Mixed Membership Graph Clustering via Systematic Edge Query

Shahana Ibrahim and Xiao Fu

School of Electrical Engineering and Computer Science
Oregon State University
Corvallis, OR 97331, USA

July 13, 2021

Abstract

This work considers clustering nodes of a largely incomplete graph. Under the problem setting, only a small amount of queries about the edges can be made, but the entire graph is not observable. This problem finds applications in large-scale data clustering using limited annotations, community detection under restricted survey resources, and graph topology inference under hidden/removed node interactions. Prior works tackled this problem from various perspectives, e.g., convex programming-based low-rank matrix completion and active query-based clique finding. Nonetheless, many existing methods are designed for estimating the single-cluster membership of the nodes, but nodes may often have mixed (i.e., multi-cluster) membership in practice. Some query and computational paradigms, e.g., the random query patterns and nuclear norm-based optimization advocated in the convex approaches, may give rise to scalability and implementation challenges. This work aims at learning mixed membership of nodes using queried edges. The proposed method is developed together with a systematic query principle that can be controlled and adjusted by the system designers to accommodate implementation challenges—e.g., to avoid querying edges that are physically hard to acquire. Our framework also features a lightweight and scalable algorithm with membership learning guarantees. Real-data experiments on crowdclustering and community detection are used to showcase the effectiveness of our method.

1 Introduction

Graph clustering (GC) is of broad interest in data science, which aims at associating the nodes of a graph with different clusters in an unsupervised manner [1]. The GC technique is well-motivated, since network data frequently arise in various applications (e.g., in social network analysis, brain signal processing, and biological/ecological data mining).

In the past two decades, much effort was invested onto dealing with the GC problem; see, e.g., [2–6]. Nonetheless, some major challenges arise in the era of big data. Notably, many graph data are highly incomplete for various reasons. For example, a large social network follower-followee data could contain billions of nodes, which translates to $\approx 10^{18}$ edges. Data acquisition at such a scale is a highly nontrivial task. In many cases, instead of collecting edge information of the entire network, data analysts have to *sample* edges of interest, and use the sampled network to perform GC [7,8]. Graph sampling (or, edge query)-based network analysis is also well-motivated for other reasons—e.g., in community detection of networks where edges are intentionally removed or hidden (e.g., terrorist networks or radical group networks) [9] and in biological/ecological networks where acquiring the complete edge information is too resource-consuming [10,11].

A number of works considered GC under incomplete edge observation. For example, the works in [12–14] used uniformly random edge queries and proposed convex low-rank matrix completion-based algorithms for GC. Another line of work [15–17] proposed active edge query-based methods in which the learning algorithms query the edges on-the-fly, in order to reduce the number of queries needed.

The approaches in [12–17] are insightful, and showed that edge query-based GC is a largely viable task, in both theory and practice. Nonetheless, a number of challenges remain. First, the methods in [12–17] were not designed to estimate multiple membership of the nodes. However, each node is often associated with different clusters in real-world networks (e.g., a person in a co-author network could belong to the machine learning and statistics communities simultaneously). Second, the theoretical guarantees in [12–17] were also established under single membership models—and oftentimes based on random query strategies (see, e.g., [12–14]). Note that in certain applications (e.g., field surveys based graph/network analysis [18]), systematic edge query strategies may be more preferred. Third, the existing methods are not always scalable. In particular, the convex optimization approaches in [12–14] recast the edge query-based graph clustering problem as a nuclear norm minimization problem, which entails N^2 (where N is the number of nodes) optimization variables—making it hard to handle real-world large-scale graphs.

Contributions. In this work, we offer an alternative framework for learning the node membership from an incomplete graph. Some notable features of our framework are as follows:

• **Systematic Edge Query Strategy.** Our algorithm design comes with a *systematic* edge query scheme, under which the node membership can be provably learned. As discussed, systematic query may be more preferable than random query strategies in a variety of applications.

• **Guaranteed Mixed Membership Identification.** Unlike existing provable edge-query based graph clustering methods in [12–17] whose goals are to learn single membership of nodes belonging to disjoint clusters, we model the undirected adjacency graphs using a mixed membership model that allows nodes to be associated with multiple overlapping clusters. Our model is reminiscent of the *mixed membership stochastic block* (MMSB) model in overlapped community detection [2]. Accordingly, we offer mixed membership identification guarantees using queried edges. Another contribution is that we derive a new sufficient condition for mixed membership identification. Compared to existing MMSB identifiability conditions, e.g., those in [19–22], our new condition is more closely tied to key characterizations of the GC problem, e.g., the Dirichlet prior of node membership, the graph size, and the cluster-cluster interaction intensity.

• **Scalable and Lightweight Algorithm.** We propose a simple procedure that only consists of the truncated *singular value decomposition* (SVD) of small matrices and a Gram–Schmidt-type greedy algorithm for *simplex-structured matrix factorization* (SSMF). This procedure is computationally very economical, especially when compared to the convex optimization approaches in [12–14, 23]. We also provide sample complexity and noise robustness analyses for the proposed framework.

• **Evaluation on Real-World Datasets.** We conduct a series of experiments on real-world datasets. First, we consider the query-based crowdclustering task [12, 13], using annotator-labeled graph data to assist image clustering. The data used in our evaluation was uploaded to the *Amazon Mechanical Turk* (AMT) platform and labeled by unknown human annotators. The acquired AMT data has been made publicly available[1]. Second, we also use co-authorship network datasets, *Digital Bibliography & Library Project* (DBLP) and *Microsoft Academic Graph* (MAG), to evaluate our method on community detection tasks.

An initial and limited version of this work was published in the proceedings of IEEE ICASSP 2021 [24]. In this journal version, we additionally include (1) detailed identifiability analysis of the proposed algorithm (Theorem 1 and its proof); (2) the integrated performance characterization of the SVD and SSMF stages (Theorem 2 and its proof); (3) a new application, namely, crowdclustering; (4) a series of new real-data experiments; (5) and a newly acquired dataset for crowdclustering that is made publicly available.

Notation. $x, \boldsymbol{x}, \boldsymbol{X}$ denote a scalar, vector and matrix, respectively. $\kappa(\boldsymbol{X})$ denotes the condition number of the matrix $\boldsymbol{X}$ and is given by $\kappa(\boldsymbol{X}) = \frac{\sigma_{\max}(\boldsymbol{X})}{\sigma_{\min}(\boldsymbol{X})}$ where $\sigma_{\max}$ and $\sigma_{\min}$ are the largest and smallest nonzero singular values of $\boldsymbol{X}$, respectively. $\boldsymbol{X} \geq \boldsymbol{0}$ denotes that all the elements of $\boldsymbol{X}$ are nonnegative. $\|\boldsymbol{X}\|_2$ represents the 2-norm of matrix $\boldsymbol{X}$, i.e., $\|\boldsymbol{X}\|_2 = \sigma_{\max}(\boldsymbol{X})$. $\|\boldsymbol{X}\|_{\mathrm{F}}$ denotes the Frobenius norm of $\boldsymbol{X}$. $\|\boldsymbol{X}\|_*$

[1] https://github.com/shahanaibrahimosu/mixed-membership-graph-clustering

denotes the nuclear norm of $\boldsymbol{X}$. $\text{svd}_K(\boldsymbol{X})$ denotes the top-K singular value decomposition (SVD) of $\boldsymbol{X}$. $\text{range}(\boldsymbol{X})$ denotes the subspace spanned by the columns of $\boldsymbol{X}$. $\boldsymbol{X}(i,j)$ and $x_{i,j}$ both denote the element of $\boldsymbol{X}$ from its ith row and jth column. $\boldsymbol{X}(:,j)$ and $\boldsymbol{x}_j$ both denote the jth column of $\boldsymbol{X}$. $\|\boldsymbol{x}\|_2$ and $\|\boldsymbol{x}\|_1$ denote ℓ_2 and ℓ_1 norms of vector $\boldsymbol{x}$, respectively. $^\top$ and † denote transpose and pseudo-inverse, respectively. $|\mathcal{C}|$ denotes the cardinality of set $\mathcal{C}$. $\cup$ denotes the union operator for sets. $[T]$ represents the set of positive integers up to T, i.e., $[T] = \{1, \ldots, T\}$. $\text{Diag}(\boldsymbol{x})$ is a diagonal matrix that holds the entries of $\boldsymbol{x}$ as its diagonal elements.

2 Problem Statement

Consider N data entities (e.g., persons in a social network or image data in a sample-sample similarity network) that are from K clusters. We allow a node to belong to multiple clusters; i.e., we consider the case where the clusters have overlaps and the nodes admits *mixed membership* [2]. Assume that the nth entity belongs to cluster k with probability $m_{k,n}$, where

$$\sum_{k=1}^{K} m_{k,n} = 1,\ m_{k,n} \geq 0. \tag{1}$$

Then, the vector $\boldsymbol{m}_n = [m_{1,n}, \ldots, m_{K,n}]^\top$ is referred to as the *membership vector* of data entity n, since it reflects the association of entity n with different clusters. All such vectors together constitute the *membership matrix* $\boldsymbol{M} = [\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N] \in \mathbb{R}^{K \times N}$.

In GC, the entities are the nodes of the graph, and the relationship between the nodes are represented as the edges of the graph. In this work, we consider undirected graphs that are represented as symmetric adjacency matrices, i.e., $\boldsymbol{A} \in \{0,1\}^{N \times N}$, where each entry $\boldsymbol{A}(i,j)$ encodes the pairwise relationship between nodes i and j in terms of binary values (i.e., $\boldsymbol{A}(i,j) \in \{0,1\}$). Our goal is to learn the membership vector $\boldsymbol{m}_n$ for each node from limited edge queries about the adjacency matrix $\boldsymbol{A}$ (i.e., the number of edges queried is much smaller than total number of edges, i.e., $N(N-1)/2$).

2.1 Motivating Examples

Our problem setting is motivated by a couple of important applications, namely, the crowdclustering problem and incomplete graph based overlapping community detection.

• **Example 1: Crowdclustering.** *Crowdclustering* is a technique that clusters data samples with the assistance from crowdsourced annotators [5,12,13,23,25,26] [15–17]. In crowdclustering, the data entities are presented to the annotators in pairs. The annotators determine if the pair of entities are from the same category or not; i.e., the annotators apply a rule such that $\boldsymbol{A}(i,j) = 1$ if entities i, j $(i < j)$ are believed to be from the same cluster, and $\boldsymbol{A}(i,j) = 0$ otherwise. Instead of asking the annotators to determine the exact membership of the nth data entity (i.e., the $\boldsymbol{m}_n$ vector), the above mentioned rule only asks the annotators to output binary labels (i.e., if two entities are similar or not). This annotation paradigm is arguably more accurate than exact membership labeling, since it can work under much less expertise.

Crowdclustering amounts to learning $\boldsymbol{m}_n$ from $\boldsymbol{A}$, which is a GC problem. The challenge is that annotating the full adjacency graph requires $O(N^2)$ pairs to be compared and labeled—a heavy workload if large data sets are considered. Hence, edge query-based GC techniques are natural to handle the crowdclustering task. This way, only a subset of the $O(N^2)$ sample pairs needs to be compared and annotated.

• **Example 2: Edge Sampling-Based Overlapping Community Detection.** The mixed membership modeling in (1) and the learning problem of interest can also be applied to *overlapping community detection* (OCD) [27]—since the mixed membership setting implies that the communities have overlaps. Classic OCD algorithms work with statistical generative models, e.g., the MMSB [2, 22] and the *Bayesian nonnegative matrix factorization model* [28]. The mixed membership identifiability guarantees were established under full observation of $\boldsymbol{A}$ in [19–22]. Mixed membership identification for OCD under sampled edges is of great

interest for applications like field survey based community analysis [18] or community detection in link-removed/hidden networks (e.g., terrorist networks) [9]. In both cases, one may not be able to observe the entire $\boldsymbol{A}$ due to reasons such as resource limitations and difficulty of edge acquisition. However, theoretical guarantees and provable algorithms for this problem have been elusive.

2.2 Prior Work

A theoretical challenge associated with our problem is as follows: Is it possible to learn the membership vector $\boldsymbol{m}_n$ from the binary matrix $\boldsymbol{A}$? This gives rise to a membership *identifiability* problem. If $\boldsymbol{A}$ is fully observed, answering the identifiability question amounts to understanding theoretical guarantees for similarity graph-based membership learning techniques, e.g., MMSB identification and spectral clustering—which has been thoroughly studied in the machine learning community; see, e.g., [19–22, 29]. However, when $\boldsymbol{A}$ is only partially observed, the identifiability problem has not been fully understood.

To handle the GC problem with partial observations, the work in [12, 13] models the generating process of $\boldsymbol{A}$ using the SBM followed by a random edge query stage for data acquisition. The SBM can be summarized as follows. Assume that $\boldsymbol{B} \in \mathbb{R}^{K\times K}$ represents a cluster-cluster similarity matrix which is symmetric and $\boldsymbol{B}(p,q)$ represents the probability that cluster p is connected with cluster q. In addition, assume that the nodes are connected through their membership identities. Then, the probability that $\boldsymbol{A}(i,j) = 1$ is $\boldsymbol{P}(i,j) = \boldsymbol{m}_i^\top \boldsymbol{B}\boldsymbol{m}_j$, i.e., $\boldsymbol{A}(i,j) \sim \mathsf{Bernoulli}\left(\boldsymbol{m}_i^\top \boldsymbol{B}\boldsymbol{m}_j\right)$ where $i<j$—the adjacency matrix $\boldsymbol{A}$ is sampled from Bernoulli distributions specified by the entries of the matrix $\boldsymbol{P} = \boldsymbol{M}^\top \boldsymbol{B}\boldsymbol{M}$. Note that under SBM, the membership vector $\boldsymbol{m}_i$ is always a unit vector; i.e., only single membership and disjoint clusters are considered under SBM.

To recover $\boldsymbol{M}$ from partially observed $\boldsymbol{A}$ under SBM, the work in [12–14, 30] use the following convex program and its variants:

$$\begin{aligned} \underset{\boldsymbol{L}\in[0,1]^{N\times N},\boldsymbol{S}}{\text{minimize}} \ & \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{S}\|_1 \\ \text{subject to } & \boldsymbol{L}(i,j) + \boldsymbol{S}(i,j) = \boldsymbol{A}(i,j),\ \forall (i,j) \in \boldsymbol{\Omega}. \end{aligned} \tag{2}$$

In the above, $\boldsymbol{L}$ is used to model the low-rank component of $\boldsymbol{A}$, $\boldsymbol{S}$ is a sparse error matrix, and $\boldsymbol{\Omega}$ denotes the set of pairs (i,j) such that the corresponding $\boldsymbol{A}(i,j)$'s are observed. Notably, it was shown in [12–14] that if $\boldsymbol{A}$ follows the SBM, then the above convex program (and its variants) recovers $\boldsymbol{A}$ up to certain errors with provable guarantees using *random* edge queries. The results from [12–14] have opened many doors for edge query-based graph clustering, but several caveats exist. In particular, the recoverability is established based on single membership models, but the mixed membership case is of more interest. In addition, the optimization problem involves $O(N^2)$ variables (memory$\approx 3,000$GB when $N = 10^6$), and thus is hardly scalable. Also, the provable guarantees in [12–14] rely on random edge query, which may not be easy to implement under some scenarios, as we discussed.

Query-based GC was also studied in another line of work [15–17]. There, the focus is developing active/online edge query-based methods that can discover the clusters on-the-fly, while attaining certain information-theoretic query lower bounds. Unlike the methods in [12–14] that take a latent model identification perspective, the algorithms in [15–17] are more from a successive clique (i.e., fully connected subgraph) finding perspective (although some connections to the SBM were also made). Similar to the convex approaches, the methods in [15–17] are designed to learn single membership.

3 Proposed Approach

In this work, we relax the SBM assumption on $\boldsymbol{P} = \boldsymbol{M}^\top \boldsymbol{B}\boldsymbol{M}$ by allowing the nodes to have mixed membership, as advocated in [2, 22, 31]. Hence, the constraints on $\boldsymbol{M}$ becomes

$$\mathbf{1}^\top \boldsymbol{M} = \mathbf{1}^\top, \quad \boldsymbol{M} \geq \mathbf{0}; \tag{3}$$

i.e., $\boldsymbol{m}_n$ resides in the probability simplex, instead of being the vertices of the simplex as in the SBM. Under the mixed membership assumption in (3), the following Bernoulli model, i.e.,

$$\begin{cases} \boldsymbol{A}(i,j) \sim \mathsf{Bernoulli}\left(\boldsymbol{m}_i^\top \boldsymbol{B}\boldsymbol{m}_j\right), \ i<j, \\ \boldsymbol{A}(i,j) = \boldsymbol{A}(j,i), \ i>j, \\ \boldsymbol{A}(i,i) = 0, \ i \in [N], \end{cases} \tag{4}$$

is adopted in our generative model for the complete adjacency matrix. Note that $\boldsymbol{m}_i^\top \boldsymbol{B}\boldsymbol{m}_j$ physically corresponds to the probability that nodes i and j admit an edge (i.e., $\mathsf{Pr}(\boldsymbol{A}(i,j)=1) = \boldsymbol{m}_i^\top \boldsymbol{B}\boldsymbol{m}_j$), and thus this Bernoulli model makes intuitive sense; see more discussions in [2,22,31].

3.1 Systematic Edge Query

Our goal is to learn $\boldsymbol{M}$ from controlled edge sampling strategies. To proceed, we first divide the nodes into L disjoint groups $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_L$ such that $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_L = [N]$. Let $\boldsymbol{A}_{\ell,m} \in \mathbb{R}^{|\mathcal{S}_\ell| \times |\mathcal{S}_m|}$ denote the adjacency matrix between groups of nodes indexed by $\mathcal{S}_\ell$ and $\mathcal{S}_m$. We propose an edge query principle as described in the following:

Edge Query Principle (EQP). Query the blocks of edges such that the sampled $\boldsymbol{A}_{\ell,m}$'s satisfy the following two conditions:
- For every $\ell \in [L]$, $K \le |\mathcal{S}_\ell|$ holds.
- Let $m_r \in [L]$ and $\{\ell_r\}_{r=1}^L = [L]$. For every ℓ_r, there exists a pair of indices m_r and ℓ_{r+1} where $\ell_{r+1} \neq \ell_r$ such that the edges from the blocks $\boldsymbol{A}_{\ell_r,m_r}$ and $\boldsymbol{A}_{\ell_{r+1},m_r}$ are queried.

Note that since $\boldsymbol{A}$ is symmetric, when we select a block $\boldsymbol{A}_{\ell,m}$ to be queried, it also covers $\boldsymbol{A}_{m,\ell}$ since $\boldsymbol{A}_{m,\ell} = \boldsymbol{A}_{\ell,m}^\top$. A couple of remarks are in order:

Remark 1 *The proposed EQP covers a large variety of query 'masks'—as shown in Fig. 1. Since the query pattern can be by design instead of random, this entails the flexibility to avoid querying edges that are known a priori hard to acquire, e.g., edges that may have been intentionally removed to conceal information or edges that correspond to interactions between groups that are hard to survey due to various reasons, e.g., long geographical distance.*

Remark 2 *Instead of sampling individual edges, we sample blocks of edges under the proposed EQP. As one will see, this simple block query pattern allows us to design a provable and lightweight algorithm for mixed membership learning. The proposed query strategy can be easily implemented when the query patterns are controlled by the network analysts. For example, in crowdclustering, one can dispatch blocks of sample pairs for similarity annotation. In community analysis, field surveys can be designed using EQP such that the blocks which are easier to be queried are selected.*

3.2 Connections to Matrix Completion

From a matrix completion viewpoint, our EQP is most closely related to those in [32, 33] that consider recoverable sampling patterns of matrix completion (MC). The work in [32] relates the recoverability of a low-rank matrix to the structural properties of a bipartite graph by considering the rows and columns as nodes. It establishes the "finite" (not unique) completability of the matrix based on combinatorial conditions on the observed edges in its subgraphs. In a similar spirit, the work in [33] shows that the range space of a low-rank matrix can be uniquely identified under the existence of certain patterns formed by the sampled entries.

From an MC perspective, our block sampling pattern can be considered as special cases covered by those in [32, 33]—with important distinguishing features. First, the work in [32, 33] established recoverability under the sampling pattern, but had no tractable algorithms. Our EQP features a polynomial-time

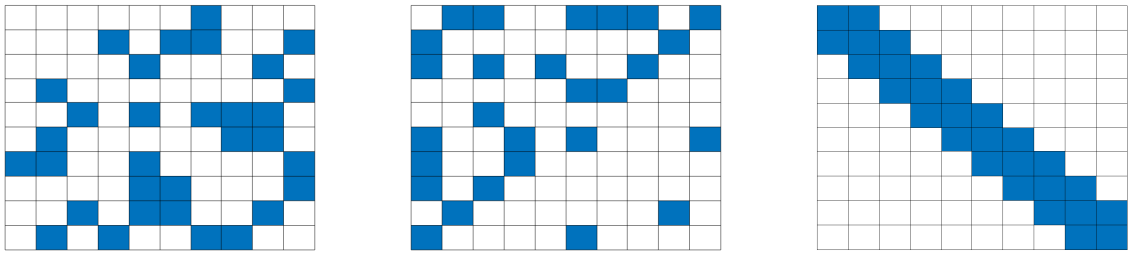

Figure 1: Some example patterns for $\boldsymbol{A}$ following EQP with $N = 1000$, $K = 5$ and $L = 10$. The shaded blue region represents the blocks queried.

lightweight algorithm via exploiting the special block sampling pattern. Second, the recoverability results in [32, 33] only cover sampled continuous real-valued low-rank matrices without noise. Our approach can provably estimate the range space of binary matrices (or, highly noisy low-rank matrices) under reasonable conditions.

3.3 Algorithm Design

In this section, we propose an algorithm under the EQP. Specifically, under the considered model in (4). we develop an algorithm that consists of simple SVD operations to estimate $\mathsf{range}(\boldsymbol{M}^\top)$ and a subsequent SSMF stage to 'extract' $\boldsymbol{M}$ from the estimated range space.

3.3.1 Main Idea – A Toy Example

To shed some light on how our algorithm approximately identifies $\mathsf{range}(\boldsymbol{M}^\top)$, let us consider the ideal case where $\boldsymbol{A}_{\ell,m} = \boldsymbol{P}_{\ell,m} = \boldsymbol{M}_\ell^\top \boldsymbol{B} \boldsymbol{M}_m$. We start by analyzing a toy example with $L = 3$ and $K \leq N/3$; see Fig. 2. We assume that the blocks $\boldsymbol{P}_{1,2} \in \mathbb{R}^{N/3 \times N/3}$, $\boldsymbol{P}_{2,2} \in \mathbb{R}^{N/3 \times N/3}$, and $\boldsymbol{P}_{3,1} \in \mathbb{R}^{N/3 \times N/3}$ are queried. Hence, by symmetry, the following blocks are known:

$$\boldsymbol{P}_{1,2} = \boldsymbol{M}_1^\top \boldsymbol{B} \boldsymbol{M}_2, \ \boldsymbol{P}_{2,2} = \boldsymbol{M}_2^\top \boldsymbol{B} \boldsymbol{M}_2, \tag{5}$$

$$\boldsymbol{P}_{2,1} = \boldsymbol{M}_2^\top \boldsymbol{B} \boldsymbol{M}_1, \ \boldsymbol{P}_{3,1} = \boldsymbol{M}_3^\top \boldsymbol{B} \boldsymbol{M}_1, \tag{6}$$

Define $\boldsymbol{C}_1 := [\boldsymbol{P}_{1,2}^\top \ , \ \boldsymbol{P}_{2,2}^\top]^\top$ and $\boldsymbol{C}_2 := [\boldsymbol{P}_{2,1}^\top \ , \ \boldsymbol{P}_{3,1}^\top]^\top$. The top-$K$ SVD of $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ can be represented as follows:

$$\boldsymbol{C}_1 = [\boldsymbol{U}_1^\top, \boldsymbol{U}_2^\top]^\top \boldsymbol{\Sigma} \, \boldsymbol{V}_2^\top, \ \boldsymbol{C}_2 = [\overline{\boldsymbol{U}}_2^\top, \overline{\boldsymbol{U}}_3^\top]^\top \overline{\boldsymbol{\Sigma}} \, \overline{\boldsymbol{V}}_1^\top. \tag{7}$$

Combining (5)-(7), and under the assumption that $\mathrm{rank}(\boldsymbol{M}_\ell) = \mathrm{rank}(\boldsymbol{B}) = K$ and $K \leq |\mathcal{S}_\ell|$, for all ℓ, one can express the bases of $\mathsf{range}(\boldsymbol{M}_1^\top)$, $\mathsf{range}(\boldsymbol{M}_2^\top)$ and $\mathsf{range}(\boldsymbol{M}_3^\top)$ as $\boldsymbol{U}_1 = \boldsymbol{M}_1^\top \boldsymbol{G}$, $\boldsymbol{U}_2 = \boldsymbol{M}_2^\top \boldsymbol{G}$, $\overline{\boldsymbol{U}}_3 = \boldsymbol{M}_3^\top \overline{\boldsymbol{G}}$, respectively, where $\boldsymbol{G} \in \mathbb{R}^{K \times K}$ and $\overline{\boldsymbol{G}} \in \mathbb{R}^{K \times K}$ are certain nonsingular matrices. Our hope is to "stitch" the bases above to have

$$\begin{aligned} \mathsf{range}(\boldsymbol{U}) &= \mathsf{range}([\boldsymbol{U}_1^\top, \boldsymbol{U}_2^\top, \boldsymbol{U}_3^\top]^\top) \\ &= \mathsf{range}([\boldsymbol{M}_1, \boldsymbol{M}_2, \boldsymbol{M}_3]^\top), \end{aligned} \tag{8}$$

with $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ in (7) and a certain $\boldsymbol{U}_3$. Note that $\overline{\boldsymbol{U}}_3$ cannot be directly combined with $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ to attain the above, since $\boldsymbol{G} = \overline{\boldsymbol{G}}$ does not generally hold. To fix this, we define the following operation: $\boldsymbol{U}_3 := \overline{\boldsymbol{U}}_3 \overline{\boldsymbol{U}}_2^\dagger \boldsymbol{U}_2$. It is not hard to see that

$$\overline{\boldsymbol{U}}_3 \overline{\boldsymbol{U}}_2^\dagger \boldsymbol{U}_2 = \boldsymbol{M}_3^\top \overline{\boldsymbol{G}} \times \left(\boldsymbol{M}_2^\top \overline{\boldsymbol{G}}\right)^\dagger \times \boldsymbol{M}_2^\top \boldsymbol{G} = \boldsymbol{M}_3^\top \boldsymbol{G},$$

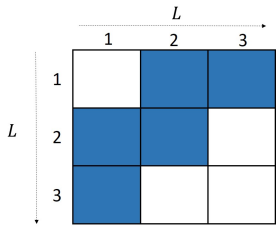

Figure 2: An illustrative case for subspace identifiability analysis. The shaded blue region represents the blocks queried.

