Journal of Machine Learning Research 27 (2026) 1-73 Submitted 4/24; Revised 2/26; Published 2/26

Two-way Node Popularity Model for Directed and Bipartite Networks

Bing-Yi Jing * BINGYIJING@CUHK.EDU.CN
School of Artificial Intelligence
The Chinese University of Hong Kong, Shenzhen
Shenzhen Loop Area Institute
Shenzhen 518172*, China*

Ting Li *,† LITING@SUSTECH.EDU.CN
Department of Statistics and Data Science
Southern University of Science and Technology
Shenzhen 518055*, China*

Jiangzhou Wang *,† WANGJZ695@SZU.EDU.CN
School of Mathematical Sciences, Institute of Statistical Sciences
Shenzhen University
Shenzhen 518060*, China*

Ya Wang *,† WANGYA@SUSTECH.EDU.CN
Department of Statistics and Data Science
Southern University of Science and Technology
Shenzhen 518055*, China*

Editor: Jie Peng

Abstract

There has been increasing research attention on community detection in directed and bipartite networks. However, these studies often fail to consider the popularity of nodes in different communities, which is a common phenomenon in real-world networks. To address this issue, we propose a new probabilistic framework called the Two-Way Node Popularity Model (TNPM). The TNPM also accommodates edges from different distributions within a general sub-Gaussian family. We introduce the Delete-One-Method (DOM) for model fitting and community structure identification, and provide a comprehensive theoretical analysis with novel technical skills dealing with sub-Gaussian generalization. Additionally, we propose the Two-Stage Divided Cosine Algorithm (TSDC) to handle large-scale networks more efficiently. Our proposed methods offer multi-folded advantages in terms of estimation accuracy and computational efficiency, as demonstrated through extensive numerical studies. We apply our methods to two real-world applications, uncovering interesting findings.

Keywords: Bipartite network, Community detection, Directed network, Node popularity

1. Introduction

Community detection is a valuable tool for understanding the structure of a network and has been applied in various fields, including biology (Calderer and Kuijjer, 2021; Li et al., 2021), social science (Wu et al., 2020; Jing et al., 2022), and global trading analysis (Jing et al., 2021). While

∗. The authors are listed in alphabetical order and all authors contributed equally to this work.
†. Co-Corresponding authors.

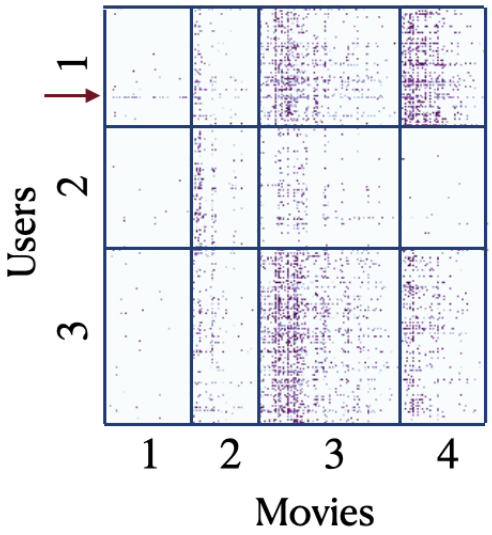


Figure 1: Adjacency matrix of the MovieLens 100K dataset, reordered according to the clusters identified by the proposed TSDC method. Blue lines denote the boundaries between clusters.

several models have been proposed for community detection in undirected networks, the study of community detection in directed and bipartite networks is relatively limited.

One reason is that directed networks are more complex than undirected ones as they involve both outgoing and incoming links. Therefore, traditional definitions of clustering problems, such as intra-cluster and inter-cluster edge density, cannot be extended directly to directed networks (Zhang et al., 2021). Bipartite networks, on the other hand, have nodes divided into two sets and edges that only connect nodes from different sets. This feature violates the assumption of symmetric relationships between nodes in undirected scenarios.

To address this gap, several studies have been conducted. The pseudo-likelihood approach has been used to identify out- and in-community structures in Amini et al. (2013) and Wang et al. (2021). Rohe et al. (2016) proposes the Stochastic co-Blockmodel (ScBM) and its extension, the Degree Corrected ScBM (DC-ScBM), which considers degree heterogeneity to model directed networks. Wang et al. (2020) analyzes the theoretical guarantees for the algorithm D-SCORE (Ji and Jin, 2016) and its variants designed under DC-ScBM. Zhou and Amini (2019) studies spectral clustering algorithms designed by a data-driven regularization of the adjacency matrix under ScBM. In Zhang et al. (2021), authors embed nodes with concentration restrictions to help identify communities.

However, all of the above methods overlook the heterogeneous popularities of nodes across different communities. Such structure has been widely observed and discussed in undirected networks (Sengupta and Chen, 2018; Noroozi et al., 2021b). Addressing this, Sengupta and Chen (2018) proposes the Popularity Adjusted Stochastic Block Model (PABM), which models the edge probability between two nodes as a product of node popularity parameters. PABM provides a flexible way of modeling the probability of connections and allows nodes in the same community to exhibit heterogeneous popularities across different communities.

Diverse node popularity patterns are also widely present in directed or bipartite scenarios. One motivating example is the MovieLens 100K dataset (Harper and Konstan, 2015), which depicts a bipartite network where entries represent user-to-movie ratings. Illustrated in Figure 1, the adjacency matrix is organized according to clustering results, with cluster boundaries marked by blue lines. This visualization reveals that group 1 users predominantly prefer movies in categories 3 and 4, though there's limited but notable interest in categories 1 and 2, with a few exceptions (highlighted by a red arrow) showing curiosity in category 1. Such diversity in node popularities reveals various consumer patterns and contributes to understanding the different behaviors of users with respect to movies from different categories.

Existing algorithms for undirected networks are not easily extendable to directed and bipartite scenarios. For example, the algorithm proposed in Sengupta and Chen (2018) is limited to networks with a small number of communities (less than 3). On the other hand, while the Sparse Subspace Clustering (SSC) algorithm proposed in Noroozi et al. (2021b) can handle large-scale networks, it is sensitive to noise, as demonstrated by the simulation results in Section 5. Moreover, our numerical studies indicate that naively applying these methods to directed and bipartite networks often leads to poor performance.

In this paper, we introduce the Two-Way Node Popularity Model (TNPM), a comprehensive probabilistic framework designed to model directed and bipartite networks with community structures and node popularities. Moreover, the TNPM allows each link to be generated from different distributions within the sub-Gaussian family. The new model presents significant challenges in model fitting due to the use of two distinct sets of scaling parameters to characterize node popularities for the out- and in-communities separately. Our main contributions are summarized as follows:

- We propose the TNPM to model directed and bipartite networks with community structures and node popularities. The model also generalizes link distributions to the sub-Gaussian family, making it applicable to binary, discrete-valued, and weighted networks.
- To fit the model, we develop two estimation procedures: (i) the Delete-One-Method (DOM), an alternating optimization algorithm, and (ii) the Two-Stage Divided Cosine Algorithm (TSDC), a computationally lightweight community detection method built on a novel Block Cosine Similarity. Empirically, both outperform state-of-the-art baselines. Furthermore, to enhance scalability, we embed DOM within popular subsampling frameworks, enabling divide-and-conquer inference on large networks.
- We establish theoretical guarantee under the TNPM. For the DOM, we prove the consistency of the estimated mean matrix and the recovered out- and in-community labels. We also derive the error bounds for the subsampled DOM. For the TSDC, we show that it admits at least one Pure Strategy Nash Equilibrium (PSNE) and that the true community structure forms a PSNE with high probability, providing a theoretical justification for the algorithm.

To the best of our knowledge, this is the first systematic study of directed and bipartite networks that considers node popularities, and the study on sub-Gaussian generalization might be of independent interest.

The rest of the paper is organized as follows. Section 2 introduces the Two-Way Node Popularity Model (TNPM). In Section 3, we propose the Delete-One-Method (DOM) and the Two-Stage Divided Cosine Algorithm (TSDC). We explore the theoretical properties in Section 4. Section 5 and 6 present extensive simulations and real data applications to demonstrate the advantages of the proposed methods.

2. Two-way Node Popularity Model

Consider a bipartite network $\mathcal{G}(\mathcal{N}, \mathcal{M}, E)$, where $\mathcal{N}$ and $\mathcal{M}$ are two disjoint sets of nodes indexed by $1, \ldots, n$ and $1, \ldots, m$, respectively, and $E \subseteq \mathcal{N} \times \mathcal{M}$ denotes the set of edges connecting nodes exclusively between the two sets. Notably, directed networks can be viewed as a special case of bipartite networks when $\mathcal{N} = \mathcal{M}$, with edges interpreted as directed links between nodes. Therefore, we adopt bipartite networks as our primary modeling framework throughout this paper, noting that all proposed algorithms and theoretical results naturally extend to the directed setting.

Let $A = (A_{ij})_{i,j=1}^{n,m}$ denote the adjacency matrix of the network, where A_{ij} represents the weight of the edge from node i in set $\mathcal{N}$ to node j in set $\mathcal{M}$. We use $A_{i\cdot}$ to denote the i-th row of matrix A, and $A_{\cdot j}$ to denote the j-th column. The community structure corresponding to nodes in set $\mathcal{N}$ is referred to as the out-community, while that of nodes in set $\mathcal{M}$ is referred to as the in-community. Let K and L denote the number of out-communities and in-communities, respectively. The distinct community blocks are denoted by $\mathcal{N}_k$ and $\mathcal{M}_\ell$ for all $k = 1, \ldots, K$ and $\ell = 1, \ldots, L$.

For brevity, we introduce the following notations. For any set Ω, let $|\Omega|$ denote its cardinality. For any numbers a and b, define $a \wedge b = \min(a, b)$, and let $\lfloor a \rfloor$ denote the greatest integer less than or equal to a. For any integer I, define $[I] = \{1, 2, \ldots, I\}$. We define $n_k = |\mathcal{N}_k|$ and $m_\ell = |\mathcal{M}_\ell|$ for all $k \in [K]$ and $\ell \in [L]$. For any two column (or row) vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, define the inner product as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^\top \boldsymbol{y}$ (or $\boldsymbol{x}\boldsymbol{y}^\top$), and define the cosine similarity as

$$\cos(\boldsymbol{x}, \boldsymbol{y}) = \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\|\boldsymbol{x}\|_2 \, \|\boldsymbol{y}\|_2},$$

where $\|\cdot\|_2$ denotes the Euclidean norm. Furthermore, let $\boldsymbol{c} \in [K]^n$ and $\boldsymbol{z} \in [L]^m$ denote the vectors of out-community and in-community assignments, respectively, such that $c_i = k$ if and only if node $i \in \mathcal{N}_k$, and $z_j = \ell$ if and only if node $j \in \mathcal{M}_\ell$. Let $\mathcal{N}_{n,K}$ and $\mathcal{M}_{m,L}$ denote the sets of clustering matrices $C \in \{0,1\}^{n \times K}$ and $Z \in \{0,1\}^{m \times L}$, respectively, where $C_{ik} = 1$ if and only if $i \in \mathcal{N}_k$, and $Z_{j\ell} = 1$ if and only if $j \in \mathcal{M}_\ell$.

To capture node popularity separately in rows and columns, we propose the *Two-Way Node Popularity Model (TNPM)*:

$$P_{ij} \triangleq \mathbb{E}\,(A_{ij}) = V_{i\boldsymbol{z}_j} \widetilde{V}_{j\boldsymbol{c}_i}, \tag{1}$$

where $V_{i\ell}$ for $i \in [n], \ell \in [L]$, and $\widetilde{V}_{jk}$ for $j \in [m],\ k \in [K]$ represent node popularity parameters. Specifically, $V_{i\ell}$ quantifies the popularity of out-node i within the in-community $\mathcal{M}_\ell$, while $\widetilde{V}_{jk}$ captures the popularity of in-node j within the out-community $\mathcal{N}_k$. Given the community assignments $\boldsymbol{c}$ and $\boldsymbol{z}$, the entries A_{ij} are assumed to be mutually independent and sub-Gaussian with mean $\mathbb{E}(A_{ij}) = P_{ij}$. This framework accommodates a wide range of distributions commonly encountered in real data, such as Bernoulli, Binomial, and Gaussian, without requiring prior specification of a particular distributional form.

The expectation of the adjacency matrix A under the TNPM model exhibits a block-wise rank-one structure, which forms the core foundation for constructing algorithms to fit TNPM. Specifically, consider a rearranged version $P(\boldsymbol{c}, \boldsymbol{z})$ of the matrix P, where rows and columns are ordered according to community memberships. For example, nodes in out- (or in-) community 1 occupy the first n_1 (or m_1) rows (or columns), nodes in community 2 occupy the next n_2 (or m_2), and so forth. We denote the (k, ℓ)-th block of $P(\boldsymbol{c}, \boldsymbol{z})$ as $P^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z})$. This submatrix $P^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z}) \in \mathbb{R}^{n_k \times m_\ell}$ corresponds to node interactions between the community pair $(\mathcal{N}_k, \mathcal{M}_\ell)$. Each entry of this block

can be written as

$$P_{ij}^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z}) = V_{i_k\ell}\widetilde{V}_{j_\ell k},$$

where i_k and j_ℓ denote the i-th and j-th elements of $\mathcal{N}_k$ and $\mathcal{M}_\ell$, respectively, with $i \in [n_k]$ and $j \in [m_\ell]$. As a result, each block $P^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})$ is a rank-one matrix, and it can be expressed as

$$P^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z}) = \Lambda^{(k,\ell)}\left[\widetilde{\Lambda}^{(\ell,k)}\right]^T, \tag{2}$$

where the vectors $\Lambda^{(k,\ell)}$ and $\widetilde{\Lambda}^{(\ell,k)}$ are defined element-wise as

$$\Lambda_i^{(k,\ell)} = V_{i_k\ell} \quad \text{for } i \in [n_k], \qquad \widetilde{\Lambda}_j^{(\ell,k)} = \widetilde{\Lambda}_{j_\ell k} \quad \text{for } j \in [m_\ell].$$

Accordingly, the entire matrix $P(\boldsymbol{c},\boldsymbol{z})$ can be written in block form as

$$P(\boldsymbol{c},\boldsymbol{z}) = \begin{bmatrix} \Lambda^{(1,1)}\left(\widetilde{\Lambda}^{(1,1)}\right)^T & \Lambda^{(1,2)}\left(\widetilde{\Lambda}^{(2,1)}\right)^T & \cdots & \Lambda^{(1,L)}\left(\widetilde{\Lambda}^{(L,1)}\right)^T \\ \Lambda^{(2,1)}\left(\widetilde{\Lambda}^{(1,2)}\right)^T & \Lambda^{(2,2)}\left(\widetilde{\Lambda}^{(2,2)}\right)^T & \cdots & \Lambda^{(2,L)}\left(\widetilde{\Lambda}^{(L,2)}\right)^T \\ \vdots & \vdots & \ddots & \vdots \\ \Lambda^{(K,1)}\left(\widetilde{\Lambda}^{(1,K)}\right)^T & \Lambda^{(K,2)}\left(\widetilde{\Lambda}^{(2,K)}\right)^T & \cdots & \Lambda^{(K,L)}\left(\widetilde{\Lambda}^{(L,K)}\right)^T \end{bmatrix}. \tag{3}$$

Rank-one structures have also been identified in undirected networks (Noroozi et al., 2021a,b). However, detecting these structures without prior knowledge of node memberships remains a significant challenge, particularly when the values of K and/or L are large. Direct application of existing methods often leads to suboptimal fitting performance and high computational costs, as demonstrated in our empirical studies.

Remark 1. *The TNPM develops the PABM to bipartite and directed networks. Koo et al. (2023) showed that PABM is a special case of the Generalized Random Dot Product Graph (GRDPG, Rubin-Delanchy et al., 2022). A natural question is how TNPM relates to GRDPG or the Random Dot Product Graph (RDPG, Young and Scheinerman, 2007). Under TNPM, the mean matrix can be reformulated as $P = XY^\top$ (see Appendix A.6 for a detailed derivation), with X and Y denoting row and column latent vectors, whereas RDPG and GRDPG correspond to $P = XX^\top$ and $P = XI_{p,q}X^\top$, respectively, with $I_{p,q}$ defined in Rubin-Delanchy et al. (2022). Thus, while sharing the latent-space spirit of RDPG/GRDPG and the popularity adjustment of PABM, TNPM remains distinct and broadens these frameworks to bipartite and directed settings.*

3. Methodology

In this section, we introduce the Delete-One-Method (DOM), its subsampled version and the Two-Stage Divided Cosine Algorithm (TSDC) for model fitting and community detection under TNPM.

3.1 The Delete-One-Method

Motivated by the observed block rank-one structure in equation (3), we propose the following objective function:

$$\text{Loss}(\boldsymbol{c},\boldsymbol{z},\Lambda,\widetilde{\Lambda},K,L) = \sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z}) - \Lambda^{(k,\ell)}\left[\widetilde{\Lambda}^{(\ell,k)}\right]^T\right\|_F^2, \tag{4}$$

where $A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})$ denotes the (k,ℓ)-th block of the matrix $A(\boldsymbol{c},\boldsymbol{z})$, which is the rearranged version of A according to the clustering assignments $\boldsymbol{c}$ and $\boldsymbol{z}$.

To address the identifiability issue in recovering $\Lambda^{(k,\ell)}$ and $\widetilde{\Lambda}^{(\ell,k)}$, we introduce $\Theta^{(k,\ell)}$ as the notation for $\Lambda^{(k,\ell)}\left[\widetilde{\Lambda}^{(\ell,k)}\right]^T$ and focus on estimating the uniquely defined rank-one matrix $\Theta^{(k,\ell)}$. Additionally, since the numbers of communities K and L are typically unknown, we follow the approach in Noroozi et al. (2021a,b) and introduce a penalty on K and L to prevent overfitting by selecting too many communities. Consequently, our optimization problem can be formulated as:

$$\left(\widehat{\Theta},\hat{\boldsymbol{c}},\hat{\boldsymbol{z}},\widehat{K},\widehat{L}\right)=\underset{\Theta,\boldsymbol{c},\boldsymbol{z},K,L}{\arg\min}\left\{\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})-\Theta^{(k,\ell)}\right\|_F^2+\text{Pen}(n,m,K,L)\right\} \tag{5}$$

$$\text{s.t.}\quad \text{rank}\left(\Theta^{(k,\ell)}\right)=1\quad \text{for all } k\in[K],\ \ell\in[L],$$

where $\widehat{\Theta}$ is the block matrix with components $\widehat{\Theta}^{(k,\ell)}$, and $\text{Pen}(n,m,K,L)$ is the penalty term, which will be defined in Section 4.2.

If $\hat{\boldsymbol{c}}$, $\hat{\boldsymbol{z}}$, $\widehat{K}$, and $\widehat{L}$ were known, the optimal solution to problem (5) would be given by the rank-one approximations $\widehat{\Theta}^{(k,\ell)}$ of the submatrices $A^{(k,\ell)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})$. These approximations can be expressed as:

$$\widehat{\Theta}^{(k,\ell)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})=\Pi_{\hat{u},\hat{v}}\left(A^{(k,\ell)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})\right)=\hat{\sigma}_1^{(k,\ell)}\hat{u}^{(k,\ell)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})\left(\hat{v}^{(\ell,k)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})\right)^T, \tag{6}$$

where $\hat{\sigma}_1^{(k,\ell)}$ is the largest singular value of $A^{(k,\ell)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})$, and $\hat{u}^{(k,\ell)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})$, $\hat{v}^{(\ell,k)}(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})$ are the corresponding singular vectors. The operator $\Pi(\cdot)$ denotes the rank-one projection. Substituting (6) into (5), the optimization problem becomes:

$$\left(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}},\widehat{K},\widehat{L}\right)=\underset{\boldsymbol{c},\boldsymbol{z},K,L}{\arg\min}\left\{\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})-\widehat{\Theta}^{(k,\ell)}\right\|_F^2+\text{Pen}(n,m,K,L)\right\}. \tag{7}$$

To obtain $(\hat{\boldsymbol{c}},\hat{\boldsymbol{z}})$, one must solve (7) for each pair (K,L), resulting in:

$$(\hat{\boldsymbol{c}}_K,\hat{\boldsymbol{z}}_L)=\underset{\substack{\boldsymbol{c}\in[K]^n\\ \boldsymbol{z}\in[L]^m}}{\arg\min}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})-\Pi_{\hat{u},\hat{v}}\left(A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})\right)\right\|_F^2, \tag{8}$$

and then determine $\hat{K}$ and $\hat{L}$ as:

$$(\widehat{K},\widehat{L})=\arg\min_{K,L}\left\{\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}\left(\hat{\boldsymbol{c}}_K,\hat{\boldsymbol{z}}_L\right)-\Pi_{\hat{u},\hat{v}}\left(A^{(k,\ell)}\left(\hat{\boldsymbol{c}}_K,\hat{\boldsymbol{z}}_L\right)\right)\right\|_F^2+\text{Pen}(n,m,K,L)\right\}. \tag{9}$$

We now consider the optimization of problem (8) given fixed values of K and L. For notational simplicity, we denote

$$\text{Loss}(\boldsymbol{c},\boldsymbol{z})=\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})-\Pi_{\hat{u},\hat{v}}\left(A^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z})\right)\right\|_F^2$$

in the following discussion. The optimization problem (8) constitutes the core component of the fitting algorithm, and solving it is often NP-hard. Therefore, developing efficient approximation methods for optimizing (8) is of critical importance. To address this challenge, we propose an alternating update algorithm and integrate a Delete-One-Method (DOM) within the iteration process, resulting in a significant reduction in computational complexity.

Specifically, at the t-th iteration, given the pair $\left(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)}\right)$, we update $\boldsymbol{c}$ and $\boldsymbol{z}$ alternately to obtain $\boldsymbol{c}^{(t+1)}$ and $\boldsymbol{z}^{(t+1)}$, respectively. In detail, given $\boldsymbol{c}^{(t)}$ and $\boldsymbol{z}^{(t)}$, the sub-optimization for updating $\boldsymbol{c}$ can be written as:

$$\boldsymbol{c}^{(t+1)} = \underset{\boldsymbol{c}\in[K]^n}{\arg\min} \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z}^{(t)}) - \Pi_{\hat{u},\hat{v}}(A^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z}^{(t)})) \right\|_F^2. \tag{10}$$

In particular, for each $i \in [n]$, we have:

$$c_i^{(t+1)} = \underset{\tilde{\boldsymbol{c}}_i^{(t)}\in[K]}{\arg\min} \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A^{(k,\ell)}(\tilde{\boldsymbol{c}}^{(t)}, \boldsymbol{z}^{(t)}) - \Pi_{\hat{u},\hat{v}}(A^{(k,\ell)}(\tilde{\boldsymbol{c}}^{(t)}, \boldsymbol{z}^{(t)})) \right\|_F^2, \tag{11}$$

where $\tilde{c}_p^{(t)} = c_p^{(t)}$ for all $p \neq i$.

The optimization problem in (11) is computationally intensive, as it requires evaluating the Frobenius norm for each block. To mitigate this, we reformulate the objective by leveraging a leave-one-out strategy. Specifically, the objective can be further expressed as:

$$\begin{aligned} c_i^{(t+1)} = \underset{\tilde{\boldsymbol{c}}_i^{(t)}\in[K]}{\arg\min} \Bigg\{ & \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A_{(\mathcal{N}_k,\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})} - \Pi_{\hat{u},\hat{v}}(A_{(\mathcal{N}_k,\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})}) \right\|_F^2 \\ & - \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A_{(\mathcal{N}_k\setminus\{i\},\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})} - \Pi_{\hat{u},\hat{v}}(A_{(\mathcal{N}_k\setminus\{i\},\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})}) \right\|_F^2 \Bigg\}. \end{aligned} \tag{12}$$

Here, $A_{(\mathcal{N}_k,\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})}$ denotes the (k,ℓ)-th block of the matrix based on the current community assignments $\tilde{\boldsymbol{c}}^{(t)}$ and $\boldsymbol{z}^{(t)}$. This notation facilitates the delete-one strategy by enabling direct manipulation of matrix sub-blocks. Thus, the simplified expression becomes:

$$\begin{aligned} c_i^{(t+1)} = \underset{\tilde{\boldsymbol{c}}_i^{(t)}\in[K]}{\arg\min} \Bigg\{ & \sum_{\ell=1}^{L} \left\| A_{(\mathcal{N}_{\tilde{c}_i^{(t)}},\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})} - \Pi_{\hat{u},\hat{v}}(A_{(\mathcal{N}_{\tilde{c}_i^{(t)}},\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})}) \right\|_F^2 \\ & - \sum_{\ell=1}^{L} \left\| A_{(\mathcal{N}_{\tilde{c}_i^{(t)}}\setminus\{i\},\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})} - \Pi_{\hat{u},\hat{v}}(A_{(\mathcal{N}_{\tilde{c}_i^{(t)}}\setminus\{i\},\mathcal{M}_\ell)}^{(\tilde{\boldsymbol{c}}^{(t)},\boldsymbol{z}^{(t)})}) \right\|_F^2 \Bigg\}. \end{aligned} \tag{13}$$

Similarly, the sub-optimization for updating $\boldsymbol{z}$ can be described as:

$$\begin{aligned} z_j^{(t+1)} = \underset{\tilde{z}_j\in[L]}{\arg\min} \Bigg\{ & \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A_{(\mathcal{N}_k,\mathcal{M}_\ell)}^{(\boldsymbol{c}^{(t+1)},\tilde{\boldsymbol{z}}^{(t)})} - \Pi_{\hat{u},\hat{v}}(A_{(\mathcal{N}_k,\mathcal{M}_\ell)}^{(\boldsymbol{c}^{(t+1)},\tilde{\boldsymbol{z}}^{(t)})}) \right\|_F^2 \\ & - \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A_{(\mathcal{N}_k,\mathcal{M}_\ell\setminus\{j\})}^{(\boldsymbol{c}^{(t+1)},\tilde{\boldsymbol{z}}^{(t)})} - \Pi_{\hat{u},\hat{v}}(A_{(\mathcal{N}_k,\mathcal{M}_\ell\setminus\{j\})}^{(\boldsymbol{c}^{(t+1)},\tilde{\boldsymbol{z}}^{(t)})}) \right\|_F^2 \Bigg\}. \end{aligned} \tag{14}$$

Consequently, for each $j \in [m]$, the update for $z_j^{(t+1)}$ is:

$$z_j^{(t+1)} = \underset{\tilde{z}_j^{(t)} \in [L]}{\arg\min} \left\{ \sum_{k=1}^{K} \left\| A^{(\boldsymbol{c}^{(t+1)}, \tilde{\boldsymbol{z}}^{(t)})}_{(\mathcal{N}_k, \mathcal{M}_{\tilde{z}_j^{(t)}})} - \Pi_{\hat{u},\hat{v}}(A^{(\boldsymbol{c}^{(t+1)}, \tilde{\boldsymbol{z}}^{(t)})}_{(\mathcal{N}_k, \mathcal{M}_{\tilde{z}_j^{(t)}})}) \right\|_F^2 \right. \\ \left. - \sum_{k=1}^{K} \left\| A^{(\boldsymbol{c}^{(t+1)}, \tilde{\boldsymbol{z}}^{(t)})}_{(\mathcal{N}_k, \mathcal{M}_{\tilde{z}_j^{(t)}} \setminus \{j\})} - \Pi_{\hat{u},\hat{v}}(A^{(\boldsymbol{c}^{(t+1)}, \tilde{\boldsymbol{z}}^{(t)})}_{(\mathcal{N}_k, \mathcal{M}_{\tilde{z}_j^{(t)}} \setminus \{j\})}) \right\|_F^2 \right\}. \tag{15}$$

The complete procedure of the DOM algorithm is summarized in the following Algorithm 1.

Algorithm 1 Delete-One-Method (DOM) Algorithm

Input: Adjacency matrix A, cluster numbers K, L, tolerance ϵ, max iterations $iter_{\max}$
Output: Final estimated community labels $\hat{\boldsymbol{c}} = \boldsymbol{c}^{(t)}$, $\hat{\boldsymbol{z}} = \boldsymbol{z}^{(t)}$

Initialize: $\left(\boldsymbol{c}^{(0)}, \boldsymbol{z}^{(0)}\right)$ and set $t = 0$
while $t < iter_{\max}$ **do**
  for $i = 1$ to n **do**
    Update $c_i^{(t+1)}$ via equation (13) with $\{c_1^{(t)}, \ldots, c_{i-1}^{(t)}\}$ replaced by $\{c_1^{(t+1)}, \ldots, c_{i-1}^{(t+1)}\}$
  end for
  for $j = 1$ to m **do**
    Update $z_j^{(t+1)}$ via equation (15) with $\{z_1^{(t)}, \ldots, z_{j-1}^{(t)}\}$ replaced by $\{z_1^{(t+1)}, \ldots, z_{j-1}^{(t+1)}\}$, and with $\boldsymbol{c}^{(t)}$ replaced by $\boldsymbol{c}^{(t+1)}$
  end for
  if $\frac{\left|\text{Loss}(\boldsymbol{c}^{(t+1)}, \boldsymbol{z}^{(t+1)}) - \text{Loss}(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)})\right|}{\text{Loss}(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)})} < \epsilon$ **then**
    break
  end if
  $t \leftarrow t + 1$
end while

In the initialization step of Algorithm 1, we employ a combination of singular value decomposition (SVD) and K-means clustering to derive the initial label vectors $\boldsymbol{c}^{(0)}$ and $\boldsymbol{z}^{(0)}$. Given the adjacency matrix A, we first perform SVD such that $A \approx X \Sigma Y^T$, where $X \in \mathbb{R}^{n \times K}$ and $Y \in \mathbb{R}^{m \times L}$ represent the extracted row and column features, respectively. We then apply K-means clustering to X and Y to obtain the initial row cluster labels $\boldsymbol{c}^{(0)}$ and column cluster labels $\boldsymbol{z}^{(0)}$. Simulation studies have shown that this initialization procedure provides more reliable results compared to other naive initialization strategies.

Remark 2. *In Algorithm 1, the change in the objective function is efficiently evaluated through a gain index, eliminating the need to recompute the entire objective function. Since many terms cancel out and can be omitted from the computation, the overall computational burden of the algorithm is significantly reduced. Similar idea is also adopted in undirected network analysis, for example, in Karrer and Newman (2011).*

Remark 3. *In Algorithm 1, the numbers of communities K and L are assumed to be known and provided as inputs. When they are unknown, one can estimate them using $(\widehat{K}, \widehat{L})$, obtained by solving the optimization problem* (9) *for each candidate pair (K, L), which can be equivalently*

written as

$$(\widehat{K}, \widehat{L}) = \arg\min_{K,L} \left\{ \|\widehat{P} - A\|_F^2 + \mathrm{Pen}(n, m, K, L) \right\}, \tag{16}$$

where $\widehat{P}$ denotes the estimated expectation of the network adjacency matrix.

3.2 Subsampled Version of DOM

The Delete-One-Method (DOM) algorithm has demonstrated effectiveness for community detection in medium-scale networks. For large-scale networks, we propose a divide-and-conquer strategy that enables community detection through subsampling.

This approach involves performing community detection on multiple small, randomly sampled subgraphs, followed by the aggregation of local clustering results into a global consensus. Several general-purpose subsampling-based algorithms, such as those proposed in Mukherjee et al. (2021); Chakrabarty et al. (2025), fall under this framework. These methods are flexible in that they can incorporate virtually any base community detection algorithm, including DOM, by dividing a large network into manageable subgraphs. In this subsection, we focus on integrating DOM with the Subsampling on Networks (SONNET) framework introduced in Chakrabarty et al. (2025). In addition, we also consider two other subsampling-based methods: the Piecewise Averaged Community Estimation (PACE) and the Global Alignment of Local Estimates (GALE), proposed by Mukherjee et al. (2021). Details of their integration with the DOM algorithm are provided in Appendix C. While initially developed for undirected networks, these methods are extended here to accommodate directed and bipartite networks.

The integration of DOM into the SONNET framework consists of four main stages: *division*, *detection*, *stitching*, and *repetition*.

Division Step: Let $\mathcal{N}$ and $\mathcal{M}$ denote the sets of out-nodes and in-nodes, respectively. These sets are divided into overlapping and disjoint subsets. Specifically, $o^{(out)}$ nodes from $\mathcal{N}$ and $o^{(in)}$ nodes from $\mathcal{M}$ are selected to form the overlapping subsets $\mathcal{S}_0^{(out)}$ and $\mathcal{S}_0^{(in)}$. The remaining nodes in $\mathcal{N}$ and $\mathcal{M}$ are then randomly and uniformly partitioned into s disjoint subsets, denoted by $\widetilde{\mathcal{S}}_1^{(out)}, \ldots, \widetilde{\mathcal{S}}_s^{(out)}$ and $\widetilde{\mathcal{S}}_1^{(in)}, \ldots, \widetilde{\mathcal{S}}_s^{(in)}$, respectively. Each subnetwork is constructed by combining the overlapping subset with one of the disjoint subsets:

$$\mathcal{S}_i^{(out)} = \mathcal{S}_0^{(out)} \cup \widetilde{\mathcal{S}}_i^{(out)},\ \mathcal{S}_i^{(in)} = \mathcal{S}_0^{(in)} \cup \widetilde{\mathcal{S}}_i^{(in)},\ \text{for } i \in [s].$$

These combined subsets define the node sets for each subnetwork, and the corresponding adjacency submatrices are extracted for community detection.

