Consensus Knowledge Graph Learning via Multi-view Sparse Low-rank Block Model

Tianxi Cai\textsuperscript{1,2}, Dong Xia\textsuperscript{3}, Luwan Zhang\textsuperscript{1} and Doudou Zhou\textsuperscript{4}

\textsuperscript{1}Department of Biostatistics, Harvard T.H. Chan School of Public Health, Boston MA
\textsuperscript{2}Department of Biomedical Informatics, Harvard Medical School, Boston MA
\textsuperscript{3}Department of Mathematics, Hong Kong University of Science and Technology, Kowloon, Hong Kong
\textsuperscript{4}Department of Statistics and Data Science, National University of Singapore

* alphabetical order

Abstract

Network analysis has been a powerful tool to unveil relationships and interactions among a large number of objects. Yet its effectiveness in accurately identifying important node-node interactions is challenged by the rapidly growing network size, with data being collected at an unprecedented granularity and scale. Common wisdom to overcome such high dimensionality is collapsing nodes into smaller groups and conducting connectivity analysis on the group level. Dividing efforts into two phases inevitably opens a gap in consistency and drives down efficiency. Consensus learning emerges as a new normal for common knowledge discovery with multiple data sources available. In this paper, we propose a unified multi-view sparse low-rank block model (msLBM) framework, which enables simultaneous grouping and connectivity analysis by combining multiple data sources. The msLBM framework efficiently represents overlapping information across large scale concepts and accommodates different types of heterogeneity across sources. Both features are desirable when analyzing high dimensional electronic health record (EHR) datasets from multiple health systems. An estimating procedure based on the alternating minimization algorithm is proposed. Our theoretical results demonstrate that a consensus knowledge graph can be more accurately learned by leveraging multi-source datasets, and statistically optimal rates can be achieved under mild conditions. Applications to the real world EHR data suggest that our proposed msLBM algorithm can more reliably reveal network structure among clinical concepts by effectively combining summary level EHR data from multiple health systems.

Key words: multi-view, consensus network, low-rank block model, sparse random effect
1 Introduction

Network analysis that unveils connectivity and interactions among a large number of objects is a problem of great importance with wide applications in social sciences, genomics, clinical medicine, and beyond (Goh et al., 2007; Nabieva et al., 2005; Luscombe et al., 2004; Scott, 1988, e.g.). As data are being collected at an unprecedented granularity and scale, it is now possible to study the structure of large networks. However, it is challenging to accurately infer the network structure in the presence of high dimensionality, especially when many nodes represent highly similar entities and multiple data sources are available. A simple approach to overcome the high dimensionality and overlapping entities is to collapse similar nodes into groups. With groupings given as a priori, network connectivity analysis is subsequently performed on the group level to improve interpretability and reproducibility. For example, inferences for gene regulatory networks in genomics are often made on the pathway level that generally represents a group of functionally related genes (Kelley and Ideker, 2005; Xia et al., 2018, e.g.). Brain function network analyses are often performed on groups of voxels localized within a small region having a common neurological function (Shaw et al., 2007; Chen et al., 2017; Lu et al., 2017, e.g.). However, for many applications, the group structure is unknown and needs to be estimated together with the network structure. In natural language processing (NLP), synonymous terms should be grouped yet such grouping structure varies by context and is not generally available. In association studies linking current procedural terminology (CPT) codes to clinical outcomes, many procedures are clinically equivalent yet the currently available grouping of CPT codes is extremely crude (Agency for Healthcare Research and Quality, 2019).

Despite the potentially large sample size, network structure inferred from a single data source can be influenced by the characteristics or generation process of the data itself. As more data sources become available, it is highly desirable to synthesize information from multiple sources, often termed as views, to jointly infer about a consensus network structure. The network structure achieved through consensus may differ from those derived from individual data sources, as each source might exhibit unique mechanisms or patterns specific to it, potentially leading to biases. This debiasing for common knowledge discovery is particularly important when dealing with inherently heterogeneous data sources. A prime example comes from knowledge extraction using Electronic Health Records (EHR) data. The EHR system contains rich longitudinal phenotypic information from millions of patients. The EHR data is a valuable source for learning medical knowledge networks linking each specific disease condition with co-morbidities, diagnostic laboratory measurements, procedures, and treatment. Constructing a consensus network using data from multiple EHR systems could potentially remove bias due to different patient populations, physician training, and practices, as well as how or when the encodings are performed. However, the between-view heterogeneity also imposes methodological challenges to accurately learning network structure.

To overcome these challenges, we propose in this paper a unified framework that can efficiently combine multi-view data to simultaneously group entities and infer about network structure. We
study methods for integrating several adjacency matrices from different views. Our underlying assumption is that after debiasing and standardizing each matrix to have unit diagonals, they converge toward a common latent correlation matrix \( C \). The primary objective is to uncover a latent group structure within \( C \). The complexity of this task arises from the variability in the unknown standardization weights and the noise levels present in the observed adjacency matrices, which can differ across views. We refer to this variability as ‘heterogeneity’. Our proposed estimation algorithm addresses this heterogeneity by identifying the signal-to-noise ratio specific to each view. This approach is distinct from conventional methods that merge information across views before determining the structure of \( C \). Specifically, we aim to learn a consensus network that reflects shared knowledge using a collection of \( m \) independently-observed graphs on \( n \) common vertex set \( \mathcal{V} = \{v_j\}_{j \in [n]} \),

\[
\mathcal{G}_{[m]} = \left\{ \mathcal{G}_s = \{ \mathcal{V}, W_s \} : \mathcal{V} = \{v_j\}_{j \in [n]}, \ W_s = [W_s(j_1, j_2)]_{j_1, j_2 \in [n]} \right\}_{s \in [m]},
\]

where \([n] = \{1, \ldots, n\}\), and \( W_s(j_1, j_2) \in \mathbb{R} \) is the observed edge weight between node \( v_{j_1} \) and node \( v_{j_2} \) from the \( s \)th view. The vertex set \( \mathcal{V} \) admits a latent grouping structure shared across \( m \) views in that there exists a unique non-overlapping \( K \)-partition:

\[
\mathcal{V} = \bigcup_{k=1}^K \mathcal{V}_k, \mathcal{V}_k \cap \mathcal{V}_l = \emptyset, \forall 1 \leq k < l \leq K,
\]

which can be equivalently represented by a 0/1 matrix \( Z = [Z(j, k)]_{j \in [n]}^{k \in [K]} \), where \( Z(j, k) = I(v_j \in \mathcal{V}_k) \). To model the network structure while accommodating heterogeneity across views, we assume a flexible multi-view sparse low-rank model for \( \{W_s\}_{s=1}^m \). This model utilizes a shared sparse consensus matrix \( C \), which encapsulates the common network structure across all views. Simultaneously, it accommodates heterogeneity by varying the degrees of nodes to reflect distinct characteristics of each view and integrates sparse differences between views to capture the unique biases inherent to each one. The consensus matrix \( C \) can be further decomposed into \( C = Z\Omega Z^T \), where \( \Omega \) is a group-level sparse and low-rank weight matrix. Our goal is to simultaneously learn \( Z \) and \( C \) from \( \mathcal{G}_{[m]} \) in the presence of heterogeneity.

The proposed framework is particularly appealing for knowledge graph modeling with multi-view data for several reasons. To illustrate this, consider our motivating example of knowledge extraction with multi-view EHR data where the nodes represent clinical concepts including disease conditions, signs/symptoms, diagnostic laboratory tests, procedures, and treatments. First, nodes within a group can effectively represent stochastically equivalent and interchangeable medical terms. For example, the clinical concepts “coronary artery disease” and “coronary heart disease” are used interchangeably by physicians but are mapped to two separate clinical concept unique identifies in the unified medical language system (UMLS) (Bodenreider, 2004). Second, the structure of the group-level dependency captured by \( \Omega \) can be used to infer clinical knowledge about a disease. The
consensus graph is particularly appealing as it removes biases from individual healthcare systems. Third, the improved estimation of the low-rank weight matrix $\Omega$ and $Z$ through consensus learning also leads to a more accurate embedding representation for the nodes.

In a special case where $\Omega$ is full-rank and each entry is non-negative and upper bounded by 1, $C$ reduces to the well-known stochastic block model (SBM) (Holland et al., 1983) since $C$ characterizes the underlying Bernoulli distribution for entries in the observed adjacency matrix. More broadly, recovering $\Omega$ shows a direct effort tapping into the network dynamics. For example, in clinical practices, complex diseases are often accompanied by a series of symptoms that may need multiple concurrent treatments. Therefore, learning a knowledge network of disease would greatly help support decision-making toward precision medicine. Lastly, the decomposition on $C$ embodies efficient vector representations that enable groups of node embeddings. This provides a new embedding technique applicable in many areas, such as proteins, DNA sequences, and fMRI, to expand their existing embedding family serving broader research needs (Asgari and Mofrad, 2015; Nguyen et al., 2016; Choi et al., 2016; Ng, 2017; Vodrahalli et al., 2018, e.g.).