Algorithm 1 `Proposed Algorithm: BeQuec`

Require: $L, K, \{\boldsymbol{A}_{\ell_r,m_r}\}_{r=1}^{L}, \{\boldsymbol{A}_{\ell_{r+1},m_r}\}_{r=1}^{L-1}$;
1: $T \leftarrow \lfloor L/2 \rfloor$;
2: $\boldsymbol{C}_T \leftarrow [\boldsymbol{A}_{\ell_T,m_T}^\top, \boldsymbol{A}_{\ell_{T+1},m_T}^\top]^\top$;
3: $[\boldsymbol{U}_{\ell_T}^\top, \boldsymbol{U}_{\ell_{T+1}}^\top]^\top \boldsymbol{\Sigma}_T \boldsymbol{V}_{m_T}^\top \leftarrow \mathrm{svd}_K(\boldsymbol{C}_T)$;
4: $[\boldsymbol{U}_{\mathrm{ref2}}^\top, \boldsymbol{U}_{\mathrm{ref1}}^\top]^\top \leftarrow [\boldsymbol{U}_{\ell_T}^\top, \boldsymbol{U}_{\ell_{T+1}}^\top]^\top$; ▷ Init. temp. variables
5: **for each** $r = T+1 : 1 : L-1$ **do** ▷ Est. $\{\boldsymbol{U}_{\ell_r}\}_{r=T+2}^{L}$
6: $\quad \boldsymbol{U}_{\ell_{r+1}} \leftarrow \mathtt{PairStitch}(\boldsymbol{A}_{\ell_r,m_r}, \boldsymbol{A}_{\ell_{r+1},m_r}, \boldsymbol{U}_{\mathrm{ref1}})$
7: $\quad \boldsymbol{U}_{\mathrm{ref1}} \leftarrow \boldsymbol{U}_{\ell_{r+1}}$;
8: **end for each**
9: **for each** $r = T : -1 : 2$ **do** ▷ Est. $\{\boldsymbol{U}_{\ell_r}\}_{r=1}^{T-1}$
10: $\quad \boldsymbol{U}_{\ell_{r-1}} \leftarrow \mathtt{PairStitch}(\boldsymbol{A}_{\ell_{r-1},m_{r-1}}, \boldsymbol{A}_{\ell_r,m_{r-1}}, \boldsymbol{U}_{\mathrm{ref2}})$
11: $\quad \boldsymbol{U}_{\mathrm{ref2}} \leftarrow \boldsymbol{U}_{\ell_{r-1}}$;
12: **end for each**
13: $\widehat{\boldsymbol{U}} \leftarrow [\boldsymbol{U}_1^\top, \ldots, \boldsymbol{U}_L^\top]^\top$;
14: $\widehat{\boldsymbol{G}} \leftarrow \mathtt{SPA}(\widehat{\boldsymbol{U}}, K)$ [35]; ▷ Est. $\boldsymbol{G}$ under (9)
15: $\widehat{\boldsymbol{M}} \leftarrow \widehat{\boldsymbol{G}}^{-\top}\widehat{\boldsymbol{U}}^\top$ (or constrained least squares);
16: **return** $\widehat{\boldsymbol{M}}$

which leads to (8). The idea conveyed by this simple example can be recursively applied to cover a general L block case under the proposed EQP, and the range space estimation accuracy can be guaranteed even when $\boldsymbol{A}$ is a noisy (binary) version of $\boldsymbol{P}$—which will be detailed in the next part. After $\boldsymbol{U}^\top = \boldsymbol{G}^\top \boldsymbol{M}$ is obtained, extracting $\boldsymbol{M}$ from $\boldsymbol{U}^\top$ is the so-called SSMF problem, which has been extensively studied [34]; see a brief discussion in Sec. 3.4.2 and the supplementary material (Sec. L).

3.3.2 Proposed Algorithm

In a similar spirit as alluded in the toy example, we propose Algorithm 1 (which is referred to as the *block edge query-based graph clustering* (`BeQuec`) algorithm). One can see that the `BeQuec` algorithm only consists of top-K SVD and least squares, which can be applied to very large graphs as long as K is of a moderate size. The SSMF stage (carried out by the *successive projection algorithm* (`SPA`) [35]) is a Gram–Schmidt type algorithm that is also scalable (see Algorithm 3 in the supplementary material in Sec. L). Note that in Algorithm 1, $\boldsymbol{U}_{\mathrm{ref1}}$ and $\boldsymbol{U}_{\mathrm{ref2}}$ are temporary variables defined to pass the subspace estimated in the current iteration to the subsequent iteration. We further explain the two major stages of `BeQuec`, namely, the range space estimation stage and the membership extraction stage as follows:

3.4 Key Ingredients of `BeQuec`

3.4.1 Range Space Estimation (Lines 1-13)

In a nutshell, `BeQuec` uses the idea from the toy example in an iterative fashion on the queried blocks $\boldsymbol{A}_{\ell_r,m_r}$ and $\boldsymbol{A}_{\ell_{r+1},m_r}$ for $r = 1, \ldots, L-1$—to estimate $\boldsymbol{U}^\top = \boldsymbol{G}^\top \boldsymbol{M}$, i.e., the range space of $\boldsymbol{M}^\top$. We start the

Algorithm 2 `PairStitch`

Require: $\boldsymbol{A}_{\ell,m}, \boldsymbol{A}_{\ell',m}, \boldsymbol{U}_{\rm ref}$
1: $\boldsymbol{C} \leftarrow [\boldsymbol{A}_{\ell,m}^\top,\ \boldsymbol{A}_{\ell',m}^\top]^\top$;
2: $[\overline{\boldsymbol{U}}_\ell^\top, \overline{\boldsymbol{U}}_{\ell'}^\top]^\top \boldsymbol{\Sigma} \boldsymbol{V}_m^\top \leftarrow \mathrm{svd}_K(\boldsymbol{C})$;
3: $\boldsymbol{U}_\ell \leftarrow \overline{\boldsymbol{U}}_\ell \overline{\boldsymbol{U}}_{\ell'}^\dagger \boldsymbol{U}_{\rm ref}$; ▷ "Stitch" $\boldsymbol{U}_\ell$ and $\boldsymbol{U}_{\rm ref}$
4: **return** $\boldsymbol{U}_\ell$.

iterations from $r = \lfloor L/2 \rfloor$ and perform the subspace stitching of the blocks in the ascending and descending orders, respectively—which helps reduce the subspace estimation error from the overall procedure, as will be seen in the analysis of Theorem 2.

3.4.2 SSMF for Membership Extraction (Lines 14-15)

If $\boldsymbol{U}$ is perfectly estimated, the second stage estimates $\boldsymbol{M}$ from the following SSMF model:

$$\boldsymbol{U}^\top = \boldsymbol{G}^\top \boldsymbol{M},\ \boldsymbol{M} \geq \boldsymbol{0},\ \boldsymbol{1}^\top \boldsymbol{M} = \boldsymbol{1}^\top, \tag{9}$$

where $\boldsymbol{G} \in \mathbb{R}^{K \times K}$ is nonsingular. The SSMF problem is the cornerstone of a number of core tasks in machine learning and signal processing, e.g., hyperspectral unmixing and topic modeling, which admits a plethora of algorithms for provably estimating $\boldsymbol{M}$ under certain conditions [34]. To be precise, line 14 of Algorithm 1 invokes an SSMF algorithm, namely, `SPA` [35]. Under the model in (9), `SPA` guarantees the identifiability of $\boldsymbol{M}$ utilizing the so-called *anchor node assumption* (ANC). Here, "identifiability" of $\boldsymbol{M}$ means that learning $\boldsymbol{M}$ up to a row permutation ambiguity can be guaranteed; see [36–38].

In essence, the ANC is identical to the so-called *separability* condition [39] from the SSMF and nonnegative matrix factorization (NMF) literature:

Definition 1 *(ANC/Separability) [21, 39, 40] The nonnegative matrix $\boldsymbol{M} \in \mathbb{R}^{K \times N}$ is said to satisfy the separability condition/ANC if there exists an index set $\boldsymbol{\Lambda} = \{q_1, \ldots, q_K\}$ such that $\boldsymbol{M}(:, q_k) = \boldsymbol{e}_k$ for all $k \in [K]$. When there exists $\boldsymbol{\Lambda} = \{q_1, \ldots, q_K\}$ such that $\|\boldsymbol{M}(:, q_k) - \boldsymbol{e}_k\|_2 \leq \varepsilon$ for all $k \in [K]$, then $\boldsymbol{M}$ is said to satisfy the ε-separability condition.*

In the context of GC, the ANC means that there exists a node q_k (i.e., an anchor node) that only belongs to cluster k for $k \in [K]$. Under the ANC, one can see that if $\boldsymbol{\Lambda}$ is known, then $\boldsymbol{G} = \boldsymbol{U}(\boldsymbol{\Lambda}, :)$ can be easily "read out" from the rows of $\boldsymbol{U}$, which also enables estimating $\boldsymbol{M}$ using (constrained) least squares as well (see line 15 in Algorithm 1). The `SPA` algorithm identifies $\boldsymbol{\Lambda}$ using a Gram-Schmidt-like procedure in K iterations and its per-iteration complexity is $O(NK^2)$ operations [34, 35]; also see Sec. L in the supplementary material for details.

Remark 3 *A number of MMSB learning algorithms rely on the ANC to establish identifiability of $\boldsymbol{M}$; see, e.g., [19–21]. However, ANC does not reflect many important aspects of the GC problem, other than the existence of some anchor nodes. In the next subsection, we will present a sufficient condition that has a different flavor, and is arguably more intuitive in the context of GC (e.g., graph size and the cluster-cluster interaction pattern). Another remark is that `SPA` is not the only algorithm for separable SSMF/NMF; see more options in [41–46]. Nonetheless, `SPA` strikes a good balance between complexity and accuracy.*

3.5 Performance Analysis

3.5.1 Algorithm Complexity

Note that Algorithm 1 only consists of top-K SVD on small blocks that have a size of $2(N/L) \times (N/L)$ assuming $|\mathcal{S}_\ell| = N/L$ for all ℓ, which has a complexity of $O((N/L)K^2)$ flops—a similar complexity order of

the SPA stage. In terms of memory, the process never needs to instantiate any $N \times N$ matrix that is used in the convex programming based methods [cf. (2)]. Instead, the variables involved are with the size of $N \times K$, thereby being economical in terms of memory (since the number of clusters K is often much smaller than N).

3.5.2 Accuracy

We first use the ideal case where $\boldsymbol{A}(i,j) = \boldsymbol{P}(i,j)$ to analyze the key idea behind our proposed approach. Building on the understanding to the ideal noiseless case, the binary observation case, i.e, the model in (4), will be analyzed by treating it as a noisy version of the ideal case, with extensive care paid to the noise induced by the Bernoulli observation process.

To be specific, we have the following subspace identification result:

Theorem 1 *(Ideal Case) Assume that $\boldsymbol{A}_{\ell,m} = \boldsymbol{P}_{\ell,m} = \boldsymbol{M}_\ell^\top \boldsymbol{B} \boldsymbol{M}_m \in \mathbb{R}^{|\mathcal{S}_\ell| \times |\mathcal{S}_m|}$ holds true for all $\ell, m \in [L]$ and $\mathrm{rank}(\boldsymbol{M}_\ell) = \mathrm{rank}(\boldsymbol{B}) = K$. Suppose that the $\boldsymbol{A}_{\ell,m}$'s are queried according to the proposed EQP. Then, the output $\widehat{\boldsymbol{U}}$ by Algorithm 1 satisfies $\mathsf{range}(\widehat{\boldsymbol{U}}) = \mathsf{range}(\boldsymbol{M}^\top)$.*

The proof of the theorem is relegated to Appendix A. The proof showcases how the range spaces of $\boldsymbol{M}_\ell^\top$'s are 'stitched' together to recover $\mathsf{range}(\boldsymbol{M}^\top)$, if $\boldsymbol{A}_{\ell,m} = \boldsymbol{P}_{\ell,m}$.

However, the Bernoulli parameters (i.e., the $\boldsymbol{P}_{\ell,m}$'s) are not observed in practice. Instead, one observes $\boldsymbol{A}_{\ell,m}$'s with binary values, following the Bernoulli observation model in (4). The Bernoulli observation process can be understood as a noisy data acquisition process. To characterize the performance under this noisy case, we consider the following assumption that is also reminiscent of the classic MMSB model [2]:

Assumption 1 *The membership vectors $\boldsymbol{m}_n$ $\forall n \in [N]$ are independently drawn from the Dirichlet distribution with parameter $\boldsymbol{\nu} = [\nu_1, \ldots, \nu_K]^\top$, i.e., $\boldsymbol{m}_n \sim \mathsf{Dir}(\boldsymbol{\nu})$ for $n \in [N]$.*

We first connect the Dirichlet model with ANC/separability using the following proposition:

Proposition 1 *Let $\mu > 0, \varepsilon > 0$. Under Assumption 1, assume that the number of nodes N satisfies*

$$N = \Omega\left((G(\varepsilon, \boldsymbol{\nu}))^{-2} \log\left(K/\mu\right) \right), \tag{10}$$

where

$$G(\varepsilon, \boldsymbol{\nu}) = \frac{\Gamma(\sum_{i=1}^K \nu_i)}{\prod_{i=1}^K \Gamma(\nu_i)} \min_k \frac{\varepsilon^{\sum_{i \neq k} \nu_i}}{\prod_{i \neq k} \nu_i}, \tag{11}$$

and the gamma function $\Gamma(z) = \int_0^\infty x^{z-1} \exp(-x) dx$ for $z > 0$. Then, with probability of at least $1 - \mu$, $\boldsymbol{M}$ satisfies the ε-separability condition.

The proof can be found in Appendix B. This result translates the ANC to a condition that relies on the graph size N. To explain the condition in (10), it is critical to understand $G(\varepsilon, \boldsymbol{\nu})$. The value of $G(\varepsilon, \boldsymbol{\nu})$ is affected by how well $\boldsymbol{m}_n$'s are spread in the probability simplex. To illustrate this, consider a case with $N = 1000$, $K = 3$ and $\varepsilon = 0.1$. Table 1 tabulates the value of $G(\varepsilon, \boldsymbol{\nu})$ under this case for different Dirichlet parameters $\boldsymbol{\nu}$, namely, $[0.5, 0.5, 0.5]^\top$, $[2.0, 0.5, 0.5]^\top$, and $[3.0, 3.0, 3.0]^\top$. The corresponding $\boldsymbol{m}_n$'s distributions in the probability simplex are shown in Fig. 3. One can see from Table 1 and (10) that $\boldsymbol{\nu} = [0.5, 0.5, 0.5]^\top$ has a larger G-function value and thus requires a smaller N to satisfy the ε-separability condition. This is consistent with the illustration in Fig. 3.

Under Assumption 1, if $|\mathcal{S}_\ell| \geq K$, then $\mathrm{rank}(\boldsymbol{M}_\ell) = K$ holds with probability one [47, 48]. This means that the following also holds with probability one:

$$\gamma := \max_\ell \kappa(\boldsymbol{M}_\ell) < \infty.$$

To proceed, we adopt the following definition of node degree that is widely used in network analysis [49–51]:

Table 1: The value of the function $G(\varepsilon, \boldsymbol{\nu})$ for different Dirichlet parameter $\boldsymbol{\nu}$ fixing $K = 3$ and $\varepsilon = 0.1$.

$\boldsymbol{\nu}$	$G(\varepsilon, \boldsymbol{\nu})$
$[0.5, 0.5, 0.5]^\top$	0.045
$[2.0, 0.5, 0.5]^\top$	8.4×10^{-4}
$[3.0, 3.0, 3.0]^\top$	7.0×10^{-5}

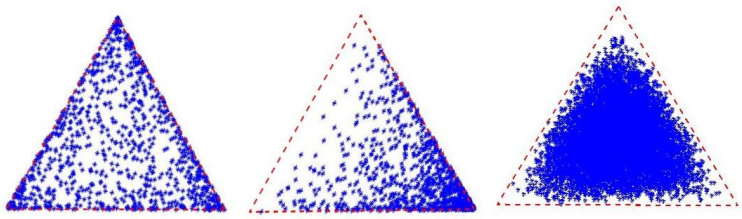

Figure 3: The columns of $\boldsymbol{M} \in \mathbb{R}^{3\times 1000}$ generated via the Dirichlet distribution with $\boldsymbol{\nu} = [0.5, 0.5, 0.5]^\top$ (left), $\boldsymbol{\nu} = [2.0, 0.5, 0.5]^\top$ (middle) and $\boldsymbol{\nu} = [3.0, 3.0, 3.0]^\top$ (right) spread in the 2-dimensional probability simplex.

Definition 2 *The degree of node i is the number of "similar nodes" it has in the adjacency graph; i.e.,* $\mathsf{degree}(i) = \sum_{j=1}^{N} \boldsymbol{A}(i,j)$, *where* $\boldsymbol{A} \in \{0,1\}^{N\times N}$.

Using Assumption 1, Proposition 1, the notations γ and $\mathsf{degree}(i)$, we state the main theorem:

Theorem 2 *(Binary Observation Case) Assume that the matrix $\boldsymbol{A}$ is generated following (3) and (4) and $\mathrm{rank}(\boldsymbol{B}) = K$. Also assume that $\boldsymbol{A}_{\ell,m}$'s are queried following the EQP, with $|\mathcal{S}_\ell| = N/L$ for every ℓ, where N/L is an integer. Let $\rho := \max_{i,j} \boldsymbol{P}(i,j)$ be the maximal entry of $\boldsymbol{P}$. Suppose that $\rho = \Omega(L\log(N/L)/N)$ and $L = O(\rho N/d)$ where d is the maximal degree of all the nodes. Also assume that*

$$N = \Omega\left(\max\left(L^2, \frac{\log\left(NK/L^2\right)}{G(\varepsilon,\boldsymbol{\nu})^2}, \frac{(K\gamma^2)^L \rho \kappa^2(\boldsymbol{B})}{\sigma^2_{\min}(\boldsymbol{B})}\right)\right) \tag{12}$$

for a certain constant $0 < \varepsilon = O\left(\frac{1}{K\gamma^3}\right)$. Then, the output $\widehat{\boldsymbol{U}}$ and $\widehat{\boldsymbol{M}}$ by Algorithm 1 satisfy the following with probability of at least $1 - O(L^2/N)$:

$$\|\widehat{\boldsymbol{U}} - \boldsymbol{U}\boldsymbol{O}\|_{\mathrm{F}} = \zeta = O\left(\frac{(K\gamma^2)^{L/2}\kappa(\boldsymbol{B})\sqrt{\rho}}{\sigma_{\min}(\boldsymbol{B})\sqrt{N/L}}\right),$$
$$\min_{\boldsymbol{\Pi}} \|\widehat{\boldsymbol{M}} - \boldsymbol{\Pi}\boldsymbol{M}\|_{\mathrm{F}} = O\left(K^2\sigma_{\max}(\boldsymbol{M})\gamma^3 \max(\varepsilon, \zeta)\right),$$

where $\boldsymbol{U}$ is an orthogonal basis of $\mathsf{range}(\boldsymbol{M}^\top)$, $\boldsymbol{O} \in \mathbb{R}^{K\times K}$ is an orthogonal matrix and $\boldsymbol{\Pi} \in \mathbb{R}^{K\times K}$ is a permutation matrix.

The proof can be found in Appendix C, which is an integration of (i) range space estimation accuracy under Bernoulli observations, (ii) least squares perturbation analysis, and (iii) noise robustness analysis of SPA. In particular, the conditions on ρ and d are needed for range space estimation under binary observations, while the conditions on N are used across parts (i)-(iii).

Theorem 2 can be understood as follows. First, under the presumed model, the proposed method favors larger graph sizes (i.e., larger N's) and larger query blocks (i.e., smaller L's). Particularly, the nested condition on N in (12) means that N should be larger than a certain threshold, since $N \geq E_1 \log(E_2 N)$ for

Table 2: The subspace distance between $\widehat{\boldsymbol{U}}$ and $\boldsymbol{U}$ for `BeQuec` in ideal and binary observation cases; $K = 5$; $L = 10$; $\boldsymbol{\nu} = [\frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}]^\top$.

Graph Size (N)	Ideal Case	Binary Obs. Case
	Dist	Dist
2000	2.48×10^{-14}	0.3214
4000	4.86×10^{-14}	0.2080
8000	2.63×10^{-14}	0.1412
10000	1.28×10^{-13}	0.1257

positive constants E_1 and E_2 holds if N is sufficiently large. Second, if the nodes tend to be more focused on a single cluster (i.e., ν_k is small for every k), the G-function's value in (11) becomes larger (cf. Fig. 3 and Table 1). Then, our method can reach a good accuracy level under smaller N's and larger L's. Third, the cluster-cluster interaction matrix $\boldsymbol{B}$ also matters: If the intra-cluster interactions dominate all node interactions and the cluster activities are balanced (i.e., $\boldsymbol{B}(k,k) \approx \boldsymbol{B}(j,j)$ for $j \neq k \in [K]$), then the condition number of $\boldsymbol{B}$ is expected to be small, thereby making membership learning easier.

Theorem 2 also sheds light on the impact of L. Ideally, one hopes to increase L since using a larger L means that the EQP uses fewer queries. With a proper L, the number of queries under the proposed EQP, i.e., $\Omega(N^2/L)$, can approach NK, which is the number of parameters to characterize any (unstructured) K-dimensional subspace in an N-dimensional space. However, L cannot be too large, as a large L may lead to the EQP condition $K \leq |\mathcal{S}_\ell| = N/L$ being violated. In addition, a larger L makes the error bound in Theorem 2 looser, since the factor $(K\gamma^2)^{L/2}$ increases with L. Hence, the theorem reveals an important trade-off that is of interest to query designers.

Remark 4 *Given the underlying SSMF structure of each block $\boldsymbol{A}_{\ell,m}$, a natural question is as follows: Instead of first estimating* range$(\boldsymbol{M}^\top)$ *and then estimating $\boldsymbol{M}$, why not estimate $\boldsymbol{M}_\ell$ (or $\boldsymbol{M}_m$) from $\boldsymbol{A}_{\ell,m}$ (using SSMF algorithms on the range space of $\boldsymbol{A}_{\ell,m}$, i.e., $\boldsymbol{U}_{\ell,m} \approx \boldsymbol{M}_\ell^\top \boldsymbol{G}_m$) and then stitch them together to recover $\boldsymbol{M}$? This is not infeasible—but is prone to failure. One reason is that this route needs that every block $\boldsymbol{M}_\ell$ satisfies the ANC (at least approximately). This is much more unrealistic than requiring the entire $\boldsymbol{M}$ to satisfy the ANC (which is also reflected in Proposition 1).*

4 Experiments

For both synthetic and real data experiments, we compare our algorithm with a number of state-of-the-art mixed membership learning algorithms, namely, *geometric intuition-based nonnegative matrix factorization* (`GeoNMF`) [19], *community detection via minimum volume simplex identification* (`CD-MVSI`) [22], and *community detection via Bayesian nonnegative matrix factorization* (`CD-BNMF`) [28]. These baselines are applied onto the queried blocks (e.g., $\boldsymbol{A}_{\ell,m}$) to estimate $\boldsymbol{M}_\ell$ and $\boldsymbol{M}_m$, as discussed in Remark 4. The estimated $\boldsymbol{M}_\ell$'s are aligned (by a nontrivial aligning procedure) to unify the row permutation ambiguity. This step is necessary since the estimated $\boldsymbol{M}_\ell$'s from different blocks are subject to different row permutation ambiguities; see details in Appendix M.

We have two different sets of real data experiments, i.e., crowdclustering of dog breed image data and community detection in co-author networks. For real data experiments, we additionally apply another mixed membership learning algorithm, namely, *sequential projection after cleaning* (`SPACL`) [20], two versions of the spectral clustering algorithm, namely, the unnormalized and normalized spectral clustering algorithms—denoted as `SC (unnorm.)` and `SC (norm.)`, respectively [52] and k`-means`. The k`-means` algorithm employed in these methods are repeated with 20 random initializations. We also use a convex optimization-based graph clustering algorithm with edge queries (denoted as `Convex-Opt`) [12] as a benchmark. We follow the setup in [12] and apply k`-means` directly on the adjacency graphs [denoted as

Table 3: The MSE, RE and averaged SRC of $\boldsymbol{M}$ for $K = 5$, $L = 10$ and $\boldsymbol{\nu} = [\frac{1}{5}\frac{1}{5}\frac{1}{5}\frac{1}{5}\frac{1}{5}]^\top$.