Detection Step: In the detection step, the DOM algorithm is applied independently to each subnetwork to obtain local clustering estimates. These estimates are represented by matrices $\widehat{C}^{(i)} \in \mathbb{R}^{n \times K}$ and $\widehat{Z}^{(i)} \in \mathbb{R}^{m \times L}$ for each $i \in [s]$, where entries corresponding to nodes absent from the i-th subgraph are padded with zeros.

Stitching step. Before stitching the clustering results across subnetworks, it is essential to first resolve the label permutation ambiguity inherent in independently estimated community labels. To this end, we perform an *alignment* step that matches the cluster labels of each subnetwork to those of a fixed reference subnetwork, which we take to be the first subnetwork ($i = 1$). This alignment is

based on the overlapping node subsets $\mathcal{S}_0$ and $\widetilde{\mathcal{S}}_0$, shared across subnetworks. Specifically, for each $i \in \{2, \ldots, s\}$, we compute permutation matrices

$$E_i^{(out)} = \mathcal{D}\left(\widehat{C}^{(i)}_{\mathcal{S}_0^{(out)}}, \widehat{C}^{(1)}_{\mathcal{S}_0^{(out)}}\right), E_i^{(in)} = \mathcal{D}\left(\widehat{Z}^{(i)}_{\mathcal{S}_0^{(in)}}, \widehat{Z}^{(1)}_{\mathcal{S}_0^{(in)}}\right),$$

where $\mathcal{D}(\cdot, \cdot)$ denotes a greedy matching algorithm (see Algorithm 7 in Appendix C) that aligns the labels by maximizing agreement over the overlapping nodes. The resulting permutation matrices $E_i^{(out)} \in \mathbb{R}^{K \times K}$ and $E_i^{(in)} \in \mathbb{R}^{L \times L}$ are then used to align the labels in the i-th subnetwork. Specifically, we align the estimated clustering matrices by multiplying them with the corresponding permutation matrices. After alignment, we compute the temporary aggregated matrices $\widehat{C}^{(\text{temp})}$ and $\widehat{Z}^{(\text{temp})}$ as follows:

$$\widehat{C}^{(\text{temp})}_{\mathcal{N}\setminus\mathcal{S}_0^{(out)}} = \sum_{i=1}^{s} \widehat{C}^{(i)}_{\mathcal{N}\setminus\mathcal{S}_0^{(out)}} E_i^{(out)}, \quad \text{and} \quad \widehat{C}^{(\text{temp})}_{\mathcal{S}_0^{(out)}} \text{ is a matrix of zeros,} \tag{17}$$

$$\widehat{Z}^{(\text{temp})}_{\mathcal{M}\setminus\mathcal{S}_0^{(in)}} = \sum_{i=1}^{s} \widehat{Z}^{(i)}_{\mathcal{M}\setminus\mathcal{S}_0^{(in)}} E_i^{(in)}, \quad \text{and} \quad \widehat{Z}^{(\text{temp})}_{\mathcal{S}_0^{(in)}} \text{ is a matrix of zeros.} \tag{18}$$

Repetition Step. To further enhance the stability and robustness of the community detection procedure, we introduce a repetition step. In each repetition, the non-overlapping out-node set $\mathcal{N} \setminus \mathcal{S}_0^{(out)}$ and in-node set $\mathcal{M} \setminus \mathcal{S}_0^{(in)}$ are randomly and uniformly repartitioned into s disjoint subnetworks. The DOM algorithm is then applied to each newly formed subnetwork. For each repetition, the resulting clustering matrices are aligned with the previously aggregated matrices via the stitching procedure, ensuring consistency across partitions. These aligned matrices are cumulatively aggregated to update the clustering information for the non-overlapping nodes.

The final clustering matrices for the out-nodes and in-nodes, denoted by $\widehat{C}^{(\text{final})}$ and $\widehat{Z}^{(\text{final})}$ respectively, are computed by averaging all aligned matrices across all repetitions, including the initial round. The final global community labels, $\hat{\boldsymbol{c}}$ for out-nodes and $\hat{\boldsymbol{z}}$ for in-nodes, are obtained by selecting the index of the maximum entry in each row:

$$\hat{c}_i = \arg\max_{k \in [K]} \widehat{C}^{(\text{final})}_{ik}, \quad \hat{z}_j = \arg\max_{\ell \in [L]} \widehat{Z}^{(\text{final})}_{j\ell}, \quad \text{for } i \in [n], j \in [m],$$

with random tie-breaking applied when multiple indices attain the maximum. The complete procedure is outlined in the following Algorithm 2.

To facilitate theoretical analysis, we introduce a simplified version of the above algorithm, referred to as SimpleSONNET-DOM, following the framework of Chakrabarty et al. (2025). In this simplified variant, the node labels in the overlapping regions are directly inherited from the first subnetwork rather than being determined via majority voting. Furthermore, the repetition step is omitted. A detailed description of SimpleSONNET-DOM is provided in Appendix C.1 (see Algorithm 4).

3.3 Two-Stage Divided Cosine Algorithm (TSDC)

In order to handle huge-scale networks more efficiently, we propose a new algorithm, Two-Stage Divided Cosine Algorithm (TSDC), leveraging the block rank-one structures of the networks. To elucidate the rationale of the proposed TSDC method, we first adopt a population-level perspective.

Algorithm 2 Subsampled DOM Integration within the SONNET Framework

Input: Adjacency matrix A, number of subgraphs s, overlap sizes $o^{(out)}$, $o^{(in)}$, number of repetitions R

Output: Final estimated community labels $\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}$

Division Step: Select overlapping subsets $\mathcal{S}_0^{(out)} \subset \mathcal{N}$ of size $o^{(out)}$, $\mathcal{S}_0^{(in)} \subset \mathcal{M}$ of size $o^{(in)}$
Partition $\mathcal{N} \setminus \mathcal{S}_0^{(out)}$ into s disjoint subsets $\widetilde{\mathcal{S}}_1^{(out)}, \ldots, \widetilde{\mathcal{S}}_s^{(out)}$
Partition $\mathcal{M} \setminus \mathcal{S}_0^{(in)}$ into s disjoint sets $\widetilde{\mathcal{S}}_1^{(in)}, \ldots, \widetilde{\mathcal{S}}_s^{(in)}$
for $i = 1$ to s **do**
  Define subnetwork: $\mathcal{S}_i^{(out)} = \mathcal{S}_0^{(out)} \cup \widetilde{\mathcal{S}}_i^{(out)}$, $\mathcal{S}_i^{(in)} = \mathcal{S}_o^{(in)} \cup \widetilde{\mathcal{S}}_i^{(in)}$
  Detection Step: Apply DOM to $A_{\mathcal{S}_i^{(out)}, \mathcal{S}_i^{(in)}}$ to obtain $\widehat{C}^{(i)}$, $\widehat{Z}^{(i)}$
end for
Stitching Step:
for $i = 2$ to s **do**
  Compute permutations: $E_i^{(out)} = \mathcal{D}\left(\widehat{C}^{(i)}_{\mathcal{S}_0^{(out)}}, \widehat{C}^{(1)}_{\mathcal{S}_0^{(out)}}\right)$, $E_i^{(in)} = \mathcal{D}\left(\widehat{Z}^{(i)}_{\mathcal{S}_0^{(in)}}, \widehat{Z}^{(1)}_{\mathcal{S}_0^{(in)}}\right)$
end for
Aggregate: Compute temporary aggregated matrices $\widehat{C}^{(\text{temp})}$ and $\widehat{Z}^{(\text{temp})}$ via (17) and (18)
Repetition Step:
for $r = 1$ to R **do**
  Repartition $\mathcal{N} \setminus \mathcal{S}_0^{(out)}$ into new disjoint subsets $\widetilde{\mathcal{S}}_{1r}^{(out)}, \ldots, \widetilde{\mathcal{S}}_{sr}^{(out)}$
  Repartition $\mathcal{M} \setminus \mathcal{S}_0^{(in)}$ into new disjoint subsets $\widetilde{\mathcal{S}}_{1r}^{(in)}, \ldots, \widetilde{\mathcal{S}}_{sr}^{(in)}$
  for $i = 1$ to s **do**
    Apply DOM to $A_{\widetilde{\mathcal{S}}_{ir}^{(out)}, \widetilde{\mathcal{S}}_{ir}^{(in)}}$ to obtain extended clustering matrices $\widehat{C}^{(i,r)}$, $\widehat{Z}^{(i,r)}$
    Compute $E_{i,r}^{(out)} = \mathcal{D}\left(\widehat{C}^{(i,r)}_{\widetilde{\mathcal{S}}_{ij}^{(out)}}, \widehat{C}^{(\text{temp})}_{\widetilde{\mathcal{S}}_{ij}^{(out)}}\right)$, $E_{i,r}^{(in)} = \mathcal{D}\left(\widehat{Z}^{(i,r)}_{\widetilde{\mathcal{S}}_{ij}^{(in)}}, \widehat{Z}^{(\text{temp})}_{\widetilde{\mathcal{S}}_{ij}^{(in)}}\right)$
    Align: $\widehat{C}^{(i,r)} \leftarrow \widehat{C}^{(i,r)} E_{i,r}^{(out)}$, $\widehat{Z}^{(i,r)} \leftarrow \widehat{Z}^{(i,r)} E_{i,r}^{(in)}$
  end for
end for
Set $\widehat{C}^{(\text{final})}$ and $\widehat{Z}^{(\text{final})}$ as:

$$\widehat{C}^{(\text{final})}_{\mathcal{N}\setminus\mathcal{S}_0^{(out)}} = \tfrac{1}{R+1}\left(\widehat{C}^{(\text{temp})} + \textstyle\sum_{r=1}^{R}\sum_{i=1}^{s}\widehat{C}^{(i,r)}\right), \quad \widehat{C}^{(\text{final})}_{\mathcal{S}_0^{(out)}} = \tfrac{1}{s}\displaystyle\sum_{i=1}^{s}\widehat{C}^{(i)}_{\mathcal{S}_0^{(out)}}$$

$$\widehat{Z}^{(\text{final})}_{\mathcal{M}\setminus\mathcal{S}_0^{(in)}} = \tfrac{1}{R+1}\left(\widehat{Z}^{(\text{temp})} + \textstyle\sum_{r=1}^{R}\sum_{i=1}^{s}\widehat{Z}^{(i,r)}\right), \quad \widehat{Z}^{(\text{final})}_{\mathcal{S}_0^{(in)}} = \tfrac{1}{s}\displaystyle\sum_{i=1}^{s}\widehat{Z}^{(i)}_{\mathcal{S}_0^{(in)}}$$

Assign final labels $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{z}}$ with
$\hat{c}_i = \arg\max_{k\in[K]} \widehat{C}^{(\text{final})}_{ik}$, $\hat{z}_j = \arg\max_{\ell\in[L]} \widehat{Z}^{(\text{final})}_{j\ell}$ for all $i \in [n]$, $j \in [m]$

Specifically, we assume that $\boldsymbol{c}$ and $\boldsymbol{z}$ represent the true out- and in-community membership vectors, respectively. Then, consider any two distinct out-nodes, denoted by i and j, and denote by $\mathcal{M}_\ell = \{j \in [m] \mid z_j = \ell\}$ the set of in-nodes belonging to the true in-community ℓ.

If both nodes i and j belong to the same out-community, say $\mathcal{N}_k$, we may assume that they correspond to the i'-th and j'-th nodes within $\mathcal{N}_k$, respectively. According to equation (2), we have:

$$\mathbb{E}(A_{i,\mathcal{M}_\ell}) = P^{(k,\ell)}_{i'\cdot}(\boldsymbol{c}, \boldsymbol{z}) = \Lambda^{(k,\ell)}_{i'}\left(\widetilde{\Lambda}^{(\ell,k)}\right)^T,$$

and

$$\mathbb{E}(A_{j,\mathcal{M}_\ell}) = P^{(k,\ell)}_{j'\cdot}(\boldsymbol{c},\boldsymbol{z}) = \Lambda^{(k,\ell)}_{j'}\left(\widetilde{\Lambda}^{(\ell,k)}\right)^T.$$

Consequently, their cosine similarity satisfies

$$\cos\big(\mathbb{E}(A_{i,\mathcal{M}_\ell}), \mathbb{E}(A_{j,\mathcal{M}_\ell})\big) = 1.$$

On the other hand, if nodes i and j belong to distinct out-communities, say $\mathcal{N}_k$ and $\mathcal{N}_{k'}$ with $k \neq k'$, we assume that i corresponds to the i'-th node in $\mathcal{N}_k$ and j corresponds to the j'-th node in $\mathcal{N}_{k'}$. Then, according to equation (2), we have:

$$\mathbb{E}(A_{i,\mathcal{M}_\ell}) = P^{(k,\ell)}_{i'\cdot}(\boldsymbol{c},\boldsymbol{z}) = \Lambda^{(k,\ell)}_{i'}\left(\widetilde{\Lambda}^{(\ell,k)}\right)^T,$$

and

$$\mathbb{E}(A_{j,\mathcal{M}_\ell}) = P^{(k',\ell)}_{j'\cdot}(\boldsymbol{c},\boldsymbol{z}) = \Lambda^{(k',\ell)}_{j'}\left(\widetilde{\Lambda}^{(\ell,k')}\right)^T.$$

Under the assumption that the vectors $\widetilde{\Lambda}^{(\ell,k)}$ and $\widetilde{\Lambda}^{(\ell,k')}$ are linearly independent, it follows that their cosine similarity is strictly less than one:

$$\cos\big(\mathbb{E}(A_{i,\mathcal{M}_\ell}), \mathbb{E}(A_{j,\mathcal{M}_\ell})\big) < 1.$$

Based on these observations, we know that in the population setting, one can determine whether nodes i and j belong to the same out-community by examining the value of $\cos\big(\mathbb{E}(A_{i,\mathcal{M}_\ell}), \mathbb{E}(A_{j,\mathcal{M}_\ell})\big)$.

For simplicity in the algorithmic description, in the above we have assumed that

$$\cos\big(\mathbb{E}(A_{i,\mathcal{M}_\ell}), \mathbb{E}(A_{j,\mathcal{M}_\ell})\big) \geq 0$$

for all $i, j \in [n]$ and $\ell \in [L]$, without loss of generality. Otherwise, one may equivalently consider

$$\left|\cos\big(\mathbb{E}(A_{i,\mathcal{M}_\ell}), \mathbb{E}(A_{j,\mathcal{M}_\ell})\big)\right|$$

in place of the original cosine value.

However, in practical scenarios, we do not directly observe $\mathcal{M}_\ell$, $\mathbb{E}(A_{i,\mathcal{M}_\ell})$, or $\mathbb{E}(A_{j,\mathcal{M}_\ell})$. Therefore, we approximate these quantities using their empirical counterparts. Specifically, let $\boldsymbol{z}$ represent an estimated assignment of column labels, and define $\mathcal{M}_\ell$ as the corresponding set of in-nodes assigned to community ℓ based on $\boldsymbol{z}$. Then replace $\mathbb{E}(A_{i,\mathcal{M}_\ell})$ and $\mathbb{E}(A_{j,\mathcal{M}_\ell})$ with their observed counterparts $A_{i,\mathcal{M}_\ell}$ and $A_{j,\mathcal{M}_\ell}$, respectively. Consequently, this leads us to define a similarity measure between any two out-nodes i and j as follows:

$$\mathrm{BlockCos}(A_{i\cdot}, A_{j\cdot}) = \sum_{\ell=1}^{L} \cos(A_{i,\mathcal{M}_\ell}, A_{j,\mathcal{M}_\ell}),$$

which aggregates cosine similarities across all in-communities. We refer to this measure as the Block Cosine Similarity between out-nodes i and j.

Similarly, given an estimated assignment of out-labels $\boldsymbol{c}$, we define the Block Cosine Similarity between any two in-nodes i and j as:

$$\mathrm{BlockCos}(A_{\cdot i}, A_{\cdot j}) = \sum_{k=1}^{K} \cos(A_{\mathcal{N}_k,i}, A_{\mathcal{N}_k,j}).$$

It is noteworthy that the concept of Block Cosine Similarity is introduced for the first time. Once similarities between any two out- (or in-) nodes are quantified, we can proceed to detect the out- and in-communities using a clustering framework analogous to the k-means approach. Specifically, we propose a two-stage algorithm for community detection. The first stage detects the row assignments $\boldsymbol{c}$ given $\boldsymbol{z}$ by maximizing the following objective function:

$$\boldsymbol{L}(\boldsymbol{c}, \boldsymbol{\mu} \mid \boldsymbol{z}) = \sum_{i=1}^{n} \mathrm{BlockCos}(A_{i\cdot}, \boldsymbol{\mu}_{\boldsymbol{c}_i \cdot})/L, \tag{19}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{K \times m}$ denotes the row cluster centers. In the second stage, we identify the column assignments $\boldsymbol{z}$ given $\boldsymbol{c}$ by maximizing the objective function:

$$\widetilde{\boldsymbol{L}}(\boldsymbol{z}, \tilde{\boldsymbol{\mu}} \mid \boldsymbol{c}) = \sum_{j=1}^{m} \mathrm{BlockCos}(A_{\cdot j}, \tilde{\boldsymbol{\mu}}_{\cdot \boldsymbol{z}_j})/K, \tag{20}$$

where $\tilde{\boldsymbol{\mu}} \in \mathbb{R}^{n \times L}$ denotes the column cluster centers.

The proposed objective functions (19) and (20) are maximized by alternately updating $(\boldsymbol{c}, \boldsymbol{z})$ and $(\boldsymbol{\mu}, \tilde{\boldsymbol{\mu}})$. Specifically, given $(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)})$ at the t-th iteration, we first obtain $(\boldsymbol{\mu}^{(t)}, \tilde{\boldsymbol{\mu}}^{(t)})$ by solving the following sub-optimization problems:

$$\boldsymbol{\mu}^{(t)} = \arg\max_{\boldsymbol{\mu}} \sum_{i=1}^{n} \mathrm{BlockCos}(A_{i\cdot}, \boldsymbol{\mu}_{\boldsymbol{c}_i^{(t)} \cdot})/L,$$

and

$$\tilde{\boldsymbol{\mu}}^{(t)} = \arg\max_{\tilde{\boldsymbol{\mu}}} \sum_{j=1}^{m} \mathrm{BlockCos}(A_{\cdot j}, \tilde{\boldsymbol{\mu}}_{\cdot \boldsymbol{z}_j^{(t)}})/K.$$

Through careful derivations, we obtain closed-form expressions for updating $(\boldsymbol{\mu}^{(t)}, \tilde{\boldsymbol{\mu}}^{(t)})$, which can be explicitly written as:

$$\mu_{k,\mathcal{M}_\ell}^{(t)} = \frac{1}{n_k} \sum_{i=1}^{n_k} \frac{A_{i\cdot}^{(k,\ell)}\left(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)}\right)}{\left\| A_{i\cdot}^{(k,\ell)}\left(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)}\right) \right\|_2}, \quad \tilde{\mu}_{\mathcal{N}_k,\ell}^{(t)} = \frac{1}{m_\ell} \sum_{j=1}^{m_\ell} \frac{A_{\cdot j}^{(k,\ell)}\left(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)}\right)}{\left\| A_{\cdot j}^{(k,\ell)}\left(\boldsymbol{c}^{(t)}, \boldsymbol{z}^{(t)}\right) \right\|_2}, \; k \in [K], \; \ell \in [L]. \tag{21}$$

Here, n_k and m_ℓ denote the sizes of the k-th out-community and the ℓ-th in-community, respectively. The normalization ensures that each updated centroid represents the average unit-norm direction of the corresponding block entries, thereby aligning with the cosine similarity-based objective. Combining the above cluster centers for each block, we obtain

$$\boldsymbol{\mu}^{(t)} = \left(\mu_{kj}^{(t)}\right)_{K \times m} \quad \text{and} \quad \tilde{\boldsymbol{\mu}}^{(t)} = \left(\tilde{\mu}_{i\ell}^{(t)}\right)_{n \times L}.$$

We update $\boldsymbol{c}$ and $\boldsymbol{z}$ alternately to obtain $\boldsymbol{c}^{(t+1)}$ and $\boldsymbol{z}^{(t+1)}$. Given $\boldsymbol{z}^{(t)}$, $\boldsymbol{\mu}^{(t)}$, and $\widetilde{\boldsymbol{\mu}}^{(t)}$, the optimization problem for updating $\boldsymbol{c}^{(t+1)}$ becomes

$$\hat{\boldsymbol{c}}^{(t+1)} = \arg\min_{\boldsymbol{c}} \boldsymbol{L}(\boldsymbol{c}, \boldsymbol{\mu}^{(t)} \mid \boldsymbol{z}^{(t)}).$$

Specifically, for each $i \in [n]$,

$$c_i^{(t+1)} = \arg \min_{1 \le k \le K} \text{BlockCos}(A_{i\cdot}, \boldsymbol{\mu}_{k\cdot}^{(t)})/L.$$

Similarly, the sub-optimization problem for updating $\boldsymbol{z}^{(t+1)}$ is given by

$$\hat{\boldsymbol{z}}^{(t+1)} = \arg\min_{\boldsymbol{z}} \widetilde{\boldsymbol{L}}(\boldsymbol{z}, \widetilde{\boldsymbol{\mu}}^{(t)} \mid \boldsymbol{c}^{(t+1)}).$$

Thus, for each $j \in [m]$,

$$z_j^{(t+1)} = \arg \min_{1 \le \ell \le L} \text{BlockCos}(A_{\cdot j}, \widetilde{\boldsymbol{\mu}}_{\cdot \ell}^{(t)})/K.$$

Next, given $\boldsymbol{c}^{(t+1)}$ and $\boldsymbol{z}^{(t+1)}$, we update $\boldsymbol{\mu}^{(t+1)}$ and $\widetilde{\boldsymbol{\mu}}^{(t+1)}$ according to the update rule in equation (21). The overall procedure is summarized in Algorithm 3, and we assume that the numbers of communities, K and L, are known. The initialization of $\left(\boldsymbol{c}^{(0)}, \boldsymbol{z}^{(0)}\right)$ follows the same strategy as in the DOM algorithm.

Algorithm 3 Two-stage Divided Cosine Algorithm (TSDC)

Input: Adjacency matrix A, cluster numbers K, L, tolerance ϵ, max iterations $iter_{\max}$
Output: Final estimated community labels $\hat{\boldsymbol{c}} = \boldsymbol{c}^{(t)}$, $\hat{\boldsymbol{z}} = \boldsymbol{z}^{(t)}$

Initialization: $\left(\boldsymbol{c}^{(0)}, \boldsymbol{z}^{(0)}\right)$, and set $t = 0$.
while $t < iter_{\max}$ **do**
  Update $\boldsymbol{\mu}^{(t)}$ and $\widetilde{\boldsymbol{\mu}}^{(t)}$ according to equation (21).
  for $i = 1$ to n **do**
    $c_i^{(t+1)} = \arg\min_{1 \le k \le K} \text{BlockCos}(A_{i\cdot}, \boldsymbol{\mu}_{k\cdot}^{(t)})$
  end for
  for $j = 1$ to m **do**
    $z_j^{(t+1)} = \arg\min_{1 \le \ell \le L} \text{BlockCos}(A_{\cdot j}, \widetilde{\boldsymbol{\mu}}_{\cdot \ell}^{(t)})$
  end for
  $err = \min\{\frac{|\boldsymbol{L}(\boldsymbol{c}^{(t+1)}, \boldsymbol{\mu}^{(t+1)} | \boldsymbol{z}^{(t+1)}) - \boldsymbol{L}(\boldsymbol{c}^{(t)}, \boldsymbol{\mu}^{(t)} | \boldsymbol{z}^{(t)})|}{\boldsymbol{L}(\boldsymbol{c}^{(t)}, \boldsymbol{\mu}^{(t)} | \boldsymbol{z}^{(t)})}, \frac{\widetilde{\boldsymbol{L}}(\boldsymbol{z}^{(t+1)}, \tilde{\boldsymbol{\mu}}^{(t+1)} | \boldsymbol{c}^{(t+1)}) - \widetilde{\boldsymbol{L}}(\boldsymbol{z}^{(t)}, \tilde{\boldsymbol{\mu}}^{(t)} | \boldsymbol{c}^{(t)})}{\widetilde{\boldsymbol{L}}(\boldsymbol{z}^{(t)}, \tilde{\boldsymbol{\mu}}^{(t)} | \boldsymbol{c}^{(t)})}\}$
  if $err < \epsilon$ **then**
    break
  end if
  $t \leftarrow t + 1$
end while

Remark 4. *In terms of computational complexity, both DOM and TSDC share the same initialization cost of $O(mn)$, arising from truncated SVD and K-means clustering. Although K and L are inherently involved in these steps, they are assumed to be constants and thus do not appear explicitly. During the iterative phase, DOM requires computing rank-one approximations and leave-one-out errors over all blocks, leading to a per-iteration complexity of $O(m^2n + n^2m)$. Although the subsampled variant of DOM performs faster in practice, it still exhibits the same theoretical complexity $O(m^2n+n^2m)$ when the number of subgraphs s is bounded. In contrast, TSDC involves only vector normalization and blockwise cosine similarity computations, resulting in a much lower per-iteration complexity of $O(mn)$. Therefore, the overall computational complexity is $O(T(m^2n + n^2m))$ for DOM and $O(Tmn)$ for TSDC, where T is the number of iterations. This highlights the greater computational efficiency of TSDC, especially in large-scale settings.*

4. Consistency Results

In this section, we demonstrate the identifiability of the community structure under the TNPM and establish the consistency of the proposed algorithms. For generality, throughout this section, we assume that each entry A_{ij} of the adjacency matrix A follows a sub-Gaussian distribution with variance proxy σ_{ij}^2 when $\boldsymbol{c}$ and $\boldsymbol{z}$ are given. Specifically, given $\boldsymbol{c}$ and $\boldsymbol{z}$, it has

$$A_{ij} - \mathbb{E}(A_{i,j}|\boldsymbol{c}, \boldsymbol{z}) \sim \text{subG}(\sigma_{ij}^2), \quad i \in [n], j \in [m], \tag{22}$$

where σ_{ij}^2 satisfies that

$$\mathbb{E}[\exp(t\{A_{ij} - \mathbb{E}(A_{i,j}|\boldsymbol{c}, \boldsymbol{z})\})] \leq \exp\left(\frac{\sigma_{ij}^2 t^2}{2}\right), \quad t \in \mathbb{R}.$$

Note that $\mathrm{subG}\left(\sigma^2\right)$ denotes a class of distributions rather than a distribution. Therefore, the notation is slightly abused when writing some random variable $X \sim \mathrm{subG}\left(\sigma^2\right)$. All technical details of the proofs are presented in the Appendix A.

4.1 Identifiability of Community Structure

We demonstrate the identifiability of community structure under TNPM with the numbers of communities K and L assumed known. According to the model setting, the parameters Λ and $\widetilde{\Lambda}$ have the following structure

$$\Lambda = \begin{pmatrix} \Lambda^{(1,1)} & \Lambda^{(1,2)} & \dots & \Lambda^{(1,L)} \\ \Lambda^{(2,1)} & \Lambda^{(2,2)} & \dots & \Lambda^{(2,L)} \\ \vdots & \vdots & & \vdots \\ \Lambda^{(K,1)} & \Lambda^{(K,2)} & \dots & \Lambda^{(K,L)} \end{pmatrix}_{n \times L}, \ \widetilde{\Lambda} = \begin{pmatrix} \widetilde{\Lambda}^{(1,1)} & \widetilde{\Lambda}^{(1,2)} & \dots & \widetilde{\Lambda}^{(1,K)} \\ \widetilde{\Lambda}^{(2,1)} & \widetilde{\Lambda}^{(2,2)} & \dots & \widetilde{\Lambda}^{(2,K)} \\ \vdots & \vdots & & \vdots \\ \widetilde{\Lambda}^{(L,1)} & \widetilde{\Lambda}^{(L,2)} & \dots & \widetilde{\Lambda}^{(L,K)} \end{pmatrix}_{m \times K}.$$

To analyze the identifiability of the community structure under TNPM, we make the following assumptions:

Assumption A1: All elements of the matrices Λ and $\widetilde{\Lambda}$ are nonzero.

Assumption A2. The points within each community are in *general position*, meaning that for any $k \in [K]$ and any choice of L distinct rows from the matrix $[\Lambda^{(k,1)}, \Lambda^{(k,2)}, \dots, \Lambda^{(k,L)}] \in \mathbb{R}^{n_k \times L}$, the corresponding L row vectors are linearly independent. Similarly, for any $\ell \in [L]$ and any choice of K distinct rows from the matrix $[\widetilde{\Lambda}^{(\ell,1)}, \widetilde{\Lambda}^{(\ell,2)}, \dots, \widetilde{\Lambda}^{(\ell,K)}] \in \mathbb{R}^{m_\ell \times K}$, the corresponding K row vectors are linearly independent.

Assumption A3: $n \geq K^2 L$ or $m \geq L^2 K$.

Remark 5. *Assumptions A1 and A2 impose restrictions on the parameter matrices Λ and $\widetilde{\Lambda}$, and similar assumptions have been adopted in prior works such as Sengupta and Chen (2018) and Noroozi et al. (2021b). Assumption A3 provides a lower bound on the network dimensions (n, m) in terms of the number of communities K and L. Since real-world networks typically contain a large number of nodes while the number of communities remains relatively small, this condition is often satisfied in practice.*

Theorem 1. *Under the TNPM, assuming that Assumptions A1∼A3 hold, we consider the following optimization problem*

$$(\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}) = \arg \min_{\boldsymbol{c}\in[K]^n, \boldsymbol{z}\in[L]^m} Loss(\boldsymbol{c}, \boldsymbol{z}),$$

with

$$\begin{aligned} Loss(\boldsymbol{c}, \boldsymbol{z}) &= \min_{\substack{\lambda_{k\ell}\in R, k\in[K], \ell\in[L] \\ \boldsymbol{\mu}\in R^{K\times m}, \tilde{\boldsymbol{\mu}}\in R^{n\times L}}} \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| P_*^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z}) - \lambda_{k\ell}\tilde{\mu}_{\mathcal{N}_k\ell}\mu_{k\mathcal{M}_\ell} \right\|_2^2, \\ &= \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| P_*^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z}) - \Pi_1\left(P_*^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z})\right) \right\|_F^2, \end{aligned}$$

where $\boldsymbol{c}$, $\boldsymbol{z}$ denote the clustering vectors, $P_* = \mathbb{E}(A)$ and $\Pi_1(\cdot)$ denotes the best rank-one approximation. Then, we have $\hat{\boldsymbol{c}} \equiv \boldsymbol{c}^*$ and $\hat{\boldsymbol{z}} \equiv \boldsymbol{z}^*$, where $\boldsymbol{c}^*$ and $\boldsymbol{z}^*$ are the ground truth community structures, and $\equiv$ indicates that the two community label assignments coincide up to a permutation $\boldsymbol{\pi}$ on $\{1, 2, \ldots, K\}$ or $\{1, 2, \ldots, L\}$.

Remark 6. *Theorem 1 provides the conditions for identifiability and demonstrates that under these assumptions, the ground truth community structures $\boldsymbol{c}^*$ and $\boldsymbol{z}^*$ can be uniquely determined by the mean of the adjacency matrix $P_* = \mathbb{E}(A)$.*

4.2 Consistency of the Estimated Mean Matrix

In this subsection, we evaluate the error associated with the estimated expectation of the adjacency matrix, which is obtained by using the DOM algorithm.

Theorem 2. *Under the TNPM with $\sigma_{max}^2 = \max_{i\in[n], j\in[m]} \sigma_{ij}^2$, define the estimator $\left(\widehat{K}, \widehat{L}, \hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}\right)$ as the following optimization problem:*

$$\begin{aligned} \left(\widehat{K}, \widehat{L}, \hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}\right) = \arg\min_{(K,L,\boldsymbol{c},\boldsymbol{z})} \Bigg\{ &\sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\| A^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z}) - \Pi_1\left(A^{(k,\ell)}(\boldsymbol{c}, \boldsymbol{z})\right) \right\|_F^2 \\ &+ Pen(n, m, K, L) \Bigg\}, \end{aligned} \tag{23}$$

and define the estimated blockwise mean values by:

$$\widehat{P}^{(k,\ell)}(\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}) = \Pi_1\left(A^{(k,\ell)}(\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}})\right),\ k \in [\widehat{K}], \ell \in [\widehat{L}].$$

Then the estimation error, $\left\| \widehat{P} - P_* \right\|_F^2 = \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left\| \Pi_1\left(A^{(k,\ell)}(\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}})\right) - P_*^{(k,\ell)}(\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}) \right\|_F^2$, satisfies the following bounds:

$$\mathbb{P}\left\{ \frac{1}{nm} \left\| \widehat{P} - P_* \right\|_F^2 \le \frac{H_1}{nm} Pen(n, m, K_*, L_*) + \frac{H_2\sigma_{max}^2}{nm} t \right\} \ge 1 - 3e^{-t},\ \forall t > 0, \tag{24}$$

$$\frac{1}{nm}\mathbb{E}\left\| \widehat{P} - P_* \right\|_F^2 \le \frac{H_1}{nm} Pen(n, m, K_*, L_*) + \frac{3H_2\sigma_{max}^2}{nm}. \tag{25}$$

where $H_1 = \frac{1}{1-\alpha_1-4\alpha_2}$, $H_2 = \frac{2C+2/\alpha_1+2C/\alpha_2}{1-\alpha_1-4\alpha_2}$ are positive constants.