With a single view, a simple approach to achieve this goal is to first perform grouping based on scalable clustering algorithms (Shi and Malik, 2000; Ng et al., 2002; Newman, 2006; Bickel and Chen, 2009; Zhao et al., 2012, e.g.) and then learn the network structure. However, dividing efforts into two phases inevitably opens a gap that could potentially create friction in consistency and drive down efficiency in transmitting information. For the sole purpose of recovering $Z$, SBM is perhaps one of the most developed frameworks that enjoys both straightforward interpretations and good statistical properties (Rohe et al., 2011; Lei and Rinaldo, 2015; Nielsen and Witten, 2018; Abbe, 2018; Gao et al., 2017, e.g.). Central to SBM is the idea that the observed adjacency matrix is a noisy version of a rank-$K$ matrix with eigenvectors having exactly $K$ unique rows. Each unique row can be comprehended as a $K$-dimensional vector representation for nodes in that group. However, this assumption becomes too restrictive requiring the embedding dimension $r$ to be tied to the number of groups $K$. As $K$ grows (potentially with $n$), the embedding dimension desirably remains low. To this end, we generalize SBM by introducing a low-rank block model (LBM) on $C$ to allow $\Omega$ to be low-rank, thus decoupling $r$ from $K$. Extending to a multi-view setting, we further introduce a multi-view sparse low-rank block model (msLBM) to jointly model the faithfulness to the consensus and view-specific varying parts. One theoretical contribution in our proposed msLBM model is to perform a low-rank and sparse matrix decomposition with overlapping subspace on multiple noisy data sources.

Recent years have witnessed a fast-growing literature on multi-layer network analysis (Levin et al., 2017; Le et al., 2018; Tang et al., 2017; Wang et al., 2019; Jones and Rubin-Delanchy, 2020; Paul et al., 2020; Lei et al., 2020; Jing et al., 2021; Arroyo et al., 2021; Levin et al., 2022, e.g.). For example, Arroyo et al. (2021) considered multiple random dot product graphs sharing a common invariant subspace and Wang et al. (2019) decomposed the logistic-transformed multi-view expected
adjacency matrices to a common part and individual low-rank matrices. Levin et al. (2022) proposed the weighted adjacency spectral embedding under the assumption that multi-layer networks share a common connectivity probability with a potential low-rank structure, while the noise distributions of different layers can be heterogeneous. Recently, MacDonald et al. (2022) proposed a latent space multiplex networks model in which part of the latent representation is shared across all layers while heterogeneity is allowed for the other part.

Our msLBM model differs from those prior works in three crucial aspects. First, our msLBM allows node-wise heterogeneity on the consensus graph of each view/layer while the existing literature assumes SBM on each layer. Secondly, what is more important, our method allows view-wise heterogeneity on the consensus graph across different views/layers. This additional flexibility enables us to deal with heterogeneous data collected from different sources. Finally, our msLBM model introduces an additional sparse signal on each view/layer which is unexplainable by the low-rank consensus graph. Oftentimes, these sparse signals can capture uncommon network structures in each view/layer. These new ingredients in msLBM are motivated by the uniqueness of multi-view EHR data. Meanwhile, all these differences also make it more challenging to estimate the underlying consensus graph in our msLBM.

The rest of the paper is organized as follows. In Section 2, we elaborate in more detail on the proposed low-rank block model and its extension accounting for heterogeneity arising in a multi-view setting. We then propose an alternating minimization-based approach in Section 3 to learn the consensus network that is easy and fast to implement in practice. Section 4 provides all theoretical justifications. Simulations are given in Section 5 to demonstrate the efficacy and robustness of the proposed method. In Section 6, we apply the proposed method to generate a new set of clinical concept embeddings and yield a very insightful Disease-Symptom-Treatment network on Coronary Artery Disease, by integrating information from a large digital repository of journal articles and three healthcare systems. Proofs on theories in Section 4 are relegated to Appendix.

2 Multi-view Sparse Low-rank Block Model

2.1 Notations

Throughout, we use a boldfaced uppercase letter to denote a matrix and the same uppercase letter in normal font to represent its entries. We use a boldfaced lowercase letter to denote a vector and the same lowercase letter in normal font to represent its entries. Let $I_r$ denote the $r \times r$ identity matrix and $1_n$ denote the $n$-dimensional all-one vector. We let $\|\cdot\|_{\ell_2}$ denote vector $\ell_2$-norm. For any matrix $A$, let $\|A\|_r, \|A\|_F$ denote its spectral norm and Frobenius norm respectively, $\|A\|_{\ell_1} = \sum_{i,j} |A_{i,j}|$, $\|A\|_{\ell_\infty} = \max_{i,j} |A_{i,j}|$, $\lambda_j(A)$ denote its $j$th largest singular value, $A_i$ and $A_j$ respectively denote its $i$th row and $j$th column, $\text{Vec}(A)$ denote vectorizing $A$ column by column, $\kappa(A) = \lambda_1(A)/\lambda_{\text{rank}(A)}(A)$.
denote the condition number of $A$. For a set $V$, we use $\text{Card}(V)$ to denote its cardinality. We denote the set of $r \times r$ orthonormal matrices by

$$O_{r,r} := \{ O \in \mathbb{R}^{r \times r} : OO^\top = O^\top O = I_r \}.$$ 

If the entries of an orthonormal matrix $O \in O_{r,r}$ are either 0 or 1 such that each row and column contains one single nonzero entry, then we call $O$ a permutation matrix. The set of all $r \times r$ permutation matrices is denoted by $\mathcal{P}_{r,r} \subset O_{r,r}$. With slight abuse of notations, we denote the set of $n \times K$ matrices with orthonormal columns by

$$O_{n,K} := \{ X \in \mathbb{R}^{n \times K} : X^\top X = I_K \}.$$ 

Given any matrix $A \in \mathbb{R}^{n \times r}$ with rank$(A) = r$, let $\mathcal{P}_A$ denote the orthogonal projection from $\mathbb{R}^n$ to the column space of $A$, or more specifically $\mathcal{P}_A(v) = A(A^\top A)^{-1}A^\top v$ for any $v \in \mathbb{R}^n$. We denote $e_j$ the $j$-th canonical basis vector, whose dimension might change at different appearances. Denote $A \circ B$ the Hadamard product of matrix $A$ and $B$, i.e., $(A \circ B)_{ij} = A_{ij}B_{ij}$.

### 2.2 Low-rank Block Model (LBM)

We first introduce the graph model for a single view. Let $G = \{V, W\}$ denote an undirected weighted graph with vertex set $V = \{v_j\}_{j \in [n]}$ and symmetric weight matrix $W = [W(j_1, j_2)]_{j_1 \in [n], j_2 \in [n]}$ with $W(j_1, j_2) \in \mathbb{R}$ representing the connection intensity between the vertices $v_{j_1}$ and $v_{j_2}$. We assume that the graph $G$ admits a latent network structure in the sense that there exists a unique (unknown) non-overlapping $K$-partition of the vertex set $V$:

$$V = \bigcup_{k=1}^{K} V_k, \quad \text{Card}(V_k) = n_k, \quad V_k \cap V_l = \emptyset, \quad \forall 1 \leq k < l \leq K,$$

with $n = \sum_{k=1}^{K} n_k$, which can be equivalently represented by a group membership matrix

$$Z = [Z(j,k)]_{j \in [n], k \in [K]}, \quad \text{where} \quad Z(j,k) = 1(v_j \in V_k) \in \{0, 1\}.$$ 

We denote by $\mathcal{Z}_{n,K}$ the set of all possible $n \times K$ dimensional $K$-group membership matrices for $n$ nodes. Throughout, we assume $K$ is known that can grow with $n$ for high-dimensional cases for all theoretical analyses. Strategies for choosing an appropriate $K$ would be discussed in Section 6. The set of orthonormalized membership matrices is denoted by

$$\mathcal{U}_{n,K} = \left\{ U := Z \left[ \text{diag}(1_n^\top Z) \right]^{1/2} : Z \in \mathcal{Z}_{n,K} \right\} \subset O_{n,K}.$$
We assume the observed edge weight matrix $W$ can be decomposed as

$$W = HZ\Omega Z^\top H + E := HCH + E := L + E,$$

(1)

where the symmetric matrix $\Omega \in \mathbb{R}^{K \times K}$ is the group-level correlation matrix that measures the strength of connectivity between groups with diagonal entries being 1 and other entries bounded by 1, the diagonal matrix $H = \text{diag}(h_1, \cdots, h_n)$ is the degree parameter that indexes the information contained by each node, and $E$ represents the sampling error. In a special case where each entry of $\Omega$ is non-negative, and $H$’s entries are upper bounded by 1, model (1) reduces to the degree corrected stochastic block model (DCBM) (Karrer and Newman, 2011) under which $W(j_1, j_2)$ represents the probability of $v_{j_1}$ and $v_{j_2}$ being connected, and the matrix $\Omega$ is assumed to be full-rank to recover $Z$ for community detection. However, this full-rank assumption is inappropriate for knowledge graph modeling where $K$ is often large but $\Omega$ is low rank. We instead assume the following a low-rank block model (LBM) as a generalization of DCBM.

