)\|\mathbf{Y}\|$ and $\|\mathbf{U} - (\mathbf{I}+\mathbf{Y})\|_F = \|\sum_{k=2}^{\infty}\frac{1}{k!}\mathbf{Y}^k\|_F \le \|\mathbf{Y}\|_F\sum_{k=2}^{\infty}\frac{1}{k!}\|\mathbf{Y}\|^{k-2} \le (e-2)\|\mathbf{Y}\|_F^2$. As a result, Lemma 6 implies

$$\begin{aligned}
&\|\Pi_o(\mathbf{Q}(\mathbf{I}+\mathbf{Y})) - (\mathbf{I}+\mathbf{Y})\|_F \le \|\Pi_o(\mathbf{Q}(\mathbf{I}+\mathbf{Y})) - \Pi_o(\mathbf{QU})\|_F + \|\Pi_o(\mathbf{QU}) - (\mathbf{I}+\mathbf{Y})\|_F\\
\le&\frac{2}{\sigma_{\min}(\mathbf{Q}(\mathbf{I}+\mathbf{Y})) + \sigma_{\min}(\mathbf{QU})}\|\mathbf{Q}(\mathbf{U}-\mathbf{I}-\mathbf{Y})\|_F + \|\mathbf{U}-\mathbf{I}-\mathbf{Y}\|_F\\
\le&\Big(1+\frac{2\|\mathbf{Q}\|}{\sigma_{\min}(\mathbf{Q}(\mathbf{I}+\mathbf{Y})) + \sigma_{\min}(\mathbf{QU})}\Big)(e-2)\|\mathbf{Y}\|_F\\
\le&\Big(1+\frac{2\|\mathbf{Q}\|}{2\sigma_{\min}(\mathbf{Q}) - e\|\mathbf{Q}\|\|\mathbf{Y}\|}\Big)(e-2)\|\mathbf{Y}\|_F^2.
\end{aligned}$$

□

Proof of Lemma 8.

$$\begin{aligned}
&\|\Pi_o(\mathbf{X}+\mathbf{Y}) - \Pi_o(\mathbf{X})\|_F = \|(\mathbf{X}+\mathbf{Y})[(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{-0.5} - \mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-0.5}\|_F\\
=&\|\mathbf{Y}[(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{-0.5}\|_F + \|\mathbf{X}\left([(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{-0.5} - [\mathbf{X}^T\mathbf{X}]^{-0.5}\right)\|_F\\
\le&\frac{\|\mathbf{Y}\|_F}{\sigma_M(\mathbf{X}) - \|\mathbf{Y}\|} + \|\mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-0.5}[[(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{0.5} - [\mathbf{X}^T\mathbf{X}]^{0.5}][(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{-0.5}\|_F\\
\le&\frac{\|\mathbf{Y}\|_F}{\sigma_M(\mathbf{X}) - \|\mathbf{Y}\|} + \|[[(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{0.5} - [\mathbf{X}^T\mathbf{X}]^{0.5}]\|_F\|[(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{-0.5}\|\\
\le&\frac{\|\mathbf{Y}\|_F}{\sigma_M(\mathbf{X}) - \|\mathbf{Y}\|} + \sqrt{2}\|\mathbf{Y}\|_F\|[(\mathbf{X}+\mathbf{Y})^T(\mathbf{X}+\mathbf{Y})]^{-0.5}\|\\
\le&\frac{(1+\sqrt{2})\|\mathbf{Y}\|_F}{\sigma_M(\mathbf{X}) - \|\mathbf{Y}\|}.
\end{aligned}$$

Here the first and the second inequalities follow from $\|\mathbf{AB}\|_F \le \|\mathbf{A}\|_F\|\mathbf{B}\|$, and the third inequality follows from (Bhatia, 1997, (VII.39)).