The penalty term in (23) is defined as

$$Pen(n,m,K,L) = 2\tilde{\sigma}^2_{max}\left\{(1+1/\alpha_2)F_1(n,m,K,L) + (1/\alpha_1)F_2(n,m,K,L)\right\}, \tag{26}$$

where $\tilde{\sigma}^2_{max}$ is a pre-specified constant, and not smaller than $\sigma^2_{max} \triangleq \max_{i\in[n],j\in[m]}\sigma^2_{ij}$. Since σ^2_{max} is unknown in real applications, we need to choose a sufficiently large but reasonable $\tilde{\sigma}^2_{max}$. $F_1(n,m,K,L)$ and $F_2(n,m,K,L)$ are defined as

$$F_1(n,m,K,L) = C\left\{nL + mK + KL\log(2KL) + KL(n\log K + \log n + m\log L + \log m)\right\},$$
$$F_2(n,m,K,L) = n\log K + \log n + m\log L + \log m.$$

Note that the constants $\{\alpha_1, \alpha_2, C\}$ involved above are all positive values that can be calculated, provided in the proof of Theorem 2. The main motivation by introducing this penalty term is to prevent the overfitting, according to inequality (A.21) in the Appendix A.

Remark 7. *Theorem 2 establishes the consistency of the estimated mean matrix obtained via the DOM algorithm under the TNPM model, which means that as the sample size increases, the estimation error between $\widehat{P}$ and the true underlying connection probability matrix P_* vanished with high probability and in expectation. Notably, this result does not require the number of latent communities K_* and L_* to be fixed. Instead, it only requires a mild growth condition that $\frac{Pen(n,m,K_*,L_*)}{nm} \to 0$, which, according to equation (26), is guaranteed if $\frac{K_*L_*log(K_*L_*)}{n\wedge m} \to 0$. This condition ensures the model complexity grows slowly enough relative the sample size.*

Remark 8. *Theorem 2 notably extends the results of Noroozi et al. (2021b) in two key aspects. First, it significantly broadens the applicable types of networks, encompassing binary, discrete-valued, continuous-value networks, and even networks with mixture link distributions. Second, it employs a novel strategy to bound the operator norm of random matrices with sub-Gaussian entries. In contrast, following the strategy of Noroozi et al. (2021b) would require a concentration inequality for Lipschitz functions of independent sub-Gaussian random variables, analogous to the inequality used in their equation (5.59) for bounded variables in $[0,1]$. To the best of our knowledge, such a general result is not yet available and remains a challenging open problem in the current literature.*

Remark 9. *Theorem 2 does not require Assumptions A1∼A3. These assumptions are primarily imposed to ensure identifiability of the community structure by guaranteeing that only the true community partition yields rank-one sub-blocks of P_*. In contrast, Theorem 2 concerns the consistency of the estimated mean matrix $\widehat{P}$ with respect to the mean matrix P_*, which relies only that there exists a partition of nodes such that the corresponding sub-blocks of P_* are rank-one. In other words, even if the detected community structure is not correct, as long as the corresponding sub-blocks of P_* remain rank-one, the conclusion of Theorem 2 continues to hold.*

Remark 10. *Although Theorem 2 is developed under a general sub-Gaussian modeling framework, it also accommodates sparse regimes. To clarify this connection, consider the Bernoulli case as a special instance, where $\sigma^2_{ij} = P_{ij}(1-P_{ij})$, and define $\tau_n \triangleq \max_{i,j}\sigma^2_{ij}$. Then $\tau_n \asymp \max_{i,j}P_{ij}$, which coincides with the classical sparsity parameter adopted in the literature on popularity-adjusted block models (PABM), see, e.g., Sengupta and Chen (2018); Noroozi et al. (2021b). Under this interpretation, if $\|P_*\|_F^2 \asymp mn\tau_n^2$, the error bound in Theorem 2 implies that*

$\mathbb{E}\Big(\|\widehat{P} - P_*\|_F^2 / \|P_*\|_F^2\Big) \to 0$ *provided that* $\frac{mn\tau_n}{m+n} \cdot \frac{1}{KL\log(KL)} \to \infty$. *In particular, when* K *and* L *are bounded, this condition reduces to requiring that the average degree diverges. For general sub-Gaussian models, an explicit notion of sparsity analogous to the Bernoulli case is not available, but the variance proxy* τ_n *plays a similar role.*

4.3 Consistency of Community Labels

In this subsection, we evaluate the error associated with the estimated community structure obtained by using the DOM algorithm. For the convenience of theoretical analysis, we assume that the true number of communities, $K = K_*$ and $L = L_*$, is known, like Noroozi et al. (2021a,b).

Let $C_* \in \mathcal{M}_{n,K}$ and $Z_* \in \mathcal{M}_{m,L}$ denote the ground truth out-community and in-community membership matrices, respectively. Let C and Z represent alternative out-community and in-community matrices. We define the proportion of misclassified nodes by C and Z as follows:

$$\begin{aligned}
\operatorname{Err}(C, C_*) &= (2n)^{-1} \min_{\mathcal{P}_K \in \mathbb{P}_K} \|C\mathcal{P}_K - C_*\|_1 = (2n)^{-1} \min_{\mathcal{P}_K \in \mathbb{P}_K} \|C\mathcal{P}_K - C_*\|_F^2, \\
\operatorname{Err}(Z, Z_*) &= (2m)^{-1} \min_{\mathcal{P}_L \in \mathbb{P}_L} \|Z\mathcal{P}_L - Z_*\|_1 = (2n)^{-1} \min_{\mathcal{P}_L \in \mathbb{P}_L} \|Z\mathcal{P}_L - Z_*\|_F^2,
\end{aligned}$$

where $\mathbb{P}_K$ and $\mathbb{P}_L$ denote the sets of all $K \times K$ and $L \times L$ permutation matrices, respectively. Furthermore, we define the set

$$\Upsilon(C_*, Z_*, \rho_{n,m}) \triangleq \{(C, Z) \in \mathcal{M}_{n,K} \times \mathcal{M}_{m,L} : \max\{\operatorname{Err}(C, C_*), \operatorname{Err}(Z, Z_*)\} \geq \rho_{n,m}\}$$

as the collection of community assignments where the proportion of misclassified nodes is at least $\rho_{n,m} \in (0, 1)$.

Theorem 3. *Under the TNPM, define* $\sigma_{max}^2 \triangleq \max_{i\in[n], j\in[m]} \sigma_{ij}^2$. *Suppose Assumptions A1* $\sim$ *A3 hold, and let* $(\widehat{C}, \widehat{Z}) \equiv (\widehat{C}_K, \widehat{Z}_L)$ *be the community matrices defined as*

$$(\widehat{C}_K, \widehat{Z}_L) = \arg\min_{(C,Z)} \sum_{k=1}^{K} \sum_{\ell=1}^{L} \left\| A^{(k,\ell)}(C, Z) - \Pi_1\left(A^{(k,\ell)}(C, Z)\right) \right\|_F^2.$$

For $\rho_{n,m} \in (0, 1)$, *if there exist an* $\alpha_{n,m} \in (0, 1/2)$ *such that the following inequality is satisfied:*

$$\begin{aligned}
&\|P_*\|_F^2 - (1 + \alpha_{n,m}) \max_{(C,Z)\in\Upsilon(C_*,Z_*,\rho_{n,m})} \sum_{k=1}^{K} \sum_{\ell=1}^{L} \|P_*^{(k,\ell)}(C, Z)\|_{op}^2 \qquad (27) \\
\geq\ & \sigma_{max}^2 \left[\frac{H_1}{\alpha_{n,m}} \{nL + mK + KL\log(2KL) + KL(m + n)\} + \frac{H_2}{\alpha_{n,m}} KL(n\log K + m\log L) \right],
\end{aligned}$$

where H_1 *and* H_2 *are positive quantities explicitly defined in the proof of this theorem, then with probability at least* $1 - 2e^{-(n+m)}$, *the proportion of misclassified nodes by* $(\widehat{C}, \widehat{Z})$ *is at most* $\rho_{n,m}$, *i.e.,*

$$\max\left\{\operatorname{Err}\left(\widehat{C}, C_*\right), \operatorname{Err}\left(\widehat{Z}, Z_*\right)\right\} \leq \rho_{n,m}. \qquad (28)$$

Remark 11. *The condition (27) means that if the community matrices (C, Z) lie within the set where the proportion of misclassified nodes is at least $\rho_{n,m}$, then a lower bound exists on the sum of the differences between the Frobenius norms and the operator norms of the blocks $\left\{P_*^{(k,\ell)}(C, Z)\right\}$. In essence, incorrect clustering increases the ranks of these blocks, which in turn leads to a larger gap between their operator and Frobenius norms.*

Remark 12. *The quantity $\rho_{n,m}$ depends on both n and m, and may converge to zero at a certain rate as $n, m \to \infty$. To illustrate this concretely, we follow the framework of Noroozi et al. (2021b) and present an example to demonstrate. Specifically, consider a bipartite network with $n = 2N$ out-nodes and $m = 2M$ in-nodes, where the numbers of row and column communities are $K = L = 2$, and both row and column communities are balanced, i.e., each community contains the same number of nodes. Suppose that $\sigma_{\max}^2 = \max_{i\in[n],\, j\in[m]} \sigma_{ij}^2$, and assume further that all entries of Λ and $\widetilde{\Lambda}$ are of order δ, with $(\Lambda_{k1}, \Lambda_{k2})$ and $(\widetilde{\Lambda}_{\ell 1}, \widetilde{\Lambda}_{\ell 2})$ being column-orthogonal for $k, \ell = 1, 2$. For convenience, we additionally assume that the in-community labels are perfectly recovered, i.e., $Z = Z_*$, while only the out-community labels are subject to misclassification. In this setting, in order for* (27) *to hold, it can be shown that an overall misclassification proportion satisfying*

$$\rho_{n,m} \gtrsim \frac{\sigma_{\max}}{\delta^2}\sqrt{\frac{m+n}{nm}} \tag{29}$$

is sufficient. This expression highlights a natural signal-to-noise ratio (SNR),

$$\mathrm{SNR} \asymp \frac{\delta^2\sqrt{nm/(m+n)}}{\sigma_{\max}},$$

under which larger signal strength δ^2 and sample size $\sqrt{nm/(m+n)}$ improve recovery, while larger noise variance $\sigma_{\max}^2$ deteriorates it. Hence the requirement that $\rho_{n,m}$ does not vanish too rapidly is fully consistent with intuition. A detailed proof is provided in the Appendix A.5.

Remark 13. *Theorem 3 provides an upper bound on the misclassification rate, going beyond the conventional statement that it tends to zero as the network size increases, as it is routinely done in papers that rely on modularity maximization for clustering assignments (see, e.g., Bickel and Chen, 2009; Zhao et al., 2012; Sengupta and Chen, 2018). Similar conclusion for undirected networks is obtained in Noroozi et al. (2021b). To the best of our knowledge, this is the cutting-edge result available so far.*

Theorem 4. *(Consistency of the SimpleSONNET-DOM algorithm) Suppose that for each partition $S_q^{(out)} = S_0^{(out)} \cup \widetilde{S}_q^{(out)}, S_q^{(in)} = S_0^{(in)} \cup \widetilde{S}_q^{(in)}$, $q = 1, \ldots, s$, there exists an upper bound $\rho_{n,m} > 0$ on the proportion of misclustered nodes in each subnetwork and a probability bound $2e^{-\frac{n+m}{s}}$ such that:*

$$\mathbb{P}\left[\max\left\{\mathrm{Err}(\widehat{C}_{S_q^{(out)}}, C^*_{S_q^{(out)}}), \mathrm{Err}(\widehat{Z}_{S_q^{(in)}}, Z^*_{S_q^{(in)}})\right\} \le \rho_{n,m}\right] \ge 1 - 2e^{-\frac{n+m}{s}},$$

where $\rho_{n,m}$ is small. Then, for the output $(\widehat{C}, \widehat{Z})$ of SimpleSONNET-DOM, the overall proportion of misclustered nodes across s subnetworks with overlapping size $o^{(out)}$ satisfies:

$$\mathrm{Err}(\widehat{C}, C_*) \leq \frac{s\left(o^{(out)} + \frac{n-o^{(out)}}{s}\right)}{n} \cdot \rho_{n,m},$$
$$\mathrm{Err}(\widehat{Z}, Z_*) \leq \frac{s\left(o^{(in)} + \frac{m-o^{(in)}}{s}\right)}{m} \cdot \rho_{n,m},$$

with probability exceeding $1-(s-1)(\omega+\tilde{\omega})-(20s-16)e^{-(n+m)}$, where:

$$\omega = Ke^{-\frac{o^{(out)}}{4K(K+1)^2}}, \quad \tilde{\omega} = Le^{-\frac{o^{(in)}}{4L(L+1)^2}}.$$

Remark 14. *Theorem 4 provides an upper bound on the overall misclustering proportion of the SimpleSONNET algorithm under a divide-and-conquer framework. In particular, while clustering is performed separately on smaller subnetworks, the aggregated estimator $(\widehat{C}, \widehat{Z})$ inherits a global error bound that scales linearly with the local error rate $\rho_{n,m}$. The multiplicative factors are determined by the number of partitions s and the overlap sizes $o^{(out)}$ and $o^{(in)}$.*

4.4 Theoretical Analysis of TSDC

The TSDC algorithm can be naturally interpreted within the framework of Game Theory, which concerns the study of multi-player decision-making problems.

In a typical game-theoretic setup, there are n players, each selecting a strategy x_i from a feasible set $\mathbb{S}_i$. Each player is associated with a utility function $u_i(\boldsymbol{x})$, equivalently expressed as

$$u_i\left(x_i, x_{-i}\right), \quad x_{-i} = \left(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n\right), \tag{30}$$

where x_{-i} denotes the strategies chosen by all players except player i. The central objective in this framework is to seek an equilibrium solution, such as a Nash equilibrium, in which no player can unilaterally increase their utility by deviating from the current strategy profile.

In this subsection, we aim to establish the theoretical guarantees of the TSDC algorithm. Specifically, we present the following theorem, which encapsulates two key results. First, we show that the multi-player game induced by the TSDC algorithm admits at least one Pure Strategy Nash Equilibrium (PSNE). Second, under suitable modeling assumptions, we demonstrate that the true community structure (C_*, Z_*) constitutes a PSNE of the game with high probability. For completeness, the definition of a PSNE is provided below; for more details, one can refer to Chapter 1 of Fudenberg and Tirole (1991) and Chapter 2 of Osborne and Rubinstein (1994).

Definition 1. Consider a game with n players, where each player i chooses a *pure* (i.e., deterministic) strategy $x_i \in \mathbb{S}_i$ and is associated with a utility function $u_i(\boldsymbol{x}) = u_i(x_i, x_{-i})$, with $x_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$. A strategy profile $\boldsymbol{x}^* = (x_1^*, \ldots, x_n^*) \in \prod_{i=1}^n \mathbb{S}_i$ is called a *pure strategy Nash equilibrium* (PSNE) if, for every player $i \in \{1, \ldots, n\}$ and every alternative strategy $x_i \in \mathbb{S}_i$,

$$u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*).$$

That is, no player can increase their utility by unilaterally deviating from x_i^* while all other players keep their strategies fixed.

To proceed, we recast the TSDC algorithm within a game-theoretic framework. The underlying game consists of $(n+m+K+L)$ players: the first $(n+m)$ players represent the n in-nodes and m out-nodes of the network, while the remaining $(K+L)$ players correspond to the K row cluster centers and L column cluster centers. The sub-optimization problems faced by these players are given by

$$\mu_k = \arg\max_{\mu_k} \sum_{i=1}^{n} \mathbf{1}(c_i = k) \sum_{\ell=1}^{L} \cos(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}), \quad \forall k \in [K], \tag{31}$$

$$\tilde{\mu}_\ell = \arg\max_{\tilde{\mu}_\ell} \sum_{j=1}^{m} \mathbf{1}(z_j = \ell) \sum_{k=1}^{K} \cos(A_{\mathcal{N}_k,\ell}, \tilde{\mu}_{\mathcal{N}_k,\ell}), \quad \forall l \in [L], \tag{32}$$

$$c_i = \arg\max_{1\le k\le K} BlockCos(A_{i\cdot}, \boldsymbol{\mu}_{k\cdot}), \quad \forall i \in [n], \tag{33}$$

$$z_j = \arg\max_{1\le \ell\le L} BlockCos(A_{\cdot j}, \tilde{\boldsymbol{\mu}}_{\cdot \ell}), \quad \forall j \in [m]. \tag{34}$$

Each player corresponds to either a node or a cluster center, and the strategy space of each player consists of a discrete set of possible labels (for nodes) or continuous embeddings (for cluster centers). Specifically, for each node i, the strategy space is the set of community labels $\{1,\dots,K\}$ for in-nodes and $\{1,\dots,L\}$ for out-nodes; for each cluster center, the strategy space is the set of possible vectors in the embedding space.

The utility function for each node is defined based on the Block Cosine Similarity between its own adjacency information and the corresponding cluster center, thereby encouraging nodes to align with the center that exhibits the highest similarity. Similarly, the utility function for each cluster center reflects the aggregated similarity to the nodes assigned to it, promoting a representative embedding that best captures the features of its associated nodes. Therefore, the TSDC algorithm can be interpreted as an iterative process in which each player updates its strategy to maximize its own utility, conditioned on the current strategies of all other players. This iterative best-response dynamic seeks to reach a configuration where no player has an incentive to unilaterally deviate from their current strategy. Such a configuration corresponds precisely to a PSNE in the game-theoretic framework. In fact, the optimization procedure adopted by TSDC is an application of the classical *Iterated Best Response* (IBR) technique to search for PSNEs. At each iteration, every player sequentially chooses the best response to the current strategies of the other players, thereby progressively moving the system towards a stable equilibrium point.

We now establish the existence of a PSNE for the game induced by the TSDC algorithm and further demonstrate that, under suitable modeling assumptions, the true community structure (C_*, Z_*) constitutes a PSNE with high probability. The theoretical result is stated as follows:

Theorem 5. *Consider the multi-player game with $(n+m+K+L)$ players, where each player updates its strategy according to the optimization rules specified in equations* (31)-(34)*. This game admits at least one PSNE. Moreover, suppose that the network data follow the TNPM model, $n \asymp m$, and all row and column communities are balanced (i.e., community sizes are of the same order). Then, with probability at least $1 - O(n^{-8}) - O(m^{-8})$, the true community structure (C_*, Z_*) is a PSNE of the game.*

Remark 15. *Although Theorem 5 is formulated in game-theoretic terms and may appear different from the classical statistical framework, it is partially analogous in spirit. In classical M-estimation,*

one typically establishes that the estimator's objective function admits at least one global maximizer and that, under suitable assumptions, the objective value at the true parameter is close to its maximum in the sense that $M_n(\theta^*) \geq \sup_{\theta\in\Theta} M_n(\theta) - o_p(1)$. Similarly, Theorem 5 establishes the existence of at least one PSNE and shows that the true community structure (C_*, Z_*) constitutes a PSNE with high probability. However, unlike classical M-estimation, Theorem 5 does not necessarily imply that the PSNE attained by TSDC algorithm converges to the true labels, since the induced game may admit multiple PSNEs.

5. Simulation Studies

In this section, we evaluate the performance of the proposed methods through numerical simulations, focusing on three key aspects: the accuracy of community detection (Section 5.1), computational efficiency (Section 5.2), and the integration of our method with subsampling frameworks (Section 5.3). Throughout Sections 5.1-5.3, the number of communities is assumed to be known, while the estimation of the number of clusters is addressed separately in Section 5.4. The code is publicly available at GitHub (https://github.com/MiaCora/Two-way-Node-Popularity-Model).

5.1 Accuracy of Community Detection

The probability matrix P_* is generated under the proposed TNPM model. The adjacency matrix A is generated element-wise from Normal, Bernoulli, or Poisson distributions guided by P_*. We also consider a Normal-Bernoulli mixture to validate our theoretical results in sub-Gaussian settings and to assess performance in sparse networks. In what follows, we report simulation details and results for the TNPM model (Normal, Normal-Bernoulli mixture, and sparse networks), with additional results provided in Appendix D.1.

The number of communities is set to $(K^*, L^*) = (3, 4)$. Entries of the block matrices Λ and $\widetilde{\Lambda}$ are drawn independently from the uniform distribution on $[0, 1]$. The ground-truth out-community assignments $\boldsymbol{c}^* \in [K]^n$ and in-community assignments $\boldsymbol{z}^* \in [L]^m$ are generated from a multinomial distribution with $P(\boldsymbol{c}_i^* = k) = 1/K$ and $P(\boldsymbol{z}_j^* = \ell) = 1/L$, where $k \in [K^*]$ and $\ell \in [L^*]$. To assess performance in sparse settings, we introduce a sparsity parameter η, defined as the proportion of nonzero entries in Λ and $\widetilde{\Lambda}$. Sparsity is then induced by thresholding: the $\lfloor nL\eta \rfloor$ ($\lfloor mK\eta \rfloor$) smallest off-diagonal entries of Λ ($\widetilde{\Lambda}$) are set to zero.

We evaluate the performance of our proposed methods against some existing approaches. NEM, proposed in Zhang et al. (2021), esimates the latent community memberships through minimizing a regularized negative log-likelihood. KBCG (Kluger et al., 2003) is a co-clustering method that employs spectral clustering for bipartite data. OMPSC is a sparse subspace clustering method introduced by Noroozi et al. (2021b). COSSC and INSC both employ spectral clustering techniques, with COSSC leveraging a cosine similarity-based matrix and INSC utilizing an inner product similarity matrix. SVDK, as mentioned in Section 3, implements the K-means algorithm on the singular matrices derived from the network's adjacency matrix. Note that OMPSC, COSSC, and INSC are fundamentally developed for symmetric networks. Following the approach in Satuluri and Parthasarathy (2011), we separately apply the spectral clustering algorithm proposed in Ng et al. (2001) to the symmetric matrices AA^T and A^TA, a procedure referred to as INSC. In COSSC method, the inner product employed in constructing these matrices is replaced by a cosine similarity to better reflect angular proximity. For the OMPSC method, following Noroozi et al. (2021b), we separately apply the Orthogonal Matching Pursuit (OMP) algorithm to the adjacency matrix A and its transpose A^T,

resulting in two symmetric affinity matrices. These matrices are then used as inputs for the spectral clustering algorithm to identify communities.

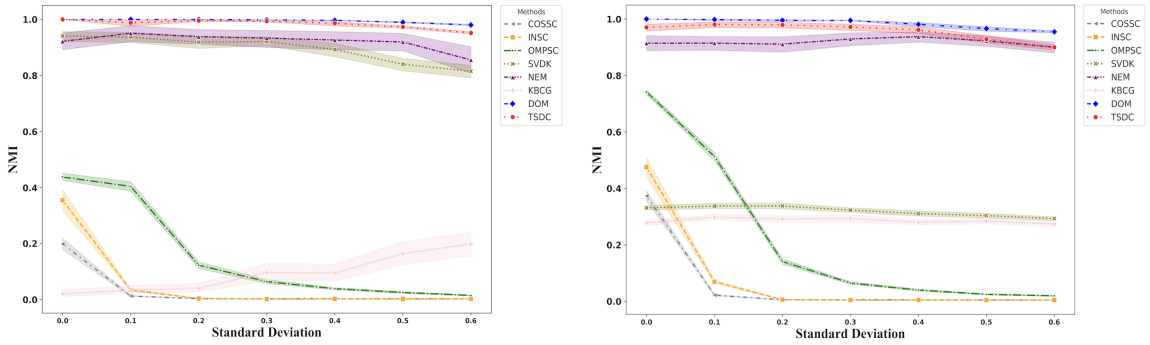
For community detection, we compare the performance of the DOM and TSDC algorithms against other techniques using three metrics: the clustering error in Wang (2010) and Zhang et al. (2021), the normalized mutual information (NMI) in Lancichinetti et al. (2009) and Zhou and Amini (2020), and the proportion of misclustered nodes in Noroozi et al. (2021a,b). Details on these metrics are accessible in Appendix D.1. This section primarily highlights the results using the NMI metric, while additional metrics results are provided in Appendix D.1. All the simulation results are based on 100 independent replications.

We explore three distinct scenarios and the simulation settings are outlined as follows:

- *Normal case*: The node counts (n, m) are set as$(600, 600)$. The adjacency matrix $A = (A_{ij})$ is generated with entries $A_{ij} \sim \mathcal{N}(P_{ij}, \sigma^2)$ with $\sigma \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$.

- *Normal-Bernoulli Mixture case*: The values of n and m range from 360 to 1320, increasing in increments of 240. The lower half of the adjacency matrix A is filled with Bernoulli variables $A_{ij} \sim \text{Ber}(P_{ij})$ for $i \in [n]$ and $j = 1, \ldots, i-1$. When $j > i$, entries follow a normal distribution $A_{ij} \sim \mathcal{N}(P_{ij}, \sigma^2)$, with σ fixed at 0.1.

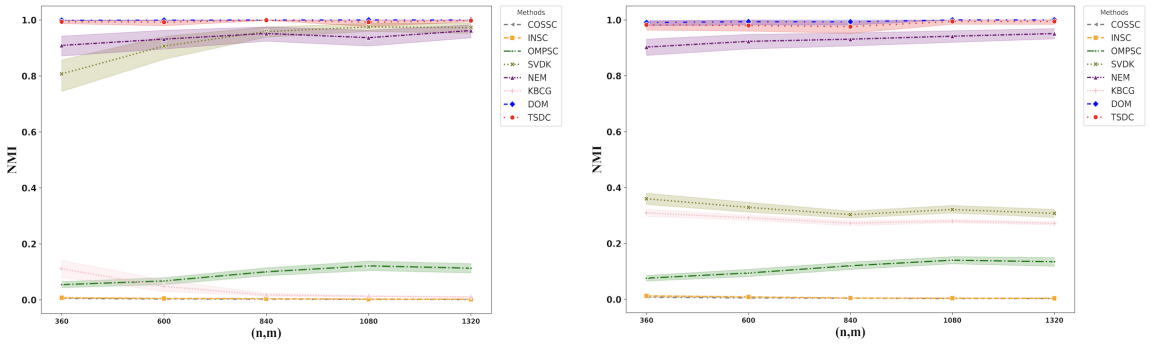
- *Sparse case*: The values of n and m range from 200 to 1000, increasing in increments of 100, while the sparsity parameter η is chosen from $\{0.3, 0.5, 0.7\}$. The adjacency matrix A is generated with Bernoulli variables $A_{ij} \sim \text{Ber}(P_{ij})$.



(a) out-community (b) in-community

Figure 2: NMI under the Normal data generation setting with $(n, m) = (600, 600)$. The left panel shows out-community clustering, and the right panel shows in-community clustering.

Figures 2 and 3 show that DOM and TSDC consistently outperform the competing methods. Appendix D.1 further corroborates these results, reporting consistent trends under both the Misclassification Score (MIS) and Clustering Error metrics. Figure 4 illustrates that increasing sparsity (η) leads to lower NMI. Additionally, DOM outperforms TSDC in smaller and sparser networks, while the gap between the two methods diminishes as the network size grows.



(a) out-community (b) in-community

Figure 3: NMI under the Normal-Bernoulli mixture setting. The left panel shows out-community clustering, and the right panel shows in-community clustering.

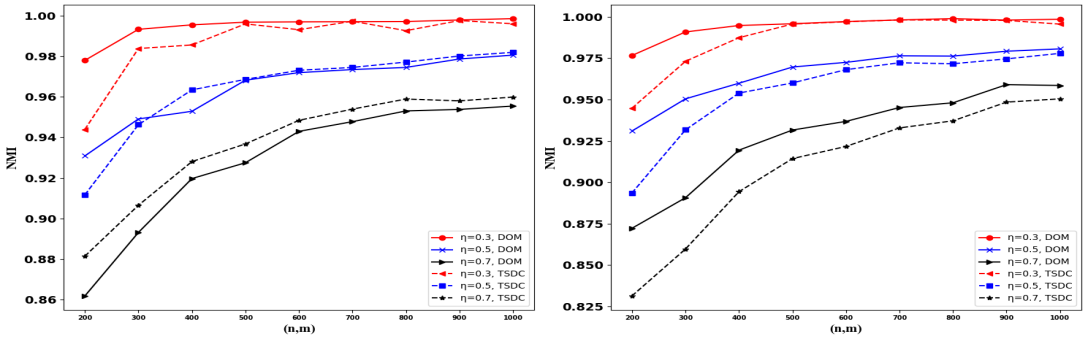


Figure 4: NMI under the sparsity setting. The left panel shows out-community clustering, and the right panel shows in-community clustering.

5.2 Computational Efficiency

We evaluate the computational efficiency of each method by recording running time (in seconds). All algorithms are implemented in Matlab and executed on a single processor of an Intel(R) Core(TM) i9-12900K CPU @ 3.20 GHz.

The matrix dimensions n and m vary from 360 to 1320 in increments of 240. Adjacency matrices $A = (A_{ij})$ are generated with $A_{ij} \sim \mathcal{N}(P_{ij}, \sigma^2)$, where $\sigma \in \{0.1, 0.5\}$. Computational efficiency is assessed using the average runtime over 100 simulations, as summarized in Table 1. The results show that TSDC is substantially faster than DOM, a finding further supported in Appendix D.2 for Bernoulli- and Poisson-generated matrices. Runtime increases with noise level, suggesting DOM may be suitable for smaller networks or non-urgent tasks, while TSDC is preferable for larger networks or time-critical applications.

σ	(n, m)	DOM	TSDC	OMPSC	COSSC	INSC	SVDK	NEM	KBCG
0.1	(360,360)	45.44	0.07	0.90	0.11	0.19	0.03	19.74	0.23
	(600,600)	173.12	0.13	2.56	0.25	0.64	0.06	42.61	0.30
	(840,840)	421.12	0.21	5.84	0.47	1.45	0.10	69.24	0.36
	(1080,1080)	892.90	0.27	11.08	0.78	2.95	0.15	106.01	0.43
	(1320,1320)	1482.88	0.57	14.34	0.97	5.41	0.19	151.61	0.51
0.5	(360,360)	66.45	0.27	0.93	0.11	0.19	0.03	20.22	0.23
	(600,600)	249.91	0.26	2.67	0.26	0.64	0.06	42.77	0.31
	(840,840)	592.11	0.37	5.96	0.49	1.46	0.10	70.21	0.37
	(1080,1080)	1219.89	0.49	11.30	0.80	2.94	0.15	107.14	0.45
	(1320,1320)	1657.89	0.59	14.87	0.98	5.38	0.19	152.85	0.53

Table 1: Running time (in seconds) for scenarios with n and m varying from 360 to 1320 in increments of 240, under the Normal data generation case for $\sigma \in \{0.1, 0.5\}$.

5.3 Subsampling-Based Community Detection in Large-Scale Networks

In this subsection, we evaluate the computational efficiency and community detection accuracy of the DOM algorithm integrated with three subsampling-based frameworks: PACE, GALE, and SONNET. Networks of size $(n, m) \in \{(1000, 1000), (2000, 2000), (5000, 5000), (8000, 8000)\}$ are considered, with adjacency matrices $A = (A_{ij})$ generated as $A_{ij} \sim \mathcal{N}(P_{ij}, \sigma^2)$ for $\sigma \in \{0.1, 0.3\}$. Runtime is measured as the average execution time (in seconds) over 100 simulation replicates, as in Section 5.2, while clustering accuracy is assessed using the NMI metric. Results for $\sigma = 0.1$ are reported in Table 2, and those for $\sigma = 0.3$ are presented in Table 3 in Appendix D.3.

Table 2 shows that the computational efficiency of DOM improves substantially when combined with subsampling-based frameworks, while community detection accuracy remains nearly unchanged. Accuracy is essentially identical under PACE and SONNET, whereas GALE performs slightly inferior, likely due to the smaller subgraph sizes $(e^{(out)}, e^{(in)})$ and the limited number of subsampling iterations s.

5.4 Unknown Number of Communities

In this section, we evaluate the performance of the community number selection method. As shown in equation (16), the community numbers K and L can be estimated using a criterion that incorporates the penalty term $\mathrm{Pen}(n, m, K, L)$ defined in (26). This designed penalty is motivated by the objective of ensuring that it exceeds the noise level with high probability (ref. (A.21)), and its construction is based on Lemma 1, which involves an unspecified constant C. Since no explicit value of C is available in the literature, it must typically be chosen large to ensure the lemma holds, which in turn often causes $\mathrm{Pen}(n, m, K, L)$ to underestimate K and L.

(n, m)	Method	Parameter Selection			Time (sec)	NMI
		$(e^{(out)}, e^{(in)})$	s			
$(1000, 1000)$	DOM	-	-		1390.2	(**1.00**,**1.00**)
	DOM+PACE	200	50		79.2	(**1.00**,**1.00**)
	DOM+GALE	200	100		146.8	(0.96,0.97)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(60,60)	20	1	78.12	(**1.00**,**1.00**)
		(60,60)	20	2	84.30	(**1.00**,**1.00**)
		(60,60)	20	5	98.19	(**1.00**,**1.00**)
$(2000, 2000)$	DOM	-	-		6594.1	(**1.00**,**1.00**)
	DOM+PACE	400	50		332.5	(**1.00**,**1.00**)
	DOM+GALE	400	100		604.8	(0.97,0.97)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(100,100)	20	1	219.14	(**1.00**,0.97)
		(100,100)	20	2	239.43	(**1.00**,**1.00**)
		(100,100)	20	5	286.14	(**1.00**,**1.00**)
$(5000, 5000)$	DOM	-	-		160114.86	(**1.00**,**1.00**)
	DOM+PACE	500	50		665.46	(0.99,0.99)
	DOM+GALE	500	100		1259.35	(0.94,0.95)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(300,300)	20	1	2415.42	(**1.00**,**1.00**)
		(300,300)	20	2	2685.90	(**1.00**,**1.00**)
		(300,300)	20	5	3618.32	(**1.00**,**1.00**)
$(8000, 8000)$	DOM	-	-		239457.32	(**1.00**,**1.00**)
	DOM+PACE	1000	50		6706.48	(0.99,**1.00**)
	DOM+GALE	1000	100		16181.03	(0.97,0.97)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(400,400)	20	1	9984.35	(**1.00**,0.97)
		(400,400)	20	2	10175.16	(**1.00**,**1.00**)
		(400,400)	20	5	10982.59	(**1.00**,**1.00**)

Table 2: Performance of DOM and its subsampling variants across different network sizes with fixed noise level $\sigma = 0.1$.