**Assumption 1 (LBM).** The graph $G = (V, W)$ satisfies (1) with $r := \text{rank}(\Omega) \leq K < n$ and $\Omega$ is a positive semi-definite matrix.

**Remark 1.** The low-rank assumption on $\Omega$ has been previously explored in latent space models, as reviewed by Athreya et al. (2018). These studies allow for distinguishing between the dimensions of embeddings and the number of clusters within a network. Notably, Tang et al. (2022) investigated the effectiveness of spectral estimators within this framework. Our model, however, diverges from these earlier approaches by not relying on the independent edge assumption (Athreya et al., 2018) and by accommodating weighted edges. Drawing inspiration from the application of learning clinical knowledge graphs, as detailed in Section 7, we adopt the assumption that the group correlation matrix $\Omega$ is positive semi-definite and symmetric. As further discussed in Section 7, this assumption is imposed simply for presentation clearness. Our methods can be easily adapted for asymmetric matrix $\Omega$, and similar theoretical results continue to hold.

Due to the additional heterogeneity $H$, the correlation matrix $C$ and the expected weight matrix $L$ can have drastically different eigenstructures. By definition and Assumption 1, there exists a matrix $U \in \mathbb{R}^{n \times r}$ such that $C = UU^\top$. Since the diagonal entries of $C$ are all ones, we have $\|e_j^\top U\| = 1$ for all $j \in [n]$. The row-wise separability of $U$ is immediately guaranteed by Lemma 1. We note that the columns of $U$ are not orthonormal.

**Lemma 1.** Under Assumption 1, for any $1 \leq k_1 < k_2 \leq K$ and $j_1 \in V_{k_1}$ and $j_2 \in V_{k_2}$, we have

$$\|U(j_1, :) - U(j_2, :)\|_{\ell_2} \geq \delta_\Omega \cdot \frac{\min_{k \in [K]} n_k^{1/2}}{\lambda_1^{1/2}(C)},$$

where $\delta_\Omega := \min_{1 \leq k_1 < k_2 \leq K} \|\Omega(k_1, :) - \Omega(k_2, :)\|_{\ell_2} > 0$.  

7
Remark 2. The condition $\delta_{\Omega} > 0$ is necessary since otherwise there exist $k_1, k_2 \in [K]$ such that $\Omega(k_1,:) = \Omega(k_2,:)$ implying that there are no differences between the $k_1$-th group and $k_2$-th group. In that case, it is more reasonable to merge these two groups.

We now study the eigenvectors of the matrix $L$. Recall that

$$L = HZ\Omega Z^\top H = (HZN^{-1}_H) \cdot (N_H\Omega N_H) \cdot (HZN^{-1}_H)^\top$$

where the diagonal matrix $N_H$ is defined by

$$N_H = \text{diag}(\tilde{h}_1, \tilde{h}_2, \ldots, \tilde{h}_K) \quad \text{where} \quad \tilde{h}_k^2 = \sum_{j \in V_k} h_j^2 \quad \text{for} \quad k = 1, \ldots, K.$$  \hspace{1cm} (2)

By definition, the matrix $HZN^{-1}_H$ has orthonormal columns in that $(HZN^{-1}_H)^\top (HZN^{-1}_H) = I_K$. Now, under Assumption 1, consider the eigendecomposition of $N_H \Omega N_H = VD \tilde{V}^\top$ with the eigenvectors $V \in \mathbb{R}^{K \times r}$ having orthonormal columns corresponding to the eigenvalues $D = \text{diag}(d_1, \ldots, d_r)$ where $d_1 = \lambda_1(D) \geq \cdots \geq d_r = \lambda_r(D) > 0$. Then we may obtain the eigendecomposition of $L$ as

$$L = \tilde{U}D\tilde{U}^\top, \quad \text{where} \quad \tilde{U} = HZN^{-1}_H V.$$  \hspace{1cm} (3)

The row-wise separability of $\tilde{U}$ is given by Lemma 2.

Lemma 2. Under Assumption 1, for any $1 \leq k_1 < k_2 \leq K$ and $j_1 \in V_{k_1}$ and $j_2 \in V_{k_2}$, we have

$$\|U(j_1,:) - U(j_2,:}\|_{\ell_2} \geq \frac{\delta_{H,\Omega}}{\lambda_1(\Omega)} \frac{\tilde{h}_{\min}}{\tilde{h}_{\max}},$$

where $\delta_{H,\Omega} := \min_{1 \leq j_1 < j_2 \leq n} \| (e_{j_1} - e_{j_2})^\top H\Omega \|_{\ell_2}$, $\tilde{h}_{\min} = \min_k \tilde{h}_k$, and $\tilde{h}_{\max} = \max_k \tilde{h}_k$.

The row separability of $\tilde{U}$ is less explicit compared to $U$. In addition to depending on the membership matrix $Z$, it depends on the row separability of $\Omega$ and the heterogeneity matrix $H$. While $\delta_{\Omega} > 0$ of Lemma 1 seems natural, the condition $\delta_{H,\Omega} > 0$ of Lemma 2 might be untrue. For instance, we have $\delta_{H,\Omega} = 0$ if $h_{j_1,\Omega(k_1,:)} = h_{j_2,\Omega(k_2,:)}$ for two vertices $j_1 \in V_{k_1}$ and $j_2 \in V_{k_2}$ in which case the vertices $j_1$ and $j_2$ are indistinguishable by the eigenvectors of $L$. Technically speaking, both $U$ and $\tilde{U}$, under reasonable conditions, can be used for the clustering of vertices. Our method directly estimates $U$ from multiple views of LBMs.

Remark 3. The matrix $\Omega$ can be rank deficient under LBMs while the DCBM assumes $\Omega$ to be full rank. Under the DCBM, the eigenspace of $E W$ is equivalent to the eigenspace of $H Z$, which admits a much simpler separability property such that $\|U(j_1,:) - U(j_2,:)\|_{\ell_2} = \sqrt{h_{j_1}^2/\tilde{h}_k^2 + h_{j_2}^2/\tilde{h}_k^2}$, if $j_1 \in V_{k_1}, j_2 \in V_{k_2}$ for $k_1 \neq k_2$. 

8
2.3 Multi-view Sparse LBM (msLBM)

We next describe the msLBM framework for learning a consensus knowledge graph using $m$ observed weighted graphs with a common vertex set $V = \{v_j\}_{j \in [n]}$.

\[
\{G_s = \{V, W_s\} : V = \{v_j\}_{j \in [n]}, W_s = (W_s(j_1, j_2))_{j_1, j_2 \in [n]}\}, \quad (4)
\]

To learn the consensus graph while accounting for the between-view heterogeneity, we propose the following msLBM

\[
W_s = H_s(Z\Omega Z^\top)H_s^\top + \Theta_s + E_s, \quad s = 1, \ldots, m, \quad (5)
\]

where both $\Theta_s \equiv \{\Theta_s\}_{s \in [m]}$ and $\Omega$ are assumed to be sparse, $E_s$ represents the sampling error from the $s$th view. Here $C = Z\Omega Z^\top$ represents the consensus graph as in Section 2.2 while the sparse bias term $\Theta_s$ reflects view-specific patterns, capturing the between view heterogeneity in knowledge graph structure. The view-specific diagonal matrices $H_s = \{H_s\}_{s \in [m]}$ in (5) capture the heterogeneity in the information content for the nodes across views. Additionally, the distributions of the error matrices $\{E_s\}_{s \in [m]}$ can be different for each view, accommodating the heterogeneity in the noise level. Here and in the sequel, we use the subscript "•" to index all views $s \in [m]$.

Rewriting the msLBM model (5) as

\[
W_s = L_s + \Theta_s + E_s \text{ with } L_s = H_s(Z\Omega Z^\top)H_s^\top, \quad s = 1, \ldots, m, \quad (6)
\]

we note that each $W_s$ can be characterized by a noise-corrupted sum of a low-rank matrix and a sparse matrix. The $m$ views $W_s = \{W_s\}_{s=1}^m$ share the common knowledge through the correlation matrix $C = Z\Omega Z^\top$ while the individual varying part goes into the sparse component. In a special case when $m = 1$ (a single noisy matrix decomposition), the model (6) is analogous to the noisy version of robust PCA model (Candès et al., 2011; Zhou and Tao, 2011, e.g.). The msLBM aims to leverage information from multiple resources, which is more challenging.