Graph Size (N)	Metric	BeQuec	GeoNMF	CD-MVSI	CD-BNMF
2000	MSE	**0.0536**	0.1060	0.8160	0.2632
	RE	**0.2403**	0.2603	0.9757	0.5330
	SRC	**0.8058**	0.7912	0.2060	0.6404
4000	MSE	**0.0236**	0.0894	0.7874	0.2185
	RE	**0.1607**	0.2464	0.9795	0.4407
	SRC	**0.8486**	0.7988	0.2556	0.7095
8000	MSE	**0.0116**	0.0331	0.7459	0.2006
	RE	**0.1141**	0.1338	0.9932	0.3492
	SRC	**0.8909**	0.8905	0.2887	0.7190
10000	MSE	0.0093	**0.0070**	0.7405	0.1668
	RE	0.1012	**0.0827**	0.9947	0.2707
	SRC	0.8992	**0.9018**	0.2916	0.7321

Table 4: The MSE, RE and averaged SRC of $\boldsymbol{M}$ for $K = 5$, $L = 10$ and $\boldsymbol{\nu} = [\frac{1}{5}\frac{1}{5}\frac{1}{5}\frac{1}{5}1]^\top$

Graph Size (N)	Metric	BeQuec	GeoNMF	CD-MVSI	CD-BNMF
2000	MSE	**0.4302**	0.4895	0.9063	0.8228
	RE	**0.6060**	0.6633	0.8417	0.8156
	SRC	**0.5235**	0.4739	0.1572	0.1832
4000	MSE	**0.3037**	0.3275	0.8975	0.7774
	RE	**0.4420**	0.4906	0.8515	0.8044
	SRC	**0.6013**	0.5678	0.1614	0.2089
8000	**MSE**	**0.1073**	0.4198	0.8969	0.8144
	RE	**0.2631**	0.5661	0.8533	0.8171
	SRC	**0.7304**	0.4948	0.1540	0.1794
10000	**MSE**	**0.0863**	0.5793	0.9013	0.8174
	RE	**0.2442**	0.5724	0.8464	0.8165
	SRC	**0.7535**	0.4082	0.1489	0.1736

k−means (graph)]. Note that in the crowdclustering experiment in Sec. 4.2 (also see Sec. 2.1 for the problem definition), we also use k−means on the original image samples [denoted as k−means (data)], since the raw data samples are available in this particular problem. The k−means (data) method serves as a basic benchmark where no annotator input is used. The source code and the real data used in our experiments are made publicly available (see the footnote in page 2).

4.1 Synthetic Data

We consider N nodes (where $N \in [2000, 10000]$) and $K = 5$ clusters. The membership vectors $\boldsymbol{m}_n$ are drawn from the Dirichlet distribution with parameter $\boldsymbol{\nu} \in \mathbb{R}^5$. The symmetric cluster-cluster interaction matrix $\boldsymbol{B} \in \mathbb{R}^{K\times K}$ is generated by randomly sampling its diagonal and non-diagonal entries from uniform distributions between 0.8 to 1 and between 0 to η, respectively. The diagonal entries are chosen to be relatively large because intra-cluster connections are supposed to be more often. The parameter $\eta \in (0, 1]$ decides the level of interactions across different clusters, which is expected to be lower than that of the intra-cluster interactions.

Metrics. To evaluate the performance of the algorithms, we use a number of standard metrics. First, we

use the subspace distance ('Dist') between between range($\widehat{\boldsymbol{U}}$) and the ground-truth range($\boldsymbol{M}^\top$), i.e.,

$$\text{Dist} = \|\boldsymbol{O}_{\boldsymbol{M}}^{\perp}\boldsymbol{Q}_{\boldsymbol{U}}\|_2,$$

where $\boldsymbol{O}_{\boldsymbol{M}}^{\perp} = \boldsymbol{I} - \boldsymbol{M}^\top(\boldsymbol{M}\boldsymbol{M}^\top)^{-1}\boldsymbol{M}$ and $\boldsymbol{Q}_{\boldsymbol{U}}$ is the orthogonal basis of $\widehat{\boldsymbol{U}}^\top$ [53]. To measure the accuracy of the estimated membership, we define the *mean squared error* (MSE) as follows [54, 55]:

$$\text{MSE} = \min_{\boldsymbol{\Pi}} \frac{1}{K}\sum_{k=1}^{K}\left\| \frac{\boldsymbol{\Pi}\boldsymbol{M}(k,:)}{\|\boldsymbol{\Pi}\boldsymbol{M}(k,:)\|_2} - \frac{\widehat{\boldsymbol{M}}(k,:)}{\|\widehat{\boldsymbol{M}}(k,:)\|_2}\right\|_2^2$$

where $\widehat{\boldsymbol{M}}$ denotes the estimate of $\boldsymbol{M}$ and $\boldsymbol{\Pi} \in \mathbb{R}^{K\times K}$ is a permutation matrix (due to the intrinsic row permutation of the outputs of SSMF algorithms such as `SPA`). As an additional reference, we also present the *relative error* (RE), i.e.,

$$\text{RE} = \min_{\boldsymbol{\Pi}} \frac{\|\boldsymbol{\Pi}\boldsymbol{M} - \widehat{\boldsymbol{M}}\|_{\text{F}}}{\|\boldsymbol{M}\|_{\text{F}}}.$$

We also use the averaged *Spearman's rank correlation coefficient* (SRC). The averaged SRC is often used in the mixed membership learning literature (e.g., [19, 22]). The SRC takes values between -1 and 1 and has a higher value if the ranking of the entries in two vectors are more similar—which is desired.

Results. We first test the identifiability claims under the ideal case (i.e., $\boldsymbol{A} = \boldsymbol{P}$). The blocks of the adjacency matrix with the leftmost query pattern in Fig. 1 is used. We choose the number of groups $L = 10$ (percentage of queried edges = 27.92%), $\eta = 0.1$ and $\boldsymbol{\nu} = [\frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}]^\top$. Table 2 shows the subspace estimation accuracy of our method measured using 'Dist'. The results are averaged from 20 random trials. One can see that the proposed method estimates the subspace of the membership matrix $\boldsymbol{M}$ very accurately in the ideal case, which verifies our subspace identifiability analysis in Theorem 1. The right column of Table 2 presents the 'Dist' measure in the binary observation case. It shows that even under the binary "quantization" of $\boldsymbol{P}$, the proposed method can still recover $\boldsymbol{U}$ to a good accuracy, if N is sufficiently large. This is consistent with Theorem 2.

Next, we consider the rightmost pattern in Fig 1 to evaluate the proposed algorithm and the baselines in the binary observation case. We fix $L = 10$ and $\eta = 0.1$. We test two different cases of $\boldsymbol{\nu}$, i.e., $\boldsymbol{\nu} = [\frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}]^\top$ and $\boldsymbol{\nu} = [\frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ 1]^\top$. The results, which are averaged from 20 random trials, are presented in Tables 3 and 4, respectively. The former case of $\boldsymbol{\nu}$ is easier, since many nodes' membership vectors tend to reside on the boundaries/corners of the probability simplex (cf. Fig. 3)—leading to the existence of many anchor nodes. Hence, every block $\boldsymbol{M}_\ell$ for $\ell \in [L]$ may satisfy the ANC, and `BeQuec` and `GeoNMF` both work reasonably well (cf. discussions in Remark 4). The latter case is more challenging due to $\nu_5 = 1$, which means that there is a cluster that has (much) fewer anchor nodes. This means that many $\boldsymbol{M}_\ell$'s do not satisfy the ANC. Consequently, `BeQuec` outperforms the baselines, since our method does not need ANC to hold on each block $\boldsymbol{M}_\ell$. Particularly, in the latter case, the best baseline `GeoNMF` outputs high MSE = 0.579 even when $N = 10,000$, while `BeQuec`'s performance is considerably better under all three evaluation metrics (with MSE = 0.086).

More synthetic data experiments with different values of η and $\boldsymbol{\nu}$ can be found in Sec. N of the supplementary material.

4.2 Crowdclustering using AMT Annotations

We consider the task of clustering a set of images of dogs into different breeds with the assistance of self-registered annotators from a crowdsourcing platform, i.e., the AMT platform.

AMT Data. We upload $N = 900$ images of $K = 5$ different breeds of dogs from the Standford Dogs Dataset [56]. The chosen breeds are Pug (180 images), Ibizan Hound (180 images), Pomeranian (180 images), Norfolk Terrier (172 images) and Newfoundland (188 images). We followed the setup in Sec. 2.1 to design the experiment. Specifically, we showed each annotator a pair of dog images and ask a simple question (query): "Do this pair of dogs belong to the same category?". If the annotator marks "Yes", then we fill

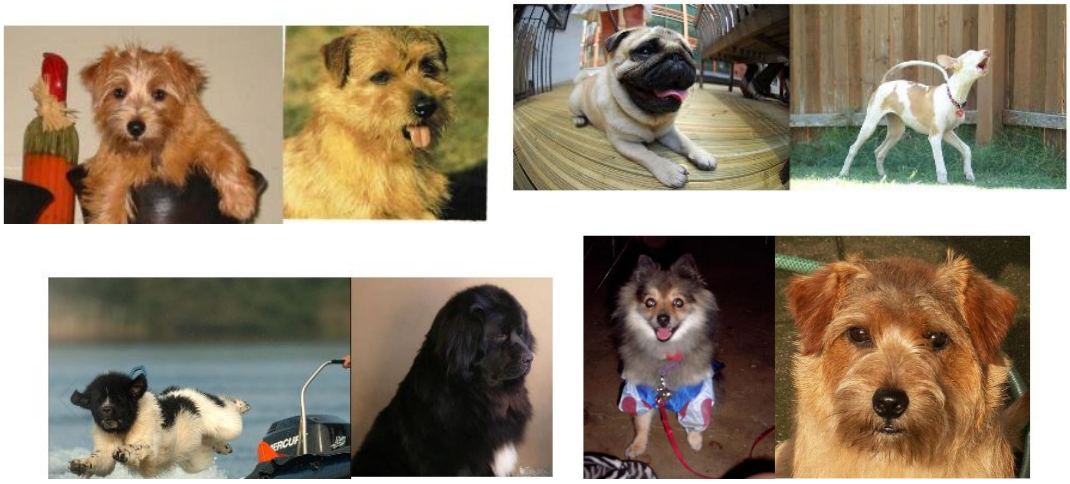

Figure 4: Some examples of the pairs presented to the AMT annotators in the crowdclustering experiment. The image pairs have true similarity labels 1,0,0,1, respectively, in clockwise order. The annotators marked 1,0,1,0, respectively.

the corresponding entry of $\boldsymbol{A}$ with "1", and otherwise "0". In this case, if one hopes to fill all the entries of $\boldsymbol{A}$, there are approximately 4×10^5 distinct pairs of images to be queried, which is costly. We selected image pairs following the block-diagonal pattern (i.e., the rightmost pattern in Fig. 1) by fixing $L = 15$. As a result, only 19.02% of the pairs (76,950) were queried[2]. Fig. 4 shows some pairs of images queried in this experiment. We obtained the responses from 642 AMT annotators. Each annotator answered (non-overlapping) 120 queries on average. The error rate of the answers given by the annotators was 9.90%.

Metric. To quantify the performance, we use standard metrics for evaluating the clustering algorithms. Specifically, since each image is only labeled with a single cluster label, we round the learned membership vectors to the closest unit vectors. This way, standard metrics, i.e., *clustering accuracy* (ACC) and *normalized mutual information* (NMI) can be used for performance characterization (see definitions of ACC and NMI in [57]). ACC ranges from 0% to 100% where 100% means perfect clustering. NMI takes values in between 0 and 1 with 1 being the best performance.

Results. Table 5 summarizes the results output by the algorithms under test, where the first and the second best results are highlighted. One can see that both the proposed method `BeQuec` and `Convex-Opt` output surprisingly high ACC and NMI results. The baselines `GeoNMF` and `SPACL` also yield reasonable ACCs/NMIs. It is somewhat expected both SBM and MMSB based methods could work, since the nodes in this case admit single membership. Nonetheless, `BeQuec` and `Convex-Opt`'s better clustering performance over other baselines may be due to various reasons. For example, the fact that `BeQuec` works under less stringent conditions may make it more resilient to modeling mismatches. The all-at-once optimization strategy of `Convex-Opt` may help avoid error propagation as in greedy methods used in other baselines. In terms of runtime, notably, `BeQuec` runs much faster (approximately 1500 times faster than `Convex-Opt`, which strikes a good balance between accuracy and efficiency for this particular task.

Another interesting observation is that `BeQuec` largely outperforms directly applying clustering methods to the data samples without annotators' assistance [cf. k`-means` (data)]. Note that k`-means` (data) sees all the data and their features (pixels in this case). It does not work well perhaps because the dogs across some breeds are naturally confusing. However, with annotating less than 20% of the data pairs' similarity (without pointing out the exact breeds), the clustering error approaches zero. This experiment may have articulated the value of this simple annotation strategy.

[2] As a result, $81.60 was paid for our annotation task to AMT, but annotating all the pairs of images for this experiment would have cost $429.02.

Table 5: Graph clustering performance on the dog breed dataset with AMT similarity annotations. $N = 900, K = 5, L = 15$. Annotation proportion = 19.02%.

Algorithms	ACC (%)	NMI	Time (s)
BeQuec (proposed)	**99.22**	**0.972**	**0.21**
GeoNMF	96.88	0.929	2.80
SPACL	97.11	0.936	**0.22**
CD-MVSI	38.02	0.199	1.30
CD-BNMF	81.89	0.683	0.28
SC (unnorm.)	22.90	0.032	1.40
SC (norm.)	94.52	0.926	1.05
k-means (graph)	65.73	0.507	1.14
k-means (data)	23.56	0.052	2034.50
Convex-Opt	**99.78**	**0.991**	354.22

Table 6: Graph clustering performance on the dog breed dataset for different annotation error rates. $N = 900, K = 5, L = 15$. Annotation proportion = 19.02%.

Algorithms	annotation error=15%		annotation error=20%	
	ACC(%)	NMI	ACC(%)	NMI
BeQuec (proposed)	**95.83**	**0.941**	**87.20**	**0.740**
GeoNMF	85.33	0.688	79.83	0.629
SPACL	84.10	0.758	**83.71**	**0.757**
CD-MVSI	42.28	0.194	38.57	0.165
CD-BNMF	67.64	0.609	66.89	0.574
SC (unnorm.)	23.50	0.033	23.68	0.036
SC (norm.)	**87.27**	**0.818**	80.27	0.732
k-means (graph)	64.96	0.511	62.48	0.475
Convex-Opt	70.98	0.853	50.57	0.582

In order to better understand the impact of annotation errors on crowdclustering, we also manually "inject" some errors to the annotations. Specifically, we randomly flip a certain percentage of the correct annotations given by the AMT annotators, and observe how this affects the performance of the algorithms.

Table 6 presents the clustering performance obtained from two different error rates. The results are averaged over 20 trials, where in each trial the flipped annotations are randomly chosen. One can observe that the proposed BeQuec method offers the most competitive clustering performance. When the annotation error rate increases from 9.9% (Table 5) to 15% and 20%, the performance degradation of BeQuec is much less obvious compared to the baselines. In particular, one can see that NMI of BeQuec decreases from 0.972 to 0.941 when the annotation error rate is changed from 9.9% to 15%. However, GeoNMF's NMI changes from 0.929 to 0.688, which is a much more drastic drop. This shows BeQuec's robustness to annotation errors.

4.3 Co-author Network Community Detection

We test the edge query-based graph clustering algorithms using the co-authorship network data from MAG [58] and DBLP [59]. In all the datasets, node-node connection (co-authorship) is represented by a binary value ("1" means that the two nodes have co-authored at least one paper and "0" means otherwise). For

Table 7: Details of the co-author network datasets used in community detection experiments

Dataset	N	K
DBLP$_{-1}$	15,075	13
DBLP$_{-2}$	18,106	16
DBLP$_{-3}$	18,646	16
DBLP$_{-4}$	18,195	16
DBLP$_{-5}$	14,982	15
MAG1	37,680	3
MAG2	19,457	3

Table 8: Averaged SRC and runtime in seconds. $L = 10$ (percentage of queried edges = 27.92%).

Datasets	BeQuec (proposed)		GeoNMF		SPACL		CD-MVSI		CD-BNMF	
	SRC	Time(s.)	SRC	Time(s.)	SRC	Time(s.)	SRC	Time(s.)	SRC	Time(s.)
DBLP$_{-1}$	**0.177**	0.47	**0.075**	15.87	0.072	0.73	0.074	0.95	0.058	5.61
DBLP$_{-2}$	**0.166**	0.38	0.089	11.13	**0.103**	1.02	0.055	0.76	0.070	8.20
DBLP$_{-3}$	**0.166**	0.46	0.076	17.18	**0.094**	1.01	0.064	0.98	0.060	8.50
DBLP$_{-4}$	**0.191**	0.48	**0.098**	15.76	0.095	0.95	0.070	0.95	0.079	8.50
DBLP$_{-5}$	**0.195**	0.41	0.084	15.65	**0.156**	0.67	0.073	0.88	0.074	5.85
MAG1	**0.125**	0.26	0.122	1.79	**0.123**	0.97	0.089	0.59	0.069	37.29
MAG2	**0.441**	0.23	0.240	4.66	**0.267**	0.50	0.249	0.53	0.122	9.92

our experiments, we use the version of these networks published by the authors of [19]. Each network is provided with ground-truth mixed membership of the nodes. In these networks, nodes represent the authors of the research papers from different fields of study. Some nodes exhibit multiple memberships, since some authors publish in different fields.

MAG Data. In our experiment, we use a subset of nodes that have a degree of at least 5. Consequently, two co-author networks, namely, MAG1 and MAG2, have 37,680 and 19,457 authors, respectively. In MAG1 and MAG2, all the authors are from 3 different fields of study, i.e., $K = 3$.

DBLP Data. The 5 DBLP datasets (denoted as DBLP1, ..., DBLP5) used in [19] have 6176, 3145, 2605, 3056, and 6269 nodes, respectively. In addition, the number of clusters present in the DBLP datasets are 6, 3, 3, 3, and 4, respectively. To make the clustering problem more challenging, we combine the smaller original DBLP networks into larger-size networks following the setting in [22]. We use the notation DBLP$_{-i}$ to denote the network where all the 5 DBLP networks except the i-th one are combined. This way, different DBLP$_{-i}$'s can be produced for testing the algorithms. In addition, each DBLP$_{-i}$, for $i = 1, \ldots, 5$, contains 13-16 clusters (see Table 7 for details), presenting more challenging scenarios.

Metric. The methods are evaluated using the averaged SRC as in the literature [19, 22]. We let all the algorithms to access only part of the network under the block-diagonal query pattern in Fig. 1. We randomize the node order in each of the 20 trials and present the averaged performance.

Results. We first test the mixed membership learning performance by setting $L = 10$ for all the MAG and DBLP datasets under test. The percentage of edges queried in this setting is 27.92%. From Table 8, one can see that the proposed BeQuec algorithm consistently outperforms the baselines in all the graphs under test. In addition, BeQuec's runtime performance is also the most competitive.

Table 9 uses MAG2 dataset to show the performance of BeQuec under various L's. We also apply the clustering algorithms used in crowdclustering to benchmark BeQuec's performance. The ACC metric is used in this table since the clustering algorithms do not output mixed membership. One can see that the performance of all algorithms decreases along with L's increase—which is consistent with our analysis

Table 9: Clustering accuracy (%) of MAG2 under various L (percentage of queried edges (%)). $N = 19457$, $K = 3$.

Alorithms	$L = 10$ (27.92%)	$L = 25$ (11.58%)	$L = 50$ (5.82%)	$L = 75$ (3.86%)	$L = 100$ (2.87%)
`BeQuec` (proposed)	**78.70**	**77.19**	**67.81**	**61.85**	**56.98**
`GeoNMF`	58.16	57.87	56.88	52.68	52.33
`SPACL`	57.49	58.17	55.25	54.22	53.07
`CD-MVSI`	53.45	21.82	14.57	13.53	11.71
`CD-BNMF`	51.19	49.35	49.37	48.99	48.97
`SC (unnorm.)`	56.97	56.49	56.38	55.88	55.51
`SC (norm.)`	**64.24**	**62.15**	**57.15**	**56.45**	**55.80**
k`-means`	54.31	53.47	53.87	53.15	51.25

in Theorem 2. For each L used, the corresponding queried percentage of edges is listed in the first row. Notably, when $L = 50$, i.e, only 5.82% of $\boldsymbol{A}$ is observed, the proposed method still outputs a reasonable clustering accuracy. This demonstrates a promising balance between the query sample complexity and the graph clustering accuracy.

5 Conclusion

In this work, we proposed a graph query scheme that enables provable graph clustering with partially observed edges. Unlike previous works relying on random graph edge query and computationally heavy convex programming, our method features a lightweight algorithm and works with systematic edge query patterns that are arguably more realistic in some applications. Our method also learns mixed membership of nodes from overlapping clusters, which improves upon existing provable graph query-based methods that work with single membership and disjoint clusters. We tested the proposed `BeQuec` algorithm on two real-world network data analytics problems, namely, crowdclustering and network community detection. `BeQuec` offers promising results in both experiments.

A potential future direction is to reduce the number of queries by allowing $N/L < K$. This may be achieved by using third-order graph statistics from the sampled blocks and a tensor decomposition framework. Using tensor-based approaches may likely present a challenging optimization problem without polynomial-time algorithms (as opposed to the matrix based framework as in this work). Studying its query complexity and computational cost trade-off presents an interesting research topic.

References

[1] S. E. Schaeffer, "Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27 – 64, 2007.

[2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing, "Mixed membership stochastic blockmodels," *Journal of Machine Learning Research*, vol. 9, no. Sep, pp. 1981–2014, 2008.

[3] D. Kuang, C. Ding, and H. Park, "Symmetric nonnegative matrix factorization for graph clustering," *Proceedings of SIAM International Conference on Data Mining*, pp. 106–117, 2012.

[4] S. Van Dongen, "Graph clustering via a discrete uncoupling process," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 1, pp. 121–141, 2008.

[5] A. Saade, M. Lelarge, F. Krzakala, and L. Zdeborová, "Clustering from sparse pairwise measurements," *IEEE International Symposium on Information Theory*, pp. 780–784, 2016.

[6] M. T. Schaub, S. Segarra, and J. N. Tsitsiklis, "Blind identification of stochastic block models from dynamical observations," *SIAM Journal on Mathematics of Data Science*, vol. 2, no. 2, pp. 335–367, 2020.

[7] J. Leskovec and C. Faloutsos, "Sampling from large graphs," *Proceedings of ACM International Conference on Knowledge Discovery and Data Mining*, p. 631–636, 2006.

[8] P. Hu and W. C. Lau, "A survey and taxonomy of graph sampling," *Computing Research Repository*, 2013.

[9] M. J. Salganik and D. D. Heckathorn, "Sampling and estimation in hidden populations using respondent-driven sampling," *Sociological Methodology*, vol. 34, no. 1, pp. 193–240, 2004.

[10] M. Malekinejad, L. Johnston, C. Kendall, L. R. Kerr, M. Rifkin, and G. Rutherford, "Using respondent-driven sampling methodology for HIV biological and behavioral surveillance in international settings: A systematic review," *AIDS and Behavior*, vol. 12, pp. 5–30, 07 2008.

[11] K. A. Gill and M. O'Neal, "Survey of Soybean Insect Pollinators: Community Identification and Sampling Method Analysis," *Environmental Entomology*, vol. 44, no. 3, pp. 488–498, 02 2015.

[12] R. Korlakai Vinayak, S. Oymak, and B. Hassibi, "Graph clustering with missing data: Convex algorithms and analysis," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2996–3004, 2014.

[13] R. Korlakai Vinayak and B. Hassibi, "Crowdsourced clustering: Querying edges vs triangles," *Advances in Neural Information Processing Systems*, vol. 29, pp. 1316–1324, 2016.

[14] Y. Chen, A. Jalali, S. Sanghavi, and H. Xu, "Clustering partially observed graphs via convex optimization," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2213–2238, 2014.

[15] A. Mazumdar and B. Saha, "Clustering with noisy queries," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, p. 5790–5801.

[16] ——, "Clustering via crowdsourcing," *arXiv preprint arXiv:1604.01839*, 2016.

[17] R. K. Vinayak, "Graph clustering: Algorithms, analysis and query design," Ph.D. dissertation, California Institute of Technology, 2018.

[18] C. Särndal, B. Swensson, and J. Wretman, *Model Assisted Survey Sampling*, ser. Springer Series in Statistics. Springer-Verlag, 2003.

[19] X. Mao, P. Sarkar, and D. Chakrabarti, "On mixed memberships and symmetric nonnegative matrix factorizations," in *International Conference on Machine Learning*, 2017, pp. 2324–2333.

[20] ——, "Estimating mixed memberships with sharp eigenvector deviations," *Journal of the American Statistical Association*, pp. 1–13, 2020.

[21] M. Panov, K. Slavnov, and R. Ushakov, "Consistent estimation of mixed memberships with successive projections," *Complex Networks*, p. 53–64, 2017.

[22] K. Huang and X. Fu, "Detecting overlapping and correlated communities without pure nodes: Identifiability and algorithm," *Proceedings of International Conference on Machine Learning*, vol. 97, pp. 2859–2868, 2019.

[23] J. Yi, R. Jin, S. Jain, T. Yang, and A. K. Jain, "Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1772–1780, 2012.

[24] S. Ibrahim and X. Fu, "Learning mixed membership from adjacency graph via systematic edge query: Identifiability and algorithm," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 5370–5374.

[25] R. G. Gomes, P. Welinder, A. Krause, and P. Perona, "Crowdclustering," *Advances in Neural Information Processing Systems*, vol. 24, pp. 558–566, 2011.

[26] J. Yi, R. Jin, A. Jain, and S. Jain, "Crowdclustering with sparse pairwise labels: A matrix completion approach," in *AAAI Workshop on Human Computation*, 2012.

[27] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Computing Surveys*, vol. 45, no. 4, Aug. 2013.

[28] I. Psorakis, S. Roberts, M. Ebden, and B. Sheldon, "Overlapping community detection using Bayesian non-negative matrix factorization," *Physical Review E*, vol. 83, p. 066114, 06 2011.

[29] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing Systems*, vol. 14, pp. 849–856, 2002.