□

Proof of Lemma 9. Let $\mathbf{U} \in \mathbb{R}^{n\times r}$ be the orthogonal matrix that has the same column space as $\mathbf{X}_0$, then

$$\|\Pi_T \mathbf{\Delta}\|_F = \|\Pi_{\mathbf{U}^\perp} \mathbf{\Delta} \Pi_{\mathbf{U}} + \Pi_{\mathbf{U}} \mathbf{\Delta}\|_F \geq \|\Pi_{\mathbf{U}^\perp} \mathbf{\Delta} \Pi_{\mathbf{U}}\|_F = \|\Pi_{\mathbf{U}^\perp} \mathbf{\Delta} \mathbf{U}\|_F,$$

and

$$\Pi_{T,\perp} \mathbf{\Delta} = \Pi_{\mathbf{U}^\perp} \mathbf{\Delta} \mathbf{U} (\mathbf{U}^T (\mathbf{X}_0 + \mathbf{\Delta}) \mathbf{U})^{-1} \mathbf{U}^T \mathbf{\Delta}^T \Pi_{\mathbf{U}^\perp}.$$

As a result,

$$\|\Pi_{T,\perp} \mathbf{\Delta}\|_F \leq \frac{\|\Pi_{\mathbf{U}^\perp} \mathbf{\Delta} \mathbf{U}\|_F^2}{\sigma_r(\mathbf{U}^T(\mathbf{X}_0 + \mathbf{\Delta})\mathbf{U})} \leq \frac{\|\Pi_T \mathbf{\Delta}\|_F^2}{\sigma_r(\mathbf{X}_0) - \|\mathbf{\Delta}\|}$$

□

References

Emmanuel Abbe. Community detection and stochastic block models: Recent developments. Journal of Machine Learning Research, 18(177):1–86, 2018. URL http://jmlr.org/papers/v18/16-480.html.

P.-A. Absil and I. V. Oseledets. Low-rank retractions: a survey and new results. Computational Optimization and Applications, 62(1):5–29, Sep 2015. ISSN 1573-2894. doi: 10.1007/s10589-014-9714-4. URL https://doi.org/10.1007/s10589-014-9714-4.

P.A. Absil, R. Mahony, and R. Sepulchre. Optimization Algorithms on Matrix Manifolds. Princeton University Press, 2009. ISBN 9781400830244. URL https://books.google.com/books?id=NSQGQeLN3NcC.

Alberto Aleta and Yamir Moreno. Multilayer networks in a nutshell. Annual Review of Condensed Matter Physics, 10(1):45–62, Mar 2019. ISSN 1947-5462. doi: 10.1146/annurev-conmatphys-031218-013259. URL http://dx.doi.org/10.1146/annurev-conmatphys-031218-013259.

Rajendra Bhatia. Matrix Analysis. Number 169 in Graduate Texts in Mathematics. Springer, New York, 1997.

Monika Bhattacharjee, Moulinath Banerjee, and George Michailidis. Change point estimation in a dynamic stochastic block model. ArXiv:1812.03090, 2018.

Sharmodeep Bhattacharyya and Shirshendu Chatterjee. General community detection with optimal recovery conditions for multi-relational sparse networks with dependent layers, 2020.

W.M. Boothby and W.M. Boothby. An Introduction to Differentiable Manifolds and Riemannian Geometry, Revised. Pure and Applied Mathematics. Elsevier Science, 2003. ISBN 9780121160517. URL https://books.google.com/books?id=DFYs99E-IFYC.

Piotr Brodka, Anna Chmiel, Matteo Magnani, and Giancarlo Ragozini. Quantifying layer similarity in multiplex networks: a systematic study. Royal Society Open Science, 5(8): 171747, 2018. doi: 10.1098/rsos.171747. URL https://royalsocietypublishing.org/doi/abs/10.1098/rsos.171747.

Randy L. Buckner and Lauren M. DiNicola. The brain's default network: updated anatomy, physiology and evolving insights. Nature Reviews Neuroscience, pages 1–16, 2019.

Xiaobo Chen, Han Zhang, Yue Gao, Chong-Yaw Wee, Gang Li, Dinggang Shen, and the Alzheimer's Disease Neuroimaging Initiative. High-order resting-state functional connectivity network for mci classification. Human Brain Mapping, 37(9):3282–3296, 2016. doi: 10.1002/hbm.23240. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/hbm.23240.

Eric C. Chi, Brian J. Gaines, Will Wei Sun, Hua Zhou, and Jian Yang. Provable convex co-clustering of tensors. Journal of Machine Learning Research, 21(214):1–58, 2020. URL http://jmlr.org/papers/v21/18-155.html.

Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora. Structural reducibility of multilayer networks. Nature Communications, 6(1):6864, Apr 2015. ISSN 2041-1723. doi: 10.1038/ncomms7864. URL https://doi.org/10.1038/ncomms7864.

Daniele Durante, Nabanita Mukherjee, and Rebecca C. Steorts. Bayesian learning of dynamic multilayer networks. Journal of Machine Learning Research, 18(43):1–29, 2017. URL http://jmlr.org/papers/v18/16-391.html.

Alan Edelman, Tomás A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints. SIAM Journal on Matrix Analysis and Applications, 20(2): 303–353, 1998. doi: 10.1137/S0895479895290954. URL https://doi.org/10.1137/S0895479895290954.

Jean Gallier. Basics of Classical Lie Groups: The Exponential Map, Lie Groups, and Lie Algebras, pages 367–414. Springer New York, New York, NY, 2001. ISBN 978-1-4613-0137-0. doi: 10.1007/978-1-4613-0137-0_14. URL https://doi.org/10.1007/978-1-4613-0137-0_14.

Aditya Gangrade, Praveen Venkatesh, Bobak Nazer, and Venkatesh Saligrama. Testing changes in communities for the stochastic block model. ArXiv:1812.00769, 2018.

G.H. Golub and C.F. Van Loan. Matrix Computations. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 2013. ISBN 9781421407944. URL https://books.google.it/books?id=X5YfsuCWpxMC.

John C. Gower and Garmt B. Dijksterhuis. Procrustes problems, volume 30 of Oxford Statistical Science Series. Oxford University Press, Oxford, UK, January 2004. URL http://oro.open.ac.uk/2736/.

Rungang Han, Yuetian Luo, Miaoyan Wang, and Anru R. Zhang. Exact clustering in tensor block model: Statistical optimality and computational limit, 2021.

Shaobo Han and David B. Dunson. Multiresolution tensor decomposition for multiple spatial passing networks. ArXiv:1803.01203, 2018.

Xuelei Hu and Lei Xu. A comparative study of several cluster number selection criteria. In Jiming Liu, Yiu-ming Cheung, and Hujun Yin, editors, Intelligent Data Engineering and Automated Learning, pages 195–202, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. ISBN 978-3-540-45080-1.

Bing-Yi Jing, Ting Li, Zhongyuan Lyu, and Dong Xia. Community detection on mixture multilayer networks via regularized tensor decomposition. The Annals of Statistics, 49 (6):3181 – 3205, 2021. doi: 10.1214/21-AOS2079. URL https://doi.org/10.1214/21-AOS2079.

Ta-Chu Kao and Mason A. Porter. Layer communities in multiplex networks. Journal of Statistical Physics, 173(3-4):1286–1302, Aug 2017. ISSN 1572-9613. doi: 10.1007/s10955-017-1858-z. URL http://dx.doi.org/10.1007/s10955-017-1858-z.

Mikko Kivela, Alex Arenas, Marc Barthelemy, James P. Gleeson, Yamir Moreno, and Mason A. Porter. Multilayer networks. Journal of Complex Networks, 2(3):203–271, 07 2014. ISSN 2051-1329. doi: 10.1093/comnet/cnu016. URL https://doi.org/10.1093/comnet/cnu016.

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

A. Kumar, Y. Sabharwal, and S. Sen. A simple linear time (1 + epsiv;)-approximation algorithm for k-means clustering in any dimensions. In 45th Annual IEEE Symposium on Foundations of Computer Science, pages 454–462, Oct 2004a. doi: 10.1109/FOCS.2004.7.

Amit Kumar, Yogish Sabharwal, and Sandeep Sen. A simple linear time $(1+\epsilon)$-approximation algorithm for k-means clustering in any dimensions. pages 454–462, 01 2004b. doi: 10.1109/FOCS.2004.7.

Can M. Le and Elizaveta Levina. Estimating the number of communities in networks by spectral methods. jul 2015. URL http://arxiv.org/abs/1507.00827.