We impose the following assumption on the noise $E_s$, which implies that its entries have zero means, equal variances, and have sub-Gaussian tails. This sub-Gaussian condition is mild, which easily holds under various special and useful distributions. For instance, in the case that $E_s$ is also sparse but denser than $\Theta_s$ and $\Omega_s$, as shown in Section 6, we can assume $E_s(j_1, j_2) \overset{i.i.d.}{\sim} (1 - \pi_s) \cdot 1_{\{x=0\}} + \pi_s \cdot N(0, \sigma_s^2)$ for a small $\pi_s \in (0, 1)$.

**Assumption 2.** For $s = 1, \ldots, m$, there exists a $\sigma_s > 0$ such that $E_s(j_1, j_2)$ are i.i.d. and

\[
\mathbb{E}E_s(j_1, j_2) = 0, \quad \text{Var}\{E_s(j_1, j_2)\} = \sigma_s^2, \quad \text{and} \quad \mathbb{E}\exp\{t \cdot E_s(j_1, j_2)\} \leq \exp\{t^2 \sigma_s^2\}, \quad \forall \ t \in \mathbb{R}
\]

for all $1 \leq j_1 < j_2 \leq n$.

**Remark 4.** Our analysis primarily centers on scenarios where the model (6) is correctly specified.
Nonetheless, our model can be viewed as a robust alternative to some existing models. Specifically, in instances when $\Theta_s = 0$, our model includes the stochastic block model (Holland et al., 1983), the degree corrected stochastic block model (Karrer and Newman, 2011), and the random dot product graphs (Young and Scheinerman, 2007) as special cases. The inclusion of additional sparse bias parameters $\Theta_s$ in our model offers increased flexibility and adaptability.

Identifiability. It is well recognized in the low-rank plus sparse matrix/tensor literature (Candès et al., 2011; Cai et al., 2022) that the low-rank part $L_s$ and sparse part $\Theta_s$ are not identifiable if $L_s$ is also very sparse. To ensure the identifiability of msLBM, we assume that the column space of $L_s$ is incoherent for all $s = 1, \cdots, m$. A symmetric rank-$r$ matrix $L \in \mathbb{R}^{n \times n}$ with the eigendecomposition of the form $UDU^T$ with $U^\top U = I_r$ is said to be incoherent with constant $\mu_0$ if

$$\max_{1 \leq j \leq n} \| e_j^\top U \| \leq \sqrt{\mu_0 r/n},$$

where $e_j$ denotes the $j$-th canonical basis vector in $\mathbb{R}^n$.

Assumption 3. There exists a $\kappa_0 > 1$ so that $\lambda_1(C)/\lambda_r(C) \leq \kappa_0^{1/2}$ and $\lambda_1(H_s^2)/\lambda_n(H_s^2) \leq \kappa_0^{1/2}$ for all $s \in [m]$.

Basically, Assumption 3 requires the matrices $C$ and $H_s$ for $s \in [m]$ to be well-conditioned. Interestingly, incoherence can be automatically guaranteed by this assumption.

Lemma 3. Under Assumption 3, let $U_s$ be the top-$r$ left eigenvectors of $L_s$, then $U_s$ is incoherent with constant $\kappa_0^2$.

Remark 5. By Lemma 3, the low-rank matrix $L_s$ is incoherent and distinguishable from the sufficiently sparse matrix $\Theta_s$. As shown in Theorem 1, in the noiseless case, if the number of non-zero entries of $\Theta_s$ is smaller than $n$, the low-rank matrix $L_s$ and sparse matrix $\Theta_s$ are distinguishable. By setting $\sigma_s \equiv 0$, Theorem 1 implies that $L_s$ and $\Theta_s$ can be exactly recovered. It is worth pointing out that this condition is sufficient but not necessary. The model may be identifiable even if the cardinality of $\Theta_s \gg n$, a scenario we leave for future exploration. In the presence of the noise $E_s$, we will show in Theorem 1 in Section 4 that $L_s$ can be consistently estimated in the sense that the relative error approaches zero as $n \to \infty$.

3 Multi-view Consensus Graph Learning

To estimate the model parameters under the msLBM (5) with observed $W_\bullet$, we first assume the rank $r$ is known, and discuss the estimation of $r$ later. Denote

$$\mathcal{F}_{n,r,\kappa_1} := \left\{ A \in \mathbb{R}^{n \times r} : \| A(i,:) \| = 1, \forall i \in [n] \text{ and } \lambda_1(A) \leq \kappa_1^{1/4} \lambda_r(A) \right\}$$
the set of all rank-$r$ well-conditioned correlation matrix, and $D_{n,\kappa_1} := \{ H = \text{diag}(H(1,1), ..., H(n,n)) : H(i,i) > 0, \forall i \in [n] \text{ and } \lambda_1(H) \leq \kappa_1^{1/2} \lambda_n(H) \}$. The constraint on condition number enforces incoherent solutions just as implied by Lemma 3. We can treat $\kappa_1$ as a tuning parameter satisfying $\kappa_1 > \kappa_0$. Our algorithm for estimating $Z, \Omega, \Theta, H$ includes two key steps. We first obtain estimates

$$(\hat{U}, \hat{H}_s, \hat{\Theta}_s) := \arg\min_{U \in F_{n,r,\kappa_1}, H_s \in D_{n,\kappa_1}, \Theta_s \in \mathbb{R}^{n \times n}} \mathcal{L}(U, H_s, \Theta_s)$$

with

$$\mathcal{L}(U, H_s, \Theta_s) := \frac{1}{2} \sum_{s=1}^{m} \alpha_s \| W_s - \Theta_s - H_s U U^\top H_s \|_F^2 + \sum_{s=1}^{m} \lambda_s \| \Theta_s \|_{\ell_1}. \quad (7)$$

Here $\hat{\mathbf{U}} \hat{\mathbf{U}}^\top$ is an estimate for $Z\mathbf{\Omega}Z^\top$. In the second step, we recover $Z$ and $\Omega$ based on $\hat{\mathbf{U}}$ via clustering. Here the positive $\alpha_s, \lambda_s$'s are tuning parameters with $\sum_{s=1}^{m} \alpha_s = 1$ and $\| \cdot \|_{\ell_1}$ norm is used to promote sparse solutions for $\{ \Theta_s \}_{s=1}^{m}$. The weights $\alpha_s$ can be chosen to reflect the noise levels in $\{ E_s \}_{s \in [m]}$ and the information content levels $\mathbf{H}_s$. For example, if the noise levels $\sigma_s$ for $s \in [m]$ are known, a natural choice of $\alpha_s$ is $\alpha_s = \sigma_s^{-2} / \sum_{l=1}^{m} \sigma_l^{-2}$, which is optimal as shown in Theorem 1.

The objective function (7) is highly non-convex, which is often solvable only locally. In Section 3.1 an alternating minimization algorithm to optimize for (7) assuming that good initializations $\hat{\mathbf{U}}(0), \hat{\mathbf{H}}(0)$ and $\hat{\mathbf{\Theta}}(0)$ have been obtained. In Section 3.2, we propose a procedure for obtaining a warm start. We detail the clustering algorithm for estimating $Z$ and $\Omega$ in Section 3.3. A data-driven approach for choosing the tuning parameters is discussed in Section 3.4.

### 3.1 Alternative Minimization

Suppose that we obtain reasonably good initializations $\hat{\mathbf{U}}(0), \hat{\mathbf{H}}(0)$ and $\hat{\mathbf{\Theta}}(0)$. In Section 3.2, we shall introduce a computationally efficient method for obtaining these initializations. To solve (7), our algorithm iteratively updates $\hat{\mathbf{U}}, \hat{\mathbf{H}}_s, \hat{\mathbf{\Theta}}_s$ by alternating minimization. The detailed implementations of these iterations are presented in Sections 3.1.1, 3.1.2, and 3.1.3. In Section 3.1.4, we introduce a fast but inexact updating algorithm of $\hat{\mathbf{U}}$ that scales smoothly to large datasets.