Following Noroozi et al. (2021b), we therefore adopt a reduced penalty $\widetilde{\mathrm{Pen}}(n, m, K, L)$ which retains only the dominant term of $\mathrm{Pen}(n, m, K, L)$:

$$\widetilde{\mathrm{Pen}}(n, m, K, L) = \frac{3\|A - \widehat{P}\|_2^2}{2nm} KL\big(n \ln K + m \ln L\big), \tag{35}$$

where $\|A-\widehat{P}\|_2^2/(nm)$ serves as a rough estimate of $\tilde{\sigma}^2_{\max}$ in $\mathrm{Pen}(n, m, K, L)$. The simulations follow the *Normal* setting in Section 5.1, with true community numbers $(K^*, L^*) \in \{(3, 3), (4, 3), (4, 4)\}$. Network sizes are set to $(n, m) \in \{(360, 360), (600, 600)\}$, and adjacency matrices are generated as $A_{ij} \sim \mathcal{N}(P_{ij}, \sigma^2)$ with $\sigma \in \{0.1, 0.2, 0.3\}$. Candidate pairs are taken from $(K, L) \in \{2, 3, 4, 5\} \times \{2, 3, 4, 5\}$.

Table 3 reports the proportion of correctly identifying the true community numbers over 100 replications. The results show that, in most cases, $(\widehat{K}, \widehat{L}) = (K^*, L^*)$, providing empirical support for the reliability of the proposed method in recovering the true number of communities in practice.

		DOM			TSDC		
(n, m)	(K^*, L^*)	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$
(360,360)	(3,3)	**1.00**	**1.00**	**1.00**	**1.00**	0.97	0.99
	(3,4)	**1.00**	0.98	0.99	0.93	0.93	0.91
	(4,3)	0.98	0.99	0.99	0.94	0.94	0.91
(600,600)	(3,3)	**1.00**	**1.00**	**1.00**	0.99	**1.00**	**1.00**
	(3,4)	**1.00**	**1.00**	0.99	0.94	0.92	0.90
	(4,3)	**1.00**	**1.00**	0.98	0.96	0.95	0.91

Table 3: Frequencies of the estimated community numbers $(\widehat{K}, \widehat{L})$ coinciding with the true values (K^*, L^*) under the DOM and TSDC algorithms.

6. Real Data Applications

In this section, we demonstrate the use of the DOM and TSDC algorithms on two real datasets: the Worldwide Food Trading Networks and MovieLens 100K dataset.

6.1 Worldwide Food Trading Networks

The Worldwide Food Trading Networks is collected by De Domenico et al. (2015) and is available at `http://www.fao.org`. Our analysis focuses on trading data in 2010, specifically on two product categories: cereals and cigarettes. We exclude countries with negligible trading volume (the first quantile), remaining 95 countries. Following the logarithmic transformation, we derive two directed 95×95 networks.

Before applying the DOM and TSDC algorithms, we first determine the numbers of out- and in-communities. Using our proposed estimation method (35), we obtain $(K, L) = (2, 3)$ for both networks under DOM. With TSDC, the estimates are $(K, L) = (2, 3)$ for the cereal network and

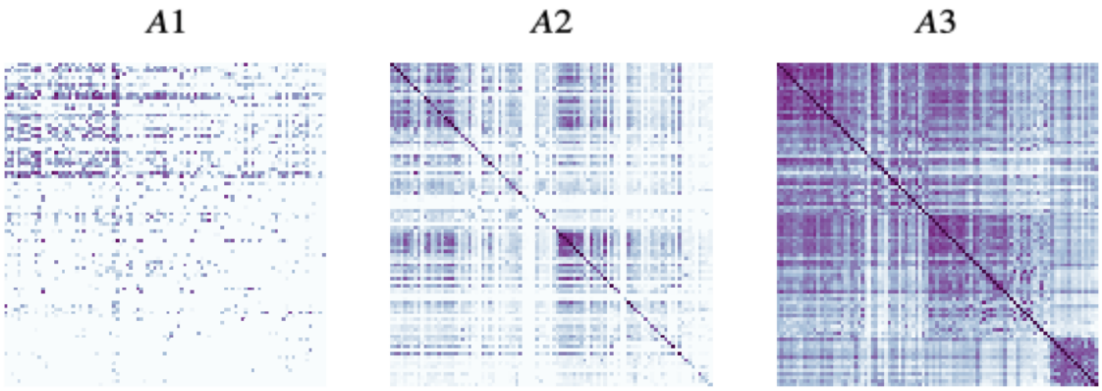


Figure 5: Panel A1 shows the original cereal trading network, while Panels A2 and A3 present the row- and column-wise block cosine similarity matrices, with nodes ordered by the detected clustering labels.

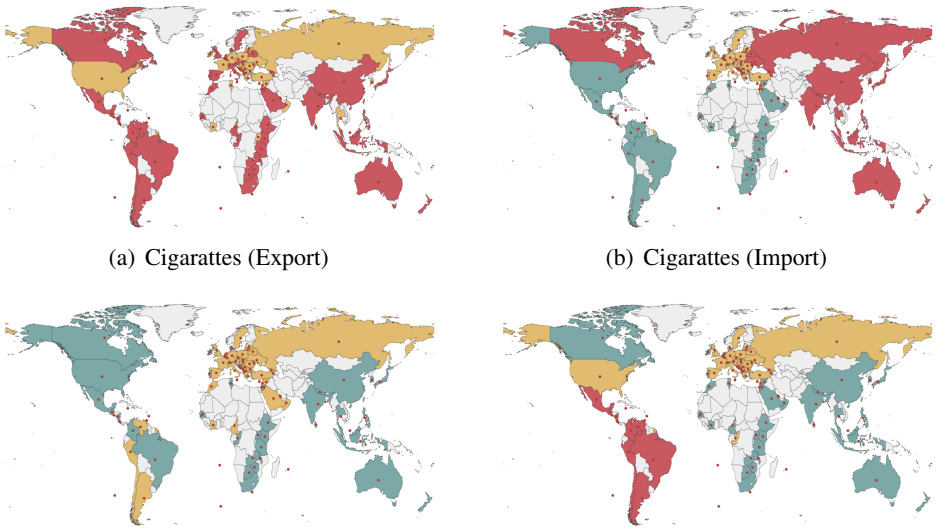
$(K, L) = (2, 2)$ for the cigarette network. For comparison, we also apply the maximum eigenvalue ratio criterion of Qi et al. (2022), which suggests $K = L = 1$. The corresponding singular value plots of the adjacency matrices, supporting this conclusion, are shown in Figure A.6 (Appendix E.1). Taking all the above into account, we finally adopt $(K, L) = (2, 3)$. Since the ground-truth community labels are unknown, direct quantitative evaluation is inherently difficult. Following the strategies adopted in the literature Jing et al. (2021) and Noroozi et al. (2021b), we assess our results by examining the interpretability of the recovered block structures and by visually comparing the estimated latent embeddings across methods, where clearer and more well-separated clustering patterns suggest a better capture of the underlying community structure. Results from the DOM algorithm are reported here, with those from TSDC provided in Appendix E.1.

Figures 5-6 present the heatmaps of the original networks together with the reordered block cosine similarity matrices obtained by the DOM algorithm. The block cosine similarity matrices reveal clear community structures: exporting countries split into two groups and importing countries into three, illustrating the TNPM's ability to detect network communities.

Figure 7 shows country clusters, with identical colors indicating the same cluster and grey areas denoting excluded countries. Panel (b) illustrates cigarette import clusters, revealing pronounced regional trade patterns. European nations form a major cluster of tobacco importers, while another cluster consists of China, India, Indonesia, and other Southeast Asian and Oceanian countries, primarily sourcing tobacco from Brazil, the United States, Canada, and Argentina. A separate cluster includes the United States together with selected South American and African countries, underscoring the strategic advantages of regional trading such as reduced transportation costs and faster delivery times.

We further compare our methods with two other approaches, NEM (Zhang et al., 2021) and KBCG (Kluger et al., 2003). In the absence of ground-truth community labels, we evaluate performance via visual inspection of the estimated latent embeddings. As shown in Figures A.11-

Figure 6: Panel B1 shows the original cigarette trading network, while Panels B2 and B3 present the row- and column-wise block cosine similarity matrices, with nodes ordered by the detected clustering labels.



(a) Cigarattes (Export)

(b) Cigarattes (Import)

(c) Cereals (Export)

(d) Cereals (Import)

Figure 7: World maps of detected communities: (a) cigarette exports, (b) cigarette imports, (c) cereal exports, and (d) cereal imports.

A.12 (Appendix E.1), the embeddings obtained from our approaches exhibit the most distinct block structures, whereas the results from NEM and KBCG appear less well separated and reveal weaker clustering patterns. This suggests that our proposed methods capture the underlying community structures in the trading networks more effectively.

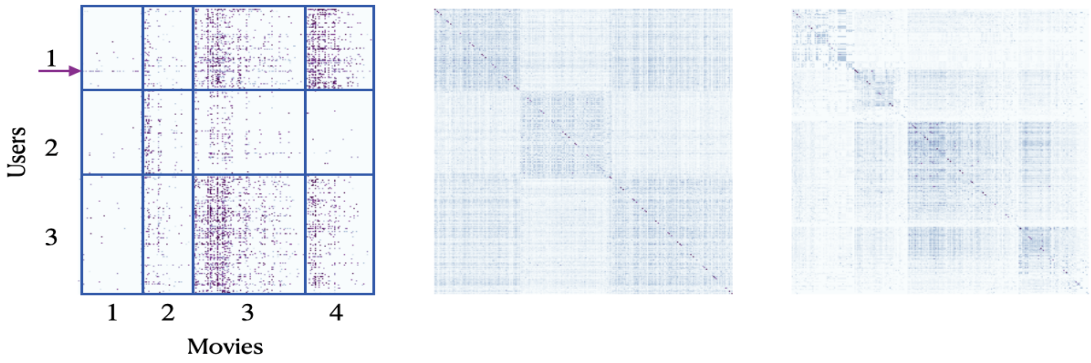


Figure 8: Heatmap of the MovieLens matrix A (left) and the reordered block cosine similarity matrices for rows (middle) and columns (right).

6.2 MovieLens 100K Dataset

The MovieLens 100K dataset, documented by Harper and Konstan (2015), is collected by the GroupLens Research of the MovieLens website (`movielens.umn.edu`), and is accessible at `https://grouplens.org/datasets/movielens/100k`. This dataset contains 10,000 ratings from 943 users across 1682 movies, leading to the construction of a 943×1682 rating matrix A, where each element $A_{i,j}$ denotes the rating from 1 to 5 given by user i to movie j. The movies are categorized into 19 genres, including "Adventure," "Action," and "Animation," among others, with 833 movies categorized in a single genre and the rest in multiple genres.

We employ the proposed algorithm to bicluster the MovieLens 100K dataset. Following the parameters set forth in Flynn and Perry (2020) and Zhao et al. (2022), we establish $K = 3$ for user clusters and $L = 4$ for movie clusters. We present the TSDC's results here and leave the DOM's in Appendix E.2.

Figure 8 displays the data matrix heatmap and block cosine similarity matrix, with nodes organized by TSDC algorithm-detected community assignments. The left panel reveals that nodes within the same user cluster exhibit diverse patterns of node popularity across different movie clusters. For instance, in user cluster 1, one individual shows a notable preference for cluster 1 movies, whereas the rest of user cluster 1 members provide significantly fewer ratings for these films. Furthermore, our results reveal distinct consumer behavior patterns. Significantly, users in cluster 3 predominantly review movies within cluster 3, while movies in cluster 4 are primarily reviewed by users in cluster 1. The middle and right panels clearly illustrate the distinct block structures and emphasize the suitability of the TNPM for modeling the dataset.

We investigate the relationship between the estimated movie clusters and the actual movie categories in the MovieLens dataset. A direct comparison is challenging due to the large number of categories and their overlaps. To mitigate this issue, we focus on a subset of 833 films, each belonging to a single unique category. We first evaluate performance using the clustering error (Zhang et al., 2021), which quantifies the discrepancy between the estimated and ground-truth community labels. The resulting errors are 0.384 for DOM, 0.388 for TSDC, 0.604 for NEM, and 0.527 for KBCG, with the two proposed methods exhibiting smaller errors than the competing approaches. In

addition, following Flynn and Perry (2020); Zhao et al. (2024), we also employ the chi-squared test to further examine the behavior of different methods. Chi-squared tests of independence on the corresponding contingency tables yield p-values of 4.39×10^{-8} for DOM and 3.003×10^{-11} for TSDC, which are substantially smaller than those reported for other methods, including 0.0415 (Flynn and Perry, 2020), 2.656×10^{-7} (Zhao et al., 2022), 0.0077 for NEM, and 0.0036 for KBCG. These results indicate that the clusters identified by our methods exhibit a stronger alignment with the ground-truth categories and achieve superior performance relative to the alternatives.

Acknowledgments

Jing's work was partially supported by the National Natural Science Foundation of China (Grant No. 12371290). Li's work was partially supported by the National Natural Science Foundation of China (Grant No. 12301360). Wang's work was supported by the National Natural Science Foundation of China (Grant No. 12301355); the Basic and Applied Basic Research Foundation of Guangdong Province (Grant No. 2024A1515011510); and the China Postdoctoral Science Foundation (Grant Nos. 2022T150293 and 2021M701588).

Appendix A. Proofs of the Main Theoretical Results

A.1 Proof of Theorem 1.

Note that

$$Loss(\boldsymbol{c},\boldsymbol{z}) = \sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z}) - \Pi_1(P^{(k,\ell)}(\boldsymbol{c},\boldsymbol{z}))\right\|_2^2.$$

To facilitate the proof, we divide the analysis into four distinct cases.
(1) If $\boldsymbol{z} \equiv \boldsymbol{z}^*$ and $\boldsymbol{c} \equiv \boldsymbol{c}^*$ (where $\equiv$ denotes that the community label assignments on both sides correspond to the same communities), then

$$Loss(\boldsymbol{c},\boldsymbol{z}) = 0.$$

(2) If $\boldsymbol{z} \equiv \boldsymbol{z}^*$ but $\boldsymbol{c} \not\equiv \boldsymbol{c}^*$, then there exist k, k', and k'' with $k' \neq k''$ such that $[\boldsymbol{c}=k]\cap[\boldsymbol{c}^*=k'] \neq \emptyset$ and $[\boldsymbol{c}=k]\cap[\boldsymbol{c}^*=k''] \neq \emptyset$. For convenience in the proof, we assume without loss of generality that $[\boldsymbol{c}=k] \subset [\boldsymbol{c}^*=k'] \cup [\boldsymbol{c}^*=k'']$. Moreover, let $[\boldsymbol{c}=k]\cap[\boldsymbol{c}^*=k'] = \{i_1, i_2, \ldots, i_S\}$ and $[\boldsymbol{c}=k]\cap[\boldsymbol{c}^*=k''] = \{i''_1, i''_2, \ldots, i''_T\}$, where $i'_s < i''_t$ for all $s \in [S]$ and $t \in [T]$. Thus, we obtain

$$P^{(k,1)}(\boldsymbol{c},\boldsymbol{z}) = P^{(k,1)}(\boldsymbol{c},\boldsymbol{z}^*) = \begin{pmatrix} V_{i'_1 1}(\widetilde{\Lambda}^{(1,k')})^T \\ V_{i'_2 1}(\widetilde{\Lambda}^{(1,k')})^T \\ \vdots \\ V_{i'_S 1}(\widetilde{\Lambda}^{(1,k')})^T \\ V_{i''_1 1}(\widetilde{\Lambda}^{(1,k'')})^T \\ V_{i''_2 1}(\widetilde{\Lambda}^{(1,k'')})^T \\ \vdots \\ V_{i''_T 1}(\widetilde{\Lambda}^{(1,k'')})^T \end{pmatrix} = \begin{pmatrix} V_{i'_1 1} & 0 \\ V_{i'_2 1} & 0 \\ \vdots & \vdots \\ V_{i'_S 1} & 0 \\ 0 & V_{i''_1 1} \\ 0 & V_{i''_2 1} \\ \vdots & \vdots \\ 0 & V_{i''_T 1} \end{pmatrix} \begin{pmatrix} (\widetilde{\Lambda}^{(1,k')})^T \\ (\widetilde{\Lambda}^{(1,k'')})^T \end{pmatrix}. \quad (A.1)$$

By Assumptions A1 and A2, we know that all elements of the matrices Λ and $\widetilde{\Lambda}$ are nonzero, and that

$$\operatorname{rank}\begin{pmatrix} (\widetilde{\Lambda}^{(1,k')})^T \\ (\widetilde{\Lambda}^{(1,k'')})^T \end{pmatrix} = 2.$$

Together with (A.1), this implies

$$\begin{aligned} &\operatorname{rank}(P^{(k,1)}(\boldsymbol{c},\boldsymbol{z})) = 2 > 1, \\ &\left\| P^{(k,1)}(\boldsymbol{c},\boldsymbol{z}) - \Pi_1(P^{(k,1)}(\boldsymbol{c},\boldsymbol{z})) \right\|_2^2 > 0. \end{aligned}$$

Consequently, if $\boldsymbol{z} \equiv \boldsymbol{z}^*$ but $\boldsymbol{c} \not\equiv \boldsymbol{c}^*$, we have

$$Loss(\boldsymbol{c},\boldsymbol{z}) > 0.$$

(3) If $\boldsymbol{z} \not\equiv \boldsymbol{z}^*$ but $\boldsymbol{c} \equiv \boldsymbol{c}^*$, then by an argument analogous to case (2) we conclude that

$$Loss(\boldsymbol{c},\boldsymbol{z}) > 0.$$

(4) If $\boldsymbol{z} \not\equiv \boldsymbol{z}^*$ and $\boldsymbol{c} \not\equiv \boldsymbol{c}^*$, our goal is to show that

$$Loss(\boldsymbol{c},\boldsymbol{z}) > 0.$$

By Assumption A3, we know that $n_{\max} \geq KL$ or $m_{\max} \geq LK$, where $n_{\max} = \max\{n_1, n_2, \ldots, n_K\}$ and $m_{\max} = \max\{m_1, m_2, \ldots, m_L\}$. Without loss of generality, assume that $n_1 = n_{\max} \geq KL$, where $n_1 = |[\boldsymbol{c}^* = 1]|$ denotes the cardinality of the row index set $[\boldsymbol{c}^* = 1]$. Hence, there exists $k \in [K]$ such that

$$|[\boldsymbol{c} = k] \cap [\boldsymbol{c}^* = 1]| \geq L. \tag{A.2}$$

Now consider a set of L distinct nodes $\mathbf{I} \triangleq \{i_1, i_2, \ldots, i_L\} \subset [\boldsymbol{c} = k] \cap [\boldsymbol{c}^* = 1]$. Since the nodes in $[\boldsymbol{c}^* = 1]$ are in general position, the rank of the submatrix $\Lambda_{\mathbf{I}}$ is L, i.e.,

$$\operatorname{rank}(\Lambda_{\mathbf{I}}) = L, \tag{A.3}$$

where

$$\Lambda_{\mathbf{I}} \triangleq \begin{pmatrix} \Lambda^{(i_1,1)} & \Lambda^{(i_1,2)} & \ldots & \Lambda^{(i_1,L)} \\ \Lambda^{(i_2,1)} & \Lambda^{(i_2,2)} & \ldots & \Lambda^{(i_2,L)} \\ \vdots & \vdots & \ldots & \vdots \\ \Lambda^{(i_L,1)} & \Lambda^{(i_L,2)} & \ldots & \Lambda^{(i_L,L)} \end{pmatrix}. \tag{A.4}$$

On the other hand, since $\boldsymbol{z} \not\equiv \boldsymbol{z}^*$, there exists some $\ell \in [L]$ such that the node set $[\boldsymbol{z} = \ell]$ is not pure. That is, there exist distinct $\ell', \ell'' \in [L]$ with $\ell' \neq \ell''$ such that

$$\begin{aligned} &[\boldsymbol{z} = \ell] \cap [\boldsymbol{z}^* = \ell'] \neq \emptyset, \\ &[\boldsymbol{z} = \ell] \cap [\boldsymbol{z}^* = \ell''] \neq \emptyset. \end{aligned}$$

Thus, by (A.3) and Assumptions A2$\sim$A3, we obtain

$$\operatorname{rank}(P_{\mathbf{I},[\boldsymbol{z}=\ell]}) \geq 2,$$

which further implies

$$\mathrm{rank}(P_{[\boldsymbol{c}=k],[\boldsymbol{z}=\ell]}) \geq \mathrm{rank}(P_{\mathbf{I},[\boldsymbol{z}=\ell]}) \geq 2.$$

Therefore,

$$Loss(\boldsymbol{c}, \boldsymbol{z}) > 0.$$

Finally, combining the results from cases (1)-(4), we complete the proof. ■

A.2 Proof of Theorem 2.

Set $\Xi = A - P_*$. By the construction of $\left(\widehat{K}, \widehat{L}, \widehat{C}, \widehat{Z}\right)$ in (23), we have

$$\left\|A(\widehat{C}, \widehat{Z}) - \widehat{P}(\widehat{C}, \widehat{Z})\right\|_F^2 + Pen(n, m, \widehat{K}, \widehat{L}) \leq \|A - P_*\|_F^2 + Pen(n, m, K_* L_*),$$

which implies that

$$\begin{aligned}
&\left\|\widehat{P}(\widehat{C}, \widehat{Z}) - P_*(\widehat{C}, \widehat{Z})\right\|_F^2 \\
\leq\ & 2\langle A(\widehat{C}, \widehat{Z}) - P_*(\widehat{C}, \widehat{Z}), P_*(\widehat{C}, \widehat{Z}) - \widehat{P}(\widehat{C}, \widehat{Z})\rangle + Pen(n, m, K_*, L_*) - Pen(n, m, \widehat{K}, \widehat{L}) \\
\leq\ & 2\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left\langle \Xi^{(k,\ell)}(\widehat{C}, \widehat{Z}), \Pi_{\hat{u},\hat{v}}\left\{\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\}\right\rangle \\
&+2\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left\langle \Xi^{(k,\ell)}(\widehat{C}, \widehat{Z}), \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\} - P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\rangle \\
&+2\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left\langle \Xi^{(k,\ell)}(\widehat{C}, \widehat{Z}), \Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\}\right\rangle \\
&+Pen(n, m, K_*, L_*) - Pen(n, m, \widehat{K}, \widehat{L}),
\end{aligned} \tag{A.5}$$

where $\Pi_{u,v}$ is defined as $\Pi_{u,v}(M) = (uu^T M(vv^T))$, and $(\tilde{u}, \tilde{v}) = \left(\tilde{u}^{(k,\ell)}(\widehat{C}, \widehat{Z}), \tilde{v}^{(k,\ell)}(\widehat{C}, \widehat{Z})\right)$ are the singular vectors of $P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})$ corresponding to the largest singular value of $P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})$. Similarly, $(\hat{u}, \hat{v}) = \left(\hat{u}^{(k,\ell)}(\widehat{C}, \widehat{Z}), \hat{v}^{(k,\ell)}(\widehat{C}, \widehat{Z})\right)$ are the singular vectors of $A^{(k,\ell)}(\widehat{C}, \widehat{Z})$ corresponding to the largest singular value of $A^{(k,\ell)}(\widehat{C}, \widehat{Z})$.

We now decompose the right-hand side of (A.5) into three parts and bound each separately. Define

$$\begin{aligned}
\Delta_1^{(k,\ell)(\widehat{C},\widehat{Z})} &\triangleq \left\langle \Xi^{(k,\ell)}(\widehat{C}, \widehat{Z}), \Pi_{\hat{u},\hat{v}}\left\{\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\}\right\rangle \\
&= \left\|\Pi_{\hat{u},\hat{v}}\left\{\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\}\right\|_F^2 \\
&= \left\|\Pi_{\hat{u},\hat{v}}\left\{\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\}\right\|_{op}^2 \\
&\leq \left\|\Pi\left\{\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\}\right\|_{op}^2.
\end{aligned}$$

Therefore,

$$\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_1^{(k,\ell)}(\widehat{C},\widehat{Z})\right| \le \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Pi\left\{\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\|_{op}^2, \tag{A.6}$$

which together with Lemma 2 yields that, for any $t \ge 0$, with probability at least $1-e^{-t}$,

$$\begin{aligned}&\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_1^{(k,\ell)}(\widehat{C},\widehat{Z})\right| \qquad (A.7)\\ \le\ & C\sigma_{max}^2\Big\{n\widehat{L}+m\widehat{K}+\widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L})+\widehat{K}\widehat{L}(n\log\widehat{K}+\log n+m\log\widehat{L}+\log m+t)\Big\},\end{aligned}$$

where C is an absolute constant. Next, define

$$\Delta_2^{(k,\ell)}(\widehat{C},\widehat{Z}) \triangleq \left\langle \Xi^{(k,\ell)}(\widehat{C},\widehat{Z}), \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\rangle,$$

and let $\widetilde{A}(\widehat{C},\widehat{Z})$ and $\widetilde{B}(\widehat{C},\widehat{Z})$ be the K-by-L matrices with their (k,ℓ)-elements defined by

$$\begin{aligned}&\widetilde{A}^{(k,\ell)}(\widehat{C},\widehat{Z}) = \Xi^{(k,\ell)}(\widehat{C},\widehat{Z}),\\ &\widetilde{B}^{(k,\ell)}(\widehat{C},\widehat{Z}) = \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z}).\end{aligned}$$

Then

$$\begin{aligned}\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_2^{(k,\ell)}(\widehat{C},\widehat{Z})\right| &= \left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\langle \widetilde{A}^{(k,\ell)}(\widehat{C},\widehat{Z}), \widetilde{B}^{(k,\ell)}(\widehat{C},\widehat{Z})\right\rangle\right|\\ &= \left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \widetilde{B}(\widehat{C},\widehat{Z})\right\rangle\right|.\end{aligned}$$

If $\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F = 0$, then

$$\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_2^{(k,\ell)}(\widehat{C},\widehat{Z})\right| = 0. \tag{A.8}$$

Otherwise,

$$\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_2^{(k,\ell)}(\widehat{C},\widehat{Z})\right| = \left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F \cdot \left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\rangle\right|. \tag{A.9}$$

For $\left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\rangle\right|$, we observe that if (K,L,C,Z) are fixed, then $\frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}$ is fixed. Moreover, for any (K,L,C,Z), we have

$$\left\|\frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\|_F = 1.$$

We can also observe that for any (K, L, C, Z), the permuted matrix $\widetilde{A}(C, Z) = \Xi(C, Z)$ has independent, mean zero, sub-Gaussian errors. Then, by using the Lemma 3 and applying the method of union bound, we know that for any $t \geq 0$, it holds that

$$\begin{aligned}
& \mathbb{P}\left\{\left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\rangle\right|^2 \geq 2\sigma_{max}^2\left(n\log\widehat{K}+\log n+m\log\widehat{L}+\log m+t\right)\right\} \\
\leq\ & \mathbb{P}\left[\bigcup_{\substack{K\in[n]\\L\in[m]}}\bigcup_{\substack{C\in\mathbb{M}_{n,K}\\Z\in\mathbb{M}_{m,L}}}\left\{\left|\left\langle \widetilde{A}(C,Z), \frac{\widetilde{B}(C,Z)}{\left\|\widetilde{B}(C,Z)\right\|_F}\right\rangle\right|^2 \geq 2\sigma_{max}^2\left(n\log K+\log n+m\log L+\log m+t\right)\right\}\right] \\
\leq\ & 2\sum_{K=1}^{n}\sum_{\ell=1}^{m}K^nL^me^{-(n\log K+\log n+m\log L+\log m+t)} \\
=\ & 2e^{-t}.
\end{aligned}$$

Thus, for any $t \geq 0$, with high probability of at least $1-2e^{-t}$, it has

$$\left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\rangle\right|^2 \leq 2\sigma_{max}^2\left(n\log\widehat{K}+\log n+m\log\widehat{L}+\log m+t\right). \quad \text{(A.10)}$$

For $\widetilde{B}(\widehat{C},\widehat{Z})$ with its elements defined as $\widetilde{B}^{(k,\ell)}(\widehat{C},\widehat{Z}) = \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})$, we can derive the following inequality

$$\begin{aligned}
\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F^2 &= \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
&\leq \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
&\leq \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left[\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 + \left\|\Pi_{\hat{u},\hat{v}}\left\{\Xi^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\}\right\|_F^2\right] \\
&= \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right) + \Xi^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
&= \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Pi_{\hat{u},\hat{v}}\left\{A^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
&= \sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\widehat{P}^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right) - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
&= \left\|\widehat{P}-P_*\right\|_F^2,
\end{aligned}$$

where the first inequality is based on the fact that $\Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\}$ is the best rank-one approximation of $P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)$. Thus, we have shown that

$$\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F^2 \leq \left\|\widehat{P}-P_*\right\|_F^2. \quad \text{(A.11)}$$

Then, combing equations (A.9), (A.10) and (A.11) yields that for any $t \geq 0$ and $\alpha_1 \geq 0$, with high probability at least $1 - 2e^{-t}$, it holds that

$$\begin{aligned}\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_2^{(k,\ell)}(\widehat{C},\widehat{Z})\right| &= \left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F \cdot \left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\rangle\right| \\ &\leq \frac{\alpha_1}{2}\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F^2 + \frac{1}{2\alpha_1}\left|\left\langle \widetilde{A}(\widehat{C},\widehat{Z}), \frac{\widetilde{B}(\widehat{C},\widehat{Z})}{\left\|\widetilde{B}(\widehat{C},\widehat{Z})\right\|_F}\right\rangle\right|^2 \\ &\leq \frac{\alpha_1}{2}\left\|\widehat{P}-P_*\right\|_F^2 + \frac{1}{\alpha_1}\sigma_{max}^2\left(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t\right).\end{aligned}$$

Thus, we have shown that for any $\alpha_1 > 0$ and $t \geq 0$, with high probability at least $1-2e^{-t}$, it holds that

$$\left|\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\Delta_2^{(k,\ell)}(\widehat{C},\widehat{Z})\right| \leq \frac{\alpha_1}{2}\left\|\widehat{P}-P_*\right\|_F^2 + \frac{1}{\alpha_1}\sigma_{max}^2\left(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t\right). \quad \text{(A.12)}$$

Finally, define

$$\Delta_3^{(k,\ell)}(\widehat{C},\widehat{Z}) = \left\langle \Xi^{(k,\ell)}(\widehat{C},\widehat{Z}), \Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\rangle.$$

Then by utilizing the formula (6.3) of Giraud (2015) on page 123, we know that

$$\begin{aligned}\left|\Delta_3^{(k,\ell)}(\widehat{C},\widehat{Z})\right| &= \left|\left\langle \Xi^{(k,\ell)}(\widehat{C},\widehat{Z}), \Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\rangle\right| \\ &\leq \left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{(2,r)} \cdot \left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\|_{(2,r)}, \quad \text{(A.13)}\end{aligned}$$

where

$$r = \min\left\{rank\left\{\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}, rank\left[\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right]\right\} \leq 2, \quad \text{(A.14)}$$

and $\|\cdot\|_{(2,r)}$ is the Ky-Fan $(2,r)$ norm. Additionally, for any matrix E, we know that

$$\|E\|_{(2,r)}^2 = \sum_{j=1}^{r}\sigma_j^2(E) \leq \|E\|_F^2, \quad \text{(A.15)}$$

where $\sigma_j^2(E)$ denotes the j-th singular value of E.