[30] S. Oymak and B. Hassibi, "Finding dense clusters via "low rank + sparse" decomposition," *Computing Research Repository*, 04 2011.

[31] A. Anandkumar, R. Ge, D. Hsu, and S. M. Kakade, "A tensor approach to learning mixed membership community models," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2239–2312, 2014a.

[32] F. J.Király, L. Theran, and R. Tomioka, "The algebraic combinatorial approach for low-rank matrix completion," *Journal of Machine Learning Research*, vol. 16, no. 41, pp. 1391–1436, 2015.

[33] D. L. Pimentel-Alarcón, N. Boston, and R. D. Nowak, "A characterization of deterministic sampling patterns for low-rank matrix completion," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 623–636, 2016.

[34] X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, "Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 36, no. 2, pp. 59–80, 2019.

[35] N. Gillis and S. Vavasis, "Fast and robust recursive algorithms for separable nonnegative matrix factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 698–714, April 2014.

[36] T.-H. Chan, C.-Y. Chi, Y.-M. Huang, and W.-K. Ma, "A convex analysis-based minimum-volume enclosing simplex algorithm for hyperspectral unmixing," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4418 –4432, Nov. 2009.

[37] X. Fu, K. Huang, and N. D. Sidiropoulos, "On identifiability of nonnegative matrix factorization," *IEEE Signal Process. Lett.*, vol. 25, no. 3, pp. 328–332, 2018.

[38] K. Huang, N. Sidiropoulos, and A. Swami, "Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition," *IEEE Trans. Signal Process.*, vol. 62, no. 1, pp. 211–224, 2014.

[39] D. Donoho and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts?" in *Advances in Neural Information Processing Systems*, vol. 16, 2003.

[40] S. Ibrahim, X. Fu, N. Kargas, and K. Huang, "Crowdsourcing via pairwise co-occurrences: Identifiability and algorithms," *Advances in Neural Information Processing Systems*, vol. 32, pp. 7847–7857, 2019.

[41] A. Kumar, V. Sindhwani, and P. Kambadur, "Fast conical hull algorithms for near-separable nonnegative matrix factorization," *Proceedings of International Conference on Machine Learning*, vol. 28, pp. 231–239, 2012.

[42] T. Mizutani, "Ellipsoidal rounding for nonnegative matrix factorization under noisy separability," *Journal of Machine Learning Research*, vol. 15, no. 29, pp. 1011–1039, 2014.

[43] X. Fu and W.-K. Ma, "Robustness analysis of structured matrix factorization via self-dictionary mixed-norm optimization," *IEEE Signal Process. Lett.*, vol. 23, no. 1, pp. 60–64, 2016.

[44] N. Gillis, "Successive nonnegative projection algorithm for robust nonnegative blind source separation," *SIAM Journal on Imaging Sciences*, vol. 7, no. 2, pp. 1420–1450, 2014.

[45] X. Fu, W.-K. Ma, T.-H. Chan, and J. M. Bioucas-Dias, "Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 6, pp. 1128–1141, 2015.

[46] X. Fu, K. Huang, B. Yang, W.-K. Ma, and N. D. Sidiropoulos, "Robust volume minimization-based matrix factorization for remote sensing and document clustering," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6254–6268, 2016.

[47] N. D. Sidiropoulos, E. E. Papalexakis, and C. Faloutsos, "Parallel randomly compressed cubes: A scalable distributed architecture for big tensor decomposition," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 57–70, 2014.

[48] N. D. Sidiropoulos and A. Kyrillidis, "Multi-way compressed sensing for sparse low-rank tensors," *IEEE Signal Process. Lett*, vol. 19, no. 11, pp. 757–760, 2012.

[49] U. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, p. 395–416, Dec. 2007.

[50] J. Scott, "Social network analysis," *Sociology*, vol. 22, no. 1, pp. 109–127, 1988.

[51] J. Lei and A. Rinaldo, "Consistency of spectral clustering in stochastic block models," *Annals of Statistics*, vol. 43, no. 1, pp. 215–237, 02 2015.

[52] Jianbo Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[53] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.

[54] X. Fu, W.-K. Ma, K. Huang, and N. D. Sidiropoulos, "Blind separation of quasi-stationary sources: Exploiting convex geometry in covariance domain," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2306–2320, 2015.

[55] X. Fu, S. Ibrahim, H.-T. Wai, C. Gao, and K. Huang, "Block-randomized stochastic proximal gradient for low-rank tensor factorization," *IEEE Trans. Signal Process.*, vol. 68, pp. 2170–2185, 2020.

[56] A. Khosla, N. Jayadevaprakash, B. Yao, and L. Fei-Fei, "Novel dataset for fine-grained image categorization," in *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

[57] C. Lu, J. Tang, M. Lin, L. Lin, S. Yan, and Z. Lin, "Correntropy induced l2 graph for robust subspace clustering," *IEEE International Conference on Computer Vision*, pp. 1801–1808, 2013.

[58] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B. Hsu, and K. Wang, "An overview of microsoft academic service (MAS) and applications," *Proceedings of International Conference on World Wide Web*, p. 243–246, 2015.

[59] M. Ley, "The DBLP computer science bibliography: Evolution, research issues, perspectives," *Proceedings of International Symposium on String Processing and Information Retrieval*, pp. 1–10, 2002.

[60] A. B. Frigyik, A. Kapila, and M. Gupta, "Introduction to the dirichlet distribution and related processes," *Dept. Elect. Eng., Univ. Washington, Seattle, WA, USA*, 2010.

[61] A. Gouda and T. Szántai, "On numerical calculation of probabilities according to dirichlet distribution," *Annals of Operations Research*, vol. 177, pp. 185–200, 2010.

[62] M. J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge University Press, 2019, vol. 48.

[63] Y. Yu, T. Wang, and R. J. Samworth, "A useful variant of the Davis–Kahan theorem for statisticians," *Biometrika*, vol. 102, no. 2, pp. 315–323, 04 2014.

[64] P. Wedin, "Perturbation theory for pseudo-inverses," *BIT Numerical Mathematics*, vol. 13, pp. 217–232, 1973.

[65] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. USA: Cambridge University Press, 2012.

[66] U. Araújo, B. Saldanha, R. Galvão, T. Yoneyama, H. Chame, and V. Visani, "The successive projections algorithm for variable selection in spectroscopic multicomponent analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 57, no. 2, pp. 65–73, 2001.

[67] R. Jonker and T. Volgenant, "Improving the hungarian assignment algorithm," *Operations Research Letters*, vol. 5, no. 4, pp. 171–175, 1986.

A Proof of Theorem 1

Algorithm 1 aims to output $\widehat{\boldsymbol{U}}$ such that $\mathsf{range}(\widehat{\boldsymbol{U}}) = \mathsf{range}([\boldsymbol{M}_1, \ldots, \boldsymbol{M}_L]^\top)$. It implies that $\widehat{\boldsymbol{U}} = [\boldsymbol{U}_1^\top, \ldots, \boldsymbol{U}_L^\top]^\top = [\boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_L]^\top \boldsymbol{G} = \boldsymbol{M}^\top \boldsymbol{G}$, for a certain nonsingular $\boldsymbol{G} \in \mathbb{R}^{K\times K}$. To show this, we start by considering the below four blocks for any $r \in [L-2]$:

$$\boldsymbol{P}_{\ell_r, m_r} = \boldsymbol{M}_{\ell_r}^\top \boldsymbol{B} \boldsymbol{M}_{m_r}, \tag{13a}$$

$$\boldsymbol{P}_{\ell_{r+1}, m_r} = \boldsymbol{M}_{\ell_{r+1}}^\top \boldsymbol{B} \boldsymbol{M}_{m_r}, \tag{13b}$$

$$\boldsymbol{P}_{\ell_{r+1}, m_{r+1}} = \boldsymbol{M}_{\ell_{r+1}}^\top \boldsymbol{B} \boldsymbol{M}_{m_{r+1}}, \tag{13c}$$

$$\boldsymbol{P}_{\ell_{r+2}, m_{r+1}} = \boldsymbol{M}_{\ell_{r+2}}^\top \boldsymbol{B} \boldsymbol{M}_{m_{r+1}}. \tag{13d}$$

Algorithm 1 defines the following matrices in rth and $(r+1)$th iterations, respectively (recall that we have assumed $\boldsymbol{A}_{\ell,m} = \boldsymbol{P}_{\ell,m} = \boldsymbol{M}_\ell^\top \boldsymbol{B} \boldsymbol{M}_m$ for any $\ell, m \in [L]$ in the ideal case):

$$\boldsymbol{C}_r := [\boldsymbol{P}_{\ell_r, m_r}^\top, \boldsymbol{P}_{\ell_{r+1}, m_r}^\top]^\top, \ \boldsymbol{C}_{r+1} := [\boldsymbol{P}_{\ell_{r+1}, m_{r+1}}^\top \boldsymbol{P}_{\ell_{r+2}, m_{r+1}}^\top]^\top.$$

The top-K SVDs of $\boldsymbol{C}_r$ and $\boldsymbol{C}_{r+1}$ can be represented as follows:

$$\boldsymbol{C}_r = [\boldsymbol{U}_{\ell_r}^\top, \boldsymbol{U}_{\ell_{r+1}}^\top]^\top \boldsymbol{\Sigma}_r \boldsymbol{V}_{m_r}^\top, \tag{14a}$$

$$\boldsymbol{C}_{r+1} = [\overline{\boldsymbol{U}}_{\ell_{r+1}}^\top, \overline{\boldsymbol{U}}_{\ell_{r+2}}^\top]^\top \boldsymbol{\Sigma}_{r+1} \boldsymbol{V}_{m_{r+1}}^\top. \tag{14b}$$

Combining (13) and (14), and using the assumption that $\mathrm{rank}(\boldsymbol{M}_\ell) = \mathrm{rank}(\boldsymbol{B}) = K$ and $K \le |\mathcal{S}_\ell|$ for every ℓ, the following equations hold:

$$\boldsymbol{U}_{\ell_r} = \boldsymbol{M}_{\ell_r}^\top \boldsymbol{G}_r, \quad \boldsymbol{U}_{\ell_{r+1}} = \boldsymbol{M}_{\ell_{r+1}}^\top \boldsymbol{G}_r, \tag{15a}$$

$$\overline{\boldsymbol{U}}_{\ell_{r+1}} = \boldsymbol{M}_{\ell_{r+1}}^\top \overline{\boldsymbol{G}}_{r+1}, \quad \overline{\boldsymbol{U}}_{\ell_{r+2}} = \boldsymbol{M}_{\ell_{r+2}}^\top \overline{\boldsymbol{G}}_{r+1}, \tag{15b}$$

where $\boldsymbol{G}_r \in \mathbb{R}^{K\times K}$ and $\overline{\boldsymbol{G}}_{r+1} \in \mathbb{R}^{K\times K}$ are certain nonsingular matrices. The equations in (15) imply that $\boldsymbol{U}_{\ell_r}$, $\boldsymbol{U}_{\ell_{r+1}}$ and $\overline{\boldsymbol{U}}_{\ell_{r+2}}$ are the bases of $\mathsf{range}(\boldsymbol{M}_{\ell_r}^\top)$, $\mathsf{range}(\boldsymbol{M}_{\ell_{r+1}}^\top)$ and $\mathsf{range}(\boldsymbol{M}_{\ell_{r+2}}^\top)$, respectively. Since $\boldsymbol{G}_r = \overline{\boldsymbol{G}}_{r+1}$ does not generally hold, the basis $\overline{\boldsymbol{U}}_{\ell_{r+2}}$ cannot be directly combined with the bases $\boldsymbol{U}_{\ell_r}$ and

$\boldsymbol{U}_{\ell_{r+1}}$ to obtain range$([\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}, \boldsymbol{M}_{\ell_{r+2}}]^\top)$. Therefore, we are interested in obtaining the basis defined as below:

$$\boldsymbol{U}_{\ell_{r+2}} := \boldsymbol{M}_{\ell_{r+2}}^\top \boldsymbol{G}_r. \tag{16}$$

In order to identify $\boldsymbol{U}_{\ell_{r+2}}$ as defined in (16), we use the second equality in (15a) and the first equality in (15b) to construct the following:

$$\overline{\boldsymbol{U}}_{\ell_{r+1}}^\dagger \boldsymbol{U}_{\ell_{r+1}} = \overline{\boldsymbol{G}}_{r+1}^{-1} \boldsymbol{G}_r. \tag{17}$$

Combining the second equality in (15b) with (17), we have

$$\overline{\boldsymbol{U}}_{\ell_{r+2}} \overline{\boldsymbol{U}}_{\ell_{r+1}}^\dagger \boldsymbol{U}_{\ell_{r+1}} = \left(\boldsymbol{M}_{\ell_{r+2}}^\top \overline{\boldsymbol{G}}_{r+1}\right) \left(\overline{\boldsymbol{G}}_{r+1}^{-1} \boldsymbol{G}_r\right) = \boldsymbol{M}_{\ell_{r+2}}^\top \boldsymbol{G}_r.$$

Hence, the following holds:

$$\boldsymbol{U}_{\ell_{r+2}} = \overline{\boldsymbol{U}}_{\ell_{r+2}} \overline{\boldsymbol{U}}_{\ell_{r+1}}^\dagger \boldsymbol{U}_{\ell_{r+1}} = \boldsymbol{M}_{\ell_{r+2}}^\top \boldsymbol{G}_r, \tag{18}$$

where $\overline{\boldsymbol{U}}_{\ell_{r+2}}$ and $\overline{\boldsymbol{U}}_{\ell_{r+1}}$ are obtained from the top-K SVD of $\boldsymbol{C}_{r+1}$ and $\boldsymbol{U}_{\ell_{r+1}}$ is obtained from the top-K SVD of $\boldsymbol{C}_r$.

Similarly, $\boldsymbol{U}_{\ell_{r-1}}$ can be identified using the top-K SVDs of $\boldsymbol{C}_r$ and $\boldsymbol{C}_{r-1}$ by the following:

$$\boldsymbol{U}_{\ell_{r-1}} = \overline{\boldsymbol{U}}_{\ell_{r-1}} \overline{\boldsymbol{U}}_{\ell_r}^\dagger \boldsymbol{U}_{\ell_r} = \boldsymbol{M}_{\ell_{r-1}}^\top \boldsymbol{G}_r, \tag{19}$$

where $\boldsymbol{U}_{\ell_r}$ is obtained from the top-K SVD of $\boldsymbol{C}_r$ and $\overline{\boldsymbol{U}}_{\ell_{r-1}}$ and $\overline{\boldsymbol{U}}_{\ell_r}$ are obtained from the top-K SVD of $\boldsymbol{C}_{r-1} := [\boldsymbol{P}_{\ell_{r-1},m_{r-1}}^\top, \boldsymbol{P}_{\ell_r,m_{r-1}}^\top]^\top$.

Algorithm 1 first applies top-K SVD for $\boldsymbol{C}_T$ where $T = \lfloor L/2 \rfloor$ and the bases $\boldsymbol{U}_{\ell_T}$ and $\boldsymbol{U}_{\ell_{T+1}}$ are identified (lines 2-3). By letting $\boldsymbol{G}_T = \boldsymbol{G}$, we thus obtain

$$\boldsymbol{U}_{\ell_T} = \boldsymbol{M}_{\ell_T}^\top \boldsymbol{G} \quad \text{and} \quad \boldsymbol{U}_{\ell_{T+1}} = \boldsymbol{M}_{\ell_{T+1}}^\top \boldsymbol{G}. \tag{20}$$

Next, we use induction to show how the iterations in Algorithm 1 (lines 5-8) identify $\boldsymbol{U}_{\ell_{T+2}}, \ldots, \boldsymbol{U}_L$. The induction hypothesis is that before any iteration at $r = T+1, T+2, \ldots, L-1$, the basis $\boldsymbol{U}_{\ell_r}$ is identified from the previous iteration such that $\boldsymbol{U}_{\ell_r} = \boldsymbol{M}_{\ell_r}^\top \boldsymbol{G}$. Note that the induction hypothesis holds true for $r = T+1$ from lines 2-3 of Algorithm 1 by performing top-K SVD of $\boldsymbol{C}_T$. From (18), we have seen that given $\boldsymbol{U}_{\ell_r}$, one can identify $\boldsymbol{U}_{\ell_{r+1}}$ using the matrices resulted from the top-K SVD of $\boldsymbol{C}_r$. Therefore, by induction, we can establish that Algorithm 1 identifies

$$\boldsymbol{U}_{\ell_r} = \boldsymbol{M}_{\ell_r}^\top \boldsymbol{G}, \text{ for every } r \in \{T+2, \ldots, L\}. \tag{21}$$

Similarly, using induction via employing the results in (19) and (20), we can establish that lines 9-12 of Algorithm 1 identifies

$$\boldsymbol{U}_{\ell_r} = \boldsymbol{M}_{\ell_r}^\top \boldsymbol{G}, \text{ for every } r \in \{1, \ldots, T-1\}. \tag{22}$$

Combining (20), (21), and (22), one can see that Algorithm 1 identifies $\widehat{\boldsymbol{U}} = [\boldsymbol{U}_1^\top, \boldsymbol{U}_2^\top, \ \ldots \ , \boldsymbol{U}_L^\top]^\top$ such that range$(\widehat{\boldsymbol{U}})$ = range$([\boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_L]^\top)$.

B Proof of Proposition 1

B.1 Preliminaries of Dirichlet Distribution

Let $\mathcal{P}^K = \{\boldsymbol{z} \in \mathbb{R}^K | \boldsymbol{z}^\top \boldsymbol{1} = 1, \boldsymbol{z} \geq 0\}$ denote the $(K-1)$-dimensional probability simplex. The Dirichlet distribution with parameter $\boldsymbol{\nu} = [\nu_1, \ldots, \nu_K]^\top$ has the following probability density function (PDF):

$$f(\boldsymbol{z}_{-k}; \boldsymbol{\nu}) = \frac{\Gamma(\sum_{i=1}^K \nu_i)}{\prod_{i=1}^K \Gamma(\nu_i)} (1 - \sum_{i \neq k} z_i)^{\nu_k - 1} \prod_{i \neq k} z_i^{\nu_i - 1}, \tag{23}$$

for any $k \in [K]$ where $\boldsymbol{z}_{-k} = [z_1, \ldots, z_{k-1}, z_{k+1}, \ldots, z_K]^\top$ and $[z_1, \ldots, z_K]^\top \in \mathcal{P}^K$ [60]. The choice of k in (23) is arbitrary since $\sum_{i \neq k} z_i = 1 - z_k$ holds for any k. The cumulative distribution function (CDF) of the Dirichlet distribution is given by

$$\begin{aligned} F(\boldsymbol{z}_{-k}; \boldsymbol{\nu}) &= \int_0^{z_1} \cdots \int_0^{z_{k-1}} \int_0^{z_{k+1}} \cdots \int_0^{z_K} f(\boldsymbol{z}_{-k}; \boldsymbol{\nu}) d\boldsymbol{z}_{-k} \\ &= \frac{\Gamma(\sum_{i=1}^K \nu_i)}{\prod_{i=1}^K \Gamma(\nu_i)} L_k(\boldsymbol{m}, \boldsymbol{\nu}) \prod_{i \neq k} \frac{z_i^{\nu_i}}{\nu_i}, \end{aligned} \tag{24}$$

where the function $L_k(\boldsymbol{m}, \boldsymbol{\nu})$ is known as the Lauricella function of the first kind and $L_k(\boldsymbol{m}, \boldsymbol{\nu}) > 1$ for any k [61]. The CDF $F(\boldsymbol{z}_{-k}; \boldsymbol{\nu})$ measures the cumulative probability starting from the k-th vertex of the probability simplex till each coordinate $i \neq k$ reaches z_i.

B.2 Proof of the Proposition

According to Assumption 1, $\boldsymbol{m}_n \in \mathcal{P}^K$ holds for all $n \in [N]$ and the vectors $\boldsymbol{m}_n$ are sampled randomly from a Dirichlet distribution with parameter $\boldsymbol{\nu}$. Using Assumption 1, we aim to analyze the conditions under which the matrix $\boldsymbol{M} \in [\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N]$ satisfies ε-separability (see Definition 1), i.e., there exists $\boldsymbol{\Lambda} = \{q_1, \ldots, q_K\}$ such that for $k = 1, \ldots, K$,

$$\|\boldsymbol{m}_{q_k} - \boldsymbol{e}_k\|_2 \leq \varepsilon, \ \boldsymbol{m}_{q_k} \in \mathcal{P}^K. \tag{25}$$

The criterion in (25) means that there exist $\boldsymbol{m}_{q_k} \in \mathcal{P}_k(\varepsilon)$ for $k = 1, \ldots, K$ where

$$\mathcal{P}_k(\varepsilon) = \{\boldsymbol{z} \in \mathcal{P}^K | \|\boldsymbol{z} - \boldsymbol{e}_k\|_2 \leq \varepsilon\}. \tag{26}$$

Next, we proceed to check under what conditions there exists at least one $\boldsymbol{m}_n$ among $\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N$ such that $\boldsymbol{m}_n \in \mathcal{P}_k(\varepsilon)$. For this, let us define two random variables as below:

$$E_{n,k} = \begin{cases} 1, & \text{if } \boldsymbol{m}_n \in \mathcal{P}_k(\varepsilon) \\ 0, & \text{otherwise.} \end{cases} \tag{27}$$

$$H_k = \sum_{n=1}^N E_{n,k}. \tag{28}$$

In order to characterize the probability that there exists at least one $\boldsymbol{m}_n$ among $\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N$ such that $\boldsymbol{m}_n \in \mathcal{P}_k(\varepsilon)$, we need to characterize $\mathsf{Pr}\,(H_k > 0)$. To achieve this, we invoke Chernoff-Hoeffding bound [62]:

Theorem 3 *Let $E_{1,k}, \ldots, E_{N,k}$ be independent bounded random variables such that $E_{n,k} \in [a_{n,k}, b_{n,k}]$. Then for any $t > 0$,*

$$\mathsf{Pr}\left(\sum_{n=1}^N E_{n,k} - \sum_{n=1}^N \mathbb{E}[E_{n,k}] \leq -t\right) \leq e^{\frac{-2t^2}{\sum_{n=1}^N (b_{n,k} - a_{n,k})^2}}.$$

By letting $t = \sum_{n=1}^N \mathbb{E}[E_{n,k}]$ and using the fact that $E_{n,k} \in [0, 1]$, we get the below from Theorem 3:

$$\mathsf{Pr}\,(H_k > 0) \geq 1 - e^{-\frac{2\left(\sum_{n=1}^N \mathbb{E}[E_{n,k}]\right)^2}{N}}. \tag{29}$$

Next, we need to characterize the term $\mathbb{E}[E_{n,k}]$ which appears in the R.H.S. of (29). From the definition of the random variable $E_{n,k}$ in (27), we can see that

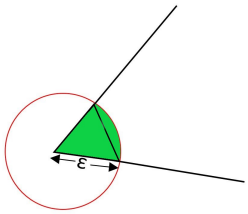

Figure 5: The figure shows any vertex $k \in \{1,2,3\}$ of the probability simplex $\mathcal{P}^3$. The green shaded region represents $\mathcal{P}_k(\varepsilon)$ (see (26)) which is the intersection of the probability simplex $\mathcal{P}^3$ and a Euclidean ball of radius ε with its center at $\boldsymbol{e}_k$. The small triangle near the vertex is 2-dimensional probability simplex with edge lengths ε; i.e, it intersects the co-ordinate axes at $\frac{\varepsilon}{\sqrt{2}}\boldsymbol{e}_k$. It can be observed that the volume of the green shaded region is larger than the volume of the small triangle.

$$\begin{aligned}
\mathbb{E}[E_{n,k}] = \mathsf{Pr}(\boldsymbol{m}_n \in \mathcal{P}_k(\varepsilon)) &= \int_{\boldsymbol{z}\in\mathcal{P}_k(\varepsilon)} f(\boldsymbol{z}_{-k};\boldsymbol{\nu})d\boldsymbol{z}_{-k}, \\
&\geq \int_{z_1=0}^{\frac{\varepsilon}{\sqrt{2}}} \cdots \int_{z_{k-1}=0}^{\frac{\varepsilon}{\sqrt{2}}} \int_{z_{k+1}=0}^{\frac{\varepsilon}{\sqrt{2}}} \cdots \int_{z_K=0}^{\frac{\varepsilon}{\sqrt{2}}} f(\boldsymbol{z}_{-k};\boldsymbol{\nu})d\boldsymbol{z}_{-k} \\
&= F(\boldsymbol{z}^*_{-k};\boldsymbol{\nu}) \geq \frac{\Gamma(\sum_{i=1}^K \nu_i)}{\prod_{i=1}^K \Gamma(\nu_i)} \prod_{i\neq k} \frac{\left(\frac{\varepsilon}{\sqrt{2}}\right)^{\nu_i}}{\nu_i} \\
&= \frac{\Gamma(\sum_{i=1}^K \nu_i)}{\prod_{i=1}^K \Gamma(\nu_i)} \frac{(\varepsilon/\sqrt{2})^{\sum_{i\neq k}\nu_i}}{\prod_{i\neq k}\nu_i} := G_k(\varepsilon,\boldsymbol{\nu}),
\end{aligned} \tag{30}$$

where $\boldsymbol{z}^*_{-k} = [z^*_1,\ldots,z^*_{k-1},z^*_{k+1},z^*_K]^\top = [\frac{\varepsilon}{\sqrt{2}},\ldots,\frac{\varepsilon}{\sqrt{2}}]^\top$. The first inequality is obtained by generalizing the geometric property as shown in Fig. 5 for any $K-1$ dimensional probability simplex and the second inequality is obtained by combining (24) and the property that $L_k(\boldsymbol{m},\boldsymbol{\alpha}) > 1$ for any k [61].