#### 3.1.1 Estimate low-rank factor $U$

Suppose that, at $t$-th iteration, provided with $\hat{\mathbf{\Theta}}(t)$ and $\hat{\mathbf{H}}_s(t)$, we update $\hat{\mathbf{U}}(t+1)$ by solving the following minimization problem:

$$\hat{\mathbf{U}}(t+1) = \arg\min_{U \in F_{n,r,\kappa_1}} \sum_{s=1}^{m} \alpha_s \| W_s - \hat{\Theta}_s(t) - \hat{\mathbf{H}}_s(t) U U^\top \hat{\mathbf{H}}_s(t) \|_F^2, \quad (8)$$

Here

$$\hat{\mathbf{U}}(t) = \arg\min_{U \in F_{n,r,\kappa_1}} \sum_{s=1}^{m} \alpha_s \| W_s - \hat{\Theta}_s(t) - \hat{\mathbf{H}}_s(t) U U^\top \hat{\mathbf{H}}_s(t) \|_F^2,$$
which has no closed-form solution. However, problem (8) can be recast to a weighted low-rank approximation problem. Denoting by $\mathbf{h}_s^{(t)} \in \mathbb{R}^n$ the diagonal entries of $\mathbf{H}_s^{(t)}$, we then have

$$\sum_{s=1}^{m} \alpha_s \left\| \mathbf{W}_s - \hat{\Theta}_s^{(t)} - \mathbf{H}_s^{(t)} \mathbf{U} \mathbf{U}^T \hat{H}_s^{(t)} \right\|_F^2$$

$$= \left\langle (\mathbf{U} \mathbf{U}^T) \circ (\mathbf{U} \mathbf{U}^T), \sum_{s=1}^{m} \alpha_s (\mathbf{h}_s^{(t)} \mathbf{h}_s^{(t)\top}) \circ (\mathbf{h}_s^{(t)} \mathbf{h}_s^{(t)\top}) \right\rangle - 2 \left\langle \mathbf{U} \mathbf{U}^T, \sum_{s=1}^{m} \alpha_s (\mathbf{h}_s^{(t)} \mathbf{h}_s^{(t)\top}) \circ (\mathbf{W}_s - \hat{\Theta}_s^{(t)}) \right\rangle.$$

Then

$$\hat{\mathbf{U}}^{(t+1)} = \arg\min_{\mathbf{U} \in \mathcal{F}_{n, r, \kappa_1}} \left\| (\mathbf{U} \mathbf{U}^T) \circ \mathbf{X}^{(t)} - \mathbf{Y}^{(t)} \right\|_F^2,$$

where $\mathbf{X}^{(t)} \in \mathbb{R}^{n \times n}$ and $\mathbf{Y}^{(t)} \in \mathbb{R}^{n \times n}$ satisfy

$$\mathbf{X}^{(t)} \circ \mathbf{X}^{(t)} = \sum_{s=1}^{m} \alpha_s (\mathbf{h}_s^{(t)} \mathbf{h}_s^{(t)\top}) \circ (\mathbf{h}_s^{(t)} \mathbf{h}_s^{(t)\top}), \quad \text{and} \quad \mathbf{X}^{(t)} \circ \mathbf{Y}^{(t)} = \sum_{s=1}^{m} \alpha_s (\mathbf{h}_s^{(t)} \mathbf{h}_s^{(t)\top}) \circ (\mathbf{W}_s - \hat{\Theta}_s^{(t)}).$$

The optimization in (9) can be solved as a weighted low-rank approximation problem (WLRA) via existing algorithms including the gradient descent algorithm and EM procedure (Srebro and Jaakkola, 2003). The upper bound of the condition number, $\kappa_1$, in (9) is a tuning parameter which addresses regularity concerns.

### 3.1.2 Estimate $\mathbf{H}_s$

Provided with $\hat{\mathbf{U}}^{(t+1)} \hat{\mathbf{U}}^{(t+1)\top}$ and $\hat{\Theta}_s^{(t)}$ at the $t$-th iteration, we can estimate $\mathbf{H}_s$ by minimizing (7), which is equivalent to

$$\hat{\mathbf{H}}_s^{(t+1)} = \arg\min_{\mathbf{H} \in \mathcal{D}_{n, \kappa_1}} \left\| \mathbf{W}_s - \hat{\Theta}_s^{(t)} - \mathbf{H} (\hat{\mathbf{U}}^{(t+1)} \hat{\mathbf{U}}^{(t+1)\top}) \right\|_F^2 \quad \text{for} \quad s \in [m]. \quad (10)$$

The problem (10) is a weighted rank-1 approximation of $\mathbf{W}_s - \hat{\Theta}_s^{(t)}$, which generally has no closed-form solution. We propose to use an alternative direction method of multipliers (ADMM) type algorithm to solve the problem (10). By decoupling the two $\mathbf{H}$’s in (10), we write

$$\min_{\mathbf{H}_1, \mathbf{H}_2 \in \mathcal{D}_{n, \kappa_1}} \left\| \mathbf{W}_s - \hat{\Theta}_s^{(t)} - \mathbf{H}_1 (\hat{\mathbf{U}}^{(t+1)} \hat{\mathbf{U}}^{(t+1)\top}) \mathbf{H}_2 \right\|_F^2 \quad \text{s.t.} \quad \mathbf{H}_1 = \mathbf{H}_2. \quad (11)$$

Problem (11) becomes easy when fixing either one of $\mathbf{H}_1$ and $\mathbf{H}_2$. Toward that end, we propose Algorithm 1 to solve the problem (11). Note that we set in Algorithm 1 the input $\mathbf{W} = \mathbf{W}_s - \hat{\Theta}_s^{(t)}$, $\mathbf{C} = \hat{\mathbf{U}}^{(t+1)} \hat{\mathbf{U}}^{(t+1)\top}$. The output of Algorithm 1 is the estimate $\hat{\mathbf{H}}_s^{(t+1)}$.

We remark that the parameter $\lambda$ in Algorithm 1 is for regularization, commonly used in ADMM
Algorithm 1 ReH(\(\mathbf{W}, \mathbf{C}, \mathbf{H}_0\))

1: Input: \(\hat{\mathbf{W}} = \mathbf{W}_s - \hat{\Theta}_s^{(t)}\), the estimated correlation matrix \(\hat{\mathbf{C}} = \hat{\mathbf{U}}^{(t+1)}\hat{\mathbf{U}}^{(t+1)\top}\), an initialization \(\mathbf{H}_0 = \hat{\mathbf{H}}_s^{(t)}\); the maximal iterations \(\text{iter}_{\text{max}}\) and the tolerance parameter \(\epsilon_{\text{tol}} > 0\); condition number \(\kappa_1\).
2: Set \(\mathbf{H}_1 = \mathbf{H}_2 = \mathbf{H}_0\), \(\lambda = 1\) and \(t = 0\)
3: while \(t < \text{iter}_{\text{max}}\) do
4: \(t = t + 1\)
5: \(\text{diag}(\mathbf{H}_1)_j = \frac{\sum_{h=1}^n \hat{W}_{jh} \text{diag}(\mathbf{H}_2)_h + \lambda \text{diag}(\mathbf{H}_2)_h}{\sum_{h=1}^n (\hat{C}_{jh} \text{diag}(\mathbf{H}_2)_h)^2 + \lambda} \)
6: \(\text{diag}(\mathbf{H}_2)_j = \frac{\sum_{h=1}^n \hat{W}_{jh} \text{diag}(\mathbf{H}_1)_h + \lambda \text{diag}(\mathbf{H}_1)_h}{\sum_{h=1}^n (\hat{C}_{jh} \text{diag}(\mathbf{H}_1)_h)^2 + \lambda} \)
7: If \(\|\mathbf{H}_1 - \mathbf{H}_2\|_F \leq \epsilon_{\text{tol}}\) then break \(\lambda = \lambda + 1\)
9: end while 
10: Choose the smallest \(\delta > 0\) such that the condition number of \(\mathbf{H}_1^{(t+1)} = \frac{\mathbf{H}_1 + \mathbf{H}_2}{2} + \delta \mathbf{I}\) is smaller than \(\kappa_1\).
11: Output: \(\hat{\mathbf{H}}_s^{(t+1)}\).

Plugging 

type algorithms. It is empirically important because some entries of \(\mathbf{H}_1\) and \(\mathbf{H}_2\) are often small, making the algorithm unstable on large-scale computations. The regularization \(\delta \mathbf{I}\) added in the last step enforces a restriction on the condition number, ensuring it remains bounded. The parameter \(\kappa_1\) needs to be larger than \(\kappa_0\). In practice, an estimate of \(\kappa_0\) is obtained using the individual data matrix \(\mathbf{W}_s\). For all \(s \in [m]\), we find the best rank-\(r\) approximation of \(\mathbf{W}_s\), diagonally scale it to a correlation matrix, and estimate \(\kappa_0\) using the largest one of the condition numbers of these correlation matrices.