Then by combing equations (A.13), (A.14) and (A.15), and considering that $\|E\|_{(2,2)} \leq \sqrt{2}\|E\|_{op}$ for any matrix E, we can derive the following inequality

$$\left|\Delta_3^{(k,\ell)}(\widehat{C},\widehat{Z})\right| \leq \sqrt{2}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op} \cdot \left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\|_F. \quad \text{(A.16)}$$

Thus, by (A.16), we know that for any $\alpha_2 > 0$, it holds that

$$\begin{aligned}&\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left|\Delta_3^{(k,\ell)}(\widehat{C},\widehat{Z})\right| \\ \leq\ & \frac{1}{\alpha_2}\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2 + \frac{\alpha_2}{2}\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\|_F^2. \quad \text{(A.17)}\end{aligned}$$

By using Lemma 2, we know that for any $t \geq 0$, with high probability at least $1 - e^{-t}$, it holds that

$$\begin{aligned}
&\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2 \\
\leq\ & C\sigma_{max}^2\Big\{n\widehat{L} + m\widehat{K} + \widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L}) + \widehat{K}\widehat{L}(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t)\Big\}, \quad \text{(A.18)}
\end{aligned}$$

where C is an absolute constant. On the other hand, we know that

$$\begin{aligned}
&\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\|_F^2 \\
\leq\ & 2\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 + 2\left\|\Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
\leq\ & 4\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
\leq\ & 4\left[\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 + \left\|\Pi_{\hat{u},\hat{v}}\left\{\Xi^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\}\right\|_F^2\right] \\
=\ & 4\left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right) + \Xi^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
=\ & 4\left\|\Pi_{\hat{u},\hat{v}}\left\{A^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right)\right\} - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2 \\
=\ & 4\left\|\widehat{P}^{(k,\ell)}\left(\widehat{C},\widehat{Z}\right) - P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_F^2,
\end{aligned}$$

which yields that

$$\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left\|\Pi_{\hat{u},\hat{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\} - \Pi_{\tilde{u},\tilde{v}}\left\{P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\}\right\|_F^2 \leq 4\left\|\widehat{P} - P_*\right\|_F^2. \quad \text{(A.19)}$$

Thus by combing (A.17), (A.18) and (A.19), we can get that for any $\alpha_2 > 0$ and $t \geq 0$, it has

$$\begin{aligned}
&\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}} \left|\Delta_3^{(k,\ell)}(\widehat{C},\widehat{Z})\right| \\
\leq\ & \frac{C\sigma_{max}^2}{\alpha_2}\Big\{n\widehat{L} + m\widehat{K} + \widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L}) + \widehat{K}\widehat{L}(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t)\Big\} \\
&+2\alpha_2\left\|\widehat{P} - P_*\right\|_F^2. \quad \text{(A.20)}
\end{aligned}$$

Finally, combing the equations (A.5), (A.7), (A.12) and (A.20) yields that for any $\alpha_1 > 0$, $\alpha_2 > 0$ and $t \geq 0$, with high probability at least $1 - 3e^{-t}$, it holds that

$$\begin{aligned}
&(1 - \alpha_1 - 4\alpha_2)\left\|\widehat{P} - P_*\right\|_F^2 \\
\leq\ & 2C\sigma_{max}^2\Big\{n\widehat{L} + m\widehat{K} + \widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L}) + \widehat{K}\widehat{L}(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t)\Big\} \\
&+\frac{2}{\alpha_1}\sigma_{max}^2\left(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t\right) \\
&+\frac{2C\sigma_{max}^2}{\alpha_2}\Big\{n\widehat{L} + m\widehat{K} + \widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L}) + \widehat{K}\widehat{L}(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t)\Big\} \\
&+Pen(n,m,K_*,L_*) - Pen(n,m,\widehat{K},\widehat{L}). \quad \text{(A.21)}
\end{aligned}$$

By invoking the assumption of Theorem 2, we know that

$$
\begin{aligned}
&Pen(n,m,K,L) = \tilde{\sigma}^2_{max}\{(2+\frac{2}{\alpha_2})F_1(n,m,K,L) + \frac{2}{\alpha_1}F_2(n,m,K,L)\} \text{ with } \tilde{\sigma}^2_{max} \geq \sigma^2_{max},\\
&F_1(n,m,K,L) = C\left\{nL + mK + KL\log(2KL) + KL(n\log K + \log n + m\log L + \log m)\right\},\\
&F_2(n,m,K,L) = n\log K + \log n + m\log L + \log m.
\end{aligned}
$$

Combining the above equation with (A.21), we obtain

$$
(1-\alpha_1-4\alpha_2)\left\|\widehat{P}-P_*\right\|_F^2 \leq Pen(n,m,K_*,L_*) + \sigma^2_{max}\left(\frac{2}{\alpha_1}+2C+\frac{2C}{\alpha_2}\right)t.
$$

By setting $H_1 = \frac{1}{1-\alpha_1-4\alpha_2}$, $H_2 = \frac{2C+2/\alpha_1+2C/\alpha_2}{1-\alpha_1-4\alpha_2}$ and letting α_1 and α_2 be small enough such that $1-\alpha_1-4\alpha_2 > 0$, we can get that for any $t \geq 0$, with high probability at least $1-3e^{-t}$, it holds that

$$
\frac{1}{nm}\left\|\widehat{P}-P_*\right\|_F^2 \leq \frac{H_1}{nm}Pen(n,m,K_*,L_*) + \frac{H_2\sigma^2_{max}}{nm}t. \tag{A.22}
$$

Thus, we have proved the first part (24) of Theorem 2.

In order to derive (25), we define $\xi = \left\|\widehat{P}-P_*\right\|_F^2 - H_1 Pen(n,m,K_*,L_*)$. By noting that for any $t \geq 0$, $\mathbb{P}\left\{\left\|\widehat{P}-P_*\right\|_F^2 \geq H_1 Pen(n,m,K_*,L_*) + H_2\sigma^2_{max}t\right\} \leq 3e^{-t}$, we obtain that

$$
\begin{aligned}
& \mathbb{E}\left\|\widehat{P}-P_*\right\|_F^2\\
= \;& \mathbb{E}\left\{\left\|\widehat{P}-P_*\right\|_F^2 - H_1 Pen(n,m,K_*,L_*)\right\} + H_1 Pen(n,m,K_*,L_*)\\
= \;& \mathbb{E}\xi + H_1 Pen(n,m,K_*,L_*)\\
\leq \;& \mathbb{E}\xi I(\xi \geq 0) + H_1 Pen(n,m,K_*,L_*)\\
= \;& H_2\sigma^2_{max}\int_0^\infty \mathbb{P}(\xi \geq H_2\sigma^2_{max}t)dt + H_1 Pen(n,m,K_*,L_*)\\
\leq \;& H_2\sigma^2_{max}\int_0^\infty 3e^{-t}dt + H_1 Pen(n,m,K_*,L_*)\\
= \;& 3H_2\sigma^2_{max} + H_1 Pen(n,m,K_*,L_*).
\end{aligned}
$$

This yields that

$$
\frac{1}{nm}\mathbb{E}\left\|\widehat{P}-P_*\right\|_F^2 \leq \frac{H_1}{nm}Pen(n,m,K_*,L_*) + \frac{3H_2\sigma^2_{max}}{nm}. \tag{A.23}
$$

Combining (A.22) and (A.23) completes the proof. ■

A.3 Proof of Theorem 3.

By the definition of $(\widehat{C}, \widehat{Z})$, we have

$$\begin{aligned}&\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\widehat{C},\widehat{Z})-\Pi_1\left(A^{(k,\ell)}(\widehat{C},\widehat{Z})\right)\right\|_F^2\\ \leq&\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(C_*,Z_*)-\Pi_1\left(A^{(k,\ell)}(C_*,Z_*)\right)\right\|_F^2,\end{aligned} \tag{A.24}$$

where $\Pi_1(X)$ denotes the best rank-one approximation of X. It is also known that for any matrix X, one has

$$\|X-\Pi_1(X)\|_F^2=\|X\|_F^2-\|\Pi_1(X)\|_F^2=\|X\|_F^2-\|X\|_{op}^2.$$

Hence, by using the property $\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(C,Z)\right\|_F^2=\|A\|_F^2$ for any $C\in\mathcal{M}_{n,K}$ and $Z\in\mathcal{M}_{m,L}$, we can rewrite (A.24) as

$$\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2\ \geq\ \sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2. \tag{A.25}$$

Next, we divide the rest part of the proof into three steps as follows.

Step1. To lower bound $\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2$. By setting $\Xi\triangleq A-P_*$, it can be noted that for any $k\in[K]$ and $\ell\in[L]$, one has

$$\left\|A^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}\geq\left\|P_*^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}-\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}, \tag{A.26}$$

which follows from the property that for any three matrices X_1, X_2, and X_3 satisfying $X_1=X_2+X_3$, it holds that

$$\|X_2\|_{op}\geq\|X_1\|_{op}-\|X_3\|_{op}.$$

Since $a\geq b-c$ implies $a^2\geq\frac{b^2}{\tau+1}-\frac{c^2}{\tau}$ for any positive numbers a,b,c and τ, we can derive, after taking sum over the indices k and ℓ in (A.26), that

$$\begin{aligned}&\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2\\ \geq&\frac{1}{\tau+1}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2-\frac{1}{\tau}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2\\ =&\frac{1}{\tau+1}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{F}^2-\frac{1}{\tau}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2\\ =&\frac{1}{\tau+1}\|P_*\|_F^2-\frac{1}{\tau}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2,\end{aligned} \tag{A.27}$$

where we have used the fact that matrices $P_*^{(k,\ell)}(C_*, Z_*)$ are of rank one, and

$$\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(C_*, Z_*)\right\|_F^2 = \|P_*(C_*, Z_*)\|_F^2 = \|P_*\|_F^2 .$$

Hence, by invoking (A.25) and (A.27), we obtain that for any $\tau > 0$, it holds that

$$\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 \geq \frac{1}{\tau+1}\|P_*\|_F^2 - \frac{1}{\tau}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(C_*, Z_*)\right\|_{op}^2 . \tag{A.28}$$

Step2. In this step, we aim to obtain the upper bound of $\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2$. Note that for any $k \in [K]$ and $\ell \in [L]$, it holds that

$$\left\|A^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op} \leq \left\|P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op} + \left\|\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op} .$$

This implies that for any $\tau_0 > 0$, it holds that

$$\left\|A^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 \leq (1+\tau_0)\left\|P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 + \frac{1}{1+\tau_0}\left\|\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 ,$$

which is based the fact that $a \leq b + c$ implies that $a^2 \leq (1+\tau_0)b^2 + (1+\frac{1}{\tau_0})c^2$ for any positive number a, b, c and τ_0. Thus, we know that for any $\tau_0 > 0$, it holds that

$$\begin{aligned}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|A^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 \leq & (1+\tau_0)\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 \\ & + (1+\frac{1}{\tau_0})\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 .\end{aligned} \tag{A.29}$$

Stpe3. In this step, we aim to combine the obtained results in the above two steps and invoke several concentration inequalities to reach the required conclusion. Specifically, by combing (A.28) and (A.29), for any $\tau > 0$ and $\tau_0 > 0$, we have

$$\begin{aligned}& \frac{1}{\tau+1}\|P_*\|_F^2 - \frac{1}{\tau}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(C_*, Z_*)\right\|_{op}^2 \\ \leq\ & (1+\tau_0)\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 + (1+\frac{1}{\tau_0})\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 ,\end{aligned}$$

which can be reformulated as

$$\begin{aligned}& (1+\tau_0)(1+\tau)\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 \\ \geq & \|P_*\|_F^2 - \frac{1+\tau}{\tau}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(C_*, Z_*)\right\|_{op}^2 - \frac{(1+\tau_0)(1+\tau)}{\tau_0}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(\widehat{C}, \widehat{Z})\right\|_{op}^2 .\end{aligned} \tag{A.30}$$

In the following, we try to upper bound the following two quantities

$$\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2 \quad \text{and} \quad \sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2.$$

First, by applying Lemma 1 to $\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2$, we know that for any $t\geq 0$, we have

$$\mathbb{P}\left[\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\left(C_*,Z_*\right)\right\|_{op}^2 \leq C\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KLt\right\}\right]\geq 1-e^{-t}, \quad \text{(A.31)}$$

where C is an absolute positive constant.

Next, for $\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2$, we use the Lemma 1 with t replaced by $n\log K+m\log L+t$ and apply the union bound to obtain that for any $t\geq 0$, it has

$$\begin{aligned}&\mathbb{P}\Big[\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2 \leq C\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KL(n\log K+m\log L+t)\right\}\Big]\\ \geq&1-K^nL^me^{-(n\log K+m\log L+t)},\\ =&1-e^{-t},\end{aligned} \quad \text{(A.32)}$$

where C is the same absolute constant as above.

Finally, combing (A.30), (A.31) and (A.32) yields that for any $\tau>0,\tau_0>0$ and $t=n+m\geq 0$, with probability at least $1-2e^{-(n+m)}$, it holds that

$$\begin{aligned}&(1+\tau_0)(1+\tau)\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2\\ \geq&\|P_*\|_F^2-\frac{1+\tau}{\tau}C\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KL(m+n)\right\}\\ &-\frac{(1+\tau_0)(1+\tau)}{\tau_0}C\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KL(n\log K+m\log L+m+n)\right\},\end{aligned} \quad \text{(A.33)}$$

where C is the same absolute constant as above. Let $\tau=\tau_0\in(0,1]$ and $(1+\tau_0)(1+\tau)=1+\alpha_{n,m}$. Then, we know that

$$\begin{cases}\tau=\sqrt{1+\alpha_{n,m}}-1,\\ \frac{1}{\tau}=\frac{1}{\sqrt{1+\alpha_{n,m}}-1}=\frac{\sqrt{1+\alpha_{n,m}}+1}{\alpha_{n,m}},\\ \frac{1+\tau}{\tau}=\frac{\sqrt{1+\alpha_{n,m}}}{\alpha_{n,m}}\leq\frac{\sqrt{2}(\sqrt{2}+1)}{\alpha_{n,m}}\\ \frac{(1+\tau)^2}{\tau}=(1+\alpha_{n,m})\cdot\frac{\sqrt{1+\alpha_{n,m}}+1}{\alpha_{n,m}}\leq\frac{2(\sqrt{2}+1)}{\alpha_{n,m}}\end{cases}$$

Then, from (A.33), we know that with probability at least $1-2e^{-(n+m)}$, it holds that

$$\begin{aligned}
&\|P_*\|_F^2-(1+\alpha_{n,m})\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2\\
\leq&\frac{\sqrt{2}(\sqrt{2}+1)C}{\alpha_{n,m}}\cdot\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KL(m+n)\right\}\\
&+\frac{2(\sqrt{2}+1)C}{\alpha_{n,m}}\cdot\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KL(n\log K+m\log L+m+n)\right\}.
\end{aligned}$$

Furthermore, we let

$$\begin{aligned}
H_1&=4(\sqrt{2}+1)C,\\
H_2&=2(\sqrt{2}+1)C.
\end{aligned}$$

Thus, we can get that

$$\begin{aligned}
&\|P_*\|_F^2-(1+\alpha_{n,m})\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|P_*^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2\\
\leq&\sigma_{max}^2\left[\frac{H_1}{\alpha_{n,m}}\left\{nL+mK+KL\log(2KL)+KL(m+n)\right\}+\frac{H_2}{\alpha_{n,m}}KL(n\log K+m\log L)\right].
\end{aligned}\tag{A.34}$$

Thus, by combing (A.34) and the assumption (27) of the theorem, we can conclude that for any solution $(\widehat{C},\widehat{Z})\equiv(\widehat{C}_K,\widehat{Z}_L)$ of (7), with probability at least $1-e^{-(n+m)}$, it holds that

$$(\widehat{C},\widehat{Z})\notin\Upsilon(C_*,Z_*,\rho_{n,m}),$$

which means that with probability at least $1-2e^{-(n+m)}$, the proportion of misclassified nodes by $(\widehat{C},\widehat{Z})$ is at most $\rho_{n,m}$, i.e.,

$$\max\left\{\mathrm{Err}\left(C,C_*\right),\mathrm{Err}\left(Z,Z_*\right)\right\}\leq\rho_{n,m},$$

which concludes the entire proof of this theorem. ■

A.4 Proof of Theorem 4.

A.4.1 THE EXISTENCE OF A PURE STRATEGY NASH EQUILIBRIUM (PSNE)

This result primarily relies on the celebrated Nash's existence theorem for pure strategy equilibria in infinite games, as proved independently by Glicksberg (1952), Fan (1952) and Debreu (1952). The detailed statement and conditions of this theorem can be found in Lemma 4.

To apply the result of Lemma 4, we first reformulate the original $(n+m+K+L)$-player game into an equivalent game. This reformulation is necessary because the original game does not satisfy the technical conditions required by Lemma 4, particularly the assumptions of compact and convex strategy spaces, as well as continuous and concave utility functions. The equivalent formulation of the game is described as follows: instead of discrete strategy selections, each player now selects a probabilistic strategy over possible actions, thereby smoothing the optimization landscape.

The corresponding sub-optimization problems for these players are defined as:

$$\mu_k = \arg\max_{\mu_k} \sum_{i=1}^{n} \tau_{ik} \sum_{\ell=1}^{L} \frac{A_{i,\mathcal{M}_\ell} \cdot \mu_{k,M_\ell}}{\|A_{i,\mathcal{M}_\ell}\|_2}, \quad \|\mu_{k,\mathcal{M}_\ell}\|_2 \leq 1, \quad \forall k \in [K], \tag{A.35}$$

$$\tilde{\mu}_\ell = \arg\max_{\tilde{\mu}_\ell} \sum_{j=1}^{m} \tilde{\tau}_{j\ell} \sum_{k=1}^{K} \frac{A_{\mathcal{N}_k,\ell} \cdot \tilde{\mu}_{\mathcal{N}_k,\ell}}{\|A_{\mathcal{N}_k,\ell}\|_2}, \quad \|\tilde{\mu}_{\mathcal{N}_k,\ell}\|_2 \leq 1, \quad \forall \ell \in [L], \tag{A.36}$$

$$\tau_i = \arg\max_{\tau_i} \sum_{k=1}^{K} \tau_{ik} \sum_{\ell=1}^{L} \cos(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}), \quad \tau_{ik} \geq 0, \quad \sum_{k=1}^{K} \tau_{ik} = 1, \quad \forall i \in [n], \tag{A.37}$$

$$\tilde{\tau}_j = \arg\max_{\tilde{\tau}_j} \sum_{\ell=1}^{L} \tilde{\tau}_{j\ell} \sum_{k=1}^{K} \cos(A_{\mathcal{N}_k,\ell}, \tilde{\mu}_{\mathcal{N}_k,\ell}), \quad \tilde{\tau}_{j\ell} \geq 0, \quad \sum_{\ell=1}^{L} \tilde{\tau}_{j\ell} = 1, \quad \forall j \in [m], \tag{A.38}$$

where $\tau_i = (\tau_{i1}, \tau_{i2}, \ldots, \tau_{iK})$, $\tilde{\tau}_j = (\tilde{\tau}_{j1}, \tilde{\tau}_{j2}, \ldots, \tilde{\tau}_{jL})^T$, $\mathcal{M}_\ell = \{j \in [m] : \arg\max_{1\leq\ell\leq L} \tilde{\tau}_{j\ell} = \ell,\ \forall j = 1, 2, \ldots, m\}$ and $\mathcal{N}_k = \{i \in [n] : \arg\max_{1\leq k\leq K} \tau_{ik} = k,\ \forall i = 1, 2, \ldots, n\}$. It is easy to verify that for each player, optimizing its utility in the reformulated game yields the same outcome as in the original formulation. Therefore, the two games are equivalent in terms of strategic behavior and equilibrium structure.

Furthermore, in the reformulated game, the strategy space for each player is compact and convex, and the utility function is continuous in all variables and concave with respect to each player's own strategy. Thus, by invoking Lemma 4 (Debreu-Glicksberg-Fan Theorem), we conclude that the above game admits at least one Pure Strategy Nash Equilibrium (PSNE).

A.4.2 TRUE COMMUNITY STRUCTURE (C_*, Z_*) IS A PSNE WITH HIGH PROBABILITY

We aim to prove that $(C, Z) = (C_*, Z_*)$ is a Pure Strategy Nash Equilibrium (PSNE) of the game. That is, for any player i, given other players' strategies are fixed, player i cannot improve their utility by unilaterally changing its own strategy.

We describe the optimization problems involved in the game as follows:

$$\begin{cases} \mu_{k,\mathcal{M}_\ell} = \arg \max\limits_{\|\mu_{k,\mathcal{M}_\ell}\|_2 \leq 1} \left(\sum\limits_{i=1}^{n} \mathbf{1}(C_i = k) \dfrac{A_{i,\mathcal{M}_\ell}}{\|A_{i,\mathcal{M}_\ell}\|_2} \right) \mu_{k,\mathcal{M}_\ell}^T \\ \tilde{\mu}_{\mathcal{N}_k,\ell} = \arg \max\limits_{\|\tilde{\mu}_{\mathcal{N}_k,\ell}\|_2 \leq 1} \left(\sum\limits_{j=1}^{m} \mathbf{1}(Z_j = \ell) \dfrac{A_{\mathcal{N}_k,\ell}^T}{\|A_{\mathcal{N}_k,\ell}\|_2} \right) \tilde{\mu}_{\mathcal{N}_k,\ell} \\ C_i = \arg \max\limits_{1\leq k\leq K} \sum\limits_{\ell=1}^{L} \cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}\right) \\ Z_j = \arg \max\limits_{1\leq \ell\leq L} \sum\limits_{k=1}^{K} \cos\left(A_{\mathcal{N}_k,\ell}, \tilde{\mu}_{\mathcal{N}_k,\ell}\right) \end{cases}$$

Our goal is to prove that the following result holds with overwhelming probability:

$$C_i^* = \arg\max_{1\le k\le K} \sum_{\ell=1}^{L} \cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}\right), \quad \mathcal{M}_\ell = \{j \in [m] \mid Z_j^* = \ell,\ \forall j = 1, 2, \ldots, m\}$$
$$= \arg\max_{1\le k\le K} \sum_{\ell=1}^{L} \frac{A_{i,\mathcal{M}_\ell}\mu_{k,\mathcal{M}_\ell}^T}{\|A_{i,\mathcal{M}_\ell}\|_2 \cdot \|\mu_{k,\mathcal{M}_\ell}\|_2}.$$

In fact, it suffices to prove that, with overwhelming probability, the following holds:

$$\sum_{\ell=1}^{L} \frac{A_{i,\mathcal{M}_\ell}\mu_{C_i^*,\mathcal{M}_\ell}^T}{\|A_{i,\mathcal{M}_\ell}\|_2 \cdot \|\mu_{C_i^*,\mathcal{M}_\ell}\|_2} > \sum_{\ell=1}^{L} \frac{A_{i,\mathcal{M}_\ell}\mu_{k',\mathcal{M}_\ell}^T}{\|A_{i,\mathcal{M}_\ell}\|_2 \cdot \|\mu_{k',\mathcal{M}_\ell}\|_2}, \quad \forall k' \neq C_i^*. \tag{A.39}$$

Assuming $C_i^* = k$ and $\mathcal{M}_\ell = \{j \in [m] \mid Z_j^* = \ell\}$ for all $j = 1, 2, \ldots, m$, the matrix $A_{i,\mathcal{M}_\ell}$ can be decomposed as

$$A_{i,\mathcal{M}_\ell} = \left(V^{(k,\ell)}\right)_{i'} \left(\widetilde{V}^{(\ell,k)}\right)^T + \mathcal{E}_{i,\mathcal{M}_\ell},$$

where $i' = \pi(i)$ indicates that the out-node i corresponds to the i'-th row in $\mathcal{N}_k$, with $\mathcal{N}_k = \{i \in [n] \mid C_i^* = k,\ \forall i = 1, \ldots, n\}$. Here, $\left(V^{(k,\ell)}\right)_{i'}$ denotes the i'-th entry of the vector $V^{(k,\ell)} \in \mathbb{R}^{n_k}$, and it satisfies $\left(V^{(k,\ell)}\right)_{i'} \asymp \tau \asymp 1$. The noise term $\mathcal{E}_{i,\mathcal{M}_\ell} = (\mathcal{E}_{ij})_{j\in M_\ell}$ consists of mean-zero sub-Gaussian random variables $\mathcal{E}_{ij}$ with variance proxy σ_{ij}^2, such that

$$\mathbb{E}\left(\mathcal{E}_{ij}^2\right) \le \sigma_{ij}^2 \le \sigma^2.$$

The main idea behind proving (A.39) is to use a series of concentration inequalities. In what follows, we will systematically analyze the concentration properties of the terms involved in (A.39) and subsequently integrate all intermediate results to achieve the desired conclusion.
(i) Note that the row-cluster center is defined as

$$\mu_{k,\mathcal{M}_\ell} = \frac{1}{n_k} \sum_{i\in\mathcal{N}_k} \frac{A_{i,\mathcal{M}_\ell}}{\|A_{i,\mathcal{M}_\ell}\|_2},$$

with

$$\|A_{i,\mathcal{M}_\ell}\|_2^2 = \left\| \left(V^{(k,\ell)}\right)_{i'} \left(\widetilde{V}^{(\ell,k)}\right)^T + \mathcal{E}_{i,\mathcal{M}_\ell} \right\|_2^2$$
$$= \left\{ \left(V^{(k,\ell)}\right)_{i'} \right\}^2 \left\| \widetilde{V}^{(\ell,k)} \right\|_2^2 + \|\mathcal{E}_{i,\mathcal{M}_\ell}\|_2^2 + 2 \left(V^{(k,\ell)}\right)_{i'} \left(\widetilde{V}^{(\ell,k)}\right)^T \mathcal{E}_{i,\mathcal{M}_\ell},$$

where

$$\left\{ \left(V^{(k,\ell)}\right)_{i'} \right\}^2 \left\| \widetilde{V}^{(\ell,k)} \right\|_2^2 \asymp \left\{ \left(V^{(k,\ell)}\right)_{i'} \right\}^2 m_l \tau^2, \tag{A.40}$$

the noise norm is

$$\|\mathcal{E}_{i,\mathcal{M}_\ell}\|_2^2 = \sum_{j\in\mathcal{M}_\ell} \epsilon_{ij}^2$$

where ϵ_{ij}s are sub-Gaussian with mean zero and $\mathbb{E}\epsilon_{ij}^2 \leq \sigma_{ij}^2 \leq \sigma^2$. Hence, we know that ϵ_{ij}^2 is sub-exponential. Thus, by invoking a concentration inequality for sum of independent sub-exponential variables, we know that for any $t > 0$, the following concentration inequality holds:

$$\mathbb{P}\left(\left|\sum_{j\in\mathcal{M}_\ell}\epsilon_{ij}^2 - \mathbb{E}\sum_{j\in\mathcal{M}_\ell}\epsilon_{ij}^2\right| \geq t\right) \leq 2\exp\left\{-\frac{1}{512}\left(\frac{t^2}{m_\ell\sigma^4}\wedge\frac{t}{\sigma^2}\right)\right\},$$

where the symbol $\wedge$ denotes the minimum of the two quantities. Thus, by taking $t \asymp \sqrt{m_\ell \log m_\ell}\cdot\sigma^2$, we can get that with probability exceeding $1 - O(\frac{1}{m_\ell^{10}})$, it has

$$\left|\sum_{j\in\mathcal{M}_\ell}\epsilon_{ij}^2 - \mathbb{E}\sum_{j\in\mathcal{M}_\ell}\epsilon_{ij}^2\right| \lesssim \sqrt{m_\ell\log m_\ell}\cdot\sigma^2,$$

which yields that

$$\|\mathcal{E}_{i,\mathcal{M}_\ell}\|_2^2 = \sum_{j\in\mathcal{M}_\ell}\epsilon_{ij}^2 \leq \{m_\ell + c\sqrt{m_\ell\log m_\ell}\}\sigma^2, \tag{A.41}$$

where c is an absolute constant. In the other hand, $\left(\widetilde{V}^{(\ell,k)}\right)^T\mathcal{E}_{i,\mathcal{M}_\ell}$ is sub-Gaussian with variance proxy $\|\widetilde{V}^{(\ell,k)}\|_2^2\sigma^2 \lesssim m_\ell\tau^2\sigma^2$ and with mean zero. Thus, by invoking the concentration inequality for the sub-Gaussian random variables, we know that

$$\mathbb{P}\left(\left|\left(\widetilde{V}^{(\ell,k)}\right)^T\mathcal{E}_{i,\mathcal{M}_\ell}\right| \geq t\right) \leq 2\exp(-c\frac{t^2}{m_\ell\tau^2\sigma^2}).$$

Then, by taking $t \asymp \sqrt{m_\ell\log m_\ell\cdot(\tau^2\sigma^2)}$, we know that with probability exceeding $1 - O(\frac{1}{m_\ell^{10}})$, it holds that

$$\left|\left(V^{(k,\ell)}\right)_{i'}\left(\widetilde{V}^{(\ell,k)}\right)^T\mathcal{E}_{i,\mathcal{M}_\ell}\right| \lesssim \tau\sqrt{m_\ell\log m_\ell\cdot(\tau^2\sigma^2)}. \tag{A.42}$$

Thus, by combining (A.40), (A.41) and (A.42), we know that with probability exceeding $1 - O(\frac{1}{m_\ell^{10}})$, it holds that

$$\|A_{i,\mathcal{M}_\ell}\|_2 \asymp \sqrt{m_\ell(\tau^4+\sigma^2)}.$$

Next, we recall that

$$\mu_{k,\mathcal{M}_\ell} = \frac{1}{n_k}\sum_{i\in\mathcal{N}_k}\frac{A_{i,\mathcal{M}_\ell}}{\|A_{i,\mathcal{M}_\ell}\|_2}. \tag{A.43}$$

Denote $a_{i\ell} \triangleq \sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2\left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \Sigma_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}\cdot(1+o(1))$, then we know that

$$\mathbb{P}(\|A_{i,\mathcal{M}_\ell}\|_2 = a_{i\ell}, \forall i\in\mathcal{N}_k) \geq 1 - O(\frac{n_k}{m_\ell^{10}}), \tag{A.44}$$

which goes to 1 if $\frac{n_k}{m_\ell^{10}} \to 0$. Then we know that when $\frac{n_k}{m_\ell^{10}} \to 0$, μ_{kM_ℓ} has the same statistical property with the following $\mu_{k\ell}$, which is defined as

$$\mu_{k\ell} = \frac{1}{n_k} \sum_{i \in \mathcal{N}_k} \frac{A_{i,\mathcal{M}_\ell}}{a_{i\ell}} = \frac{1}{n_k} \sum_{i \in \mathcal{N}_k} \frac{\left(V^{(k,\ell)}\right)_{i'} \left(\widetilde{V}^{(\ell,k)}\right)^T}{a_{i\ell}} + \frac{1}{n_k} \sum_{i \in \mathcal{N}_k} \frac{\mathcal{E}_{i,\mathcal{M}_\ell}}{a_{i\ell}}.$$

Furthermore,

$$\mathbb{E}\mu_{k\ell} = \frac{1}{n_k} \sum_{i \in \mathcal{N}_k} \frac{\left(V^{(k,\ell)}\right)_{i'} \left(\widetilde{V}^{(\ell,k)}\right)^T}{a_{i\ell}} \triangleq \alpha_{k\ell} \cdot \left(\widetilde{V}^{(\ell,k)}\right)^T, \tag{A.45}$$

with $\alpha_{k\ell} \triangleq \frac{1}{n_k} \sum_{i \in \mathcal{N}_k} \frac{\left(V^{(k,\ell)}\right)_{i'}}{a_{i\ell}} \asymp \frac{1}{\sqrt{m_\ell}}$, where $i' = \pi(i)$. For all $j \in \mathcal{M}_\ell$, define $S_j^{(k,\ell)}$ as

$$S_j^{(k,\ell)} \triangleq \frac{1}{n_k} \sum_{i \in \mathcal{N}_k} \frac{\mathcal{E}_{ij}}{a_{i\ell}},$$

which is sub-Gaussian with variance proxy:

$$\frac{1}{n_k^2} \sum_{i \in \mathcal{N}_k} \frac{\sigma_{ij}^2}{a_{i\ell}^2} \asymp \frac{\sigma^2}{n_k m_\ell (\tau^4 + \sigma^2)} (1 + o(1)).$$

Then, by the property of sub-Gaussian tail bound, we know that

$$\mathbb{P}\left(\left|S_j^{(k,\ell)}\right| \geq t\right) \leq 2 \exp\left(-\frac{t^2}{\frac{2}{n_k^2} \sum_{i \in \mathcal{N}_k} \frac{\sigma_{ij}^2}{a_{i\ell}^2}}\right).$$

Furthermore, we take

$$t \asymp \frac{1}{n_k} \sqrt{\sum_{i \in \mathcal{N}_k} \frac{\sigma_{ij}^2}{a_{i\ell}^2}} \cdot \sqrt{\log n_k},$$

which yields that, with probability exceeding $1 - O(\frac{1}{n_k^{10}})$, it has

$$\left|S_j^{(k,\ell)}\right| \lesssim \frac{\sqrt{\log n_k}}{n_k} \sqrt{\sum_{i \in \mathcal{N}_k} \frac{\sigma_{ij}^2}{a_{i\ell}^2}} \asymp \sqrt{\frac{\log n_k}{n_k m_\ell}} \cdot \sqrt{\frac{\sigma^2}{\tau^4 + \sigma^2}} \cdot (1 + o(1)). \tag{A.46}$$

Thus, by combing (A.43), (A.44), (A.45) and (A.46), we can get that with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{m_\ell}{n_k^{10}})$, it holds that

$$\mu_{kM_\ell} = \left\{ \frac{1 + o(1)}{n_k} \sum_{i \in N_k} \frac{\left(V^{(k,\ell)}\right)_{i'}}{a_{i\ell}} \right\} \cdot \left(\widetilde{V}^{(\ell,k)}\right)^T = \alpha_{kl}\{1 + o(1)\} \cdot \left(\widetilde{V}^{(\ell,k)}\right)^T, \tag{A.47}$$

$$\|\mu_{kM_\ell}\|_2 = |\alpha_{kl}| \|\widetilde{V}^{(\ell,k)}\|_2 \{1 + o(1)\}. \tag{A.48}$$

Recall that with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}})$, it has

$$\|A_{i,\mathcal{M}_\ell}\|_2 = \{1+o(1)\}\sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}, \quad \forall i \in \mathcal{N}_k. \tag{A.49}$$