Applying (30) in (29), we get the probability such that there exists at least one $\boldsymbol{m}_n$ such that $\boldsymbol{m}_n \in \mathcal{P}_k(\varepsilon)$ as below:

$$\mathsf{Pr}\left(H_k > 0\right) \geq 1 - e^{-2N(G_k(\varepsilon,\boldsymbol{\nu}))^2}.$$

To characterize the corresponding probability for all the K vertices, we can apply the union bound to obtain the below

$$\begin{aligned}
\mathsf{Pr}\left(\bigcap_{k=1}^K (H_k > 0)\right) &\geq 1 - \sum_{k=1}^K e^{-2N(G_k(\varepsilon,\boldsymbol{\nu}))^2} &(31)\\
&\geq 1 - Ke^{-2N(G(\varepsilon,\boldsymbol{\nu}))^2}, &(32)
\end{aligned}$$

where $G(\varepsilon,\boldsymbol{\nu}) = \min_k G_k(\varepsilon,\boldsymbol{\nu})$.

Eq. (31) implies that with probability greater than or equal to $1 - Ke^{-2N(G(\varepsilon,\boldsymbol{\nu}))^2}$, the matrix $\boldsymbol{M} = [\boldsymbol{m}_1,\ldots,\boldsymbol{m}_N]$ satisfies ε-separability. By letting $\mu = Ke^{-2N(G(\varepsilon,\boldsymbol{\nu}))^2}$, one can see that if $N \geq \log\left(\frac{K}{\mu}\right)/(2G(\varepsilon,\boldsymbol{\nu}))^2$, the matrix $\boldsymbol{M} = [\boldsymbol{m}_1,\ldots,\boldsymbol{m}_N]$ satisfies ε-separability with probbaility greater than $1-\mu$ where

$$G(\varepsilon,\boldsymbol{\nu}) = \frac{\Gamma(\sum_{i=1}^K \nu_i)}{\prod_{i=1}^K \Gamma(\nu_i)} \min_k \frac{(\varepsilon/\sqrt{2})^{\sum_{i\neq k}\nu_i}}{\prod_{i\neq k}\nu_i}.$$

C Proof of Theorem 2

The analysis for Theorem 2 consists of three major steps. We provide an overview to the proof as follows:
(A) Range Space Estimation. In this step, lines 1-13 of Algorithm 1 that learn the matrix $\boldsymbol{U}$ such that $\mathsf{range}(\boldsymbol{U}) = \mathsf{range}(\boldsymbol{M}^\top)$ is analyzed.

- **A1)** Under the binary observation model, the top-K SVD's are performed on the adjacency blocks $\boldsymbol{A}_{\ell,m}$'s not on blocks $\boldsymbol{P}_{\ell,m}$'s. The noise associated with this is characterized by using Lemmas 1 and 2.
- **A2)** Each matrix block $\boldsymbol{U}_r$ is estimated using the sub-algorithm 2. Lemmas 3-5 are derived to characterize this basis stitching operation. A number of lemmas (Lemmas 10-13) are employed to assist this step, for e.g., to characterize the condition numbers of the latent matrices.
- **A3)** The final estimation error bound for the matrix $\boldsymbol{U} = [\boldsymbol{U}_1^\top, \ldots, \boldsymbol{U}_L^\top]^\top$ is obtained by recursively applying the error bound for each $\boldsymbol{U}_r$ and by employing Lemma 6.

(B) Membership Estimation. In this step, the error bound for the membership matrix $\boldsymbol{M}$ (see lines 14-15 of Algorithm 1) is analyzed.

- **B1)** The conditions for $\boldsymbol{M}$ to approximately satisfy the ANC is characterized by utilizing Proposition 1.
- **B2)** The final estimation error bound for $\boldsymbol{M}$ is derived using Lemma 7. Lemmas 7 characterizes the combined noise in the model which is contributed by both the noise in the estimate of $\boldsymbol{U}$ and the deviation from the ANC.

(C) Putting Together. In this step, various conditions used in Lemma 1-7 and Proposition 1 are combined to derive the final conditions of Theorem 2.

C.1 Range Space Estimation

In Theorem 1, the subspace identifiability is established via successively applying the top-K SVD on $\boldsymbol{C}_r := [\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top$, i.e., $\boldsymbol{C}_r = [\overline{\boldsymbol{U}}_{\ell_r}^\top, \overline{\boldsymbol{U}}_{\ell_{r+1}}^\top]^\top \boldsymbol{\Sigma}_r \boldsymbol{V}_{m_r}^\top$. However, since $\boldsymbol{C}_r$ is not observed in practice, Algorithm 1 performs the top-K SVD on its estimates $\widehat{\boldsymbol{C}}_r$, which is defined as $\widehat{\boldsymbol{C}}_r := [\boldsymbol{A}_{\ell_r,m_r}^\top, \boldsymbol{A}_{\ell_{r+1},m_r}^\top]^\top$. The factors extracted from the top-K SVD of $\widehat{\boldsymbol{C}}_r$ are denoted as $\widehat{\overline{\boldsymbol{U}}}_{\ell_r}$ and $\widehat{\overline{\boldsymbol{U}}}_{\ell_{r+1}}$, which correspond to the 'noiseless versions' $\overline{\boldsymbol{U}}_{\ell_r}$ and $\overline{\boldsymbol{U}}_{\ell_{r+1}}$, respectively. To proceed, we invoke the following lemma to bound the estimation accuracy of $\widehat{\overline{\boldsymbol{U}}}_{\ell_r}$ and $\widehat{\overline{\boldsymbol{U}}}_{\ell_{r+1}}$.

Lemma 1 *Denote* $\sigma_1 := \max_{r\in[L-1]} \sigma_{\max}(\boldsymbol{C}_r)$ *and* $\sigma_K := \min_{r\in[L-1]} \sigma_{\min}(\boldsymbol{C}_r)$. *Assume that*

$$\mathrm{rank}(\boldsymbol{C}_r) = K, \quad \|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2 \le \|\boldsymbol{C}_r\|_2. \tag{33}$$

Then, there exists an orthogonal matrix $\boldsymbol{O}_r$ *such that*

$$\|\widehat{\overline{\boldsymbol{U}}}_{\ell_r} - \overline{\boldsymbol{U}}_{\ell_r}\boldsymbol{O}_r\|_{\mathrm{F}} \le \frac{6\sqrt{2K}\sigma_1\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2}{\sigma_K^2}. \tag{34}$$

In addition, $\|\widehat{\overline{\boldsymbol{U}}}_{\ell_{r+1}} - \overline{\boldsymbol{U}}_{\ell_{r+1}}\boldsymbol{O}_r\|_{\mathrm{F}}$ *admits the same upper bound in* (34).

The proof of the lemma is given in Sec. D in the supplemental material.

In order to utilize the bound in (34), we proceed to characterize the term $\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2$. To this end, we present the following lemma:

Lemma 2 *Let* $\rho := \max_{i,j\in[N]} \boldsymbol{P}(i,j)$. *Assume there exists a positive constant* c_0 *such that* $L \leq \frac{4\rho N}{d}$ *and* $\rho \geq \frac{Lc_0 \log(3N/L)}{3N}$. *Then, there exists another positive constant* $\tilde{c}$ *(which is a function of* c_0*) such that*

$$\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2 \leq \tilde{c}\sqrt{\rho N/L}, \quad \forall\, r \in [L-1] \tag{35}$$

holds with probability of at least $1 - \frac{L}{2N}$.

The proof of the lemma is given in Sec. E of the supplementary material.

Since $\{\ell_r\}_{r=1}^L = [L]$ according to EQP, without loss of any generality (w.l.o.g.), in the following sections of the proof, we can fix $\ell_r = r$ for every r. Combining Lemmas 1 and 2, the inequalities, i.e.,

$$\|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_{\mathrm{F}} \leq \phi \text{ and } \|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_{\mathrm{F}} \leq \phi \tag{36}$$

hold with probability of at least $1 - \frac{L}{N}$ where $\phi = \frac{6\tilde{c}\sqrt{2K\rho N}\sigma_1}{\sqrt{L}\sigma_K^2}$.

Algorithm 1 (lines 2-3) first estimates $\boldsymbol{U}_T$ and $\boldsymbol{U}_{T+1}$ by directly performing the top-K SVD to $\widehat{\boldsymbol{C}}_T = [\boldsymbol{A}_{T,m_T}^\top, \boldsymbol{A}_{T+1,m_T}^\top]^\top$. Recall that we use the notation $\boldsymbol{C}_T = [\boldsymbol{P}_{T,m_T}^\top, \boldsymbol{P}_{T+1,m_T}^\top]^\top = [\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]^\top \boldsymbol{\Sigma}_T \boldsymbol{V}_{m_T}^\top$. Therefore, by (36) and denoting $\boldsymbol{O}_T := \boldsymbol{O}$, we have

$$\|\widehat{\boldsymbol{U}}_T - \boldsymbol{U}_T\boldsymbol{O}\|_{\mathrm{F}} \leq \phi \text{ and } \|\widehat{\boldsymbol{U}}_{T+1} - \boldsymbol{U}_{T+1}\boldsymbol{O}\|_{\mathrm{F}} \leq \phi \tag{37}$$

holding with probability of at least $1 - \frac{L}{N}$, where $\widehat{\boldsymbol{U}}_T$ and $\widehat{\boldsymbol{U}}_{T+1}$ are the factors resulted from the top-K SVD of $\widehat{\boldsymbol{C}}_T$.

Next, the iterative steps (lines 5-8) in Algorithm 1 (which includes the sub-algorithm `PairStitch`) perform the below operation to estimate the basis $\boldsymbol{U}_{r+1}$ for $r = T+1, \ldots, L$ (where $T = \lfloor L/2 \rfloor$):

$$\widehat{\boldsymbol{U}}_{r+1} = \widehat{\overline{\boldsymbol{U}}}_{r+1} \widehat{\overline{\boldsymbol{U}}}_r^{\dagger} \widehat{\boldsymbol{U}}_r. \tag{38}$$

In order to analyze $\widehat{\boldsymbol{U}}_{r+1}$, we first define the following notations:

$$\alpha_{\max} := \max_r \sigma_{\max}(\boldsymbol{M}_r), \quad \alpha_{\min} := \min_r \sigma_{\min}(\boldsymbol{M}_r).$$

The above implies that $\frac{\alpha_{\max}}{\alpha_{\min}} = \max_r \kappa(\boldsymbol{M}_r) = \gamma$. Under Assumption 1, $\mathrm{rank}(\boldsymbol{M}_r) = K$ holds with probability 1. Hence, both $\alpha_{\max} < \infty$ and $\alpha_{\min} > 0$ also hold with probability one.

Using the above notations and the assumptions of Theorem 2, we have the following three lemmas:

Lemma 3 *There exists constants* $\beta_{\min} > 0$ *and* $\beta_{\max} < \infty$ *such that for every* $r \in [L-1]$, *we have*

$$\sigma_{\max}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]) \leq \beta_{\max} = \sqrt{2K}\alpha_{\max},$$
$$\sigma_{\min}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]) \geq \beta_{\min} = \alpha_{\min}.$$

Lemma 4 *Let* $\widehat{\overline{\boldsymbol{U}}}_r$ *denote the noisy estimate of* $\overline{\boldsymbol{U}}_r\boldsymbol{O}_r$ *obtained from the top-*K *SVD of* $\widehat{\boldsymbol{C}}_r$. *Suppose that* $\|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_2 \leq \frac{\alpha_{\min}}{2\beta_{\max}}$. *Then, we have* $\|\widehat{\overline{\boldsymbol{U}}}_r^{\dagger} - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^{\dagger}\|_2 \leq \frac{2\sqrt{2}\beta_{\max}^2}{\alpha_{\min}^2}\|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\|_2$.

Lemma 5 *Consider the relation in* (38). *Assume that* $\|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_2 \leq \frac{\alpha_{\min}}{2\beta_{\max}}$ *and* $\|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 \leq \frac{\alpha_{\min}}{2\beta_{\max}}$, *then we have*

$$\begin{aligned}\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_2 \leq{}& \frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 + 2\left(\frac{\alpha_{\max}}{\beta_{\min}}\right)^2\|\widehat{\overline{\boldsymbol{U}}}_r^{\dagger} - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^{\dagger}\|_2 \\ &+ 4\frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\|\widehat{\boldsymbol{U}}_r - \boldsymbol{U}_r\boldsymbol{O}\|_2.\end{aligned} \tag{39}$$

The proofs of Lemmas 3-5 can be found Sec. F, G, and H of the supplementary material, respectively. By applying Lemma 4 to the second term in (39) of Lemma 5, we get

$$\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_2 \leq \frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 + 4\sqrt{2}\left(\frac{\alpha_{\max}\beta_{\max}}{\beta_{\min}\alpha_{\min}}\right)^2\|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\|_2 + 4\frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\|\widehat{\boldsymbol{U}}_r - \boldsymbol{U}_r\boldsymbol{O}\|_2. \tag{40}$$

Employing Lemma 3, we have

$$\frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}} = \frac{\sqrt{2K}\alpha_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\alpha_{\min}} = \sqrt{2K}\gamma^2. \tag{41}$$

Combining (40) and (41) and defining $\overline{\kappa} = \sqrt{2K}\gamma^2$, we obtain

$$\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_2 \leq \overline{\kappa}\|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 + 4\sqrt{2}\overline{\kappa}^2\|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\|_2 + 4\overline{\kappa}\|\widehat{\boldsymbol{U}}_r - \boldsymbol{U}_r\boldsymbol{O}\|_2. \tag{42}$$

Hence, we obtain the following inequalities with probability of at least $1 - \frac{L}{N}$:

$$\begin{aligned}\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_{\mathrm{F}} &\leq \sqrt{K}\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_2 \\ &\leq \sqrt{K\overline{\kappa}}\|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 + 4\sqrt{2}\sqrt{K}\overline{\kappa}^2\|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\|_2 \\ &\quad + 4\sqrt{K\overline{\kappa}}\|\widehat{\boldsymbol{U}}_r - \boldsymbol{U}_r\boldsymbol{O}\|_2 \\ &\leq 5\sqrt{2K}\overline{\kappa}^2\phi + 4\overline{\kappa}\sqrt{K}\|\widehat{\boldsymbol{U}}_r - \boldsymbol{U}_r\boldsymbol{O}\|_{\mathrm{F}},\end{aligned}$$

where $\phi = \frac{6\tilde{c}\sqrt{2K\rho N}\sigma_1}{\sqrt{L}\sigma_K^2}$. The first inequality is obtained by using norm equivalence, the second inequality is obtained by applying (42) and the last inequality is by applying (36) and using the fact that $\overline{\kappa} \geq 1$.

By recursively applying the above result for any $r \in \{T+1, \ldots, L-1\}$, we further have

$$\begin{aligned}\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_{\mathrm{F}} &\leq 5\sqrt{2K}\overline{\kappa}^2\phi + 4\sqrt{K\overline{\kappa}}\Big(5\sqrt{2K}\overline{\kappa}^2\phi + 4\sqrt{K\overline{\kappa}}\ldots \\ &\qquad \big(5\sqrt{2K}\overline{\kappa}^2\phi + 4\sqrt{K\overline{\kappa}}\|\widehat{\boldsymbol{U}}_{T+1} - \boldsymbol{U}_{T+1}\boldsymbol{O}\|_{\mathrm{F}}\big)\Big) \\ &\leq 5\sqrt{2K}\overline{\kappa}^2\phi + 4\sqrt{K\overline{\kappa}}\Big(5\sqrt{2K}\overline{\kappa}^2\phi + 4\sqrt{K\overline{\kappa}}\ldots \\ &\qquad \big(5\sqrt{2K}\overline{\kappa}^2\phi + 4\sqrt{K\overline{\kappa}}\phi\big)\Big) \\ &= 5\sqrt{2K}\overline{\kappa}^2\phi\Big(1 + 4\sqrt{K\overline{\kappa}} + (4\sqrt{K\overline{\kappa}})^2 + \ldots + (4\sqrt{K\overline{\kappa}})^{r-T}\Big),\end{aligned}$$

where we have applied (37) for obtaining the last inequality. By applying the geometric sum formula, for any $r \in \{T+1, \ldots, L-1\}$, we have

$$\begin{aligned}\|\widehat{\boldsymbol{U}}_{r+1} - \boldsymbol{U}_{r+1}\boldsymbol{O}\|_{\mathrm{F}} &\leq \frac{5\sqrt{2K}\overline{\kappa}^2\phi\left((4\sqrt{K\overline{\kappa}})^{r-T+1} - 1\right)}{4\sqrt{K\overline{\kappa}} - 1} \\ &\leq \frac{5\sqrt{2K}\overline{\kappa}^2\phi(4\sqrt{K\overline{\kappa}})^{r-T+1}}{4\sqrt{K\overline{\kappa}} - 1}.\end{aligned} \tag{43}$$

Following the same derivation, one can show that for $r = 2, \ldots, T$, we have

$$\|\widehat{\boldsymbol{U}}_{r-1} - \boldsymbol{U}_{r-1}\boldsymbol{O}\|_{\mathrm{F}} \leq \frac{5\sqrt{2K}\overline{\kappa}^2\phi(4\sqrt{K\overline{\kappa}})^{T-r+2}}{4\sqrt{K\overline{\kappa}} - 1}. \tag{44}$$

Consequently, we get

$$\begin{aligned}\|\widehat{\boldsymbol{U}}-\boldsymbol{U}\boldsymbol{O}\|_{\mathrm{F}} &\leq \sum_{r=1}^{L}\|\widehat{\boldsymbol{U}}_r-\boldsymbol{U}_r\boldsymbol{O}\|_F \\ &= \sum_{r=1}^{T-1}\|\widehat{\boldsymbol{U}}_r-\boldsymbol{U}_r\boldsymbol{O}\|_F+\|\widehat{\boldsymbol{U}}_T-\boldsymbol{U}_T\boldsymbol{O}\|_{\mathrm{F}}+\|\widehat{\boldsymbol{U}}_{T+1}-\boldsymbol{U}_{T+1}\boldsymbol{O}\|_{\mathrm{F}}+\sum_{r=T+2}^{L}\|\widehat{\boldsymbol{U}}_r-\boldsymbol{U}_r\boldsymbol{O}\|_F \\ &\leq \frac{5\sqrt{2K\overline{\kappa}^2}\phi(4\sqrt{K\overline{\kappa}})^{T+1}}{(4\sqrt{K\overline{\kappa}}-1)^2}+\frac{5\sqrt{2K\overline{\kappa}^2}\phi(4\sqrt{K\overline{\kappa}})^{T+1}}{(4\sqrt{K\overline{\kappa}}-1)^2} \\ &= \frac{10\sqrt{2K\overline{\kappa}^2}\phi(4\sqrt{K\overline{\kappa}})^{L/2+1}}{(4\sqrt{K\overline{\kappa}}-1)^2},\end{aligned}$$

where the first inequality is by the triangle inequality, the second inequality is obtained by (37), (43), (44), and by the geometric sum formula.

Substituting $\phi=\frac{6\tilde{c}\sqrt{2K\rho N}\sigma_1}{\sqrt{L}\sigma_K^2}$ into the above result, we have the following result with probability at least $1-L(L-1)/N$:

$$\|\widehat{\boldsymbol{U}}-\boldsymbol{U}\boldsymbol{O}\|_{\mathrm{F}}\leq\frac{15K\tilde{c}\overline{\kappa}^2(4\sqrt{K\overline{\kappa}})^{(L/2+1)}\sigma_1\sqrt{\rho N}}{2(\sqrt{K\overline{\kappa}}-1)^2\sigma_K^2\sqrt{L}}. \tag{45}$$

Note that we have applied union bound to reach the probability $1-L(L-1)/N$, since the bound (36) derived from Lemma 2 are invoked for every $r\in[L-1]$. Next, we bound σ_1 and σ_K using the following lemma:

Lemma 6 *Assume that the columns of $\boldsymbol{M}$ are generated from any continuous distribution and there exist positive constants c and C, where $c\leq C$ depending only on distribution of the columns of $\boldsymbol{M}$. Also assume that $(N/L)\geq\frac{4}{c}\log(NK/L)$. Then, the following holds true with probability of at least $1-(3L(L-1)/N)$:*

$$\sigma_K\geq\sqrt{2}c\sigma_{\min}(\boldsymbol{B})N/L, \tag{46}$$

$$\sigma_1\leq\sqrt{2}C\sigma_{\max}(\boldsymbol{B})N/L. \tag{47}$$

The proof can be found in Sec. J of the supplementary material.

Applying Lemma 6 in (45) and substituting $\overline{\kappa}=\sqrt{2K}\gamma^2$, we get the following with probability at least $1-(4L(L-1)/N)$:

$$\|\widehat{\boldsymbol{U}}-\boldsymbol{U}\boldsymbol{O}\|_F\leq\frac{15K^2\tilde{c}C\gamma^4(4\sqrt{2}K\gamma^2)^{(L/2+1)}\kappa(\boldsymbol{B})\sqrt{\rho}}{c^2\sigma_{\min}(\boldsymbol{B})(\sqrt{2}K\gamma^2-1)^2\sqrt{2(N/L)}}:=\zeta. \tag{48}$$

C.2 Membership Estimation

Note that in the $\boldsymbol{A}=\boldsymbol{P}$ case, $\mathsf{range}(\boldsymbol{U})=\mathsf{range}(\boldsymbol{M}^\top)$. Therefore, there exists a nonsingular matrix $\boldsymbol{G}\in\mathbb{R}^{K\times K}$ such that $\boldsymbol{U}=\boldsymbol{M}^\top\boldsymbol{G}$. In the binary observation case, in order to estimate the membership matrix $\boldsymbol{M}$, Algorithm 1 applies `SPA` to the estimated $\widehat{\boldsymbol{U}}$. To analyze this step, we start by considering the following noisy NMF model:

$$\widehat{\boldsymbol{U}}^\top=\boldsymbol{O}^\top\boldsymbol{G}^\top\boldsymbol{M}+\boldsymbol{N}, \tag{49}$$

where $\boldsymbol{O}\in\mathbb{R}^{K\times K}$ is the orthogonal matrix, $\boldsymbol{M}$ satisfies simplex constraints, i.e., $\boldsymbol{M}\geq\boldsymbol{0},\boldsymbol{1}^\top\boldsymbol{M}=\boldsymbol{1}^\top$ and $\boldsymbol{N}\in\mathbb{R}^{K\times N}$ is the noise matrix, whose Euclidean norm upper bound was obtained in the previous subsection by (48) such that $\|\boldsymbol{N}\|_{\mathrm{F}}\leq\zeta$. From the noisy observation $\widehat{\boldsymbol{U}}$, one can get an estimate for $\boldsymbol{M}$ employing

SPA. However, SPA can provably work only when $\boldsymbol{M}$ satisfies ε-separability. For this, we use Proposition 1 to get the condition for ε-separability for $\boldsymbol{M}$. Then, combining the noise robustness analysis of SPA in [35] and Proposition 1, we show the following:

Lemma 7 *Consider the noisy model in (49) and assume that $\|\boldsymbol{N}\|_{\mathrm{F}} \leq \zeta$ and that the assumptions in Proposition 1 hold true. Also, suppose that*

$$\zeta \leq \frac{1}{4\sqrt{2K}\alpha_{\max}\widetilde{\kappa}(\boldsymbol{M})} \quad \textit{and} \quad \varepsilon \leq \frac{1}{4\sqrt{2K}\gamma\widetilde{\kappa}(\boldsymbol{M})}. \tag{50}$$

Then, Algorithm 1 outputs $\widehat{\boldsymbol{M}}$ such that

$$\min_{\boldsymbol{\Pi}} \|\widehat{\boldsymbol{M}} - \boldsymbol{\Pi}\boldsymbol{M}\|_{\mathrm{F}} \leq \sigma_{\max}(\boldsymbol{M})\sqrt{2K}\gamma\widetilde{\kappa}(\boldsymbol{M})\left(3\varepsilon + 4\zeta\right), \tag{51}$$

where $\widetilde{\kappa}(\boldsymbol{M}) = 1 + 160K\gamma^2$.

The proof is provided in Sec K of the supplementary material.

C.3 Putting Together

To make the bounds for $\boldsymbol{U}$ and $\boldsymbol{M}$ given by (48) and (51), respectively, legitimate, we need the assumptions in Lemmas 1-7 hold true.

The assumptions on $\boldsymbol{C}_r$ in Lemma 1 given by the equalities in (33) are satisfied if

$$\operatorname{rank}(\boldsymbol{M}_r) = \operatorname{rank}(\boldsymbol{B}) = K, \quad \forall r \in [L] \quad \text{and} \tag{52}$$

$$\tilde{c}\sqrt{\rho N/L} \leq \sigma_K. \tag{53}$$

Under Assumption 1, $\operatorname{rank}(\boldsymbol{M}_r) = K$ hold with probability one [47, 48]. Using (46), The condition (53) can be further written as

$$N/L \geq \frac{\tilde{c}^2\rho}{2\sigma^2_{\min}(\boldsymbol{B})c^2}. \tag{54}$$

The assumptions in Lemma 4 and Lemma 5 can be written as:

$$\begin{aligned}\phi = \frac{6\tilde{c}\sqrt{2K\rho N}\sigma_1}{\sqrt{L}\sigma_K^2} &\leq \frac{6\tilde{c}C\kappa(\boldsymbol{B})\sqrt{LK\rho}}{c^2\sigma_{\min}(\boldsymbol{B})\sqrt{N}} \\ &\leq \frac{\alpha_{\min}}{2\beta_{\max}} = \frac{\alpha_{\min}}{2\sqrt{2K}\alpha_{\max}} = \frac{1}{2\sqrt{2K}\gamma},\end{aligned}$$

where we have utilized Lemma 6 to obtain the first inequality and Lemma 3 to obtain the second equality. Hence, we obtain the condition such that

$$N/L \geq \frac{(6\sqrt{2}\tilde{c}C)^2\kappa^2(\boldsymbol{B})K^2\rho\gamma^2}{c^4\sigma^2_{\min}(\boldsymbol{B})}. \tag{55}$$

We can see that the condition in (54) is satisfied if (55) holds.