3.1.3 Estimate sparse individual component \(\Theta_s\)

Finally, provided with \(\hat{\mathbf{U}}^{(t+1)}\hat{\mathbf{U}}^{(t+1)\top}\) and \(\hat{\mathbf{H}}_s^{(t+1)}\) at the \(t\)-th iteration, we can estimate the sparse individual component \(\Theta_s\) by solving

\[
\hat{\Theta}_s^{(t+1)} := \arg\min_{\Theta \in \mathbb{R}^{n \times n}} \frac{\alpha_s}{2} \|\mathbf{W}_s - \Theta - \hat{\mathbf{H}}_s^{(t+1)}\hat{\mathbf{U}}^{(t+1)}\hat{\mathbf{U}}^{(t+1)\top}\hat{\mathbf{H}}_s^{(t+1)}\|_F^2 + \lambda_s \|\Theta\|_{\ell_1} \quad \text{for} \ s \in [m]. \tag{12}
\]

Problem (12) has a closed-form solution through a simple entry-wise soft-thresholding method. To this end, we propose Algorithm 2 to obtain \(\hat{\Theta}_s^{(t+1)}\) where the threshold is set at \(\tau_s = \lambda_s / \alpha_s\).

Putting together the iterative rules in Section 3.1.1, 3.1.2 and 3.1.3, we solve the problem (7) by Algorithm 3.
As a result, we can simply obtain (problem (13) amounts to a best rank-
r is attainable by a truncated eigenvalue decomposition. Indeed, observe that the solution to the
Frobenius norm in (8) becomes easy. For a fixed \( s \) which does not admit a closed-form solution. However, the optimization problem for each matrix
favorable performances.

3.1.4 Inexact but faster update of \( U \)

While the proposed update of \( \hat{U}^{(t+1)} \) via problem (8) is (at least locally) polynomial-time solvable by gradient descent, it is still quite slow on large-scale real datasets, e.g., the clinical knowledge graph example in Section 6. We observe that a simple but fast inexact update of \( \hat{U}^{(t+1)} \) yields favorable performances.

The major computation bottleneck of the problem (8) is the sum of matrix Frobenius norms which does not admit a closed-form solution. However, the optimization problem for each matrix Frobenius norm in (8) becomes easy. For a fixed \( s \) and given \( W_s, \hat{\Theta}_s \), the solution of

\[
(\hat{H}_s^{(t)}, \hat{U}_s^{(t)}) := \arg\min_{U \in \mathcal{F}_{n,r,s}, H \in \mathcal{D}_{n,s}} \|W_s - \hat{\Theta}_s^{(t)}/HUU^\top H\|_F^2
\]  

is attainable by a truncated eigenvalue decomposition. Indeed, observe that the solution to the problem (13) amounts to a best rank-\( r \) approximation of \( W_s - \hat{\Theta}_s \) by a positive semi-definite matrix, which is attainable by a truncated eigenvalue decomposition as described in Algorithm 4. As a result, we can simply obtain \( (\hat{U}_s^{(t)}, \hat{U}_s^{(t)\top}, \hat{H}_s^{(t)\top}) = \text{ReC}(W_s - \hat{\Theta}_s^{(t)}, r) \).
Algorithm 4 ReC($W, r$)

**Input:** the symmetric matrix $W$ and rank $r$.

1. Compute the eigen-decomposition: $W = \sum_{i=1}^{n} \lambda_i v_i v_i^\top$ and denote $\tilde{\lambda}_i = \lambda_i \cdot 1(\lambda_i > 0)$.
2. Get the best rank-$r$ approximation of $W = \sum_{i=1}^{r} \tilde{\lambda}_i v_i v_i^\top =: V \Sigma V^\top$.

Get $U = V \Sigma^{1/2}$ and its row normalization matrix $\tilde{U}$. The $i$ row of $\tilde{U}$ is $\tilde{U}_i = \frac{1}{\|U_i\|_2} U_i$.

4. $C = \tilde{U} \tilde{U}^\top$, $H = \text{diag}(\|U_1\|_2, \ldots, \|U_n\|_2)$.

Output: $\tilde{U} \tilde{U}^\top$.

Unlike (13), problem (8) involves the sum of multiple low-rank approximations, which admits no closed-form solution. To speed up the update of $\tilde{U}^{(t+1)}$, we turn to solve the individual problem (13) for each $s$ independently, and then to update $\tilde{U}^{(t+1)}$ as a weighted average of the correlation matrix $\tilde{U}_s^{(t)} \tilde{U}_s^{(t)^\top}$ estimated locally on each $W_s$. Then, we calculate the weighted average

$$\tilde{C}^{(t)} = \sum_{s=1}^{m} \frac{\alpha_s}{\sum_s \alpha_s} \tilde{U}_s^{(t)} \tilde{U}_s^{(t)^\top}. \quad (14)$$

Since all $\alpha_s > 0$, it is easy to check that $\tilde{C}^{(t)}$ is indeed a correlation matrix, i.e., $\tilde{C}^{(t)}$ is positively semi-definite and all diagonal entries equal 1. However, the rank of $\tilde{C}^{(t)}$ is larger than $r$. Finally, by applying Algorithm 4 on $\tilde{C}^{(t)}$, we obtain the final update by $\tilde{C}^{(t+1)} = \tilde{U}^{(t+1)} \tilde{U}^{(t+1)^\top} = \text{ReC}(\tilde{C}^{(t)}, r)$.

Equipped with this fast and inexact update of $\tilde{U}^{(t+1)}$, our estimating procedure on the large-scale dataset is summarized in Algorithm 5.

### 3.2 Warm Initialization

We next describe a procedure for obtaining $\tilde{U}^{(0)}$, $\tilde{H}^{(0)}$, and $\tilde{\Theta}^{(0)}$ as warm initializations of the iterative algorithm discussed in Section 3.1. Recall the equation (6) that amounts to a low-rank plus sparse decomposition of each $W_s$. We follow the penalized method in Tao and Yuan (2011) to estimate the low-rank matrix $L_s = H_s U U^\top H_s$ and the sparse matrix $\Theta_s$:

$$\min_{L_s, \Theta_s \in \mathbb{R}^{n \times n}} \frac{1}{2} \|W_s - L_s - \Theta_s\|_F^2 + \mu \|L_s\|_* + \tau \|\Theta_s\|_1, \quad (15)$$

where the nuclear norm $\|\cdot\|_*$ promotes low-rank solution and $\|\cdot\|_{\ell_1}$ norm promotes sparse solution. The parameters $\mu, \tau > 0$ control the rank and sparsity. The problem (15) is convex and Tao and Yuan (2011) proposed an alternating splitting augmented Lagrangian method (ASALM) to solve it. Their key idea is to reformulate (15) into the following favorable form:

$$\min_{L_s, \Theta_s, E_s \in \mathbb{R}^{n \times n}} \frac{1}{2} \|E_s\|_F^2 + \mu \|L_s\|_* + \tau \|\Theta_s\|_{\ell_1} \quad \text{s.t. } L_s + \Theta_s + E_s = W_s. \quad (16)$$
The augmented Lagrangian function of (16) is

\[
\mathcal{L}(\mathbf{L}_s, \mathbf{\Theta}_s, \mathbf{E}_s, \mathbf{\Lambda}, \beta) := \frac{1}{2}\|\mathbf{E}_s\|^2_F + \mu \|\mathbf{L}_s\|_* + \tau \|\mathbf{\Theta}_s\|_{\ell_1} \\
- \langle \mathbf{\Lambda}, \mathbf{L}_s + \mathbf{\Theta}_s + \mathbf{E}_s - \mathbf{W}_s \rangle + \frac{\beta}{2}\|\mathbf{L}_s + \mathbf{\Theta}_s + \mathbf{E}_s - \mathbf{W}_s\|^2_F,
\]

where \(\beta > 0\) is a tuning parameter. The iterative scheme of ASALM then consists of the following updates with explicit solutions at the \(k\)-th iteration:

\[
\begin{align*}
\mathbf{E}_s^{(k+1)} &\in \text{argmin}_{\mathbf{E}_s \in \mathbb{R}^{n \times n}} \frac{1}{2}\|\mathbf{E}_s\|^2_F + \frac{\beta}{2}\|\mathbf{E}_s + \mathbf{L}_s^{(k)} + \mathbf{\Theta}_s^{(k)} - \frac{1}{\beta}\mathbf{\Lambda}^{(k)} - \mathbf{W}_s\|^2_F \\
\mathbf{\Theta}_s^{(k+1)} &\in \text{argmin}_{\mathbf{\Theta}_s \in \mathbb{R}^{n \times n}} \tau \|\mathbf{\Theta}_s\|_{\ell_1} + \frac{\beta}{2}\|\mathbf{\Theta}_s + \mathbf{L}_s^{(k)} + \mathbf{E}_s^{(k+1)} - \frac{1}{\beta}\mathbf{\Lambda}^{(k)} - \mathbf{W}_s\|^2_F \\
\mathbf{L}_s^{(k+1)} &\in \text{argmin}_{\mathbf{L}_s \in \mathbb{R}^{n \times n}} \mu \|\mathbf{L}_s\|_* + \frac{\beta}{2}\|\mathbf{L}_s + \mathbf{\Theta}_s^{(k+1)} + \mathbf{E}_s^{(k+1)} - \frac{1}{\beta}\mathbf{\Lambda}^{(k)} - \mathbf{W}_s\|^2_F \\
\mathbf{\Lambda}^{(k+1)} &= \mathbf{\Lambda}^{(k)} - \beta(\mathbf{L}_s^{(k+1)} + \mathbf{\Theta}_s^{(k+1)} + \mathbf{E}_s^{(k+1)} - \mathbf{W}_s).
\end{align*}
\]