By (A.47), we can also get that with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{m_\ell}{n_k^{10}})$, it holds that

$$\begin{aligned} A_{i,\mathcal{M}_\ell}\mu_{k,\mathcal{M}_\ell}^\top &= \left\{\left(V^{(k,\ell)}\right)_{i'}\left(\widetilde{V}^{(\ell,k)}\right)^T + \mathcal{E}_{i,\mathcal{M}_\ell}\right\}\left\{\alpha_{kl}\{1+o(1)\}\cdot\left(\widetilde{V}^{(\ell,k)}\right)\right\} \\ &= \left(V^{(k,\ell)}\right)_{i'}\alpha_{kl}\{1+o(1)\}\cdot\left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \alpha_{kl}\{1+o(1)\}\cdot\mathcal{E}_{i,\mathcal{M}_\ell}\widetilde{V}^{(\ell,k)}. \end{aligned} \tag{A.50}$$

In the other hand, we know that $\mathcal{E}_{i,\mathcal{M}_\ell}\widetilde{V}^{(\ell,k)}$ is sub-Gaussian with variance proxy:

$$\sum_{j\in\mathcal{M}_\ell}\left(\widetilde{V}^{(\ell,k)}\right)_{j'}^2\sigma_{ij}^2,$$

where $j' = \tilde{\pi}(j)$ indicates that the in-node j corresponds to the j'-th column in $\mathcal{M}_\ell$. Then, by the property of sub-Gaussian tail bound, we know that

$$\mathbb{P}\left(\left|\mathcal{E}_{i,\mathcal{M}_\ell}\widetilde{V}^{(\ell,k)}\right| > t\right) \le 2\exp\left(-\frac{t^2}{2\sum_{j\in\mathcal{M}_\ell}\left(\widetilde{V}^{(\ell,k)}\right)_{j'}^2\sigma_{ij}^2}\right).$$

Furthermore, we take

$$t \asymp \sqrt{\sum_{j\in\mathcal{M}_\ell}\left(\widetilde{V}^{(\ell,k)}\right)_{j'}^2\sigma_{ij}^2}\cdot\sqrt{\log m_\ell},$$

which yields that, with probability exceeding $1 - O(\frac{1}{m_\ell^{10}})$, it has

$$\left|\mathcal{E}_{i,\mathcal{M}_\ell}\widetilde{V}^{(\ell,k)}\right| \lesssim \sqrt{\sum_{j\in\mathcal{M}_\ell}\left(\widetilde{V}^{(\ell,k)}\right)_{j'}^2\sigma_{ij}^2}\cdot\sqrt{\log m_\ell} \lesssim \sqrt{(m_\ell\log m_\ell)\tau^2\sigma^2},$$

which, combining (A.50), yields that with probability exceeding $1-O(\frac{n_k}{m_\ell^{10}})-O(\frac{m_\ell}{n_k^{10}})$, it holds that

$$A_{i,\mathcal{M}_\ell}\mu_{k,\mathcal{M}_\ell}^\top = \left(V^{(k,\ell)}\right)_{i'}\alpha_{kl}\{1+o(1)\}\cdot\left\|\widetilde{V}^{(\ell,k)}\right\|_2^2. \tag{A.51}$$

Then, by combing (A.51), (A.49) and (A.48), we can get that with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{m_\ell}{n_k^{10}})$, it holds that

$$\begin{aligned}\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}\right) =& \frac{A_{i,\mathcal{M}_\ell}\mu_{k,\mathcal{M}_\ell}^\top}{\|A_{i,\mathcal{M}_\ell}\|_2 \cdot \|\mu_{k,\mathcal{M}_\ell}\|_2} \\ =& \frac{\left(V^{(k,\ell)}\right)_{i'} \alpha_{kl} \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2}{|\alpha_{kl}|\|\widetilde{V}^{(\ell,k)}\|_2 \sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}} \cdot \{1+o(1)\}. \\ =& \frac{\left(V^{(k,\ell)}\right)_{i'} \alpha_{kl} \left\|\widetilde{V}^{(\ell,k)}\right\|_2}{|\alpha_{kl}| \sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}} \cdot \{1+o(1)\}.\end{aligned}$$

That is, when $C_i^* = k$, with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{m_\ell}{n_k^{10}})$, it holds that

$$\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}\right) = \frac{\left(V^{(k,\ell)}\right)_{i'} \alpha_{kl} \left\|\widetilde{V}^{(\ell,k)}\right\|_2}{|\alpha_{kl}| \sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}} \cdot \{1+o(1)\}. \tag{A.52}$$

In the other hand, when $C_i^* = k$ but $k \neq k'$, consider the cosine similarity

$$\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k',\mathcal{M}_\ell}\right) = \frac{A_{i,\mathcal{M}_\ell}\mu_{k',\mathcal{M}_\ell}^\top}{\|A_{i,\mathcal{M}_\ell}\|_2 \cdot \|\mu_{k',\mathcal{M}_\ell}\|_2}.$$

Note that with probability exceeding $1 - O(\frac{n_{k'}}{m_\ell^{10}}) - O(\frac{m_\ell}{n_{k'}^{10}})$, it holds that

$$A_{i,\mathcal{M}_\ell}\mu_{k',\mathcal{M}_\ell}^\top = \left\{\left(V^{(k,\ell)}\right)_{i'}\left(\widetilde{V}^{(\ell,k)}\right)^T + \mathcal{E}_{i,\mathcal{M}_\ell}\right\}\left\{\alpha_{k'l}\{1+o(1)\} \cdot \left(\widetilde{V}^{(\ell,k')}\right)\right\},$$

where $\mathcal{E}_{i,\mathcal{M}_\ell}\widetilde{V}^{(\ell,k')}$, with probability exceeding $1 - O(\frac{1}{m_\ell^{10}})$, satisfies that

$$\left|\mathcal{E}_{i,\mathcal{M}_\ell}\widetilde{V}^{(\ell,k')}\right| \lesssim \sqrt{\sum_{j\in\mathcal{M}_\ell}\left(\widetilde{V}^{(\ell,k')}\right)_{j'}^2 \sigma_{ij}^2} \cdot \sqrt{\log m_\ell} \lesssim \sqrt{(m_\ell \log m_\ell)\tau^2\sigma^2}.$$

Thus, we know that with probability exceeding $1 - O(\frac{n_{k'}}{m_\ell^{10}}) - O(\frac{m_\ell}{n_{k'}^{10}})$, it holds that

$$A_{i,\mathcal{M}_\ell}\mu_{k',\mathcal{M}_\ell}^\top = \left(V^{(k,\ell)}\right)_{i'} \alpha_{k'l}\{1+o(1)\}\left(\widetilde{V}^{(\ell,k)}\right)^T \widetilde{V}^{(\ell,k')} + \alpha_{k'l}\{1+o(1)\}O\left(\sqrt{(m_\ell \log m_\ell)\tau^2\sigma^2}\right),$$

which yields that with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{n_{k'}}{m_\ell^{10}}) - O(\frac{m_\ell}{n_{k'}^{10}})$, it holds that

$$\begin{aligned}&\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k',\mathcal{M}_\ell}\right) \\ =& \frac{\left(V^{(k,\ell)}\right)_{i'} \alpha_{k'l}\{1+o(1)\}\left(\widetilde{V}^{(\ell,k)}\right)^T \widetilde{V}^{(\ell,k')} + \alpha_{k'l}\{1+o(1)\}O\left(\sqrt{(m_\ell \log m_\ell)\tau^2\sigma^2}\right)}{\sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2} \cdot |\alpha_{k'l}|\|\widetilde{V}^{(\ell,k')}\|_2}.\end{aligned} \tag{A.53}$$

By invoking (A.52), we know that with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{m_\ell}{n_k^{10}})$, it has

$$|\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}\right)| = \frac{|\left(V^{(k,\ell)}\right)_{i'}| \left\|\widetilde{V}^{(\ell,k)}\right\|_2}{\sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}} \cdot \{1 + o(1)\}. \tag{A.54}$$

While, by invoking (A.53), with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{n_{k'}}{m_\ell^{10}}) - O(\frac{m_\ell}{n_{k'}^{10}})$, it has

$$|\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k',\mathcal{M}_\ell}\right)| \le \frac{|\left(V^{(k,\ell)}\right)_{i'}| \cdot \left|\left(\widetilde{V}^{(\ell,k)}\right)^T \frac{\widetilde{V}^{(\ell,k')}}{\|\widetilde{V}^{(\ell,k')}\|_2}\right| + O(\sqrt{\sigma^2 \log m_\ell})}{\sqrt{\left\{\left(V^{(k,\ell)}\right)_{i'}\right\}^2 \left\|\widetilde{V}^{(\ell,k)}\right\|_2^2 + \sum_{j\in\mathcal{M}_\ell}\sigma_{ij}^2}} \cdot \{1 + o(1)\}. \tag{A.55}$$

Thus, for all $k' \neq k = C_i^*$, combing (A.54), (A.55) and the assumption that $\widetilde{V}^{(\ell,k)}$ and $\widetilde{V}^{(\ell,k')}$ are linearly independent, with probability exceeding $1 - O(\frac{n_k}{m_\ell^{10}}) - O(\frac{m_\ell}{n_k^{10}}) - O(\frac{n_{k'}}{m_\ell^{10}}) - O(\frac{m_\ell}{n_{k'}^{10}})$, it has

$$|\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k,\mathcal{M}_\ell}\right)| > |\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k',\mathcal{M}_\ell}\right)|.$$

Furthermore, if we assume that $n \asymp m$ and all the row clusters and column clusters are balanced, then with probability exceeding $1 - O(\frac{1}{n^8})$, it holds that

$$|\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{C_i^*,\mathcal{M}_\ell}\right)| > |\cos\left(A_{i,\mathcal{M}_\ell}, \mu_{k',\mathcal{M}_\ell}\right)| \text{ with } k' \neq C_i^*, \ \ \forall i \in [n], \ \ell \in [L]. \tag{A.56}$$

Similarly, we can get that with probability exceeding $1 - O(\frac{1}{m^8})$, it holds that

$$|\cos\left(A_{\mathcal{N}_k,j}, \tilde{\mu}_{\mathcal{N}_k,Z_j^*}\right)| > \cos\left(A_{\mathcal{N}_k,j}, \tilde{\mu}_{\mathcal{N}_k,\ell'}\right)| \text{ with } \ell' \neq Z_j^*, \ \ \forall j \in [m], \ k \in [K]. \tag{A.57}$$

Combing (A.56) and (A.57), we know that with probability exceeding $1 - O(\frac{1}{n^8}) - O(\frac{1}{m^8})$, it holds that $(C, Z) = (C_*, Z_*)$ is the Pure Strategy Nash Equilibrium (PSNE) of the game. ■

A.5 Proof of (29) in Remark 12.

Suppose that in out-community 1, there are $N(1-\rho_1)$ nodes correctly assigned to community 1 and $N\rho_1$ nodes misclassified into community 2. Similarly, in out-community 2, there are $N(1-\rho_2)$ nodes correctly assigned to community 2 and $N\rho_2$ nodes misclassified into community 1. Hence, the overall misclassification proportion is

$$\rho_{n,m} = \frac{\rho_1 + \rho_2}{2}.$$

We suppose that

$$\Lambda = \begin{pmatrix} \Lambda^{(1,1)} & \Lambda^{(1,2)} \\ \Lambda^{(2,1)} & \Lambda^{(2,2)} \end{pmatrix}, \qquad \widetilde{\Lambda} = \begin{pmatrix} \widetilde{\Lambda}^{(1,1)} & \widetilde{\Lambda}^{(1,2)} \\ \widetilde{\Lambda}^{(2,1)} & \widetilde{\Lambda}^{(2,2)} \end{pmatrix},$$

and further decompose

$$\Lambda^{(1,1)} = (\Lambda_{11}^{(a)}, \Lambda_{11}^{(b)})^T, \quad \Lambda^{(1,2)} = (\Lambda_{12}^{(a)}, \Lambda_{12}^{(b)})^T, \quad \Lambda^{(2,1)} = (\Lambda_{21}^{(a)}, \Lambda_{21}^{(b)})^T, \quad \Lambda^{(2,2)} = (\Lambda_{22}^{(a)}, \Lambda_{22}^{(b)})^T,$$

where, for instance, $\Lambda_{11}^{(a)}$ corresponds to the node popularities of out-community 1 nodes correctly clustered into in-community 1, while $\Lambda_{11}^{(b)}$ corresponds to those misclassified from out-community 1, and the other terms are defined analogously. Let

$$\widetilde{P} = \begin{pmatrix} \widetilde{P}^{(1,1)} & \widetilde{P}^{(1,2)} \\ \widetilde{P}^{(2,1)} & \widetilde{P}^{(2,2)} \end{pmatrix} \triangleq P(C, Z).$$

Thus, for $\widetilde{P}^{(1,1)}$, we know that

$$\widetilde{P}^{(1,1)} = \begin{pmatrix} \Lambda_{11}^{(a)}(\widetilde{\Lambda}^{(1,1)})^T \\ \Lambda_{21}^{(b)}(\widetilde{\Lambda}^{(1,2)})^T \end{pmatrix} = \begin{pmatrix} \Lambda_{11}^{(a)} & \mathbf{0} \\ \mathbf{0} & \Lambda_{21}^{(b)} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} (\widetilde{\Lambda}^{(1,1)})^T \\ (\widetilde{\Lambda}^{(1,2)})^T \end{pmatrix}.$$

Since $(\widetilde{\Lambda}^{(1,1)}, \widetilde{\Lambda}^{(1,2)})$ is column-orthogonal, it follows that

$$\begin{aligned} \|\widetilde{P}^{(1,1)}\|_F^2 &= \|\Lambda_{11}^{(a)}\|_F^2 \|\widetilde{\Lambda}^{(1,1)}\|_F^2 + \|\Lambda_{21}^{(b)}\|_F^2 \|\widetilde{\Lambda}^{(1,2)}\|_F^2, \\ \|\widetilde{P}^{(1,1)}\|_{op}^2 &= \|\Lambda_{11}^{(a)}\|_F^2 \|\widetilde{\Lambda}^{(1,1)}\|_F^2. \end{aligned} \tag{A.58}$$

The identity in (A.58) holds because ρ_1 is assumed to be much smaller than $(1-\rho_1)$, which implies

$$\|\Lambda_{11}^{(a)}\|_F^2 \|\widetilde{\Lambda}^{(1,1)}\|_F^2 < \|\Lambda_{21}^{(b)}\|_F^2 \|\widetilde{\Lambda}^{(1,2)}\|_F^2.$$

Consequently,

$$\|\widetilde{P}^{(1,1)}\|_F^2 - (1+\alpha_{n,m})\|\widetilde{P}^{(1,1)}\|_{op}^2 \asymp nm\delta^4 \left\{\rho_2 - \alpha_{n,m}(1-\rho_1)\right\}.$$

By analogous arguments, we obtain

$$\begin{aligned} \|\widetilde{P}^{(1,2)}\|_F^2 - (1+\alpha_{n,m})\|\widetilde{P}^{(1,2)}\|_{op}^2 &\asymp nm\delta^4 \left\{\rho_2 - \alpha_{n,m}(1-\rho_1)\right\}, \\ \|\widetilde{P}^{(2,1)}\|_F^2 - (1+\alpha_{n,m})\|\widetilde{P}^{(2,1)}\|_{op}^2 &\asymp nm\delta^4 \left\{\rho_1 - \alpha_{n,m}(1-\rho_2)\right\}, \\ \|\widetilde{P}^{(2,2)}\|_F^2 - (1+\alpha_{n,m})\|\widetilde{P}^{(2,2)}\|_{op}^2 &\asymp nm\delta^4 \left\{\rho_1 - \alpha_{n,m}(1-\rho_2)\right\}. \end{aligned}$$

Summing over all blocks yields

$$\begin{aligned} \|P_*\|_F^2 - (1+\alpha_{n,m}) \sum_{k=1}^{K}\sum_{\ell=1}^{L} \|P_*^{(k,\ell)}(C,Z)\|_{op}^2 &= \sum_{k=1}^{K}\sum_{\ell=1}^{L} \left\{ \|\widetilde{P}^{(k,\ell)}\|_F^2 - (1+\alpha_{n,m})\|\widetilde{P}^{(k,\ell)}\|_{op}^2 \right\} \\ &\asymp nm\delta^4 \left[(\rho_1+\rho_2) - \alpha_{n,m}\{2-(\rho_1+\rho_2)\}\right] \\ &\asymp nm\delta^4 \left\{\rho_{n,m} - \alpha_{n,m}(1-\rho_{n,m})\right\}. \end{aligned}$$

Therefore, to satisfy condition (27), it suffices that there exists $\alpha_{n,m} \in (0, 1/2)$ such that

$$mn\,\delta^4 \left[\rho_{n,m} - \alpha_{n,m}(1-\rho_{n,m})\right] \gtrsim \frac{\sigma_{\max}^2}{\alpha_{n,m}}(m+n).$$

Equivalently, this requires

$$\begin{aligned}\rho_{n,m} &\gtrsim \frac{\sigma_{\max}^2\,(m+n)}{\alpha_{n,m}(1+\alpha_{n,m})\,nm\,\delta^4} \;+\; \frac{\alpha_{n,m}}{1+\alpha_{n,m}} \\ &\asymp \frac{\sigma_{\max}^2}{\delta^4}\cdot\frac{m+n}{mn}\cdot\frac{1}{\alpha_{n,m}} \;+\; \alpha_{n,m}.\end{aligned}$$

In particular, the condition is satisfied if

$$\rho_{n,m} \;\gtrsim\; \frac{\sigma_{\max}}{\delta^2}\sqrt{\frac{m+n}{nm}},$$

which completes the proof. ■

A.6 Derivation of $P = XY^\top$ in Remark 1

Let $\widetilde{P}$ be the matrix obtained by permuting the rows and columns of P so that vertices are grouped by their community memberships $c_i \in [K]$ and $z_j \in [L]$ in increasing order. Then there exist permutation matrices Π_L and Π_R such that

$$\widetilde{P} = \Pi_L^\top P \Pi_R,$$

and $\widetilde{P}$ has a $K \times L$ block structure with blocks

$$\widetilde{P}^{(k,\ell)} = \Lambda^{(k,\ell)}\big[\widetilde{\Lambda}^{(\ell,k)}\big]^\top, \qquad k \in [K],\ \ell \in [L].$$

Then we can show that

$$\widetilde{P} = \widetilde{X}\widetilde{Y}^\top$$

where

$$\widetilde{X} = \begin{bmatrix} \Lambda^{(1,1)} & \Lambda^{(1,2)} & \cdots & \Lambda^{(1,L)} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \Lambda^{(2,1)} & \cdots & \Lambda^{(2,L)} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \Lambda^{(K,1)} & \cdots & \Lambda^{(K,L)} \end{bmatrix} \in \mathbb{R}^{n\times KL},$$

$$\widetilde{Y} = \begin{bmatrix} \widetilde{\Lambda}^{(1,1)} & 0 & \cdots & 0 & \widetilde{\Lambda}^{(1,2)} & \cdots & 0 & \widetilde{\Lambda}^{(1,K)} & 0 & 0 \\ 0 & \widetilde{\Lambda}^{(2,1)} & \cdots & 0 & 0 & \widetilde{\Lambda}^{(2,2)} & 0 & 0 & \widetilde{\Lambda}^{(2,K)} & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \widetilde{\Lambda}^{(L,1)} & 0 & 0 & \widetilde{\Lambda}^{(L,2)} & 0 & 0 & \widetilde{\Lambda}^{(L,K)} \end{bmatrix} \in \mathbb{R}^{m\times KL}.$$

Finally, undoing the permutation gives

$$P = \Pi_L\,\widetilde{P}\,\Pi_R^\top = \Pi_L\,\widetilde{X}\widetilde{Y}^\top\,\Pi_R^\top = XY^\top,$$

where $X = \Pi_L\widetilde{X}$ and $Y = \Pi_R\widetilde{Y}$. ■

Appendix B. Auxiliary Lemmas and Proofs

Lemma 1. *Let the elements of $n \times m$ matrix Ξ be independent, mean zero, sub-Gaussian errors, i.e., $\Xi_{ij} \sim subG(\sigma_{ij}^2)$, for any $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. Denote by $\sigma_{max}^2 = \max_{1\leq i\leq n, 1\leq j\leq m}\{\sigma_{ij}^2\}$. The matrix Ξ is partitioned into KL sub-matrices $\Xi^{(k,\ell)}$, where $k = 1, 2, \ldots, K$ and $l = 1, 2, \ldots, L$. Then, for any $t \geq 0$, it holds that*

$$\mathbb{P}\left[\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\right\|_{op}^2 \leq C\sigma_{max}^2\left\{nL + mK + KL\log(2KL) + KLt\right\}\right] \geq 1 - e^{-t}, \tag{A.59}$$

where C is some positive absolute constant.

Proof of Lemma 1. By invoking Theorem 4.4.5 (norm of matrices with sub-Gaussian entries) on page 91 in Papaspiliopoulos (2020), we can get that, for any $\tilde{t} \geq 0$, it holds that

$$\mathbb{P}\left[\left\|\Xi^{(k,\ell)}\right\|_{op} \leq \widetilde{C}\tau_{max}\left\{\sqrt{n_k} + \sqrt{m_l} + \tilde{t}\right\}\right] \geq 1 - 2e^{-\tilde{t}},$$

where $\widetilde{C}$ is some positive absolute constant and $\tau_{max}^2 = \max_{1\leq i\leq n, 1\leq j\leq m}\left\|\Xi_{ij}^{(k,\ell)}\right\|_{\psi_2}$ with $\|\cdot\|_{\psi_2}$ denoting the sub-Gaussian norm of a sub-Gaussian random variable. On the other hand, it can be shown that

$$\tau_{max}^2 \leq 8\sigma_{max}^2.$$

Thus, we can get that

$$\mathbb{P}\left[\left\|\Xi^{(k,\ell)}\right\|_{op} \leq 2\sqrt{2}\widetilde{C}\sigma_{max}\left\{\sqrt{n_k} + \sqrt{m_l} + \tilde{t}\right\}\right] \geq 1 - 2e^{-\tilde{t}^2}.$$

Then, by setting $\tilde{t} = \sqrt{\log(2KL) + t}$, with $t \geq 0$, we can get that

$$\mathbb{P}\left[\left\|\Xi^{(k,\ell)}\right\|_{op}^2 \leq 24\widetilde{C}^2\sigma_{max}^2\left\{n_k + m_l + \log(2KL) + t\right\}\right] \geq 1 - e^{-\{\log(KL)+t\}}.$$

Then, it can be obtained that

$$\begin{aligned}
& \mathbb{P}\left[\bigcup_{k\in[K]}\bigcup_{l\in[L]}\left\{\left\|\Xi^{(k,\ell)}\right\|_{op}^2 > 24\widetilde{C}^2\sigma_{max}^2\left\{n_k + m_l + \log(2KL) + t\right\}\right\}\right] \\
\leq & \sum_{k=1}^{K}\sum_{\ell=1}^{L} e^{-\{\log(KL)+t\}} \\
= & e^{-t}.
\end{aligned}$$

Thus, we know that

$$\mathbb{P}\left[\bigcap_{k\in[K]}\bigcap_{l\in[L]}\left\{\left\|\Xi^{(k,\ell)}\right\|_{op}^2 \leq 24\widetilde{C}^2\sigma_{max}^2\left\{n_k + m_l + \log(2KL) + t\right\}\right\}\right] \geq 1 - e^{-t}.$$

So, it holds that

$$\mathbb{P}\left[\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}\right\|_{op}^2 \le C\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KLt\right\}\right] \ge 1-e^{-t}.$$

where $C = 24\widetilde{C}^2$ is a positive absolute constant, and the proof is completed. ■

Lemma 2. *Let the elements of $n\times m$ matrix Ξ be independent, mean zero, sub-Gaussian errors, i.e., $\Xi_{ij} \sim subG(\sigma_{ij}^2)$, for any $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. Let $\sigma_{max}^2 = \max_{1\le i\le n, 1\le j\le m}\{\sigma_{ij}^2\}$. Then, for any $t \ge 0$, with probability at least $1-e^{-t}$, it holds that*

$$\begin{aligned}&\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2 \qquad\qquad (A.60)\\ \le\ & C\sigma_{max}^2\Big\{n\widehat{L}+m\widehat{K}+\widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L})+\widehat{K}\widehat{L}(n\log\widehat{K}+\log n+m\log\widehat{L}+\log m+t)\Big\},\end{aligned}$$

where C is an absolute constant.

Proof of Lemma 2. By applying Lemma 1, for any fixed (K, L, C, Z), we can conclude that for any $x \ge 0$, it holds that

$$\mathbb{P}\left[\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(C,Z)\right\|_{op}^2 \le C\sigma_{max}^2\left\{nL+mK+KL\log(2KL)+KLx\right\}\right] \ge 1-e^{-x}. \quad (A.61)$$

Then, by using the union bound over $K \in [n]$, $L \in [m]$, $C \in \mathcal{M}_{n,K}$ and $Z \in \mathcal{M}_{m,L}$ and setting $x = n\log\widehat{K}+\log n+m\log\widehat{L}+\log m+t$, we can obtain that

$$\begin{aligned}&\mathbb{P}\left\{\sum_{k=1}^{\widehat{K}}\sum_{\ell=1}^{\widehat{L}}\left\|\Xi^{(k,\ell)}(\widehat{C},\widehat{Z})\right\|_{op}^2 > \Delta(\widehat{K},\widehat{L})\right\}\\ \le\ & \mathbb{P}\left[\bigcup_{\substack{K\in[n]\\L\in[m]}}\bigcup_{\substack{C\in\mathbb{M}_{n,K}\\Z\in\mathbb{M}_{m,L}}}\left\{\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(C,Z)\right\|_{op}^2 > \Delta(K,L)\right\}\right]\\ \le\ & \sum_{K=1}^{n}\sum_{\ell=1}^{m}K^nL^m\mathbb{P}\left\{\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\Xi^{(k,\ell)}(C,Z)\right\|_{op}^2 > \Delta(K,L)\right\}\\ \le\ & \sum_{K=1}^{n}\sum_{\ell=1}^{m}K^nL^me^{-(n\log K+\log n+m\log L+\log m+t)}\\ \le\ & e^{-t},\end{aligned}$$

where $\Delta(\widehat{K},\widehat{L}) = C\sigma_{max}^2\Big\{n\widehat{L}+m\widehat{K}+\widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L})+\widehat{K}\widehat{L}(n\log\widehat{K}+\log n+m\log\widehat{L}+\log m+t)\Big\}$ and $\Delta(K,L) = C\sigma_{max}^2\Big\{nL+mK+KL\log(2KL)+KL(n\log K+\log n+m\log L+\log m+$

$t)\Big\}$. Thus we know that for any $t \geq 0$, with high probability at least $1 - e^{-t}$, it holds that

$$\begin{aligned} & \sum_{k=1}^{\widehat{K}} \sum_{\ell=1}^{\widehat{L}} \left\| \Xi^{(k,\ell)}(\widehat{C}, \widehat{Z}) \right\|_{op}^2 \\ \leq \; & C\sigma_{max}^2 \Big\{ n\widehat{L} + m\widehat{K} + \widehat{K}\widehat{L}\log(2\widehat{K}\widehat{L}) + \widehat{K}\widehat{L}(n\log\widehat{K} + \log n + m\log\widehat{L} + \log m + t) \Big\}, \end{aligned}$$

where C is an absolute constant, and the entire proof concludes. ■

Lemma 3. *Assume $\xi = (\xi_1, \xi_2, \ldots, \xi_n)$ is a vector of independent, mean zero, sub-Gaussian errors with $\xi_i \sim subG(\sigma_{ij}^2)$, and $\sigma_{max}^2 = \max_{i\in[n]} \left\{ \sigma_{ij}^2 \right\}$. Assume $h = (h_1, h_2, \ldots, h_n)$ is a fixed vector with $\|h\|_F^2 = \sum_{i=1}^n h_i^2 = 1$. Then, for any $t \geq 0$, it holds that*

$$\mathbb{P}\left(\left| \xi^T h \right|^2 \geq \sigma_{max}^2 t \right) \leq 2e^{-t/2}.$$

Proof of Lemma 3. By the assumptions of Lemma 3, it can be shown that, for any $\lambda \in \mathbb{R}$, it holds that

$$\mathbb{E}e^{\lambda \xi^T h} = e^{\frac{\lambda^2}{2}\left(\sum_{i=1}^n h_i^2 \sigma_i^2 \right)} \leq e^{\frac{\lambda^2}{2}\sigma_{max}^2},$$

which means that $\xi^T h$ is a sub-Gaussian random variable with variance proxy σ_{max}^2. So, by the property of sub-Gaussian random variable, we know that for any $x \geq 0$, it holds that

$$\mathbb{P}\left(\left| \xi^T h \right|^2 \geq x \right) = \mathbb{P}\left(\left| \xi^T h \right| \geq \sqrt{x} \right) \leq 2e^{-\frac{x}{2\sigma_{max}^2}}. \tag{A.62}$$

Finally, by plugging $x = t\sigma_{max}^2$ into (A.62), we know that for any $t \geq 0$, it holds that

$$\mathbb{P}\left(\left| \xi^T h \right|^2 \geq t\sigma_{max}^2 \right) \leq 2e^{-t/2},$$

which concludes the entire proof. ■

Lemma 4 (Glicksberg (1952); Fan (1952); Debreu (1952))**.** *Consider a strategic form game*

$$\langle \mathcal{I}, (S_i)_{i\in\mathcal{I}}, (u_i)_{i\in\mathcal{I}} \rangle$$

such that for each $i \in \mathcal{I}$:

- *S_i is compact and convex;*
- *$u_i(s_i, s_{-i})$ is continuous in s_{-i};*
- *$u_i(s_i, s_{-i})$ is continuous and concave in s_i [in fact, quasi-concavity suffices].*

Then a pure strategy Nash equilibrium exists.

Appendix C. Integration of DOM with Subsampling Frameworks

C.1 Simplified DOM-Augmented Subsampling via SONNET

Algorithm 4 SimpleSONNET-DOM

Input: Adjacency matrix A, number of subgraphs s, overlap sizes $o^{(out)}$, $o^{(in)}$
Output: Final estimated community labels $\hat{\boldsymbol{c}}$, $\hat{\boldsymbol{z}}$

1: **Division Step:** Select overlapping subsets $\mathcal{S}_0^{(out)} \subset \mathcal{N}$ of size $o^{(out)}$, $\mathcal{S}_0^{(in)} \subset \mathcal{M}$ of size $o^{(in)}$
2: Partition $\mathcal{N} \setminus \mathcal{S}_0^{(out)}$ into s disjoint subsets $\widetilde{\mathcal{S}}_1^{(out)}, \ldots, \widetilde{\mathcal{S}}_s^{(out)}$
3: Partition $\mathcal{M} \setminus \mathcal{S}_0^{(in)}$ into s disjoint sets $\widetilde{\mathcal{S}}_1^{(in)}, \ldots, \widetilde{\mathcal{S}}_s^{(in)}$
4: **for** $i = 1$ to s **do**
5: Define subnetwork: $\mathcal{S}_i^{(out)} = \mathcal{S}_0^{(out)} \cup \widetilde{\mathcal{S}}_i^{(out)}$, $\mathcal{S}_i^{(in)} = \mathcal{S}_o^{(in)} \cup \widetilde{\mathcal{S}}_i^{(in)}$
6: **Detection Step:** Apply DOM to $A_{\mathcal{S}_i^{(out)}, \mathcal{S}_i^{(in)}}$ to obtain $\widehat{C}^{(i)}$, $\widehat{Z}^{(i)}$
7: **end for**
8: **Stitching Step:**
9: **for** $i = 2$ to s **do**
10: Compute permutations: $E_i^{(out)} = \mathcal{D}\left(\widehat{C}^{(i)}_{\mathcal{S}_0^{(out)}}, \widehat{C}^{(1)}_{\mathcal{S}_0^{(out)}}\right)$, $E_i^{(in)} = \mathcal{D}\left(\widehat{Z}^{(i)}_{\mathcal{S}_0^{(in)}}, \widehat{Z}^{(1)}_{\mathcal{S}_0^{(in)}}\right)$
11: Align clustering matrices: $\widehat{C}^{(i)} \leftarrow \widehat{C}^{(i)} E_i^{(out)}$, $\widehat{Z}^{(i)} \leftarrow \widehat{Z}^{(i)} E_i^{(in)}$
12: **end for**
13: Set $\widehat{C}^{(\text{final})}$ and $\widehat{Z}^{(\text{final})}$ as:
$\widehat{C}^{(\text{final})} = \widehat{C}^{(1)}_{\mathcal{S}_0^{(out)}} + \widehat{C}^{(1)}_{\widetilde{\mathcal{S}}_1^{(out)}} + \ldots + \widehat{C}^{(s)}_{\widetilde{\mathcal{S}}_s^{(out)}}$, $\widehat{Z}^{(\text{final})} = \widehat{Z}^{(1)}_{\mathcal{S}_0^{(in)}} + \widehat{Z}^{(1)}_{\widetilde{\mathcal{S}}_1^{(in)}} + \ldots + \widehat{Z}^{(s)}_{\widetilde{\mathcal{S}}_s^{(in)}}$
14: Assign final labels $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{z}}$ with
$\hat{c}_i = \arg\max_{k \in [K]} \widehat{C}^{(\text{final})}_{ik}$, $\hat{z}_j = \arg\max_{\ell \in [L]} \widehat{Z}^{(\text{final})}_{j\ell}$ for all $i \in [n]$, $j \in [m]$

C.2 DOM-Augmented Subsampling via PACE

The integration of the Delete-One-Method (DOM) with the Piecewise Averaged Community Estimation (PACE) framework proceeds in three sequential stages: subgraph selection, detection, and patching. In the subgraph selection stage, s subgraphs are constructed by independently sampling subsets of out-nodes and in-nodes from the node sets $\mathcal{N}$ and $\mathcal{M}$. For each $i = 1, \ldots, s$, we randomly choose subsets $\widetilde{\mathcal{S}}_i^{(\text{out})} \subset \mathcal{N}$ and $\widetilde{\mathcal{S}}_i^{(\text{in})} \subset \mathcal{M}$ of sizes $e^{(\text{out})}$ and $e^{(\text{in})}$, respectively. To ensure that each subgraph is well-defined, nodes with zero out-degree or in-degree in the induced submatrix $A_{(\widetilde{\mathcal{S}}_i^{(\text{out})}, \widetilde{\mathcal{S}}_i^{(\text{in})})}$ are removed, yielding the cleaned subsets $\mathcal{S}_i^{(\text{out})}$ and $\mathcal{S}_i^{(\text{in})}$.