Also, the below conditions from Lemmas 2 and 6 have to be satisfied:

$$L \leq \frac{4\rho N}{d}, \quad \rho \geq \frac{Lc_0\log(3N/L)}{3N} \tag{56a}$$

$$(N/L) \geq \frac{4}{c}\log(NK/L). \tag{56b}$$

Next, we proceed to analyze the conditions in Lemma 7. The conditions in Proposition 1 has to be satisfied by Lemma 7 as well which implies that the below has to hold:

$$N = \Omega\left((G(\varepsilon, \boldsymbol{\nu}))^{-2} \log\left(K/\mu\right)\right), \tag{57}$$

where $G(\varepsilon, \boldsymbol{\nu})$ is given by (11).

Lemma 7 has a condition on ζ given by (50). By substituting the value of ζ given by (48), the condition on ζ in (50) can be re-written as:

$$N/L \geq \frac{(30K^3\tilde{c}C\gamma^4\kappa(\boldsymbol{B})\widetilde{\kappa}(\boldsymbol{M})\alpha_{\max})^2\rho(4\sqrt{2}K\gamma^2)^{L+2}}{(c^2\sigma_{\min}(\boldsymbol{B})(K\gamma^2-1)^2)^2}. \tag{58}$$

Note that the condition on N given by (55) is satisfied if the above condition on N holds true.

Hence, if the conditions (57) and (58) are satisfied together with (52) and (56), the bounds for $\boldsymbol{U}$ and $\boldsymbol{M}$ given by Theorem 2 hold.

Next, we proceed to derive the probability with which the final bounds for $\boldsymbol{U}$ and $\boldsymbol{M}$ hold true. For this, we summarize the probabilistic events appeared in the proof as follows:

1. The first event is that the subspace perturbation error associated with every adjacency matrix blocks $\boldsymbol{C}_r, r \in [L-1]$ is bounded. This happens with probability of at least $1 - L(L-1)/N$ (see Lemma 2);
2. The second event is that the singular values σ_1 and σ_K associated with the adjacency matrix blocks $\boldsymbol{C}_r, r \in [L-1]$ are bounded. This holds with probability of at least $1 - (3L(L-1)/N)$ (see Lemma 6);
3. The third event is that there exist anchor nodes in the membership matrix $\boldsymbol{M}$ up to an error bounded by ε. This happens with probability of at least $1-\mu$ according to Proposition 1. By substituting $\mu = L(L-1)/N$ in (10) of Proposition 1, we can get that the result in Proposition 1 holds with probability of at least $1 - L(L-1)/N$.

Hence, by applying the union bound to the three event probabilities listed above, the final bounds for $\boldsymbol{U}$ and $\boldsymbol{M}$ hold true with probability of at least $1 - (5L(L-1)/N)$.

D Proof of Lemma 1

We invoke the following lemma from [63] to characterize the estimation accuracy of the factors resulting from the top-K SVD of $\widehat{\boldsymbol{C}}_r$:

Lemma 8 *[63] Let $\boldsymbol{C} \in \mathbb{R}^{m\times n}$ and $\widehat{\boldsymbol{C}} \in \mathbb{R}^{m\times n}$ have singular values $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_{\min(m,n)}$ and $\widehat{\alpha}_1 \geq \widehat{\alpha}_2 \geq \ldots \widehat{\alpha}_{\min(m,n)}$, respectively. Fix $1 \leq t \leq s \leq \operatorname{rank}(\boldsymbol{C})$ and assume that $\min(\alpha_{t-1}^2 - \alpha_t^2, \alpha_s^2 - \alpha_{s+1}^2) > 0$, where $\alpha_0^2 := \infty$ and $\alpha_{\operatorname{rank}(\boldsymbol{C})+1} := 0$. Let $q := s - t + 1$ and let $\boldsymbol{U} = \left[\boldsymbol{u}_t, \boldsymbol{u}_{t+1}, \ldots, \boldsymbol{u}_s\right] \in \mathbb{R}^{m\times q}$ and $\widehat{\boldsymbol{U}} = \left[\widehat{\boldsymbol{u}}_t, \widehat{\boldsymbol{u}}_{t+1}, \ldots, \widehat{\boldsymbol{u}}_s\right] \in \mathbb{R}^{m\times q}$ have orthonormal columns satisfying $\boldsymbol{C}^\top \boldsymbol{u}_j = \alpha_j \boldsymbol{v}_j$ and $\widehat{\boldsymbol{C}}^\top \widehat{\boldsymbol{u}}_j = \widehat{\alpha}_j \widehat{\boldsymbol{v}}_j$ for $j = t, t+1, \ldots, s$ and let $\boldsymbol{V} = \left[\boldsymbol{v}_t, \boldsymbol{v}_{t+1}, \ldots, \boldsymbol{v}_s\right] \in \mathbb{R}^{n\times q}$ and $\widehat{\boldsymbol{V}} = \left[\widehat{\boldsymbol{v}}_t, \widehat{\boldsymbol{v}}_{t+1}, \ldots, \widehat{\boldsymbol{v}}_s\right] \in \mathbb{R}^{n\times q}$ have orthonormal columns satisfying $\boldsymbol{C}\boldsymbol{v}_j = \alpha_j \boldsymbol{u}_j$ and $\widehat{\boldsymbol{C}}\widehat{\boldsymbol{v}}_j = \widehat{\alpha}_j \widehat{\boldsymbol{u}}_j$ for $j = t, t+1, \ldots, s$. Then there exists an orthogonal matrix $\boldsymbol{O} \in \mathbb{R}^{q\times q}$ such that*

$$\|\widehat{\boldsymbol{U}} - \boldsymbol{U}\boldsymbol{O}\|_{\mathrm{F}} \leq \frac{2^{3/2}(2\alpha_1 + \|\widehat{\boldsymbol{C}} - \boldsymbol{C}\|_2)\min(q^{1/2}\|\widehat{\boldsymbol{C}} - \boldsymbol{C}\|_2, \|\widehat{\boldsymbol{C}} - \boldsymbol{C}\|_{\mathrm{F}})}{\min(\alpha_{t-1}^2 - \alpha_t^2, \alpha_s^2 - \alpha_{s+1}^2)}$$

and the same upper bound holds for $\|\widehat{\boldsymbol{V}} - \boldsymbol{V}\boldsymbol{O}\|_{\mathrm{F}}$.

By letting $t = 1$ and $s = K$, using the assumption that $\mathrm{rank}(\boldsymbol{C}_r) = K$ and applying Lemma 8, we have

$$\begin{aligned}\|\widehat{\overline{\boldsymbol{U}}}_{\ell_r} - \overline{\boldsymbol{U}}_{\ell_r}\boldsymbol{O}_r\|_{\mathrm{F}} &= \frac{2^{3/2}\sqrt{K}(2\sigma_{\max}(\boldsymbol{C}_r) + \|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2)\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2}{\sigma_{\min}(\boldsymbol{C}_r)^2} \\ &\leq \frac{2^{3/2}\sqrt{K}3\sigma_{\max}(\boldsymbol{C}_r)\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2}{\sigma_{\min}(\boldsymbol{C}_r)^2},\end{aligned}$$

where $\boldsymbol{O}_r \in \mathbb{R}^{K\times K}$ is orthogonal, the first equality is due to the fact that for any matrix $\boldsymbol{X}$ of rank at least K, $\|\boldsymbol{X}\|_{\mathrm{F}} \leq \sqrt{\mathrm{rank}(\boldsymbol{X})}\|\boldsymbol{X}\|_2$ and $\mathrm{rank}(\boldsymbol{X}) \geq K$ and the last inequality is by using the assumption that $\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2 \leq \|\boldsymbol{C}_r\|_2 = \sigma_{\max}(\boldsymbol{C}_r)$.

Therefore, using the defined σ_1 and σ_K, we have for any $r \in [L-1]$,

$$\|\widehat{\overline{\boldsymbol{U}}}_{\ell_r} - \overline{\boldsymbol{U}}_{\ell_r}\boldsymbol{O}_r\|_{\mathrm{F}} \leq \frac{6\sqrt{2K}\sigma_1\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2}{\sigma_K^2}. \tag{59}$$

Note that upper bound in (59) is valid for $\|\widehat{\overline{\boldsymbol{U}}}_{\ell_{r+1}} - \overline{\boldsymbol{U}}_{\ell_{r+1}}\boldsymbol{O}_r\|_{\mathrm{F}}$ as well.

E Proof of Lemma 2

Consider the below lemma:

Lemma 9 *[51] Let $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ be the symmetric binary adjacency matrix of a random graph on N nodes in which the edges $\boldsymbol{A}(i,j)$ occur independently. Let $\overline{\boldsymbol{P}}$ be a matrix of size $N \times N$ such that the entries $\overline{\boldsymbol{P}}(i,j) \in [0,1]$. Assume that $\boldsymbol{A}(i,j) \sim$ Bernoulli$(\overline{\boldsymbol{P}}(i,j))$ for $i < j$, $\boldsymbol{A}(j,i) = \boldsymbol{A}(i,j)$ for $j > i$ and $\boldsymbol{A}(i,i) = 0$ for every $i \in [N]$. Set $\mathbb{E}[\boldsymbol{A}] = \overline{\boldsymbol{P}}$ and assume that $d \geq \max(\rho N, c_0 \log N)$ where $c_0 > 0$. Then, for any $t > 0$, there exists a constant $c_t = c(t, c_0)$ such that*

$$\|\boldsymbol{A} - \overline{\boldsymbol{P}}\|_2 \leq c_t\sqrt{d},$$

with probability greater than $1 - N^{-t}$.

To utilize Lemma 9 in our case, we analyze diagonal pattern (rightmost pattern in Fig. 1) and nondiagonal patterns (for e.g., the first and second patterns in Fig. 1) of EQP separately. For the diagonal pattern where $\ell_r = r$ and $\ell_{r+1} = m_r = r + 1$ for every r, we consider the below sub-adjacency matrix for any $r \in [L-1]$:

$$\widetilde{\boldsymbol{A}}_r = \begin{bmatrix} \boldsymbol{A}_{r,r} \ , \ \boldsymbol{A}_{r,r+1} \\ \boldsymbol{A}_{r+1,r} \ , \ \boldsymbol{A}_{r+1,r+1} \end{bmatrix},$$

where $\widetilde{\boldsymbol{A}}_r \in \mathbb{R}^{2N/L\times 2N/L}$.

Now, let us define

$$\widetilde{\boldsymbol{P}}_r = \begin{bmatrix} \boldsymbol{P}_{r,r} \ , \ \boldsymbol{P}_{r,r+1} \\ \boldsymbol{P}_{r+1,r} \ , \ \boldsymbol{P}_{r+1,r+1} \end{bmatrix}.$$

It can be noted that $\mathbb{E}[\widetilde{\boldsymbol{A}}_r] = \widetilde{\boldsymbol{P}}_r - \mathrm{diag}(\widetilde{\boldsymbol{P}}_r)$. Denoting $\overline{\boldsymbol{P}}_r := \widetilde{\boldsymbol{P}}_r - \mathrm{diag}(\widetilde{\boldsymbol{P}}_r)$ and $\rho := \max_{i,j\in[N]} \boldsymbol{P}(i,j)$, we have

$$\begin{aligned}\|\widetilde{\boldsymbol{A}}_r - \widetilde{\boldsymbol{P}}_r\|_2 &= \|\widetilde{\boldsymbol{A}}_r - \overline{\boldsymbol{P}}_r + \overline{\boldsymbol{P}}_r - \widetilde{\boldsymbol{P}}_r\|_2 \\ &= \|\widetilde{\boldsymbol{A}}_r - \overline{\boldsymbol{P}}_r - \mathrm{diag}(\widetilde{\boldsymbol{P}}_r)\|_2 \\ &\leq \|\widetilde{\boldsymbol{A}}_r - \overline{\boldsymbol{P}}_r\|_2 + \|\mathrm{diag}(\widetilde{\boldsymbol{P}}_r)\|_2 \\ &\leq \|\widetilde{\boldsymbol{A}}_r - \overline{\boldsymbol{P}}_r\|_2 + \max_i \widetilde{\boldsymbol{P}}_r(i,i) \\ &\leq \|\widetilde{\boldsymbol{A}}_r - \overline{\boldsymbol{P}}_r\|_2 + \rho. \end{aligned} \tag{60}$$

In order to bound the first term $\|\widetilde{\boldsymbol{A}}_r - \overline{\boldsymbol{P}_r}\|_2$, we can utilize Lemma 9. Hence, by assuming

$$d \geq \max(\frac{2\rho N}{L}, c_0 \log(2N/L)), \tag{61}$$

we apply Lemma 9 and (60) and obtain $\|\widetilde{\boldsymbol{A}}_r - \widetilde{\boldsymbol{P}}_r\|_2 \leq c_t\sqrt{d} + \rho$ with probability greater than $1 - (2N/L)^{-t}$. Next, we will rewrite the condition (61) in a simpler form. Suppose

$$\frac{2\rho N}{L} \geq c_0 \log(2N/L), \tag{62}$$

then the condition (61) becomes $d \geq \frac{2\rho N}{L}$. If we choose L to be small enough, then we can satisfy this condition on d, i.e., if we have

$$L \leq \frac{4\rho N}{d}, \tag{63}$$

then the condition $d \geq \frac{2\rho N}{L}$ gets automatically satisfied. Therefore, by using the assumptions (62)-(63) and fixing $t = 1$, we can apply Lemma 9 to obtain

$$\begin{aligned} \|\widetilde{\boldsymbol{A}}_r - \widetilde{\boldsymbol{P}}_r\|_2 &\leq c_1\sqrt{d} + \rho \\ &\leq 2c_1\sqrt{\rho N/L} + \rho \\ &\leq (2c_1 + 1)\sqrt{\rho N/L} \end{aligned} \tag{64}$$

with probability at least $1 - (2N/L)^{-1}$. In the above, the first inequality is by using the relation (60), the second inequality is by employing the assumed bound on L given in (63) and the third inequality is by using the fact that $\rho \leq 1$.

Then, we have

$$\begin{aligned} &\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2 \\ &= \left\|[\boldsymbol{A}_{r,r+1}^\top, \boldsymbol{A}_{r+1,r+1}^\top]^\top - [\boldsymbol{P}_{r,r+1}^\top, \boldsymbol{P}_{r+1,r+1}^\top]^\top\right\|_2 \\ &\leq \|\widetilde{\boldsymbol{A}}_r - \widetilde{\boldsymbol{P}}_r\|_2 \leq (2c_1 + 1)\sqrt{\rho N/L} \end{aligned} \tag{65}$$

holds with probability greater than $1 - (2N/L)^{-1}$, where we have applied (64) to obtain the last inequality.

In order to apply Lemma 9 for non-diagonal patterns in EQP, we start by defining $\boldsymbol{A}_r^*$ and $\boldsymbol{P}_r^*$ in the following way:

$$\boldsymbol{A}_r^* = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{A}_{\ell_r,m_r} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{A}_{\ell_{r+1},m_r} \\ \boldsymbol{A}_{\ell_r,m_r}^\top & \boldsymbol{A}_{\ell_{r+1},m_r}^\top & \boldsymbol{0} \end{bmatrix},$$

$$\boldsymbol{P}_r^* = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{P}_{\ell_r,m_r} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{P}_{\ell_{r+1},m_r} \\ \boldsymbol{P}_{\ell_r,m_r}^\top & \boldsymbol{P}_{\ell_{r+1},m_r}^\top & \boldsymbol{0} \end{bmatrix}$$

where $\boldsymbol{0}$ represents a zero matrix which is of size $N/L \times N/L$. With this definition, we can see that $\boldsymbol{A}_r^*(i,j) \sim$ Bernoulli$(\boldsymbol{P}_r^*(i,j))$ for $i < j$, $\boldsymbol{A}_r^*(j,i) = \boldsymbol{A}_r^*(i,j)$ for $j > i$, $\boldsymbol{A}_r^*(i,i) = 0$ for every i and $\mathbb{E}[\boldsymbol{A}_r^*] = \boldsymbol{P}_r^*$. Therefore, we can apply Lemma 9 by fixing $t = 1$ to result in

$$\|\boldsymbol{A}_r^* - \boldsymbol{P}_r^*\|_2 \leq c_1\sqrt{d}, \tag{66}$$

with probability greater than $1 - (3N/L)^{-1}$. Similar as before, we can derive the conditions in order to obtain (66). It can be easily shown that the sufficient conditions to be satisfied are

$$\frac{3\rho N}{L} \geq c_0 \log(3N/L), \quad L \leq \frac{6\rho N}{d}. \tag{67}$$

Then, for non-diagonal EQP patterns, we have

$$\begin{aligned}\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2 &= \left\| [\boldsymbol{A}_{\ell_r,m_r}^\top, \boldsymbol{A}_{\ell_{r+1},m_r}^\top]^\top - [\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top \right\|_2 \\ &\leq \|\boldsymbol{A}_r^* - \boldsymbol{P}_r^*\|_2 \leq c_1\sqrt{d} \leq \sqrt{6}c_1\sqrt{\rho N/L}, \end{aligned} \tag{68}$$

where we obtained the last inequality by applying the assumed bound on L as given by the second condition in (67).

Hence, from (65) and (68), we can see that $\|\widehat{\boldsymbol{C}}_r - \boldsymbol{C}_r\|_2$ is bounded for any patterns in EQP with probability of at least $1 - (2N/L)^{-1}$ if the below conditions are satisfied.

$$\frac{3\rho N}{L} \geq c_0 \log(3N/L), \quad L \leq \frac{4\rho N}{d}.$$

F Proof of Lemma 3

Consider the below relation for any $r \in [L-1]$:

$$\begin{aligned}\sigma_{\max}^2([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]) &= \|[\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]\|_2^2 \leq \|[\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]\|_{\mathrm{F}}^2 \\ &= \|\boldsymbol{M}_r\|_{\mathrm{F}}^2 + \|\boldsymbol{M}_{r+1}\|_{\mathrm{F}}^2 \\ &\leq K\|\boldsymbol{M}_r\|_2^2 + K\|\boldsymbol{M}_{r+1}\|_2^2 \leq 2K\alpha_{\max}^2, \end{aligned}$$

where we have utilized the norm equivalence for the first and second inequalities and applied the definition of $\alpha_{\max}$ for the last inequality. Hence, we have

$$\sigma_{\max}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]) \leq \sqrt{2K}\alpha_{\max} := \beta_{\max}.$$

Next, consider the below for any $\boldsymbol{x} \in \mathbb{R}^K$ such that $\|\boldsymbol{x}\|_2 = 1$:

$$\begin{aligned} &\|[\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]^\top \boldsymbol{x}\|_2^2 = \|\boldsymbol{M}_r^\top \boldsymbol{x}\|_2^2 + \|\boldsymbol{M}_{r+1}^\top \boldsymbol{x}\|_2^2 \\ \Rightarrow &\min_{\boldsymbol{x}} \|[\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]^\top \boldsymbol{x}\|_2^2 = \min_{\boldsymbol{x}} \|\boldsymbol{M}_r^\top \boldsymbol{x}\|_2^2 + \min_{\boldsymbol{x}} \|\boldsymbol{M}_{r+1}^\top \boldsymbol{x}\|_2^2 \\ \Rightarrow &\min_{\boldsymbol{x}} \|[\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]^\top \boldsymbol{x}\|_2^2 \geq \min_{\boldsymbol{x}} \|\boldsymbol{M}_r^\top \boldsymbol{x}\|_2^2 \\ &\Rightarrow \sigma_{\min}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]) \geq \sigma_{\min}(\boldsymbol{M}_r) \geq \alpha_{\min} := \beta_{\min}, \end{aligned}$$

where we have utilized the definition of $\alpha_{\min}$ for the last inequality.

G Proof of Lemma 4

We utilize the following classic result to characterize the pseudo-inverse:

Lemma 10 *[64] Consider any matrices $\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{E} \in \mathbb{R}^{m\times n}$ such that $\boldsymbol{Z} = \boldsymbol{Y} + \boldsymbol{E}$. Suppose $\mathrm{rank}(\boldsymbol{Z}) = \mathrm{rank}(\boldsymbol{Y})$. Then, we have*

$$\|\boldsymbol{Z}^\dagger - \boldsymbol{Y}^\dagger\|_2 \leq \sqrt{2}\|\boldsymbol{Y}^\dagger\|_2\|\boldsymbol{Z}^\dagger\|_2\|\boldsymbol{E}\|_2.$$

By fixing $\boldsymbol{Z} := \widehat{\overline{\boldsymbol{U}}}_r^\dagger$ and $\boldsymbol{Y} = (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger$, we can apply Lemma 10 to obtain

$$\begin{aligned} &\|\widehat{\overline{\boldsymbol{U}}}_r^\dagger - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger\|_2 \\ &\quad\leq \sqrt{2}\|(\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger\|_2\|\widehat{\overline{\boldsymbol{U}}}_r^\dagger\|_2\|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\|_2. \end{aligned} \tag{69}$$

Regarding the first term in the R.H.S. of the above equation, we have

$$\begin{aligned}\|(\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger\|_2 &= \sigma_{\max}((\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger) \\ &= \frac{1}{\sigma_{\min}(\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)} = \frac{1}{\sigma_{\min}(\overline{\boldsymbol{U}}_r)},\end{aligned} \tag{70}$$

where the last equality is due to the orthogonality of $\boldsymbol{O}_r$.

Similarly, we have

$$\|\widehat{\overline{\boldsymbol{U}}}_r^\dagger\|_2 = \sigma_{\max}(\widehat{\overline{\boldsymbol{U}}}_r^\dagger) = \frac{1}{\sigma_{\min}(\widehat{\overline{\boldsymbol{U}}}_r)}. \tag{71}$$

To proceed, we aim to get a lower bound for $\sigma_{\min}(\widehat{\overline{\boldsymbol{U}}}_r)$. For this, we have the following result:

Lemma 11 *The below relation holds true if* $\|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_2 \leq \sigma_{\min}(\overline{\boldsymbol{U}}_r)/2$ *:*

$$\sigma_{\min}(\widehat{\overline{\boldsymbol{U}}}_r) \geq \sigma_{\min}(\overline{\boldsymbol{U}}_r)/2.$$

Proof: Consider the below set of relations for any vector $\boldsymbol{x} \in \mathbb{R}^K$ satisfying $\|\boldsymbol{x}\| = 1$:

$$\begin{aligned}\|\widehat{\overline{\boldsymbol{U}}}_r\boldsymbol{x}\|_2 &= \|\overline{\boldsymbol{U}}_r\boldsymbol{O}_r\boldsymbol{x} + (\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\boldsymbol{x}\|_2 \\ &\geq \|\overline{\boldsymbol{U}}_r\boldsymbol{O}_r\boldsymbol{x}\|_2 - \|(\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\boldsymbol{x}\|_2, \\ \Rightarrow \min_{\boldsymbol{x}} \|\widehat{\overline{\boldsymbol{U}}}_r\boldsymbol{x}\|_2 &\geq \min_{\boldsymbol{x}} \|\overline{\boldsymbol{U}}_r\boldsymbol{O}_r\boldsymbol{x}\|_2 - \max_{\boldsymbol{x}} \|(\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\boldsymbol{x}\|_2, \\ \Rightarrow \sigma_{\min}(\widehat{\overline{\boldsymbol{U}}}_r) &\geq \sigma_{\min}(\overline{\boldsymbol{U}}_r) - \|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_2,\end{aligned}$$

where the first inequality is by applying the triangle inequality.

Using the assumption that $\|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_2 \leq \sigma_{\min}(\overline{\boldsymbol{U}}_r)/2$, we get $\sigma_{\min}(\widehat{\overline{\boldsymbol{U}}}_r) \geq \sigma_{\min}(\overline{\boldsymbol{U}}_r)/2$. □

Applying (70), (71) and Lemma 11 in (69), we get

$$\|\widehat{\overline{\boldsymbol{U}}}_r^\dagger - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger\|_2 \leq \frac{2\sqrt{2}}{\sigma_{\min}^2(\overline{\boldsymbol{U}}_r)}\|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)\|_2. \tag{72}$$

The final step is to characterize $\sigma_{\min}(\overline{\boldsymbol{U}}_r)$. For this, we utilize the following result:

Lemma 12 *For every* r*, we have*

$$\begin{aligned}\sigma_{\min}(\overline{\boldsymbol{U}}_r) &\geq \frac{\sigma_{\min}(\boldsymbol{M}_r)}{\sigma_{\max}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}])} = \frac{\alpha_{\min}}{\beta_{\max}}, \\ \sigma_{\max}(\overline{\boldsymbol{U}}_r) &\leq \frac{\sigma_{\max}(\boldsymbol{M}_r)}{\sigma_{\min}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}])} = \frac{\alpha_{\max}}{\beta_{\min}}.\end{aligned}$$

Proof: Recall the below set of relations (assuming $\ell_r = r$ for every r):

$$\boldsymbol{C}_r = [\boldsymbol{P}_{r,m_r}^\top, \boldsymbol{P}_{r+1,m_r}^\top]^\top = [\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]^\top \boldsymbol{B}\boldsymbol{M}_{m_r}. \tag{73}$$

The top-K SVD of $\boldsymbol{C}_r$ results in the below relation:

$$\boldsymbol{C}_r = [\overline{\boldsymbol{U}}_r^\top, \overline{\boldsymbol{U}}_{r+1}^\top]^\top \boldsymbol{\Sigma}_r \boldsymbol{V}_{m_r}^\top. \tag{74}$$

From (73) and (74), we get that there exists a nonsingular matrix $\overline{\boldsymbol{G}}_r$ such that

$$[\overline{\boldsymbol{U}}_r^\top, \overline{\boldsymbol{U}}_{r+1}^\top]^\top = [\boldsymbol{M}_r, \boldsymbol{M}_{r+1}]^\top \overline{\boldsymbol{G}}_r, \tag{75}$$

where the matrix $[\overline{\boldsymbol{U}}_r^\top, \overline{\boldsymbol{U}}_{r+1}^\top]^\top$ is semi-orthogonal. We invoke the below fact to proceed further.