The augmented Lagrangian function of (18) is

\[
\mathcal{L}(\mathbf{L}_s, \mathbf{\Theta}_s, \mathbf{E}_s, \mathbf{\Lambda}, \beta) := \frac{1}{2}\|\mathbf{E}_s\|^2_F + \mu \|\mathbf{L}_s\|_* + \tau \|\mathbf{\Theta}_s\|_{\ell_1} \\
- \langle \mathbf{\Lambda}, \mathbf{L}_s + \mathbf{\Theta}_s + \mathbf{E}_s - \mathbf{W}_s \rangle + \frac{\beta}{2}\|\mathbf{L}_s + \mathbf{\Theta}_s + \mathbf{E}_s - \mathbf{W}_s\|^2_F,
\]

where \(\beta > 0\) is a tuning parameter. The iterative scheme of ASALM then consists of the following updates with explicit solutions at the \(k\)-th iteration:
It is straightforward to see that $E_s^{(k+1)}$ has a closed-form solution and the solution $\Theta_s^{(k+1)}$ is attainable by entry-wise thresholding. Explicit solutions for $\Theta_s^{(k+1)}$ and $L_s^{(k+1)}$ can be obtained as

$$\Theta_s^{(k+1)} = S_{\tau \beta^{-1}}(Ws + \beta^{-1} \Lambda^{(k)} - E_s^{(k+1)} - L_s^{(k)})$$

and

$$L_s^{(k+1)} = Q_{\mu \beta^{-1}}(Ws + \beta^{-1} \Lambda^{(k)} - E_s^{(k+1)} - \Theta_s^{(k+1)}),$$

respectively, where for any $a > 0$ and matrix $M$ with singular value decomposition (SVD) $U \Sigma V^T$,

$$(S_a(M))_{ij} := \max\{|M_{ij}| - a, 0\} \cdot \text{sign}(M_{ij}).$$

$$(Q_a(M)) := U S_a(\Sigma) V^T.$$  \hspace{1cm} (19)  \hspace{1cm} (20)

Therefore, ASALM for (16) updates $(L_s^{(k+1)}, \Theta_s^{(k+1)}, E_s^{(k+1)})$ via the following computations in Algorithm 6.

**Algorithm 6** The k-th iteration of the extended ASALM for (16):

1. Compute $E_s^{(k+1)} = \frac{\beta}{1+\beta} (Ws + \beta^{-1} \Lambda^{(k)} - L_s^{(k)} - \Theta_s^{(k)}).$
2. Compute $\Theta_s^{(k+1)} = S_{\tau \beta^{-1}}(Ws + \beta^{-1} \Lambda^{(k)} - E_s^{(k+1)} - L_s^{(k)}).$
3. Compute $L_s^{(k+1)} = Q_{\mu \beta^{-1}}(Ws + \beta^{-1} \Lambda^{(k)} - E_s^{(k+1)} - \Theta_s^{(k+1)}).$
4. Update $\Lambda^{(k+1)} = \Lambda^{(k)} - \beta (L_s^{(k+1)} + \Theta_s^{(k+1)} + E_s^{(k+1)} - Ws).$

The initialization $\hat{U}_s^{(0)}$ is then passed to Algorithm 4 which outputs $\hat{H}_s^{(0)}$ and $\hat{U}_s^{(0)}$. Then we ensemble $\hat{U}_s^{(0)}$ via (14) and Algorithm 4 to generate the initial estimate $\hat{U}^{(0)}$. Note that the ensemble (14) relies on the weight $\alpha_s$ assigned to the $s$th view. We shall discuss the way to choose these weights in Section 3.4.

**Remark 6.** Algorithm 3 tackles a highly non-convex estimation problem, making theoretical proof of its convergence challenging. While the initial estimators demonstrate convergence (Tao and Yuan, 2011), they do not achieve the optimal rate of $O(1/m)$ associated with multi-view learning since they are trained individually for each view. Nevertheless, we posit that with a warm initialization, Algorithm 3 is likely to yield estimators that are superior to the initial ones. This assertion is supported by our simulation studies in Section 5 and real data analysis in Section 6. When additional validation data such as those from knowledge graph in the real data example, one may mitigate the convergence to a local optimum by monitoring the algorithm performance at each iteration, as elaborated in Section 3.4.
3.3 Clustering and Network Analysis

In the final step, we apply the K-means algorithm (Steinhaus, 1956) on \( \hat{U} \in \mathcal{F}_{n,r} \) from the output of Algorithm 3 to recover \( Z \) and \( \Omega \). The K-means algorithm aims to solve the optimization problem

\[
(\hat{Z}, \hat{K}) = \arg\min_{Z \in \mathcal{Z}_{n,K}, K \in \mathbb{R}^{K \times r}} \|ZK - \hat{U}\|_F^2
\]

where the \( k \)-th row of \( K \) represents the \( k \)-th centroid in the \( r \)-dimensional space. Even though the exact solution to the optimization problem in (21) is generally NP-hard (Mahajan et al., 2009), there exist efficient algorithms to find an approximate solution whose objective value is within a constant fraction of the global minimal value (Kumar et al., 2004; Awasthi et al., 2015). Therefore, given \( \varepsilon \in (0, 1) \), we calculate the \((1 + \varepsilon)\)-approximate solution:

\[
(\tilde{Z}, \tilde{K}) \in \mathcal{Z}_{n,K} \times \mathbb{R}^{K \times r}
\]

s.t. \( \|\tilde{Z}K - \hat{U}\|_F^2 \leq (1 + \varepsilon) \min_{Z \in \mathcal{Z}_{n,K}, K \in \mathbb{R}^{K \times r}} \|ZK - \hat{U}\|_F^2 \). \hspace{1cm} (22)

Although the solutions to (22) might not be unique, they all attain the same theoretical guarantees. We denote by \( \tilde{Z} \) any output from the optimization of (22). The group-level weight matrix \( \Omega \) can be naturally estimated by

\[
\tilde{\Omega} = (\tilde{Z}^\top \tilde{Z})^{-1} \tilde{Z}^\top \tilde{C}\tilde{Z}(\tilde{Z}^\top \tilde{Z})^{-1}.
\]

Remark 7. Since the matrix \( \hat{U} \) has rank \( r \), \( \tilde{\Omega} \) has rank at most \( r \). If the underlying graph is sparse, a hard thresholding procedure can be applied on \( \tilde{\Omega} \) to obtain its sparsified version. In Section 6, we will show the sparsity of \( \tilde{\Omega} \) in the real data analysis.

3.4 Tuning Parameters

In our algorithm, several tuning parameters require careful selection, including \( \alpha_s, \lambda_s \), the rank \( r \) for the objective equation (7), and the number of groups \( K \), as well as \( \mu, \tau, \) and \( \beta \) for the initialization process (15). To optimize computational efficiency, we initially determine the parameters \( \mu, \tau, \beta, \) and \( r \) through a grid search. Subsequently, we select \( \alpha_s \) and \( \lambda_s \).

Specifically, for the convex optimization problem (15), we adopt the guidelines from Tao and Yuan (2011) to set the parameters: \( \mu \) is set as \( \sqrt{n^2 + \sqrt{8}n\delta} \) as \( n^{-1/2} \), and \( \beta \) as \( an^2/\|W_s\|_{\ell_1} \) for each \( s \in [m] \). The values of \( \delta, \alpha \), and the rank \( r \) are fine-tuned using a grid search. For instance, if we have access to labels on a subset of concept pairs with their relatedness scores manually annotated (as in Section 6), we can calculate the Spearman’s rank correlation between the labeled relatedness scores and the corresponding estimated entries from \( \sum_{s=1}^{m} \hat{U}_s^{(0)}(\hat{U}_s^{(0)})^\top \). The tuning parameters that result in the highest rank correlation are then chosen. For computational efficiency, a preliminary
estimate of the rank $r$ can also be constructed based on the eigenvalue decay of $W_s$, which is a widely recognized technique for determining the rank of low-rank matrices (Jolliffe, 2005).