In the detection stage, for each subgraph that meets the minimum size requirement—namely, both the out-node and in-node subsets contain at least $e_\star^{(\text{out})}$ and $e_\star^{(\text{in})}$ nodes, respectively—the DOM algorithm is applied to estimate community memberships. The corresponding estimated assignment matrices are denoted by $\widehat{C}^{(i)} \in \mathbb{R}^{n \times K}$ and $\widehat{Z}^{(i)} \in \mathbb{R}^{m \times L}$, where rows corresponding to non-sampled nodes are padded with zeros. For subgraphs that do not satisfy the minimum size requirement, both $\widehat{C}^{(i)} \in \mathbb{R}^{n \times K}$ and $\widehat{Z}^{(i)} \in \mathbb{R}^{m \times L}$ are defined as zero matrices. Consequently, the estimated clustering (or affinity) matrices are given by

$$\widetilde{C}^{(i)} = \widehat{C}^{(i)} \big(\widehat{C}^{(i)}\big)^\top, \qquad \widetilde{Z}^{(i)} = \widehat{Z}^{(i)} \big(\widehat{Z}^{(i)}\big)^\top.$$

Algorithm 5 DOM-Augmented Subsampling via PACE

Input: Adjacency matrix A, number of subgraphs s, subgraph sizes $(e^{(out)}, e^{(in)})$, minimum subgraph sizes $(e_\star^{(out)}, e_\star^{(in)})$, thresholding parameter τ

Output: Final estimated community labels $\hat{\boldsymbol{c}}, \hat{\boldsymbol{z}}$

1: **Subgraph Selection:** For each $i = 1, \ldots, s$, randomly select subsets $\widetilde{\mathcal{S}}_i^{(out)} \subset \mathcal{N}$ and $\widetilde{\mathcal{S}}_i^{(in)} \subset \mathcal{M}$ with sizes $|\widetilde{\mathcal{S}}_i^{(out)}| = e^{(out)}$ and $|\widetilde{\mathcal{S}}_i^{(in)}| = e^{(in)}$.

2: **for** $i = 1$ to s **do**

3: From $\widetilde{\mathcal{S}}_i^{(out)}$ and $\widetilde{\mathcal{S}}_i^{(in)}$, remove nodes with zero out-degree and in-degree, respectively, in the submatrix $A_{(\widetilde{\mathcal{S}}_i^{(out)}, \widetilde{\mathcal{S}}_i^{(in)})}$. Denote the resulting sets as $\mathcal{S}_i^{(out)}$ and $\mathcal{S}_i^{(in)}$.

4: **Detection:** If $|\mathcal{S}_i^{(out)}| \geq e_\star^{(out)}$ and $|\mathcal{S}_i^{(in)}| \geq e_\star^{(in)}$, apply the DOM algorithm to $A_{(\mathcal{S}_i^{(out)}, \mathcal{S}_i^{(in)})}$ to obtain clustering matrices $\widetilde{C}^{(i)}$ and $\widetilde{Z}^{(i)}$ (padded with zeros for non-sampled nodes). Otherwise, set $\widetilde{C}^{(i)} \equiv 0$, $\widetilde{Z}^{(i)} \equiv 0$.

5: Define assignment indicators:

$$y_{j_1 j_2}^{(i,out)} = \mathbf{1}\{j_1, j_2 \in \mathcal{S}_i^{(out)}\}, \quad \forall j_1, j_2 \in \mathcal{N}; \quad y_{h_1 h_2}^{(i,in)} = \mathbf{1}\{h_1, h_2 \in \mathcal{S}_i^{(in)}\}, \quad \forall h_1, h_2 \in \mathcal{M}.$$

6: **end for**

7: **Patching:** Compute the total appearance counts:

$$Q_{j_1 j_2}^{(out)} = \sum_{i=1}^{s} y_{j_1 j_2}^{(i,out)}, \quad Q_{h_1 h_2}^{(in)} = \sum_{i=1}^{s} y_{h_1 h_2}^{(i,in)}.$$

Define the aggregated matrices:

$$\widetilde{C}_{\tau, j_1 j_2} = \frac{\mathbf{1}\{Q_{j_1 j_2}^{(out)} \geq \tau\} \sum_{i=1}^{s} \widetilde{C}_{j_1 j_2}^{(i)}}{Q_{j_1 j_2}^{(out)}}, \quad \widetilde{Z}_{\tau, h_1 h_2} = \frac{\mathbf{1}\{Q_{h_1 h_2}^{(in)} \geq \tau\} \sum_{i=1}^{s} \widetilde{Z}_{h_1 h_2}^{(i)}}{Q_{k_1 k_2}^{(in)}}.$$

8: Apply spectral clustering to $\widetilde{C}_\tau$ and $\widetilde{Z}_\tau$ to obtain community labels $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{z}}$ for out-nodes and in-nodes, respectively.

The patching stage aggregates clustering information across all subgraphs. For each pair of out-nodes $(j_1, j_2) \in \mathcal{N} \times \mathcal{N}$ and each pair of in-nodes $(h_1, h_2) \in \mathcal{M} \times \mathcal{M}$, the averaged clustering affinity matrices $\widetilde{C}_\tau$ and $\widetilde{Z}_\tau$, thresholded at level τ, are defined as

$$\widetilde{C}_{\tau, j_1 j_2} = \frac{\mathbf{1}\{Q_{j_1 j_2}^{(\mathrm{out})} \geq \tau\} \sum_{i=1}^{s} \widetilde{C}_{j_1 j_2}^{(i)}}{Q_{j_1 j_2}^{(\mathrm{out})}}, \qquad \widetilde{Z}_{\tau, h_1 h_2} = \frac{\mathbf{1}\{Q_{h_1 h_2}^{(\mathrm{in})} \geq \tau\} \sum_{i=1}^{s} \widetilde{Z}_{h_1 h_2}^{(i)}}{Q_{h_1 h_2}^{(\mathrm{in})}}.$$

Here, $Q_{j_1 j_2}^{(\mathrm{out})} = \sum_{i=1}^{s} y_{j_1 j_2}^{(i,\mathrm{out})}$ and $Q_{h_1 h_2}^{(\mathrm{in})} = \sum_{i=1}^{s} y_{h_1 h_2}^{(i,\mathrm{in})}$ denote the number of subgraphs in which nodes (j_1, j_2) and (h_1, h_2) co-occur, respectively, with

$$y_{j_1 j_2}^{(i,\mathrm{out})} = \mathbf{1}\{j_1, j_2 \in \mathcal{S}_i^{(\mathrm{out})}\}, \qquad y_{h_1 h_2}^{(i,\mathrm{in})} = \mathbf{1}\{h_1, h_2 \in \mathcal{S}_i^{(\mathrm{in})}\}.$$

The threshold parameter τ is used to filter out entries with low co-occurrence frequency and is set to 0.4 following Mukherjee et al. (2021). Finally, the spectral clustering method of Amini et al.

(2012) is applied to the aggregated matrices $\widetilde{C}_\tau$ and $\widetilde{Z}_\tau$ to obtain the final community assignments $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{z}}$ for the out-nodes and in-nodes, respectively. The complete procedure is summarized in Algorithm 5.

C.3 DOM-Augmented Subsampling via GALE

The integration of the Delete-One Method (DOM) within the Global Alignment of Local Estimates (GALE) framework enables efficient community detection in large-scale directed and bipartite networks. The proposed approach consists of four main stages: subgraph selection, community detection, traversal construction, and aggregation.

In the subgraph selection stage, s subgraphs are generated by independently sampling subsets of out-nodes and in-nodes from $\mathcal{N}$ and $\mathcal{M}$, of fixed sizes $e^{(\mathrm{out})}$ and $e^{(\mathrm{in})}$, respectively. Following the notation introduced in Section C.1, these node sets are denoted by $\{\mathcal{S}_i^{(\mathrm{out})}\}_{i=1}^s$ and $\{\mathcal{S}_i^{(\mathrm{in})}\}_{i=1}^s$. The DOM algorithm is then applied to each induced submatrix $A_{(\mathcal{S}_i^{(\mathrm{out})}, \mathcal{S}_i^{(\mathrm{in})})}$, producing the estimated assignment matrices $\widehat{C}^{(i)} \in \mathbb{R}^{n\times K}$ and $\widehat{Z}^{(i)} \in \mathbb{R}^{m\times L}$, where rows corresponding to non-sampled nodes are padded with zeros.

The traversal construction stage begins with the formation of two supergraphs of subgraphs, where each node corresponds to one of the s subgraphs and each edge indicates a substantial intersection between a pair of subgraphs. The adjacency structures of these supergraphs are encoded by the matrices $Q^{(\mathrm{out})}, Q^{(\mathrm{in})} \in \mathbb{R}^{s\times s}$, where an edge is established between the i_1-th and i_2-th subgraphs if the size of their intersection exceeds a predefined threshold:

$$Q_{i_1 i_2}^{(out)} = \begin{cases} 1, & \text{if } \left|\mathcal{S}_{i_1}^{(\mathrm{out})} \cap \mathcal{S}_{i_2}^{(\mathrm{out})}\right| \geq \left\lceil \frac{(e^{(\mathrm{out})})^2}{2n} \right\rceil \\ 0, & \text{otherwise.} \end{cases}, \quad Q_{i_1 i_2}^{(in)} = \begin{cases} 1, & \text{if } \left|\mathcal{S}_{i_1}^{(in)} \cap \mathcal{S}_{i_2}^{(in)}\right| \geq \left\lceil \frac{(e^{(in)})^2}{2m} \right\rceil \\ 0, & \text{otherwise.} \end{cases},$$

Given the constructed supergraphs $Q^{(\mathrm{out})}$ and $Q^{(\mathrm{in})}$, depth-first search (DFS) is applied to each to generate traversal sequences $\{x_1, x_2, \ldots, x_{J^{(\mathrm{out})}}\}$ and $\{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{J^{(\mathrm{in})}}\}$. These traversals define ordered sequences of subgraphs such that every subgraph is visited at least once. The sequence lengths $J^{(\mathrm{out})}$ and $J^{(\mathrm{in})}$ satisfy $J^{(\mathrm{out})}, J^{(\mathrm{in})} \in [s, 2s-1]$.

In the aggregation stage, the cluster assignments obtained from subgraphs along each traversal are progressively aligned using permutation matrices derived from matching overlapping nodes with those in previously visited subgraphs. The traversal is initialized with the clustering matrices of the first subgraph, $\widehat{C}^{(x_1)}$ and $\widehat{Z}^{(\tilde{x}_1)}$, which serve as the initial global estimates $\widehat{C}$ and $\widehat{Z}$, respectively. For each subsequent subgraph, the intersection with the union of previously visited subgraphs is identified, and the corresponding permutation matrix is computed using the greedy matching algorithm described in Algorithm 7.

Once aligned, the clustering assignments from each subgraph are selectively aggregated into the global estimates. This aggregation is controlled by a thresholding criterion designed to enhance reliability: for each node j, the contribution to $\widehat{C}_{jk}$ (or $\widehat{Z}_{jk}$) is included only if the node appears in a sufficient number of subgraphs. Specifically, the aggregated entries are updated as

$$\widehat{C}_{jk} \leftarrow \xi_j \cdot \frac{\sum_{\ell=1}^{i} \widehat{C}_{jk}^{(\ell)} \mathbf{1}\{j \in \mathcal{S}_{x_\ell}^{(\mathrm{out})}\}}{\sum_{\ell=1}^{i} \mathbf{1}\{j \in \mathcal{S}_{x_\ell}^{(\mathrm{out})}\}}, \qquad \widehat{Z}_{jk} \leftarrow \tilde{\xi}_j \cdot \frac{\sum_{\ell=1}^{i} \widehat{Z}_{jk}^{(\ell)} \mathbf{1}\{j \in \mathcal{S}_{\tilde{x}_\ell}^{(\mathrm{in})}\}}{\sum_{\ell=1}^{i} \mathbf{1}\{j \in \mathcal{S}_{\tilde{x}_\ell}^{(\mathrm{in})}\}},$$

Algorithm 6 DOM-Augmented Subsampling via GALE

Input: Adjacency matrix A, number of subgraphs s, subgraph sizes $(e^{(out)}, e^{(in)})$, alignment thresholds $\{\tau_i\}_{i=1}^s$

Output: Final estimated community labels $\hat{\boldsymbol{c}}$, $\hat{\boldsymbol{z}}$

1: **Subgraph Selection:** For each $i = 1, \ldots, s$, randomly select subsets $\mathcal{S}_i^{(out)} \subset \mathcal{N}$ and $\mathcal{S}_i^{(in)} \subset \mathcal{M}$ with sizes $|\mathcal{S}_i^{(out)}| = e^{(\mathrm{out})}$ and $|\mathcal{S}_i^{(in)}| = e^{(in)}$.

2: **for** $i = 1$ to s **do**

3: **Detection:** Apply the DOM algorithm to sub-matrix $A_{(\mathcal{S}_i^{(out)}, \mathcal{S}_i^{(in)})}$ to obtain the estimated assignment matrices $\widehat{C}^{(i)}$ and $\widehat{Z}^{(i)}$ (padded with zeros for non-sampled nodes).

4: **end for**

5: **Traversal Construction:** For both out-node and in-node subsets, construct traversals $\{\mathcal{S}_{x_1}^{(out)}, \ldots, \mathcal{S}_{x_{J^{(out)}}}^{(out)}\}$ and $\{\mathcal{S}_{\tilde{x}_1}^{(in)}, \ldots, \mathcal{S}_{\tilde{x}_{J^{(in)}}}^{(in)}\}$ over out-node subsets $\{\mathcal{S}_i^{(out)}\}_{i=1}^s$ and in-node subsets $\{\mathcal{S}_i^{(in)}\}_{i=1}^s$ by performing depth-first search (DFS) on the supergraph of subgraphs, where their adjacency structure is denoted as $Q^{(\mathrm{out})}, Q^{(\mathrm{in})} \in \mathbb{R}^{s \times s}$ with

$$Q_{i_1 i_2}^{(out)} = \begin{cases} 1, & \text{if } \left|\mathcal{S}_{i_1}^{(\mathrm{out})} \cap \mathcal{S}_{i_2}^{(\mathrm{out})}\right| \geq \left\lceil \frac{(e^{(\mathrm{out})})^2}{2n} \right\rceil \\ 0, & \text{otherwise.} \end{cases}, \quad Q_{i_1 i_2}^{(in)} = \begin{cases} 1, & \text{if } \left|\mathcal{S}_{i_1}^{(in)} \cap \mathcal{S}_{i_2}^{(in)}\right| \geq \left\lceil \frac{(e^{(in)})^2}{2m} \right\rceil \\ 0, & \text{otherwise.} \end{cases},$$

6: **Aggregation:** Set global estimate $\widehat{C} \leftarrow \widehat{C}^{(x_1)}$ and $\widehat{Z} \leftarrow \widehat{Z}^{(x_1)}$.

7: **for** $i = 2$ to $J^{(out)}$ (and $J^{(in)}$) **do**

8: **if** $\mathcal{S}_{x_i}^{(out)}$ (and $\mathcal{S}_{\tilde{x}_i}^{(in)}$) has not been visited **then**

9: (a) Define the overlap $\mathcal{S}^{(out)} = \mathcal{S}_{x_i}^{(out)} \cap \left(\cup_{\ell=1}^{i-1} \mathcal{S}_{x_\ell}^{(out)}\right)$ and $\mathcal{S}^{(in)} = \mathcal{S}_{\tilde{x}_i}^{(in)} \cap \left(\cup_{\ell=1}^{i-1} \mathcal{S}_{\tilde{x}_\ell}^{(in)}\right)$.

10: (b) Compute permutation matrices $\widehat{\Pi}_i^{(out)} = \mathcal{D}(\widehat{C}_{\mathcal{S}^{(out)}}^{(x_i)}, \widehat{C}_{\mathcal{S}^{(out)}})$ (and $\widehat{\Pi}_i^{(in)} = \mathcal{D}(\widehat{Z}_{\mathcal{S}^{(in)}}^{(\tilde{x}_i)}, \widehat{Z}_{\mathcal{S}^{(in)}})$) to align current assignments with previously visited nodes.

11: (c) Update $\widehat{C}^{(x_i)} \leftarrow \widehat{C}^{(x_i)} \widehat{\Pi}_i^{(out)}$ (and $\widehat{Z}^{(\tilde{x}_i)} \leftarrow \widehat{Z}^{(\tilde{x}_i)} \widehat{\Pi}_i^{(in)}$).

12: (d) Update $\xi_j \leftarrow \mathbf{1}\left\{\sum_{\ell \in \{1,\ldots,i\}} \mathbf{1}\{j \in \mathcal{S}_{x_\ell}^{(out)}\} > \tau_i\right\}$ for each out-node j (and update $\tilde{\xi}_j \leftarrow \mathbf{1}\left\{\sum_{\ell \in \{1,\ldots,i\}} \mathbf{1}\{j \in \mathcal{S}_{\tilde{x}_\ell}^{(in)}\} > \tau_i\right\}$ for each in-node j).

13: (e) Update

$$\widehat{C}_{jk} \leftarrow \xi_j \frac{\sum_{\ell \in \{1,\ldots,i\}} \widehat{C}_{jk}^{(\ell)} \mathbf{1}\{j \in \mathcal{S}_{x_\ell}^{(out)}\}}{\sum_{\ell \in \{1,\ldots,i\}} \mathbf{1}\{j \in \mathcal{S}_{x_\ell}^{(out)}\}} \left(\text{and} \widehat{Z}_{jk} \leftarrow \tilde{\xi}_j \frac{\sum_{\ell \in \{1,\ldots,i\}} \widehat{Z}_{jk}^{(\ell)} \mathbf{1}\{j \in \mathcal{S}_{\tilde{x}_\ell}^{(in)}\}}{\sum_{\ell \in \{1,\ldots,i\}} \mathbf{1}\{j \in \mathcal{S}_{\tilde{x}_\ell}^{(in)}\}}\right)$$

14: (f) Mark $\mathcal{S}_{x_i}^{(out)}$ (and $\mathcal{S}_{\tilde{x}_i}^{(in)}$) as visited.

15: **end if**

16: **end for**

17: Set $\hat{\boldsymbol{c}}_i = \arg\max_k \widehat{C}_{ik}$, $\hat{\boldsymbol{z}}_j = \arg\max_\ell \widehat{Z}_{j\ell}$ for $i \in [n]$ and $j \in [m]$.

Algorithm 7 Greedy Match Algorithm

Input: Two clustering matrices $O_1, O_2 \in \{0,1\}^{|S|\times K}$ with same node set S.
Output: Permutation matrix $\Pi \in \{0,1\}^{K\times K}$

Compute confusion matrix $M = O_1^\top O_2$
Initialize permutation matrix $\Pi \leftarrow \mathbf{0}_{K\times K}$
while there are rows/columns in M with nonnegative entries **do**
    (a) Find (i,j) such that $M_{ij} = \|M\|_\infty$ (break ties arbitrarily)
    (b) Set $\Pi_{ij} = 1$
    (c) Set $M_{i\cdot} \equiv -1$ and $M_{\cdot j} \equiv -1$.
end while
return Π

where $\xi_j = \mathbf{1}\Big\{\sum_{\ell=1}^{i} \mathbf{1}\{j \in \mathcal{S}_{x_\ell}^{(out)}\} > \tau_i\Big\}$ for each out-node j and $\tilde{\xi}_j = \mathbf{1}\{\sum_{\ell=1}^{i} \mathbf{1}\{j \in \mathcal{S}_{\tilde{x}_\ell}^{(in)}\} > \tau_i\}$ for each in-node j. Finally, following Section 3.3, the global community labels $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{z}}$ are obtained by assigning each node to the cluster corresponding to the largest entry in its respective row of $\widehat{C}$ and $\widehat{Z}$. The overall procedure is summarized in Algorithm 6.

Appendix D. Supplementary Simulation Results

D.1 Additional Numerical Results on Community Detection Accuracy

This section presents supplementary numerical results on community detection accuracy, evaluated using the following metrics:

- **Clustering Error (clust.err):** Let $\hat{\psi}$ denote the estimated out- or in-community assignment function, and let ψ^* denote the corresponding true community assignment function. The clustering error, denoted by clust.err$(\hat{\psi}, \psi^*)$, measures the proportion that $\hat{\psi}$ and ψ^* assign nodes i and j differently. It is defined as

$$\text{clust.err}(\hat{\psi}, \psi^*) \;=\; \frac{2}{n(n-1)} \sum_{i<j} I\Big(I\big(\hat{\psi}(i) = \hat{\psi}(j)\big) + I\big(\psi^*(i) = \psi^*(j)\big) = 1\Big).$$

- **Normalized Mutual Information (NMI):** NMI is a standard clustering accuracy measure ranging from 0 (no agreement) to 1 (perfect agreement), with larger values indicating better clustering alignment; see Lancichinetti et al. (2009); Zhou and Amini (2020).

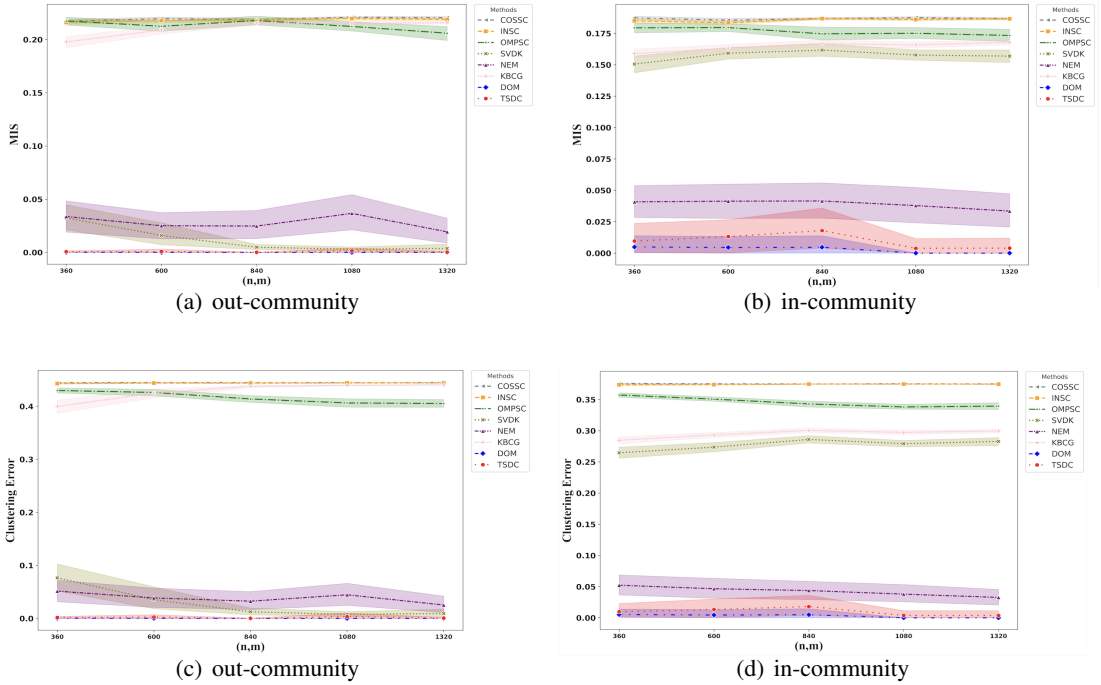
- **Proportion of Misclustered Nodes (MIS):** To quantify the proportion of misclustered nodes, we compare the true clustering matrix Z_* with any estimated clustering matrix Z. This proportion is defined as

$$\text{Err}(Z, Z_*) \;=\; \frac{1}{2n} \min_{P_K \in \mathcal{P}_K} \|ZP_K - Z_*\|_1 \;=\; \frac{1}{2n} \min_{P_K \in \mathcal{P}_K} \|ZP_K - Z_*\|_F^2,$$

  where $\mathcal{P}_K$ denotes the set of permutation matrices $P_K : \{1, 2, \ldots, K\} \to \{1, 2, \ldots, K\}$.

Figures A.1, A.2, and A.5 show community detection accuracy under the Normal, Normal-Bernoulli Mixture, and Sparsity settings, respectively, evaluated by three metrics. These results are consistent with those in Figures 2-4 of the main text.

For data generated from Bernoulli and Poisson distributions, Figures A.3 and A.4 report simulation results with node counts ranging from $n = m = 500$ to $n = m = 900$ in increments of 100. In these settings, the adjacency matrix A follows a Bernoulli distribution in the former case and a Poisson distribution in the latter. Consistent with the Normal distribution results, both DOM and TSDC outperform the competing methods in accuracy.



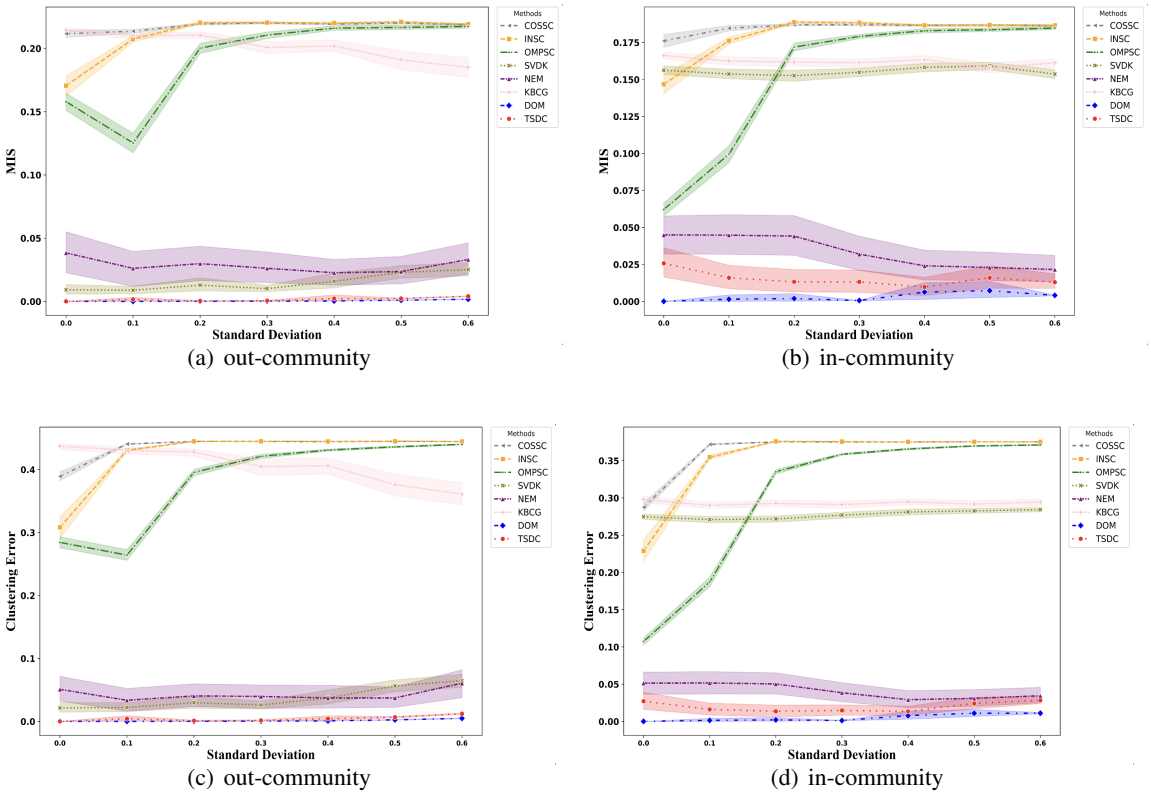
(a) out-community (b) in-community

(c) out-community (d) in-community

Figure A.1: Proportion of misclustered nodes (top) and clustering error (bottom) for the Normal-Bernoulli mixture case. The left panel reports out-community clustering, and the right panel reports in-community clustering.

D.2 Additional Numerical Results on Computational Efficiency

The running times for the Bernoulli and Poisson cases are reported in Tables 1 and 2.



(a) out-community (b) in-community

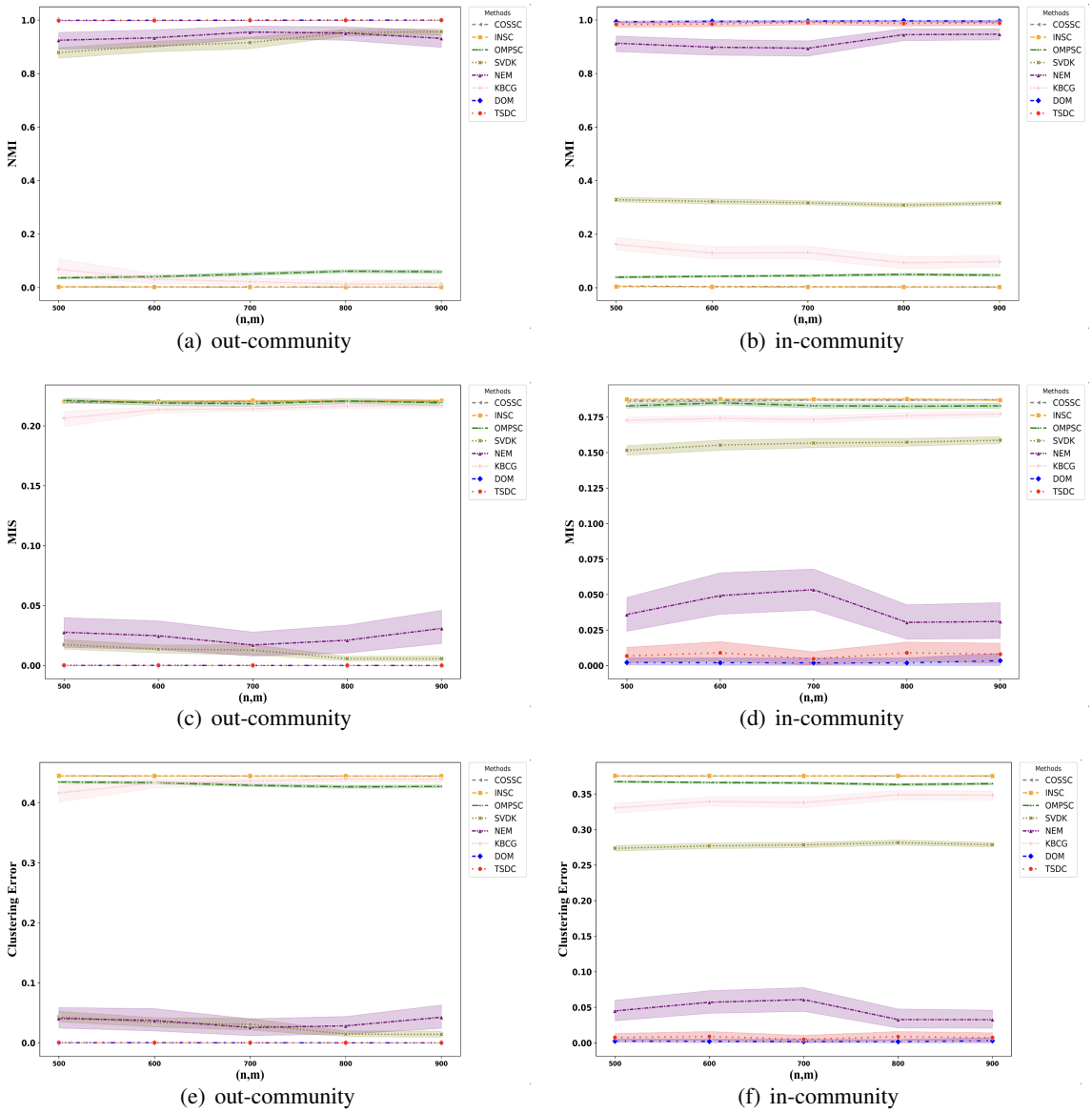
(c) out-community (d) in-community

Figure A.2: Proportion of misclustered nodes (top) and clustering error (bottom) across different standard deviations in the Normal case with $n = m = 600$. The left panels report out-community clustering, and the right panels report in-community clustering.

(n, m)	DOM	TSDC	OMPSC	COSSC	INSC	SVDK	NEM	KBCG
(500,500)	272.58	0.29	5.26	0.35	0.79	0.13	34.53	0.29
(600,600)	285.03	0.23	4.57	0.34	0.85	0.10	42.88	0.32
(700,700)	617.35	0.41	9.48	0.60	1.69	0.19	58.71	0.37
(800,800)	683.60	0.38	9.01	0.60	2.52	0.16	70.53	0.41
(900,900)	1526.51	0.70	19.81	1.10	3.85	0.35	86.04	0.43

Table 1: Running time (seconds) for the Bernoulli case.