Fact 1 *Consider the equation* $\overline{\boldsymbol{U}} = \overline{\boldsymbol{M}}^\top \overline{\boldsymbol{G}}$ *where the matrix* $\overline{\boldsymbol{U}}$ *is tall and is semi-orthogonal. Then the below holds:*

$$\sigma_{\max}(\overline{\boldsymbol{G}}) = 1/\sigma_{\min}(\overline{\boldsymbol{M}}) \quad \textit{and} \quad \sigma_{\min}(\overline{\boldsymbol{G}}) = 1/\sigma_{\max}(\overline{\boldsymbol{M}}).$$

Proof: Since $\overline{\boldsymbol{U}}$ is a semi-orthogonal matrix, we have

$$\begin{aligned} & \overline{\boldsymbol{U}}^\top \overline{\boldsymbol{U}} = \boldsymbol{I}, \\ \implies & \overline{\boldsymbol{G}}^\top \overline{\boldsymbol{M}\boldsymbol{M}}^\top \overline{\boldsymbol{G}} = \boldsymbol{I}, \\ \implies & \overline{\boldsymbol{M}\boldsymbol{M}}^\top = \overline{\boldsymbol{G}}^{-\top} \overline{\boldsymbol{G}}^{-1} = (\overline{\boldsymbol{G}\boldsymbol{G}}^\top)^{-1}. \end{aligned}$$

The above relation implies that $\sigma_{\max}(\overline{\boldsymbol{G}}) = 1/\sigma_{\min}(\overline{\boldsymbol{M}})$, $\sigma_{\min}(\overline{\boldsymbol{G}}) = 1/\sigma_{\max}(\overline{\boldsymbol{M}})$ hold true. □

Due to Fact 1, from (75), we get

$$\sigma_{\max}(\overline{\boldsymbol{G}}_r) = \frac{1}{\sigma_{\min}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}])}, \tag{76a}$$

$$\sigma_{\min}(\overline{\boldsymbol{G}}_r) = \frac{1}{\sigma_{\max}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}])}. \tag{76b}$$

Next we consider the below relation obtained from (75):

$$\overline{\boldsymbol{U}}_r = \boldsymbol{M}_r^\top \overline{\boldsymbol{G}}_r.$$

Since $\boldsymbol{M}_r$ is full row-rank, we have

$$\begin{aligned} \sigma_{\min}(\overline{\boldsymbol{U}}_r) &= \min_{\|\boldsymbol{x}\|_2=1} \|\boldsymbol{M}_r^\top \overline{\boldsymbol{G}}_r \boldsymbol{x}\|_2 \geq \min_{\|\boldsymbol{x}\|_2=1} \sigma_{\min}(\boldsymbol{M}_r) \|\overline{\boldsymbol{G}}_r \boldsymbol{x}\|_2 \\ &= \sigma_{\min}(\boldsymbol{M}_r) \min_{\|\boldsymbol{x}\|_2=1} \|\overline{\boldsymbol{G}}_r \boldsymbol{x}\|_2 = \sigma_{\min}(\boldsymbol{M}_r) \sigma_{\min}(\overline{\boldsymbol{G}}_r) \\ &= \frac{\sigma_{\min}(\boldsymbol{M}_r)}{\sigma_{\max}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}])} \geq \frac{\alpha_{\min}}{\beta_{\max}}, \end{aligned}$$

where we have applied (76) to obtain the last equality and used Lemma 3 for the last inequality.

Similarly, we can easily have,

$$\begin{aligned} \sigma_{\max}(\overline{\boldsymbol{U}}_r) &\leq \sigma_{\max}(\overline{\boldsymbol{G}}_r) \sigma_{\max}(\boldsymbol{M}_r) \\ &= \frac{\sigma_{\max}(\boldsymbol{M}_r)}{\sigma_{\min}([\boldsymbol{M}_r, \boldsymbol{M}_{r+1}])} \leq \frac{\alpha_{\max}}{\beta_{\min}}. \end{aligned}$$

□

Combining Lemma 12 with (72), we have

$$\|\widehat{\overline{\boldsymbol{U}}}_r^\dagger - (\overline{\boldsymbol{U}}_r \boldsymbol{O}_r)^\dagger\|_2 \leq \frac{2\sqrt{2}\beta_{\max}^2}{\alpha_{\min}^2} \|\widehat{\overline{\boldsymbol{U}}}_r - (\overline{\boldsymbol{U}}_r \boldsymbol{O}_r)\|_2 \tag{77}$$

H Proof of Lemma 5

For simpler notations, let us define $\boldsymbol{G}_1 := \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r$, $\boldsymbol{G}_2 := (\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger$ and $\boldsymbol{G}_3 := \boldsymbol{U}_r\boldsymbol{O}$. Also we define $\widehat{\boldsymbol{G}}_1 := \widehat{\overline{\boldsymbol{U}}}_{r+1}$, $\widehat{\boldsymbol{G}}_2 := (\widehat{\overline{\boldsymbol{U}}}_r)^\dagger$ and $\boldsymbol{G}_3 := \widehat{\boldsymbol{U}}_r\boldsymbol{O}$. With these, consider the below:

$$\begin{aligned}
\left\|\widehat{\boldsymbol{G}}_1\widehat{\boldsymbol{G}}_2\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_1\boldsymbol{G}_2\boldsymbol{G}_3\right\|_2 &= \left\|\widehat{\boldsymbol{G}}_1\widehat{\boldsymbol{G}}_2\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_1\boldsymbol{G}_2\boldsymbol{G}_3 - \widehat{\boldsymbol{G}}_1\boldsymbol{G}_2\boldsymbol{G}_3 + \widehat{\boldsymbol{G}}_1\boldsymbol{G}_2\boldsymbol{G}_3\right\|_2 \\
&= \left\|\left(\widehat{\boldsymbol{G}}_1 - \boldsymbol{G}_1\right)\boldsymbol{G}_2\boldsymbol{G}_3 + \widehat{\boldsymbol{G}}_1\left(\widehat{\boldsymbol{G}}_2\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_2\boldsymbol{G}_3\right)\right\|_2 \\
&\le \left\|\left(\widehat{\boldsymbol{G}}_1 - \boldsymbol{G}_1\right)\boldsymbol{G}_2\boldsymbol{G}_3\right\|_2 + \left\|\widehat{\boldsymbol{G}}_1\left(\widehat{\boldsymbol{G}}_2\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_2\boldsymbol{G}_3\right)\right\|_2 \\
&= \left\|\left(\widehat{\boldsymbol{G}}_1 - \boldsymbol{G}_1\right)\boldsymbol{G}_2\boldsymbol{G}_3\right\|_2 \\
&\quad + \left\|\widehat{\boldsymbol{G}}_1(\widehat{\boldsymbol{G}}_2 - \boldsymbol{G}_2)\boldsymbol{G}_3 + \widehat{\boldsymbol{G}}_1\widehat{\boldsymbol{G}}_2\left(\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_3\right)\right\|_2 \\
&\le \|\boldsymbol{G}_2\|_2\|\boldsymbol{G}_3\|_2\left\|\widehat{\boldsymbol{G}}_1 - \boldsymbol{G}_1\right\|_2 + \|\widehat{\boldsymbol{G}}_1\|_2\|\boldsymbol{G}_3\|_2\left\|\widehat{\boldsymbol{G}}_2 - \boldsymbol{G}_2\right\|_2 \\
&\quad + \|\widehat{\boldsymbol{G}}_1\|_2\|\widehat{\boldsymbol{G}}_2\|_2\left\|\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_3\right\|_2, \qquad (78)
\end{aligned}$$

where we have applied triangle inequality and the fact that $\|\boldsymbol{Y}\boldsymbol{Z}\|_2 \le \|\boldsymbol{Y}\|_2\|\boldsymbol{Z}\|_2$ to obtain the first and last inequalities, respectively.

We introduce the following lemma and then proceed to characterize $\|\widehat{\boldsymbol{G}}_1\|_2$, $\|\boldsymbol{G}_2\|_2$, $\|\widehat{\boldsymbol{G}}_2\|_2$ and $\|\boldsymbol{G}_3\|_2$.

Lemma 13 *The below relation holds for every r:*

$$\sigma_{\max}(\boldsymbol{U}_r) \le \frac{\alpha_{\max}}{\beta_{\min}}.$$

The proof of the lemma is provided in Sec. I.

- **Bound for $\|\widehat{\boldsymbol{G}}_1\|_2 = \|\widehat{\overline{\boldsymbol{U}}}_{r+1}\|_2$**

$$\begin{aligned}
\|\widehat{\boldsymbol{G}}_1\|_2 = \|\widehat{\overline{\boldsymbol{U}}}_{r+1}\|_2 &= \|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r + \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 \\
&\le \|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 + \|\overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 \\
&\le \sigma_{\max}(\overline{\boldsymbol{U}}_{r+1}) + \sigma_{\max}(\overline{\boldsymbol{U}}_{r+1}) \\
&= 2\sigma_{\max}(\overline{\boldsymbol{U}}_{r+1}) \le 2\frac{\alpha_{\max}}{\beta_{\min}}.
\end{aligned}$$

  where we have used triangle inequality for the first inequality, used the assumption that $\|\widehat{\overline{\boldsymbol{U}}}_{r+1} - \overline{\boldsymbol{U}}_{r+1}\boldsymbol{O}_r\|_2 \le \sigma_{\min}(\overline{\boldsymbol{U}}_{r+1})/2$ for the second inequality and invoked Lemma 12 for the last inequality.

- **Bound for $\|\boldsymbol{G}_2\|_2 = \|(\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger\|_2$**

$$\|\boldsymbol{G}_2\|_2 = \|(\overline{\boldsymbol{U}}_r\boldsymbol{O}_r)^\dagger\|_2 = 1/\sigma_{\min}(\overline{\boldsymbol{U}}_r) \le \frac{\beta_{\max}}{\alpha_{\min}},$$

  where we have applied Lemma 12 for the last inequality.

- **Bound for $\|\widehat{\boldsymbol{G}}_2\|_2 = \|\widehat{\overline{\boldsymbol{U}}}_r^\dagger\|_2$**

$$\|\widehat{\boldsymbol{G}}_2\|_2 = \|\widehat{\overline{\boldsymbol{U}}}_r^\dagger\|_2 = 1/\sigma_{\min}(\widehat{\overline{\boldsymbol{U}}}_r^\dagger) \le 2/\sigma_{\min}(\overline{\boldsymbol{U}}_r) \le 2\frac{\beta_{\max}}{\alpha_{\min}},$$

  where we have used applied Lemma 11 by using the assumption that $\|\widehat{\overline{\boldsymbol{U}}}_r - \overline{\boldsymbol{U}}_r\boldsymbol{O}_r\|_2 \le \sigma_{\min}(\overline{\boldsymbol{U}}_r)/2$ for the first inequality and invoked Lemma 12 for the last inequality.

- **Bound for $\|\boldsymbol{G}_3\|_2 = \|\boldsymbol{U}_r\boldsymbol{O}\|_2$**

$$\|\boldsymbol{G}_3\|_2 = \|\boldsymbol{U}_r\boldsymbol{O}\|_2 = \|\boldsymbol{U}_r\|_2 \leq \frac{\alpha_{\max}}{\beta_{\min}},$$

where we have invoked Lemma 13 for the inequality.

Applying these upper bounds in (78), we finally have

$$\begin{aligned}\left\|\widehat{\boldsymbol{G}}_1\widehat{\boldsymbol{G}}_2\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_1\boldsymbol{G}_2\boldsymbol{G}_3\right\|_2 \leq \frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\left\|\widehat{\boldsymbol{G}}_1 - \boldsymbol{G}_1\right\|_2 + 2\frac{\alpha_{\max}}{\beta_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\left\|\widehat{\boldsymbol{G}}_2 - \boldsymbol{G}_2\right\|_2 \\ + 4\frac{\beta_{\max}}{\alpha_{\min}}\frac{\alpha_{\max}}{\beta_{\min}}\left\|\widehat{\boldsymbol{G}}_3 - \boldsymbol{G}_3\right\|_2.\end{aligned}$$

I Proof of Lemma 13

Theorem 1 shows that there exists a certain nonsingular $\boldsymbol{G}$ such that for each true basis $\boldsymbol{U}_r$ for $r \in [L]$, we have:

$$\boldsymbol{U}_r = \boldsymbol{M}_r^\top \boldsymbol{G}. \tag{79}$$

Recall that, under noiseless case, Algorithm 1 first directly estimates $\boldsymbol{U}_T$ and $\boldsymbol{U}_{T+1}$ by performing the top-K SVD to $\boldsymbol{C}_T = [\boldsymbol{P}_{T,m_T}^\top, \boldsymbol{P}_{T+1,m_T}^\top]^\top$ (assuming $\ell_r = r$ for every r) where

$$\boldsymbol{C}_T = [\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]^\top \boldsymbol{\Sigma}_T \boldsymbol{V}_{m_T}^\top. \tag{80}$$

The above implies that,

$$[\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]^\top = [\boldsymbol{M}_T, \boldsymbol{M}_{T+1}]^\top \boldsymbol{G}$$

holds where $[\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]$ is semi-orthogonal. By utilizing Fact 1, we get $\sigma_{\max}(\boldsymbol{G}) = 1/\sigma_{\min}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}])$ and $\sigma_{\min}(\boldsymbol{G}) = 1/\sigma_{\max}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}])$. Using this result, from (79), we have

$$\begin{aligned}\sigma_{\max}(\boldsymbol{U}_r) &\leq \sigma_{\max}(\boldsymbol{G})\sigma_{\max}(\boldsymbol{M}_r) \\ &\leq \frac{\sigma_{\max}(\boldsymbol{M}_r)}{\sigma_{\min}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}])} \leq \frac{\alpha_{\max}}{\beta_{\min}},\end{aligned}$$

where we have utilized Lemma 3 to obtain the last inequality.

J Proof of Lemma 6

Let us first consider the below:

$$\boldsymbol{C}_r = [\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top = [\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top \boldsymbol{B}\boldsymbol{M}_{m_r}.$$

Consider the K-th eigenvalue of the symmetric matrix $\boldsymbol{C}_r^\top \boldsymbol{C}_r$:

$$\begin{aligned}&\lambda_K(\boldsymbol{C}_r^\top \boldsymbol{C}_r) \\ &= \lambda_K(\underbrace{\boldsymbol{M}_{m_r}^\top \boldsymbol{B}[\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}][\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top}_{\boldsymbol{W}_1 \in \mathbb{R}^{N/L \times K}} \underbrace{\boldsymbol{B}\boldsymbol{M}_{m_r}}_{\boldsymbol{W}_2 \in \mathbb{R}^{K \times N/L}}),\end{aligned} \tag{81}$$

where λ_K denotes the Kth eigenvalue with $\lambda_1 \geq \ldots \geq \lambda_K$.

We utilize the following fact to proceed further:

Lemma 14 (Theorem 1.3.22) *[65] Consider two matrices $\boldsymbol{W}_1 \in \mathbb{R}^{N\times K}$ and $\boldsymbol{W}_2 \in \mathbb{R}^{K\times N}$. Let $\lambda_1, \ldots, \lambda_K$ be the nonzero eigenvalues of the matrix $\boldsymbol{W}_1\boldsymbol{W}_2$. Then, the matrix $\boldsymbol{W}_2\boldsymbol{W}_1$ also holds the same set of eigenvalues.*

Combining Lemma 14 and (81), we have

$$\lambda_K(\boldsymbol{C}_r^\top \boldsymbol{C}_r) = \lambda_K(\boldsymbol{B}\boldsymbol{M}_{m_r}\boldsymbol{M}_{m_r}^\top \boldsymbol{B}[\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}][\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top). \tag{82}$$

Next, we consider the following matrix

$$\boldsymbol{H}_r := \boldsymbol{M}_{m_r}\boldsymbol{M}_{m_r}^\top = \sum_{n=1}^{N/L} \boldsymbol{M}_{m_r}(:,n)\boldsymbol{M}_{m_r}(:,n)^\top. \tag{83}$$

To proceed, we have the lemma from [21] to bound $\lambda_K(\boldsymbol{H}_r)$ and $\lambda_1(\boldsymbol{H}_r)$.

Lemma 15 *[21] Assume that the columns of $\boldsymbol{M}$ are generated from any continuous distribution. Then, there exists positive constants c and C depending only on distribution of the columns of $\boldsymbol{M}$ such that*

$$\begin{aligned}
&\mathsf{Pr}\left(\lambda_K\left(\boldsymbol{H}_r\right) \le cN/L\right) \le K\exp(-cN/4L),\\
&\mathsf{Pr}\left(\lambda_1\left(\boldsymbol{H}_r\right) \ge CN/L\right) \le \frac{K}{2^{CN/L}}.
\end{aligned}$$

To simplify the bound on the probability, let us find the conditions at which the following is satisfied:

$$K\exp(-cN/4L) \le (N/L)^{-1}, \tag{84}$$

$$\frac{K}{2^{CN/L}} \le (N/L)^{-1}. \tag{85}$$

The conditions (84) and (85) are equivalent to the following:

$$(N/L) \ge \frac{4}{c}\log(NK/L), \tag{86}$$

$$(N/L) \ge \frac{\log_2 e}{C}\log(NK/L). \tag{87}$$

We can immediately see that if (86) is satisfied, (87) is also satisfied since $c \le C$.

Therefore, we get that if $(N/L) \ge \frac{4}{c}\log(NK/L)$, with probability of at least $1-(N/L)^{-1}$,

$$\lambda_K\left(\boldsymbol{M}_{m_r}\boldsymbol{M}_{m_r}^\top\right) \ge cN/L, \tag{88a}$$

$$\lambda_1\left(\boldsymbol{M}_{m_r}\boldsymbol{M}_{m_r}^\top\right) \le CN/L. \tag{88b}$$

Using the above result, we can also have

$$\lambda_K([\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}][\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top) \ge 2cN/L, \tag{89a}$$

$$\lambda_1([\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}][\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top) \le 2CN/L, \tag{89b}$$

each holding with probability at least $1-(2N/L)^{-1}$, if $(2N/L) \ge \frac{4}{c}\log(2NK/L)$.

Thus, applying (88) and (89) in (82), we get

$$\begin{aligned}
\lambda_K(\boldsymbol{C}_r^\top \boldsymbol{C}_r) &\ge \lambda_K^2(\boldsymbol{B})\lambda_K([\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}][\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top)\lambda_K(\boldsymbol{M}_{m_r}\boldsymbol{M}_{m_r}^\top)\\
&\ge \lambda_K^2(\boldsymbol{B})c(2N/L)c(N/L) = 2\lambda_K^2(\boldsymbol{B})c^2N^2/L^2,\\
&= 2\sigma_{\min}^2(\boldsymbol{B})c^2N^2/L^2,
\end{aligned}$$

with probability at least $1-(2N/3L)^{-1}$ where we have used the facts that $\lambda_1((\boldsymbol{A}\boldsymbol{B})^{-1}) \le \lambda_1(\boldsymbol{A}^{-1})\lambda_1(\boldsymbol{B}^{-1})$ and $\lambda_1(\boldsymbol{A}^{-1}) = 1/\lambda_K(\boldsymbol{A})$ for obtaining the first inequality and used $\lambda_K(\boldsymbol{B}) = \sigma_{\min}(\boldsymbol{B})$ for the last equality.

Similarly, we can get the below with probability at least $1-(2N/3L)^{-1}$:

$$\begin{aligned}\lambda_1(\boldsymbol{C}_r^\top \boldsymbol{C}_r) &\leq \lambda_1^2(\boldsymbol{B})\lambda_1([\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}][\boldsymbol{M}_{\ell_r}, \boldsymbol{M}_{\ell_{r+1}}]^\top)\lambda_1(\boldsymbol{M}_{m_r}\boldsymbol{M}_{m_r}^\top)\\ &\leq \lambda_1^2(\boldsymbol{B})C(2N/L)C(N/L) = 2\lambda_1^2(\boldsymbol{B})C^2N^2/L^2,\\ &= 2\sigma_{\max}^2(\boldsymbol{B})C^2N^2/L^2.\end{aligned}$$

Therefore, by taking union bound over every $r \in [L-1]$, with probability at least $1-(3L(L-1)/N)$, the following equations hold simultaneously for every r:

$$\sigma_{\min}([\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top) = \sigma_{\min}(\boldsymbol{C}_r) = \sqrt{\lambda_K(\boldsymbol{C}_r^\top \boldsymbol{C}_r)} \geq \sqrt{2}\sigma_{\min}(\boldsymbol{B})cN/L, \tag{90}$$

$$\sigma_{\max}([\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top) = \sigma_{\max}(\boldsymbol{C}_r) = \sqrt{\lambda_1(\boldsymbol{C}_r^\top \boldsymbol{C}_r)} \leq \sqrt{2}\sigma_{\max}(\boldsymbol{B})CN/L. \tag{91}$$

The equations (90) and (91) immediately follow that the below holds with probability at least $1-(3L(L-1)/N)$:

$$\begin{aligned}\sigma_K &= \min_{r\in\{1,\ldots,L-1\}} \sigma_{\min}([\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top)\\ &\geq \sqrt{2}\sigma_{\min}(\boldsymbol{B})cN/L,\\ \sigma_1 &= \max_{r\in\{1,\ldots,L-1\}} \sigma_{\max}([\boldsymbol{P}_{\ell_r,m_r}^\top, \boldsymbol{P}_{\ell_{r+1},m_r}^\top]^\top)\\ &\leq \sqrt{2}\sigma_{\max}(\boldsymbol{B})CN/L.\end{aligned}$$

Combining the conditions to obtain (88) and (89), we get that the above results hold true if $(N/L) \geq \frac{4}{c}\log(NK/L)$.

K Proof of Lemma 7

We consider the given noisy NMF model:

$$\widehat{\boldsymbol{U}}^\top = \boldsymbol{O}^\top \boldsymbol{G}^\top \boldsymbol{M} + \boldsymbol{N}.$$

It is given that the assumptions in Proposition 1 hold true. This immediately means that $\boldsymbol{M}$ satisfies ε-separability (see Definition 1), i.e., there exists a set of indices $\boldsymbol{\Lambda} := \{q_1, \ldots, q_K\}$ such that $\boldsymbol{M}(:, \boldsymbol{\Lambda}) = \boldsymbol{I}_K + \boldsymbol{E}$, where $\boldsymbol{E} \in \mathbb{R}^{K\times K}$ is the error matrix with $\|\boldsymbol{E}(:,k)\|_2 \leq \varepsilon$ and $\boldsymbol{I}_K$ is the identity matrix of size $K\times K$. Without loss of generality, we let $\boldsymbol{\Lambda} = \{1, \ldots, K\}$. Therefore, we have

$$\begin{aligned}\widehat{\boldsymbol{U}}^\top &= \boldsymbol{O}^\top \boldsymbol{G}^\top \boldsymbol{M} + \boldsymbol{N} = \boldsymbol{O}^\top \boldsymbol{G}^\top [\boldsymbol{I}_K + \boldsymbol{E}\ ,\ \widetilde{\boldsymbol{M}}] + \boldsymbol{N}\\ &= \boldsymbol{O}^\top \boldsymbol{G}^\top [\boldsymbol{I}_K, \widetilde{\boldsymbol{M}}] + [\boldsymbol{O}^\top \boldsymbol{G}^\top \boldsymbol{E}, \boldsymbol{0}] + \boldsymbol{N},\end{aligned}$$

where the zero matrix $\boldsymbol{0}$ has the same dimension of $\widetilde{\boldsymbol{M}}$.

Let $\widetilde{\boldsymbol{N}} = [\boldsymbol{O}^\top \boldsymbol{G}^\top \boldsymbol{E}, \boldsymbol{0}] + \boldsymbol{N}$. This gives us the noisy NMF model as

$$\widehat{\boldsymbol{U}}^\top = \underbrace{\boldsymbol{O}^\top \boldsymbol{G}^\top}_{\boldsymbol{W}} \underbrace{[\boldsymbol{I}_K, \widetilde{\boldsymbol{M}}]}_{\boldsymbol{M}} + \widetilde{\boldsymbol{N}}. \tag{92}$$

Then, the below holds for any $k \in [K]$:

$$\begin{aligned}\|\widetilde{\boldsymbol{N}}(:,k)\|_2 &\leq \|\boldsymbol{G}\boldsymbol{O}\|_2 \|\boldsymbol{E}(:,k)\|_2 + \|\boldsymbol{N}(:,k)\|_2,\\ &\leq \sigma_{\max}(\boldsymbol{G})\varepsilon + \zeta := \widetilde{\zeta}(\boldsymbol{G}),\end{aligned} \tag{93}$$

where the first inequality is by applying triangle inequality and the second inequality is by using the orthogonality of $\boldsymbol{O}$ and also by using the given assumption $\|\boldsymbol{N}(:,k)\|_2 \leq \zeta$.

We utilize the below lemma from [35] which characterizes the estimation accuracy of SPA under the NMF model (92).

Lemma 16 *[35] Consider the noisy model in (92) and suppose that $\boldsymbol{W}$ is full rank and $\boldsymbol{M}$ satisfies the simplex constraints, i.e., $\boldsymbol{M} \geq \boldsymbol{0}, \boldsymbol{1}^\top \boldsymbol{M} = \boldsymbol{1}^\top$. Let each column of $\widetilde{\boldsymbol{N}}$ satisfies the condition $\|\widetilde{\boldsymbol{N}}(:,k)\|_2 \leq \widetilde{\zeta}$ with*

$$\widetilde{\zeta} \leq \sigma_{\min}(\boldsymbol{W}) \varrho \widetilde{\kappa}^{-1}(\boldsymbol{W}). \tag{94}$$

Then the SPA Algorithm returns indices $\widehat{\boldsymbol{\Delta}} = \{\hat{q}_1, \ldots, \hat{q}_K\}$ such that for each k,

$$\min_{\pi} \|\widehat{\boldsymbol{U}}(\hat{q}_k, :) - \boldsymbol{W}(:, \pi(k))\|_2 \leq \widetilde{\kappa}(\boldsymbol{W}) \widetilde{\zeta}, \tag{95}$$

where $\widetilde{\kappa}(\boldsymbol{W}) := 1 + 80\kappa^2(\boldsymbol{W})$, $\varrho := \min\left(\frac{1}{2\sqrt{K-1}}, \frac{1}{4}\right)$ and π is certain permutation for $k \in [K]$.