Jing Lei and Kevin Z. Lin. Bias-adjusted spectral clustering in multi-layer stochastic block models, 2020.

Jing Lei and Alessandro Rinaldo. Consistency of spectral clustering in stochastic block models. Ann. Statist., 43(1):215–237, 02 2015. doi: 10.1214/14-AOS1274. URL http://dx.doi.org/10.1214/14-AOS1274.

Jing Lei, Kehui Chen, and Brian Lynch. Consistent community detection in multi-layer network data. Biometrika, 107(1):61–73, 12 2019. ISSN 0006-3444. doi: 10.1093/biomet/asz068. URL https://doi.org/10.1093/biomet/asz068.

Gilad Lerman and Teng Zhang. lp-recovery of the most significant subspace among multiple subspaces with outliers. Constructive Approximation, 40(3):329–385, December 2014. ISSN 0176-4276. doi: 10.1007/s00365-014-9242-6.

Peter W. MacDonald, Elizaveta Levina, and Ji Zhu. Latent space models for multiplex networks with shared structure, 2021.

Pedro Mercado, Antoine Gautier, Francesco Tudisco, and Matthias Hein. The power mean laplacian for multilayer graph clustering. ArXiv:1803.00491, 2018.

B.C. Munsell, C.-Y. Wee, S.S. Keller, B. Weber, C. Elger, L.A.T. da Silva, T. Nesland, M. Styner, D. Shen, and L. Bonilha. Evaluation of machine learning algorithms for treatment outcome prediction in patients with epilepsy based on structural connectome data. NeuroImage, 118:219–230, 2015.

Subhadeep Paul and Yuguo Chen. Consistent community detection in multi-relational data through restricted multi-layer stochastic blockmodel. Electron. J. Statist., 10(2):3807–3870, 2016. doi: 10.1214/16-EJS1211. URL https://doi.org/10.1214/16-EJS1211.

Subhadeep Paul and Yuguo Chen. Spectral and matrix factorization methods for consistent community detection in multi-layer networks. Ann. Statist., 48(1):230–250, 02 2020. doi: 10.1214/18-AOS1800. URL https://doi.org/10.1214/18-AOS1800.

Marianna Pensky and Teng Zhang. Spectral clustering in the dynamic stochastic block model. Electronic Journal of Statistics, 13(1):678 – 709, 2019. doi: 10.1214/19-EJS1533. URL https://doi.org/10.1214/19-EJS1533.

Cornelis J. Stam. Modern network science of neurological disorders. Nature Reviews Neuroscience, 15(10):683–695, 2014. doi: 10.1038/nrn3801. URL https://app.dimensions.ai/details/publication/pub.1037745277.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63(2):411–423, 2001. doi: https://doi.org/10.1111/1467-9868.00293. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/1467-9868.00293.

Daren Wang, Yi Yu, and Alessandro Rinaldo. Optimal covariance change point localization in high dimension. ArXiv:1712.09912, 2017.

Junhui Wang. Consistent selection of the number of clusters via crossvalidation. Biometrika, 97(4):893–904, 12 2010. ISSN 0006-3444. doi: 10.1093/biomet/asq061. URL https://doi.org/10.1093/biomet/asq061.

Miaoyan Wang and Yuchen Zeng. Multiway clustering via tensor block models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/9be40cee5b0eee1462c82c6964087ff9-Paper.pdf.

Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. BIT Numerical Mathematics, 12(1):99–111, Mar 1972. ISSN 1572-9125. doi: 10.1007/BF01932678. URL https://doi.org/10.1007/BF01932678.

Tao Wu, Austin R Benson, and David F Gleich. General tensor spectral co-clustering for higher-order data. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/fe51510c80bfd6e5d78a164cd5b1f688-Paper.pdf.

Teng Zhang, Arthur Szlam, Yi Wang, and Gilad Lerman. Hybrid linear modeling via local best-fit flats. International Journal of Computer Vision, 100(3):217–240, Dec 2012. ISSN 1573-1405. doi: 10.1007/s11263-012-0535-6. URL https://doi.org/10.1007/s11263-012-0535-6.

Zhixin Zhou and Yizhe Zhu. Sparse random tensors: concentration, regularization and applications, 2019.