D.3 Additional Numerical Results on Subsampling-Based Community Detection

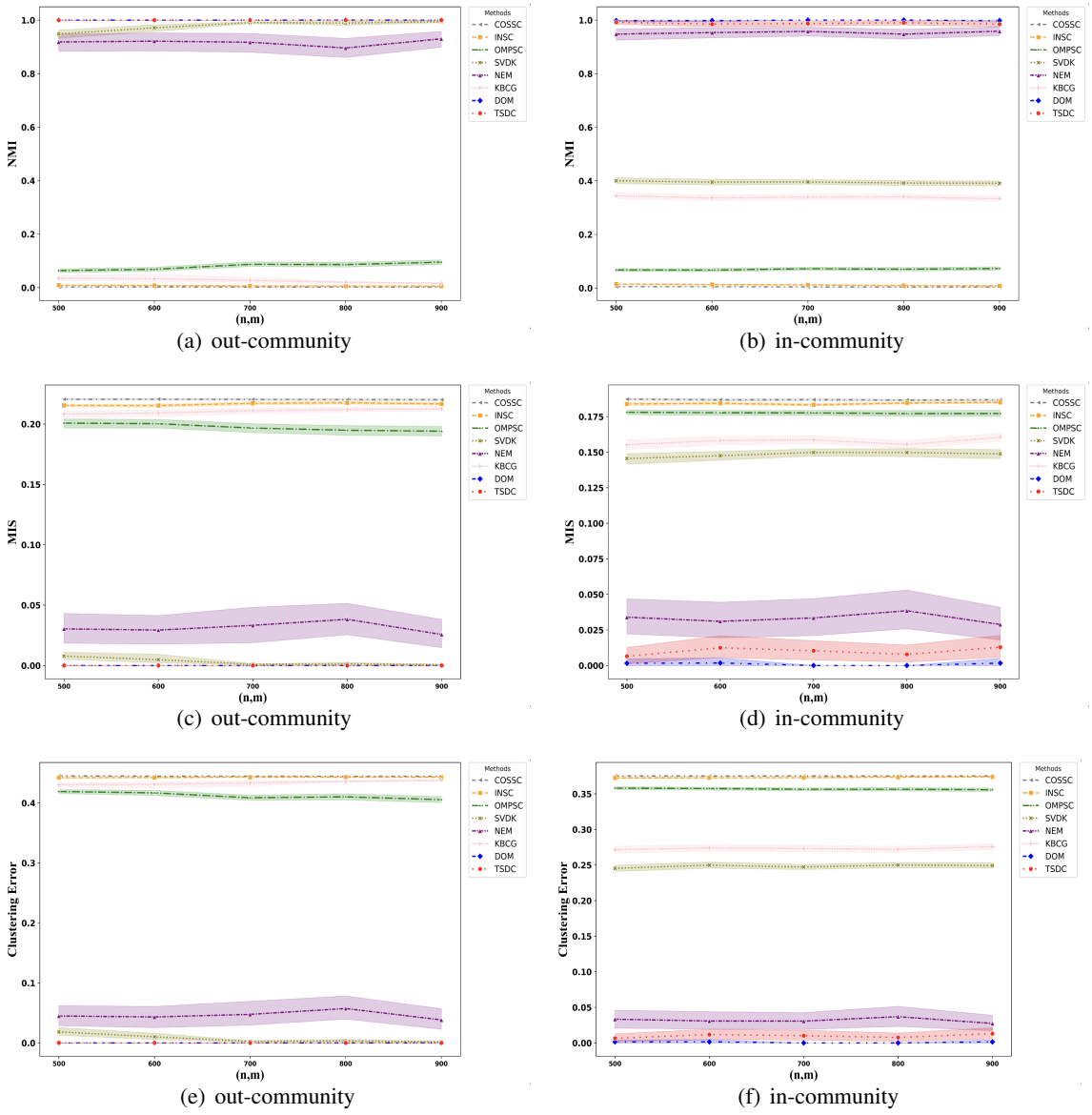


(a) out-community (b) in-community

(c) out-community (d) in-community

(e) out-community (f) in-community

Figure A.3: NMI (top), Proportion of Misclustered Nodes (middle), and Clustering Error (bottom) for adjacency matrices A derived from Bernoulli random variables. The number of nodes ranges from $n = m = 500$ to $n = m = 900$ with the increment of 100. Left panels display out-community clustering outcomes, while right panels reveal in-community results.



(a) out-community (b) in-community

(c) out-community (d) in-community

(e) out-community (f) in-community

Figure A.4: NMI (top), proportion of misclustered nodes (middle), and clustering error (bottom) for adjacency matrices A generated from Poisson random variables. The number of nodes ranges from $n = m = 500$ to $n = m = 900$ in increments of 100. The left panels report out-community clustering, and the right panels report in-community clustering.

Appendix E. Additional Results on Real Data Examples

E.1 Worldwide Food Trading Networks

Figures A.7 and A.8 show heatmaps of the original networks and the rearranged block cosine similarity matrices obtained via the TSDC algorithm for cereal and cigarette trades. The results are

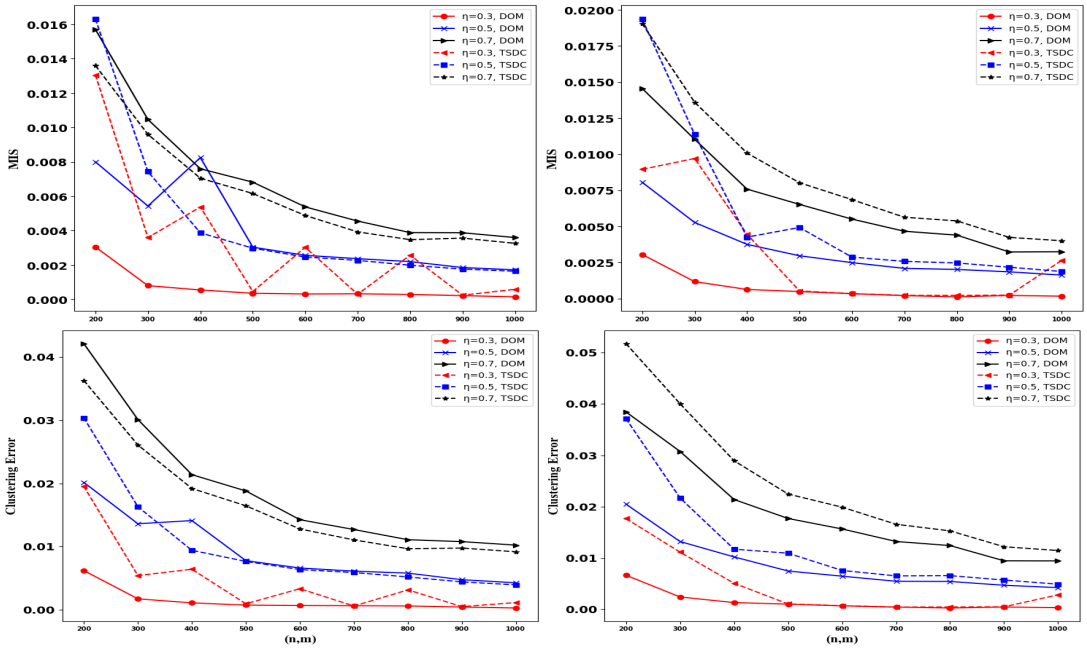


Figure A.5: Proportion of misclustered nodes (top) and clustering error (bottom) for the sparsity case. The left panel reports out-community clustering, and the right panel reports in-community clustering.

(n, m)	DOM	TSDC	OMPSC	COSSC	INSC	SVDK	NEM	KBCG
(500,500)	96.67	0.09	1.74	0.18	0.40	0.04	35.31	0.25
(600,600)	174.17	0.12	2.58	0.26	0.63	0.05	97.54	0.31
(700,700)	244.55	0.15	3.65	0.34	0.91	0.07	84.72	0.33
(800,800)	375.51	0.20	5.21	0.44	2.21	0.09	79.02	0.38
(900,900)	534.46	0.22	7.92	0.56	1.73	0.10	90.69	0.36

Table 2: Running time (seconds) for the Poisson case.

consistent with those from the DOM algorithm. Figure A.9 illustrates global trading patterns, highlighting geographic clustering with Europe as the primary tobacco importer and Asia–Oceania as the main tobacco market and production region. Panel (c) of Figure A.9, together with Figure A.10, links cereal export clusters to production levels, identifying major exporters with high yields and revealing parallels between cereal and cigarette clustering patterns driven by geographic proximity.

(n, m)	Method	Parameter Selection			Time (sec)	NMI
		$(e^{(out)}, e^{(in)})$	s			
$(1000, 1000)$	DOM	-	-		2483.5	(**1.00**,0.99)
	DOM+PACE	200	50		201.7	(0.98,0.96)
	DOM+GALE	200	100		342.5	(0.95,0.94)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(60,60)	20	1	105.89	(0.94,0.92)
		(60,60)	20	2	119.99	(0.97,0.94)
		(60,60)	20	5	135.00	(0.97,0.97)
$(2000, 2000)$	DOM	-	-		8357.24	(**1.00**,**1.00**)
	DOM+PACE	400	50		440.04	(**1.00**,**1.00**)
	DOM+GALE	400	100		782.86	(0.97,0.97)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(100,100)	20	1	302.78	(0.97,0.93)
		(100,100)	20	2	316.70	(0.98,0.98)
		(100,100)	20	5	340.72	(**1.00**,0.99)
$(5000, 5000)$	DOM	-	-		159655.26	(**1.00**,**1.00**)
	DOM+PACE	500	50		802.04	(0.97,0.98)
	DOM+GALE	500	100		1472.65	(0.94,0.96)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(300,300)	20	1	3030.53	(**1.00**,0.96)
		(300,300)	20	2	3357.83	(**1.00**,**1.00**)
		(300,300)	20	5	3779.92	(**1.00**,**1.00**)
$(8000, 8000)$	DOM	-	-		260125.47	(**1.00**,**1.00**)
	DOM+PACE	1000	50		7558.03	(0.99,0.99)
	DOM+GALE	1000	100		14767.44	(0.97,0.96)
	DOM+SONNET	$(o^{(out)}, o^{(in)})$	s	R	-	-
		(400,400)	20	1	11077.94	(**1.00**,0.96)
		(400,400)	20	2	11775.09	(**1.00**,**1.00**)
		(400,400)	20	5	13391.74	(**1.00**,**1.00**)

Table 3: Performance of DOM and its subsampling variants across network sizes with noise level $\sigma = 0.3$.

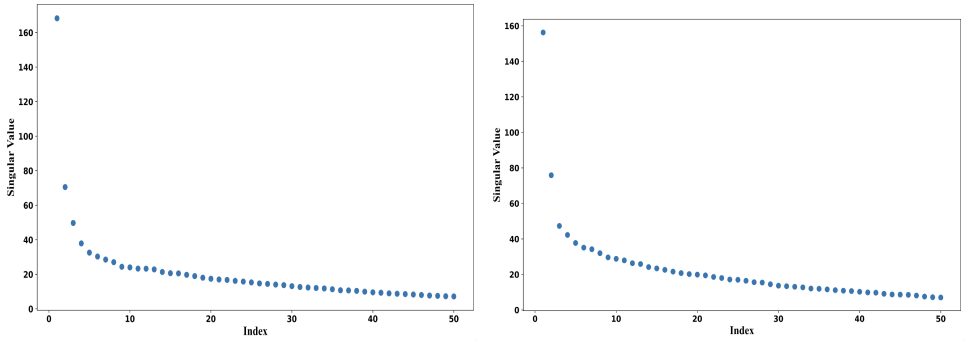


Figure A.6: Singular values of adjacency matrices: the left panel shows the first 50 for the Cereal network, and the right panel shows the first 50 for the Cigarette network.

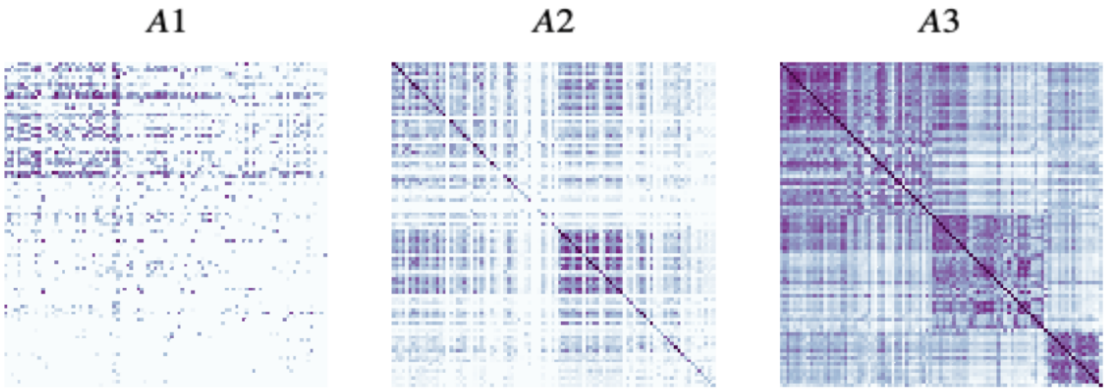


Figure A.7: Heatmap of the original Cereal network ($A1$) and the rearranged block cosine similarity matrix for rows ($A2$) and columns ($A3$).

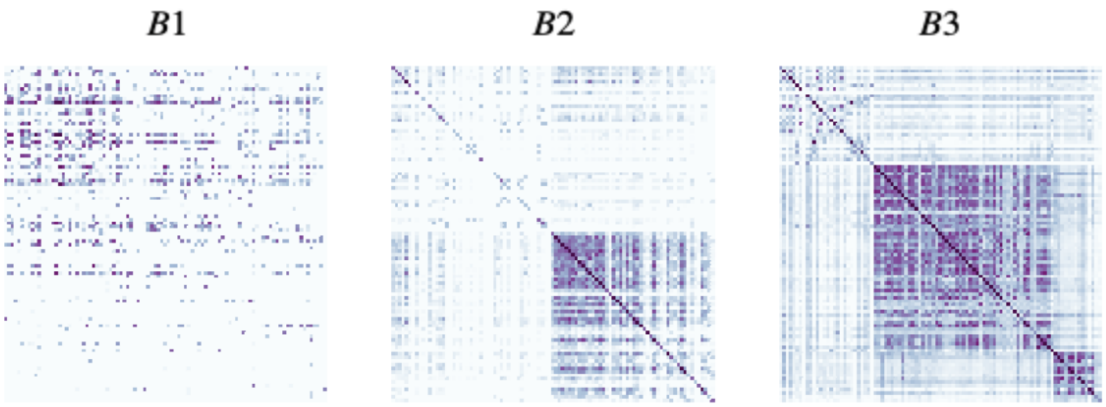
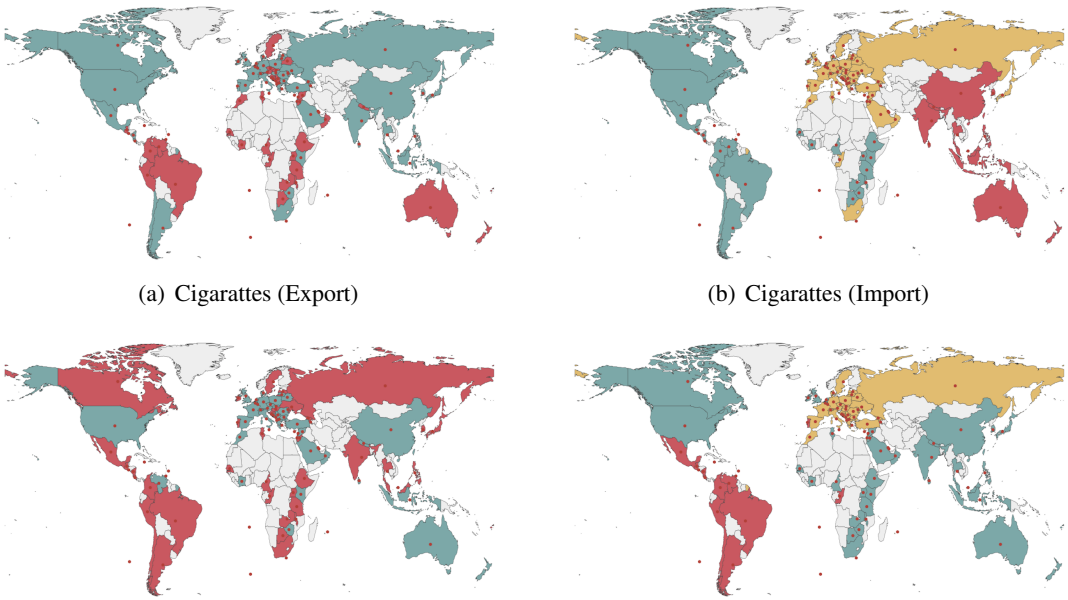


Figure A.8: Heatmap of the original Cigarette network ($B1$) and the rearranged block cosine similarity matrix for rows ($B2$) and columns ($B3$).



(a) Cigarattes (Export)

(b) Cigarattes (Import)

(c) Cereals (Export)

(d) Cereals (Import)

Figure A.9: World maps of (a) cigarette export clustering, (b) cigarette import clustering, (c) cereal export clustering, and (d) cereal import clustering.

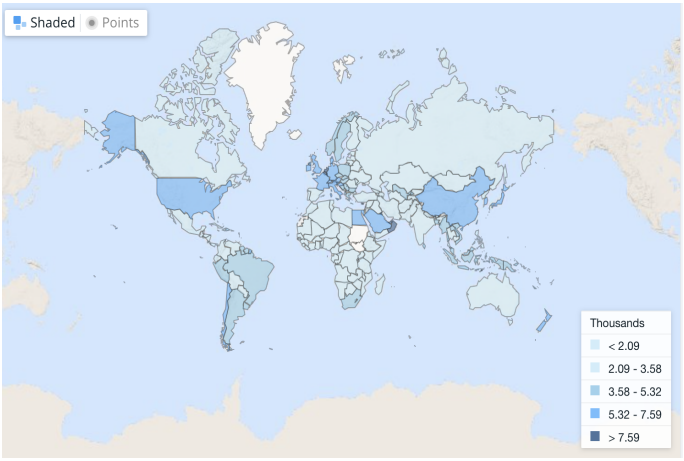
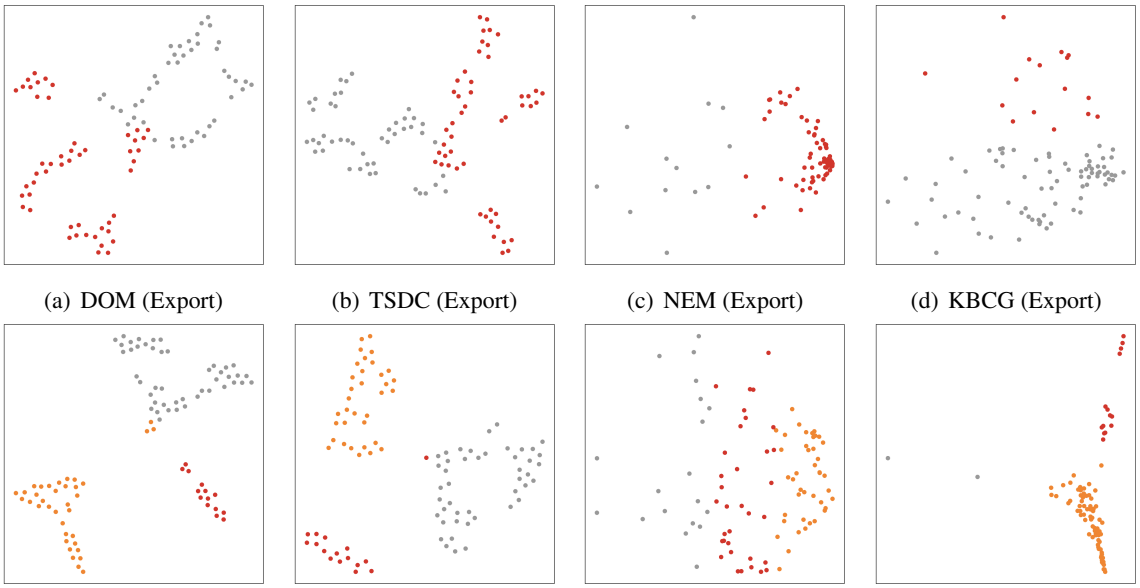


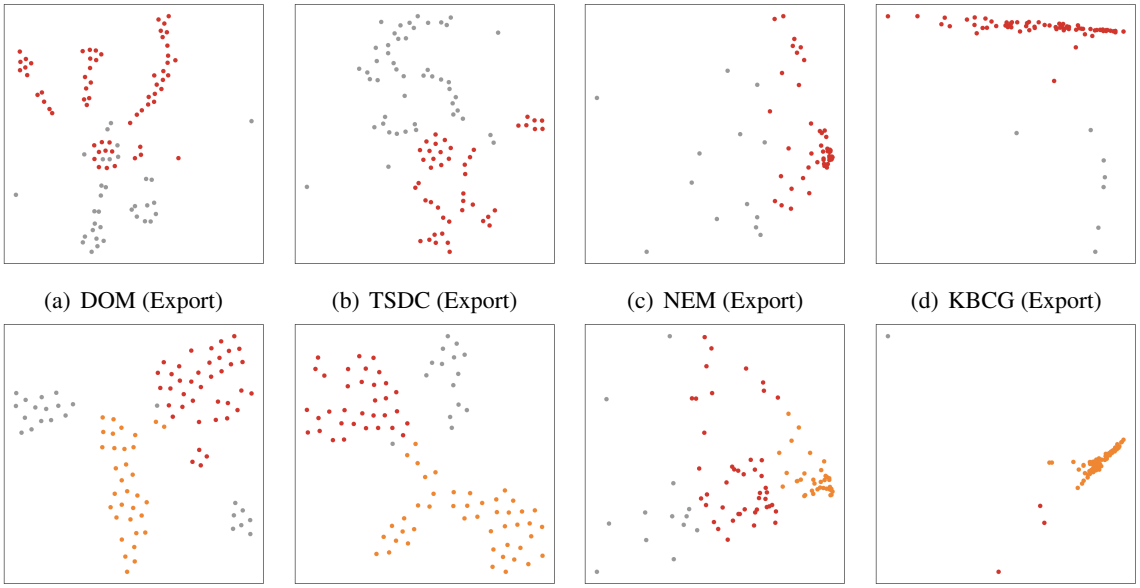
Figure A.10: Cereal yield per hectare by country in 2010.



(a) DOM (Export) (b) TSDC (Export) (c) NEM (Export) (d) KBCG (Export)

(e) DOM (Import) (f) TSDC (Import) (g) NEM (Import) (h) KBCG (Import)

Figure A.11: Latent embeddings of the cereal network under four methods-DOM, TSDC, NEM, and KBCG. For DOM and TSDC, embeddings are obtained by applying singular value decomposition (SVD) to the estimated probability matrix, extracting the left and right singular vectors of dimension $K \times L$, and then applying t-SNE (van der Maaten and Hinton, 2008) for two-dimensional visualization. For NEM, we set the latent embedding dimension to $K \times L$ and likewise apply t-SNE. For KBCG, the row and column embedding dimensions are set to 2 and 3, respectively. Panels (a)-(d) display out-node embeddings, and panels (e)-(h) display in-node embeddings; node colors indicate community memberships.



(a) DOM (Export) (b) TSDC (Export) (c) NEM (Export) (d) KBCG (Export)

(e) DOM (Import) (f) TSDC (Import) (g) NEM (Import) (h) KBCG (Import)

Figure A.12: Latent embeddings of the cigarette network under four methods-DOM, TSDC, NEM, and KBCG. The embedding construction follows the same procedure as described for the cereal network. Panels (a)-(d) display out-node embeddings, and panels (e)-(h) display in-node embeddings; node colors indicate community memberships.

E.2 MovieLens 100K Dataset

The results of applying the DOM algorithm to the MovieLens 100K dataset are shown in Figure A.13. Although the block structures in the block cosine similarity matrix are less distinct than those in Figure 8 of the main paper, they remain identifiable, indicating that the TNPM is reasonably compatible with this dataset.

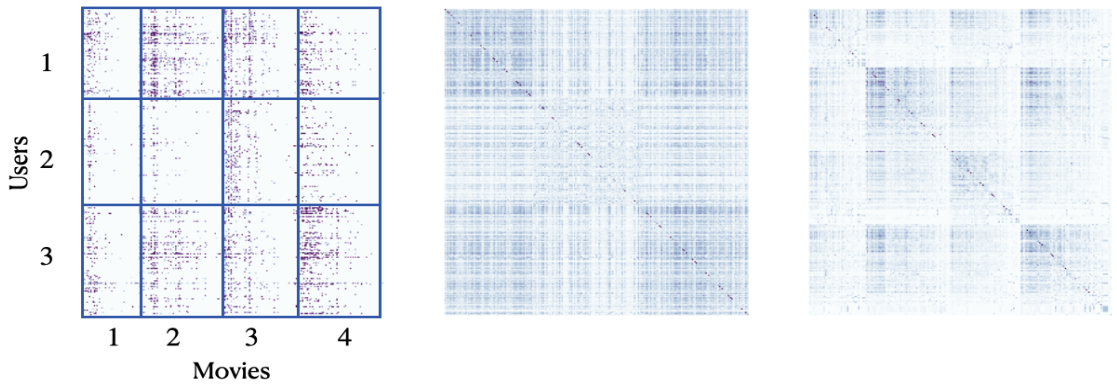


Figure A.13: Heatmap of the MovieLens matrix A (left) and the rearranged block cosine similarity matrices for rows (middle) and columns (right).

E.3 List of Countries

Figure A.14 shows the 95 selected countries.

Country List

Canada, China,_Hong_Kong_SAR, China,_Taiwan_Province_of, India, Ireland, Japan, Malaysia, Philippines, Korea, Thailand, Cameroon, Malawi, Tunisia, Uganda, Fiji, South Africa, Tanzania, Zambia, Bahrain, Botswana, Jordan, Senegal, Ethiopia. Russia, Sweden, Ukraine, Croatia, Hungary, Israel, Luxembourg, Georgia, Armenia, Latvia, Cabo_Verde, Brazil, Guatemala, Argentina, Colombia, Costa Rica, Mexico, Nicaragua, Portugal, Chile, El Salvador, Jamaica, Peru, Trinidad and Tobago, Barbados, Congo, Australia, Belgium, China, Indonesia, Lebanon, Netherlands, Saudi Arabia, Singapore Rep., United Kingdom, United States, Côte d'Ivoire, Cyprus, Kenya, Syria, New Zealand, Sri Lanka, Oman, Malta, Mauritius, Nepal, Qatar, Zimbabwe, Austria, Denmark, France, Germany, Italy, Poland, Spain, Switzerland, Turkey, Albania, Bosnia and Herz., Bulgaria, Czech Rep., Greece, Lithuania, Montenegro, Morocco, Moldova, Romania, Serbia, Slovakia, Macedonia, Belarus, Venezuela.

Figure A.14: Selected 93 countries

References

Arash A. Amini, Aiyou Chen, Peter J. Bickel, and Elizaveta Levina. Pseudo-likelihood methods for community detection in large sparse networks. *Annals of Statistics*, 41:2097–2122, 2012. URL https://api.semanticscholar.org/CorpusID:14510650.

Arash A Amini, Aiyou Chen, Peter J Bickel, and Elizaveta Levina. Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics*, 41(4):2097–2122, 2013.

Peter J Bickel and Aiyou Chen. A nonparametric view of network models and newman–girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50):21068–21073, 2009.

Genís Calderer and Marieke L Kuijjer. Community detection in large-scale bipartite biological networks. *Frontiers in Genetics*, page 520, 2021.

Sayan Chakrabarty, Srijan Sengupta, and Yuguo Chen. Subsampling-based community detection for large networks. *Statistica Sinica*, 35(3):1–42, 2025.

Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora. Structural reducibility of multilayer networks. *Nature communications*, 6(1):1–9, 2015.

Gerard Debreu. A social equilibrium existence theorem. *Proceedings of the National Academy of Sciences*, 38(10):886–893, 1952.

Ky Fan. Fixed-point and minimax theorems in locally convex topological linear spaces. *Proceedings of the National Academy of Sciences*, 38(2):121–126, 1952.

Cheryl Flynn and Patrick Perry. Profile likelihood biclustering. *Electronic Journal of Statistics*, 14 (1):731–768, 2020.

Drew Fudenberg and Jean Tirole. *Game theory*. MIT press, 1991.

Christophe Giraud. *Introduction to high-dimensional statistics*. CRC Press, 2015.

Irving L Glicksberg. A further generalization of the kakutani fixed point theorem, with application to nash equilibrium points. *Proceedings of the American Mathematical Society*, 3(1):170–174, 1952.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

Pengsheng Ji and Jiashun Jin. Coauthorship and citation networks for statisticians. *The Annals of Applied Statistics*, 10(4):1779–1812, 2016.

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.

Bingyi Jing, Ting Li, Ningchen Ying, and Xianshi Yu. Community detection in sparse networks using the symmetrized laplacian inverse matrix (slim). *Statistica Sinica*, 32(1):1, 2022.

Brian Karrer and Mark EJ Newman. Stochastic blockmodels and community structure in networks. *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, 83(1):016107, 2011.

Yuval Kluger, Ronen Basri, Joseph T Chang, and Mark Gerstein. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome research*, 13(4):703–716, 2003.

John Koo, Minh Tang, and Michael W Trosset. Popularity adjusted block models are generalized random dot product graphs. *Journal of Computational and Graphical Statistics*, 32(1):131–144, 2023.

Andrea Lancichinetti, Santo Fortunato, and János Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New journal of physics*, 11(3):033015, 2009.

Ting Li, Jianchang Hu, Shiying Wang, and Heping Zhang. Super-variants identification for brain connectivity. *Human brain mapping*, 42(5):1304–1312, 2021.

Soumendu Sundar Mukherjee, Purnamrita Sarkar, and Peter J Bickel. Two provably consistent divide-and-conquer clustering algorithms for large networks. *Proceedings of the National Academy of Sciences*, 118(44):e2100482118, 2021.

Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14, 2001.

Majid Noroozi, Marianna Pensky, and Ramchandra Rimal. Sparse popularity adjusted stochastic block model. *The Journal of Machine Learning Research*, 22(1):8671–8706, 2021a.

Majid Noroozi, Ramchandra Rimal, and Marianna Pensky. Estimation and clustering in popularity adjusted block model. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83(2):293–317, 2021b.

Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

Omiros Papaspiliopoulos. High-dimensional probability: An introduction with applications in data science, 2020.

Haobo Qi, Xuening Zhu, and Hansheng Wang. A random projection method for large-scale community detection. *arXiv preprint*, 2022.

Karl Rohe, Tai Qin, and Bin Yu. Co-clustering directed graphs to discover asymmetries and directional communities. *Proceedings of the National Academy of Sciences*, 113(45):12679–12684, 2016.

Patrick Rubin-Delanchy, Joshua Cape, Minh Tang, and Carey E. Priebe. A statistical interpretation of spectral embedding: The generalised random dot product graph. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(4):1446–1473, 2022. doi: 10.1111/rssb.12509. URL `https://doi.org/10.1111/rssb.12509`.

Venu Satuluri and Srinivasan Parthasarathy. Symmetrizations for clustering directed graphs. In *Proceedings of the 14th International Conference on Extending Database Technology*, pages 343–354, 2011.

Srijan Sengupta and Yuguo Chen. A block model for node popularity in networks with community structure. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(2): 365–386, 2018.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11):2579–2605, 2008.

Jiangzhou Wang, Jingfei Zhang, Binghui Liu, Ji Zhu, and Jianhua Guo. Fast network community detection with profile-pseudo likelihood methods. *Journal of the American Statistical Association*, pages 1–14, 2021.

Junhui Wang. Consistent selection of the number of clusters via crossvalidation. *Biometrika*, 97(4): 893–904, 2010.

Zhe Wang, Yingbin Liang, and Pengsheng Ji. Spectral algorithms for community detection in directed networks. *The Journal of Machine Learning Research*, 21(1):6101–6145, 2020.

Ling Wu, Qishan Zhang, Chi-Hua Chen, Kun Guo, and Deqin Wang. Deep learning techniques for community detection in social networks. *IEEE Access*, 8:96016–96026, 2020.

Stephen J. Young and Edward R. Scheinerman. Random dot product graph models for social networks. *Proceedings of the 5th International Conference on Algorithms and Models for the Web-Graph (WAW 2007)*, 4863:138–149, 2007. doi: 10.1007/978-3-540-77004-6_11.

Jingnan Zhang, Xin He, and Junhui Wang. Directed community detection with network embedding. *Journal of the American Statistical Association*, pages 1–11, 2021.

Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, 40(4):2266–2292, 2012.

Yunpeng Zhao, Ning Hao, and Ji Zhu. Variational estimators of the degree-corrected latent block model for bipartite networks. *arXiv preprint arXiv:2206.08465*, 2022.

Yunpeng Zhao, Ning Hao, and Ji Zhu. Variational estimators of the degree-corrected latent block model for bipartite networks. *Journal of Machine Learning Research*, 25:1–42, 2024.

Zhixin Zhou and Arash A Amini. Analysis of spectral clustering algorithms for community detection: the general bipartite setting. *The Journal of Machine Learning Research*, 20(1):1774–1820, 2019.

Zhixin Zhou and Arash A Amini. Optimal bipartite network clustering. *The Journal of Machine Learning Research*, 21(1):1460–1527, 2020.