Therefore, we can apply Lemma 16 to the the noisy model in (92) and obtain an estimate $\widehat{\boldsymbol{G}} = \widehat{\boldsymbol{U}}(\widehat{\boldsymbol{\Delta}}, :)$ by applying SPA such that

$$\|\widehat{\boldsymbol{G}} - \boldsymbol{\Pi}\boldsymbol{G}\boldsymbol{O}\|_{\mathrm{F}} \leq \sqrt{K} \widetilde{\kappa}(\boldsymbol{G}) \widetilde{\zeta}(\boldsymbol{G}). \tag{96}$$

In the above, we applied $\boldsymbol{W} = \boldsymbol{O}^\top \boldsymbol{G}^\top$, used the norm equivalence to convert from ℓ_2-norm to the Frobenius norm, used the fact that $\sigma_{\min}(\boldsymbol{G}\boldsymbol{O}) = \sigma_{\min}(\boldsymbol{G})$ and $\sigma_{\max}(\boldsymbol{G}\boldsymbol{O}) = \sigma_{\max}(\boldsymbol{G})$ due to the orthogonality of the matrix $\boldsymbol{O}$ and $\boldsymbol{\Pi}$ is a permutation matrix such that $\boldsymbol{\Pi} = \arg\min_{\widetilde{\boldsymbol{\Pi}}} \|\widehat{\boldsymbol{G}} - \widetilde{\boldsymbol{\Pi}}\boldsymbol{G}\boldsymbol{O}\|_{\mathrm{F}}$. Also, the condition in Lemma 16 can be represented as

$$\widetilde{\zeta}(\boldsymbol{G}) = \sigma_{\max}(\boldsymbol{G})\varepsilon + \zeta \leq \frac{\sigma_{\min}(\boldsymbol{G})\varrho}{\widetilde{\kappa}(\boldsymbol{G})}. \tag{97}$$

Next, we proceed to characterize the singular values of $\boldsymbol{G}$ to express the above bound in more intuitive way. To this end, we have the following lemma:

Lemma 17 *The below relations hold:*

$$\begin{aligned}
\sigma_{\max}(\boldsymbol{G}) &= 1/\sigma_{\min}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}]) \leq \frac{1}{\alpha_{\min}}, \\
\sigma_{\max}(\boldsymbol{G}) \geq \sigma_{\min}(\boldsymbol{G}) &= 1/\sigma_{\max}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}]) \geq \frac{1}{\sqrt{2K}\alpha_{\max}}, \\
\kappa(\boldsymbol{G}) &\leq \sqrt{2K}\gamma.
\end{aligned}$$

Proof 1 *Recall that, under noiseless case, Algorithm 1 first directly estimates $\boldsymbol{U}_T$ and $\boldsymbol{U}_{T+1}$ by performing the top-K SVD to $\boldsymbol{C}_T = [\boldsymbol{P}_{T,m_T}^\top, \boldsymbol{P}_{T+1,m_T}^\top]^\top$ where*

$$\boldsymbol{C}_T = [\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]^\top \boldsymbol{\Sigma}_T \boldsymbol{V}_{m_T}^\top. \tag{98}$$

The above relation implies that, $[\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]^\top = [\boldsymbol{M}_T, \boldsymbol{M}_{T+1}]^\top \boldsymbol{G}$ holds where $[\boldsymbol{U}_T^\top, \boldsymbol{U}_{T+1}^\top]^\top$ is semi-orthogonal. Applying Fact 1, we can obtain

$$\begin{aligned}
\sigma_{\max}(\boldsymbol{G}) &= 1/\sigma_{\min}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}]) \quad \textit{and} \\
\sigma_{\min}(\boldsymbol{G}) &= 1/\sigma_{\max}([\boldsymbol{M}_T, \boldsymbol{M}_{T+1}]).
\end{aligned}$$

By applying Lemma 3, we can finally get

$$\begin{aligned}
\sigma_{\max}(\boldsymbol{G}) \leq 1/\alpha_{\min}, \quad \sigma_{\min}(\boldsymbol{G}) \geq 1/(\sqrt{2K}\alpha_{\max}) \\
\kappa(\boldsymbol{G}) \leq \sqrt{2K}\gamma.
\end{aligned}$$

Applying Lemma 17, we have

$$\widetilde{\kappa}(\boldsymbol{G}) = 1 + 80\kappa^2(\boldsymbol{G}) \leq 1 + 160K\gamma^2 := \widetilde{\kappa}(\boldsymbol{M}),$$
$$\widetilde{\zeta}(\boldsymbol{G}) = \sigma_{\max}(\boldsymbol{G})\varepsilon + \zeta \leq \frac{\varepsilon}{\alpha_{\min}} + \zeta.$$

Hence the bound in (96) can be written as

$$\|\widehat{\boldsymbol{G}} - \boldsymbol{\Pi}\boldsymbol{G}\boldsymbol{O}\|_{\mathrm{F}} \leq \sqrt{K}\widetilde{\kappa}(\boldsymbol{M})(\varepsilon/\alpha_{\min} + \zeta), \tag{99}$$

and the condition (97) can be rewritten as:

$$\frac{\varepsilon}{\alpha_{\min}} + \zeta \leq \frac{\varrho}{\sqrt{2K}\alpha_{\max}\widetilde{\kappa}(\boldsymbol{M})}. \tag{100}$$

Using the given assumptions on ε and ζ, we can see that (100) gets satisfied.

Now that we have $\boldsymbol{G}$ estimated as given by (99), the next step is to estimate the matrix $\boldsymbol{M}$. Algorithm 1 estimates $\boldsymbol{M}$ via least squares. To characterize this step, we consider the following lemma:

Lemma 18 *[64] Consider the noiseless model $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H}$. Suppose that $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{W}}$ are the noisy estimates of $\boldsymbol{X}$ and $\boldsymbol{W}$, respectively. Assume that there exist constants $\phi_1, \phi_2 > 0$ such that $\|\widehat{\boldsymbol{W}} - \boldsymbol{W}\|_2 \leq \phi_1\|\boldsymbol{W}\|_2$, $\|\widehat{\boldsymbol{X}} - \boldsymbol{X}\|_2 \leq \phi_2\|\boldsymbol{X}\|_2$ and $\phi_1\kappa(\boldsymbol{W}) < 1$. Then we have*

$$\frac{\|\widehat{\boldsymbol{H}} - \boldsymbol{H}\|_2}{\|\boldsymbol{H}\|_2} \leq \frac{\kappa(\boldsymbol{W})(\phi_1 + \phi_2)}{1 - \phi_1\kappa(\boldsymbol{W})} + \kappa(\boldsymbol{W})\phi_1,$$

where $\widehat{\boldsymbol{H}}$ is the least squares solution to the problem $\widehat{\boldsymbol{X}} = \widehat{\boldsymbol{W}}\boldsymbol{H}$.

By fixing $\boldsymbol{X} := \boldsymbol{O}^\top\boldsymbol{U}^\top$, $\boldsymbol{W} := \boldsymbol{O}^\top\boldsymbol{G}^\top\boldsymbol{\Pi}^\top$ and $\boldsymbol{H} := \boldsymbol{\Pi}\boldsymbol{M}$, we can apply Lemma 18. By applying the matrix norm equivalence and considering the fact that the matrix $\boldsymbol{O}$ is orthogonal, the constants ϕ_1 and ϕ_2 in Lemma 18 takes the following values:

$$\phi_1 = \sqrt{K}\widetilde{\kappa}(\boldsymbol{M})(\varepsilon/\alpha_{\min} + \zeta)/\sigma_{\max}(\boldsymbol{G}),\ \phi_2 = \zeta/\sigma_{\max}(\boldsymbol{U}),$$

where the first equality is from (99), the second equality is by using the given assumption that $\|\widehat{\boldsymbol{U}} - \boldsymbol{U}\boldsymbol{O}\|_{\mathrm{F}} = \|\boldsymbol{N}\|_{\mathrm{F}} \leq \zeta$, and $\widetilde{\kappa}(\boldsymbol{M}) = 1 + 160\gamma^2$.

Then, by applying Lemma 18, the estimation bound for $\boldsymbol{M}$ can be expressed as

$$\frac{\|\widehat{\boldsymbol{M}} - \boldsymbol{\Pi}\boldsymbol{M}\|_2}{\|\boldsymbol{M}\|_2} \leq \frac{\kappa(\boldsymbol{G})(\phi_1 + \phi_2)}{1 - \phi_1\kappa(\boldsymbol{G})} + \kappa(\boldsymbol{G})\phi_1. \tag{101}$$

Using the given assumptions on ε and ζ, we can see that the denominator term $1 - \phi_1\kappa(\boldsymbol{G})$ is lower bounded:

$$\begin{aligned}
1 - \phi_1\kappa(\boldsymbol{G}) &= 1 - \left(\frac{\sqrt{K}\widetilde{\kappa}(\boldsymbol{M})\kappa(\boldsymbol{G})(\varepsilon/\alpha_{\min} + \zeta)}{\sigma_{\max}(\boldsymbol{G})}\right), \\
&= 1 - \left(\frac{\sqrt{K}\widetilde{\kappa}(\boldsymbol{M})(\varepsilon/\alpha_{\min} + \zeta)}{\sigma_{\min}(\boldsymbol{G})}\right) \\
&\geq 1 - \left(\sqrt{2K}\alpha_{\max}\widetilde{\kappa}(\boldsymbol{M})(\varepsilon/\alpha_{\min} + \zeta)\right) \\
&\geq 1/2.
\end{aligned} \tag{102}$$

where we have used the assumptions on ε and ζ for the last inequality. Note the result in (102) also makes sure that the condition $\phi_1\kappa(\boldsymbol{G}) < 1$ in Lemma 18 is satisfied.

By applying (102) in the bound (101), we get

$$\frac{\|\widehat{\boldsymbol{M}}-\boldsymbol{\Pi}\boldsymbol{M}\|_2}{\|\boldsymbol{M}\|_2} \leq 2\kappa(\boldsymbol{G})(\phi_1+\phi_2)+\kappa(\boldsymbol{G})\phi_1 = \kappa(\boldsymbol{G})(3\phi_1+2\phi_2)$$
$$= \kappa(\boldsymbol{G})\left(\frac{3\sqrt{K}\widetilde{\kappa}(\boldsymbol{M})(\varepsilon/\alpha_{\min}+\zeta)}{\sigma_{\max}(\boldsymbol{G})}+\frac{2\zeta}{\sigma_{\max}(\boldsymbol{U})}\right). \tag{103}$$

Consider the below to lower bound $\sigma_{\max}(\boldsymbol{U})$ where $\boldsymbol{U}^\top = \boldsymbol{G}^\top\boldsymbol{M}$:

$$\sigma_{\max}(\boldsymbol{U}) \geq \sigma_{\min}(\boldsymbol{G})\sigma_{\max}(\boldsymbol{M})$$
$$= \frac{\sigma_{\max}(\boldsymbol{M})}{\sigma_{\max}([\boldsymbol{M}_T,\boldsymbol{M}_{T+1}])} \geq 1, \tag{104}$$

where we used Lemma 17 to obtain the first equality and used the fact that $\sigma_{\max}(\boldsymbol{M}) \geq \sigma_{\max}([\boldsymbol{M}_T,\boldsymbol{M}_{T+1}])$ for the last inequality.

Applying Lemma 17 and (104), we have

$$\begin{aligned}\|\widehat{\boldsymbol{M}}-\boldsymbol{\Pi}\boldsymbol{M}\|_2 &\leq \sigma_{\max}(\boldsymbol{M})\left(3\sqrt{K}\widetilde{\kappa}(\boldsymbol{M})\left(\frac{\varepsilon}{\alpha_{\min}}+\zeta\right)\sqrt{2K}\alpha_{\max}+2\sqrt{2K}\gamma\zeta\right)\\ &= \sigma_{\max}(\boldsymbol{M})\left(3\sqrt{2}K\widetilde{\kappa}(\boldsymbol{M})\gamma\varepsilon+3\sqrt{2}K\widetilde{\kappa}(\boldsymbol{M})\alpha_{\max}\zeta+2\sqrt{2K}\gamma\zeta\right)\\ &\leq \sigma_{\max}(\boldsymbol{M})\left(3\sqrt{2}K\widetilde{\kappa}(\boldsymbol{M})\gamma\varepsilon+4\sqrt{2}K\widetilde{\kappa}(\boldsymbol{M})\gamma\zeta\right)\\ &= \sigma_{\max}(\boldsymbol{M})\sqrt{2}K\gamma\widetilde{\kappa}(\boldsymbol{M})\left(3\varepsilon+4\zeta\right),\end{aligned}$$

where $\widetilde{\kappa}(\boldsymbol{M}) = 1+160K\gamma^2$.

L Successive Projection Algorithm (SPA)

In this section, we detail the successive projection algorithm (SPA) [35, 45, 66] presented in Algorithm 3.

Assume that the separability condition holds. For the model $\boldsymbol{U}^\top = \boldsymbol{G}^\top\boldsymbol{M} \in \mathbb{R}^{K\times N}$, SPA estimates the anchor nodes $\boldsymbol{\Lambda} = \{q_1,\ldots,q_K\}$ through a series of steps as follows:

$$q_1 = \arg\max_{n\in[N]} \ \|\boldsymbol{U}(n,:)\|_2^2, \tag{105a}$$

$$q_k = \arg\max_{n\in[N]} \ \left\|\boldsymbol{P}^{\perp}_{\overline{\boldsymbol{G}}(:,1:k-1)}\boldsymbol{U}(n,:)\right\|_2^2, \ k>1, \tag{105b}$$

where $\overline{\boldsymbol{G}}(:,1:k-1) = [\boldsymbol{U}^\top(:,q_1),\ldots,\boldsymbol{U}^\top(:,q_{k-1})]$ and $\boldsymbol{P}^{\perp}_{\overline{\boldsymbol{G}}(:,1:k-1)}$ is a projector onto the orthogonal complement of $\mathrm{range}(\overline{\boldsymbol{G}}(:,1:k-1))$.

Eq. (105a) indicates that the first anchor node q_1 can be identified by finding the index of the row of $\boldsymbol{U}$ that has the maximum ℓ_2 norm. This can be understood by considering the below for any $n\in[N]$,

$$\|\boldsymbol{U}(n,:)\|_2 = \left\|\sum_{k=1}^{K}\boldsymbol{G}(k,:)\boldsymbol{M}(k,n)\right\|_2 \leq \sum_{k=1}^{K}\|\boldsymbol{G}(k,:)\boldsymbol{M}(k,n)\|_2$$
$$= \sum_{k=1}^{K}\boldsymbol{M}(k,n)\,\|\boldsymbol{G}(k,:)\|_2 \leq \max_{k=1,\ldots,K}\|\boldsymbol{G}(k,:)\|_2,$$

where the first inequality is due to triangle inequality, the second equality is due to the non-negativity of $\boldsymbol{M}$ and the last inequality is by the sum-to-one condition on the rows of $\boldsymbol{M}$. In the above, all the equalities hold simultaneously if and only if $\boldsymbol{M}(:,n) = \boldsymbol{e}_k^\top$ for a certain k, i.e., $n\in\{q_1,\ldots,q_K\}$.

Algorithm 3 Successive Projection Algorithm (SPA) [35]

Require: $\boldsymbol{U}, K$

Initialize $\widetilde{\boldsymbol{U}} \leftarrow \boldsymbol{U}$;
for each $k = 1 : K$ **do**
  $q_k \leftarrow \arg\max_{n \in [N]} \ \|\widetilde{\boldsymbol{U}}(n,:)\|_2^2$; ▷ Select anchor node
  $\boldsymbol{u}_k \leftarrow \widetilde{\boldsymbol{U}}(q_k,:)$;
  $\widetilde{\boldsymbol{U}} \leftarrow \widetilde{\boldsymbol{U}} \left(\boldsymbol{I}_K - \frac{\boldsymbol{u}_k^\top \boldsymbol{u}_k}{\|\boldsymbol{u}_k\|_2} \right)$;
end for each
$\boldsymbol{G} \leftarrow \boldsymbol{U}(\{q_1, \ldots, q_K\}, :)$;
return $\boldsymbol{G}$.

After identifying the first anchor node q_1, all the rows of $\boldsymbol{U}$ are projected onto the orthogonal complement of $\boldsymbol{U}(q_1,:)$ such that the same index q_1 will not show up in the subsequent steps. From the projected columns, the next anchor node is then identified in a similar way as shown in Eq. (105b). Following this procedure, SPA can identify the anchor node set $\boldsymbol{\Lambda}$ in K steps.

The work in [35] showed that under the following noisy model

$$\widehat{\boldsymbol{U}}^\top = \boldsymbol{G}^\top \boldsymbol{M} + \boldsymbol{N}, \ \boldsymbol{M} \geq \boldsymbol{0}, \ \boldsymbol{1}^\top \boldsymbol{M} = \boldsymbol{1}^\top, \tag{106}$$

where $\widehat{\boldsymbol{U}}$ is the noisy estimate of $\boldsymbol{U}$, SPA can still provably identify $\boldsymbol{M}$ under the separability condition if the noise $\boldsymbol{N}$ has bounded norm (see Theorem 16).

M Permutation Fixing Procedure for Baseline Algorithms

For synthetic and real data experiments, we employ the procedure in Algorithm 4 for the baseline algorithms to align the permutations of the estimated $\widehat{\boldsymbol{M}}_\ell$ over different blocks. Note that this aligning procedure is not from any of the baseline papers, but is only presented for benchmarking the baselines under the diagonal EQP pattern. To explain this procedure, let us consider the diagonal EQP pattern as shown in Fig. 6 (a toy example with $L = 3$). In order to learn the membership matrix $\boldsymbol{M} = [\boldsymbol{M}_1, \boldsymbol{M}_2, \boldsymbol{M}_3]$ using the baselines, we may run the baseline algorithm first on $\boldsymbol{A}_{1,2}$ and subsequently on $\boldsymbol{A}_{2,3}$ to obtain the below (assuming no estimation error):

$$[\boldsymbol{M}_1, \boldsymbol{M}_2] = \texttt{BaselineAlgorithm}(\boldsymbol{A}_{1,2})$$
$$[\overline{\boldsymbol{M}}_2, \overline{\boldsymbol{M}}_3] = \texttt{BaselineAlgorithm}(\boldsymbol{A}_{2,3})$$

where $\overline{\boldsymbol{M}}_2 = \boldsymbol{\Pi}\boldsymbol{M}_2$ and $\overline{\boldsymbol{M}}_3 = \boldsymbol{\Pi}\boldsymbol{M}_3$. Here, $\boldsymbol{\Pi} \in \{0,1\}^{K \times K}$ is a row permutation matrix which is unknown. In order to learn the complete membership matrix, i.e., $\boldsymbol{M} = [\boldsymbol{M}_1, \boldsymbol{M}_2, \boldsymbol{M}_3]$, we need to identify this row permutation matrix $\boldsymbol{\Pi}$. This can be achieved by using any permutation fixing algorithms such as Hungarian algorithm [67] by inputting $\boldsymbol{M}_2$ and $\overline{\boldsymbol{M}}_2 (= \boldsymbol{\Pi}\boldsymbol{M}_2)$. Algorithm 4 uses this idea in an iterative fashion to learn $\boldsymbol{M}$ using the baseline algorithms for general L case.

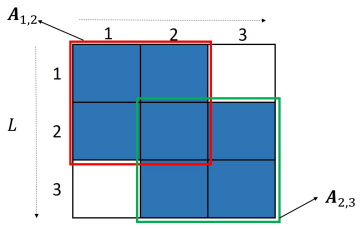

Figure 6: Adjacency matrix observed via diagonal EQP pattern ($L = 3$). The blue shaded region represents the observed blocks in the adjacency matrix.

In Algorithm 4, `BaselineAlgorithm` can be any graph clustering algorithm, for e.g., `GeoNMF`, `SPACL`, `CD-MVSI`, `CD-BNMF`, `SC (unnorm.)`, `SC (norm.)`, k-`means` etc.

Algorithm 4 `Permutation Fixing`

Require: L,K,$\{\boldsymbol{A}_{\ell,\ell}\}_{\ell=1}^{L}$, $\{\boldsymbol{A}_{\ell,\ell+1}\}_{\ell=1}^{L-1}$, `BaselineAlgorithm`.

$\boldsymbol{\Pi} \leftarrow \boldsymbol{I}_K$;
for each $\ell = 1$ **to** $L-1$ **do**
  $\boldsymbol{H}_\ell \leftarrow \begin{bmatrix} \boldsymbol{A}_{\ell,\ell} \ , \ \boldsymbol{A}_{\ell,\ell+1} \\ \boldsymbol{A}_{\ell,\ell+1}^\top \ , \ \boldsymbol{A}_{\ell+1,\ell+1} \end{bmatrix}$;
  $[\overline{\boldsymbol{M}}_\ell, \overline{\boldsymbol{M}}_{\ell+1}] \leftarrow$ `BaselineAlgorithm`($\boldsymbol{H}_\ell$,K);
  if $\ell > 1$ **then**
    $\boldsymbol{\Pi} \leftarrow$ `HugarianAlgorithm`($\overline{\boldsymbol{M}}_\ell$,$\boldsymbol{M}_\ell^{(prev)}$) [67];
  end if
  $[\boldsymbol{M}_\ell, \boldsymbol{M}_{\ell+1}] \leftarrow \boldsymbol{\Pi}^\top [\overline{\boldsymbol{M}}_\ell, \overline{\boldsymbol{M}}_{\ell+1}]$;
  $\boldsymbol{M}_\ell^{(prev)} \leftarrow \boldsymbol{M}_{\ell+1}$;
end for each
$\widehat{\boldsymbol{M}} = [\boldsymbol{M}_1, \ldots, \boldsymbol{M}_L]$;
return $\widehat{\boldsymbol{M}}$.

Table 10: $K = 5$, $L = 10$, $\boldsymbol{\nu} = [\frac{1}{5}\frac{1}{2}1\frac{1}{2}\frac{1}{5}]^\top$ and $\eta = 0.1$

Graph Size	Metric	`BeQuec`	`GeoNMF`	`CD-MVSI`	`CD-BNMF`
2000	MSE	**0.5992**	0.8277	0.7403	0.6893
	RE	**0.5378**	0.5695	0.8282	0.7194
	SRC	**0.4452**	0.2807	0.2211	0.3074
4000	MSE	**0.3123**	0.3704	0.7325	0.6662
	RE	**0.6213**	0.8930	0.8127	0.7245
	SRC	**0.5908**	0.5855	0.2165	0.3204
8000	MSE	**0.1804**	0.1859	0.7090	0.6743
	RE	**0.3021**	0.3597	0.8215	0.7251
	SRC	**0.7465**	0.7417	0.2376	0.3019
10000	MSE	0.1454	**0.1435**	0.7023	0.6626
	RE	**0.2865**	0.3580	0.8198	0.7210
	SRC	**0.7715**	0.7585	0.2490	0.3110

Table 11: $K = 5$, $L = 10$, $\boldsymbol{\nu} = [\frac{1}{5}\frac{1}{3}\frac{1}{2}\frac{1}{5}1]^\top$ and $\eta = 0.2$

Graph Size (N)	Metric	BeQuec	GeoNMF	CD-MVSI	CD-BNMF
2000	MSE	**0.7742**	0.8266	0.8793	0.8217
	RE	**0.7198**	0.8494	0.8237	0.8104
	SRC	**0.2761**	0.2737	0.1770	0.1669
4000	MSE	**0.5280**	0.3613	0.8810	0.8320
	RE	**0.5868**	0.5263	0.8336	0.8161
	SRC	**0.4894**	0.5496	0.1655	0.1431
8000	MSE	**0.1733**	0.2082	0.8875	0.8313
	RE	**0.3178**	0.3829	0.8439	0.8170
	SRC	**0.6886**	0.6618	0.1575	0.1329
10000	MSE	**0.1385**	0.2794	0.8861	0.8344
	RE	**0.2869**	0.4508	0.8444	0.8184
	SRC	**0.7124**	0.6216	0.1517	0.1245

Table 12: $K = 5$, $L = 10$, $\boldsymbol{\nu} = [\frac{1}{5}\frac{1}{3}\frac{1}{2}\frac{1}{5}1]^\top$ and $\eta = 0.3$

Graph Size (N)	Metric	BeQuec	GeoNMF	CD-MVSI	CD-BNMF
2000	MSE	**0.7838**	0.9181	0.8811	0.8494
	RE	**0.7136**	0.8997	0.8135	0.8182
	SRC	**0.2991**	0.2246	0.1634	0.1281
4000	MSE	**0.5952**	0.4236	0.8790	0.8403
	RE	**0.5970**	0.6097	0.8274	0.8159
	SRC	**0.4367**	0.5100	0.1537	0.1254
8000	MSE	**0.2748**	0.3225	0.8643	0.8467
	RE	**0.4157**	0.5030	0.8181	0.8211
	SRC	**0.6332**	0.5834	0.1607	0.1025
10000	MSE	**0.2005**	0.2770	0.8728	0.8456
	RE	**0.3385**	0.4903	0.8330	0.8214
	SRC	**0.6820**	0.6131	0.1548	0.1006

N Additional Synthetic Data Experiments

In this section, we present additional synthetic data experiments.

Tables 10-12 present the results averaged over 20 random trials for different $\boldsymbol{\nu}$'s (i.e., different Dirichlet parameters) and η's. We let entries of $\boldsymbol{\nu}$ to be different, which simulates the scenario where unbalanced clusters are present. The values of η are varied to test different levels of interactions among the clusters. In these tables, we follow the same diagonal EQP pattern as in the manuscript. Note that in most of the cases, the proposed method outperforms the baselines. One can also see that all methods' performance deteriorates when η increases from 0.1 to 0.3. Nonetheless, the proposed method keeps its margin against the baselines under all η's.