We estimate $\sigma_s^2$ as $\tilde{\sigma}_s^2 = \|W_s - \hat{L}_s^{(0)} - \hat{\Theta}_s^{(0)}\|_F^2 / n^2$. The weight parameters are then set as $\alpha_s = c_s \hat{h}_s^{-4} \tilde{\sigma}_s^{-2}$ and the regularization parameters as $\lambda_s = c \sigma_s \log^{1/2} n$, where $\hat{h}_s$ represents the average of the diagonal entries of $\hat{L}_s^{(0)}$. The constants $c_s$ and $c$ are determined again through grid search. The theoretical underpinnings for the rates of $\alpha_s$ and $\lambda_s$ are explained in Section 4.

To select the number of groups $K$ without prior grouping information, we recommend using the elbow method, which involves plotting the within-cluster sum of squared errors (WSS) against various $K$ values, and the Silhouette method. However, if some group labels are available, the sum of normalized mutual information (NMI) and adjusted Rand index (ARI) can be utilized. When only partial labels are available, such as pairs within and between groups, we suggest using a composite score defined as the sum of sensitivity and specificity to ascertain the optimal $K$. The procedure of selecting these tuning parameters is further illustrated in Sections 5 and 6.

4 Theory

In this section, we provide theoretical analyses of the performance of the estimator (7) under the msLBM model with Assumptions 1-3.

4.1 Joint Estimation Bounds for Weight and Heterogeneity Matrix

Let $\kappa_1 > \kappa_0$ be the condition number used in (7) and $\tilde{U}, \{\tilde{H}_s\}_{s=1}^m, \{\tilde{\Theta}_s\}_{s=1}^m$ be the estimators. For $\forall s \in [m]$, denote $\Delta_s = \hat{L}_s - L_s := \hat{H}_s \tilde{U} \hat{U}^\top \hat{H}_s - H_s U U^\top H_s$. Denote $\Psi_s = \text{supp}(\Theta_s)$ the support of $\Theta_s$, i.e., the locations of non-zero entries of $\Theta_s$. The joint estimation bounds for $\{\Delta_s\}_{s=1}^m$ in both Frobenius and Sup norms are as follows.

**Theorem 1.** Under Assumptions 1-3, there exist constants $C'_0, C_0, C_1, \cdots, C_8 > 0$ depending only on $\kappa_1$ such that if $m \leq C'_0 r, C_0 r^2 |\Psi_s| \leq n$ and $\lambda_s = \alpha_s \sigma_s^2 \log^{1/2} n$ is appropriately chosen, we get with probability at least $1 - mn^{-2}$ that

$$\sum_{s=1}^m \alpha_s \|\Delta_s\|_F^2 \leq C_1 r n \cdot \left( \max_{s \in [m]} \alpha_s \sigma_s^2 \right) + C_2 \sum_{s=1}^m \alpha_s \sigma_s^2 |\Psi_s| \log n$$

$$\sum_{s=1}^m \alpha_s \|\Delta_s\|_\infty^2 \leq C_3 r^3 \cdot \left( \max_{s \in [m]} \alpha_s \sigma_s^2 \right) + C_4 \sum_{s=1}^m \alpha_s \sigma_s^2 r^2 \cdot \frac{|\Psi_s| \log n}{n}$$
and

\[ \sum_{s=1}^{m} \alpha_s \| \tilde{\Theta}_s - \Theta_s \|_F^2 \leq \max_{s \in [m]} |\Psi_s| \cdot C_5 r^3 (\max_{s \in [m]} \alpha_s \sigma_s^2) + \max_{s \in [m]} |\Psi_s| \cdot C_6 \sum_{s=1}^{m} \alpha_s \sigma_s^2 \log n \]

\[ \sum_{s=1}^{m} \alpha_s \| \tilde{\Theta}_s - \Theta_s \|_{\infty}^2 \leq C_7 r^3 \cdot (\max_{s \in [m]} \alpha_s \sigma_s^2) + C_8 \sum_{s=1}^{m} \alpha_s \sigma_s^2 \log n. \]

The bounds established in Theorem 1 require no conditions on the magnitudes of the non-zero entries of the heterogeneity matrices $\Theta_s$’s. This is due to the penalty by the $\ell_1$-norm. On the other hand, to prove sharp bounds, we require that the support sizes of $\Theta_s$ are upper bounded by $O(n)$. This is a mild condition since the matrix is indeed very sparse on EHR datasets. See the real data example in Section 6. The probability bound $1 - mn^{-2}$ can be improved to $1 - (mn)^{-2}$ if we replace the $\log n$ term in the upper bounds by $\log(mn)$.

**Remark 8.** The view-wise heterogeneity matrix $H_s$ significantly enhances model flexibility. This heterogeneity is a crucial methodological innovation, enabling superior performance with real data. However, this comes at a cost of an increased number of model parameters, with $O(mn)$ attributable to estimating $\{H_s\}$. In our theoretical analysis, we concentrate on the case where $m = O(r)$, which strikes a balance between maximizing model flexibility and minimizing model complexity. This assumption is well-suited to real data applications. For example, in our real data experiment, the rank $r$ is approximately 250, while $m \leq 10$. The condition $m = O(r)$ also allows for technical convenience. The main parameter of interest, the shared correlation matrix $C$, has a model complexity of $O(\log n)$. If $m \gg r$, the model complexity is dominated by $O(mn)$, and accurately estimating the heterogeneity matrices becomes a major bottleneck in fitting the msLBM. When $m \gg r$, additional structure on heterogeneity is needed in order to better borrow information across views. For example, one may assume some of the views share similar $H_s$. This can significantly reduce the degree of heterogeneity in $\{H_s\}_{s=1}^{m}$. The estimation procedures can be modified accordingly, and we expect that parallel theoretical results can be derived.

By choosing the weight $\alpha_s \asymp m^{-1}$, Theorem 1 implies that w.h.p.,

\[ \sum_{s=1}^{m} \| \tilde{\Delta}_s \|_F^2 = O \left( rn \cdot \max_{s \in [m]} \sigma_s^2 + \sum_{s=1}^{m} \sigma_s^2 |\Psi_s| \log n \right). \]  \( (24) \)

The degrees of freedom of the parameters $\{H_s\}_{s=1}^{m}$ and $\mathbf{U} \mathbf{U}^\top$ in msLBM model is $O(rn)$. It implies that the first term in the RHS of (24) is sharp if $\sigma_s^2 \asymp \sigma^2$ for all $s \in [m]$. The second term in RHS of (24) is related to $\sum_{s=1}^{m} |\Psi_s|$, which is the model complexity of the heterogeneity matrices. Ignoring the logarithmic factor, the second term in RHS of (24) is also sharp w.r.t. the degrees of freedom.

The bound (24) actually implies the estimators $\tilde{H}_s$ are relatively consistent under mild conditions. For notional clarity, we denote $H_s = \text{diag}(h_s)$ and $\tilde{H}_s = \text{diag}(\tilde{h}_s)$ with $h_s, \tilde{h}_s \in \mathbb{R}^n$ for all $s \in [m]$.  

20
For ease of exposition, we denote $h_s = \min_{i \in [n]} |h_s(i)|$ for all $s \in [m]$. Denote $h_{\min} = \min_{s \in [m]} h_s$ and $h_{\max} = \max_{s \in [m]} h_s$. The bound (24), together with the fact that $\|L_s\|_F^2 \geq h_{\min}^4 n^2 / r$, yields

$$\frac{\sum_{s=1}^m \| \hat{L}_s - L_s \|_F^2}{\sum_{s=1}^m \| L_s \|_F^2} = O\left( \frac{r^2}{h_{\min}^4 nm} \cdot \max_{s \in [m]} \sigma_s^2 + \frac{r \sum_{s=1}^m \sigma_s^2 |\Psi_s| \log n}{h_{\min}^4 n^2 m} \right).$$

The relative error in Frobenius norm diminishes as long as $h_{\min}^4 nm \gg r^2 \max_{s \in [m]} \sigma_s^2 + n^{-1} \sum_{s=1}^m \sigma_s^2 |\Psi_s| \log n$, which holds trivially if further assuming $\max_{s \in [m]} \sigma_s^2 = O(h_{\min}^4)$. Moreover, the error rate decays as $m$ increases under the upper bound condition on the support cardinality $|\Psi_s|$.

Remark 9. Multi-view versions of stochastic block models have been studied in Agterberg et al. (2022); Lei et al. (2020); Paul et al. (2020), showing that, under suitable conditions, increasing the number of observed networks can improve the accuracy in estimating the shared model parameters. While such an improvement is also achieved by our method and model, as discussed above, a crucial difference in our msLBM model is its accommodation of between view heterogeneity. In fact, the total number of heterogeneity parameters, of order $O(mn)$, increases with the number of views $m$, unlike the models in the aforementioned works, whose complexities typically remain constant irrespective of $m$. The inclusion of these additional heterogeneity parameters in the msLBM enhances the model’s robustness, albeit at the cost of more complex estimation procedures and theoretical investigations.

4.2 Spectral Clustering Consistency

As discussed after Theorem 1, the low-rank part $L_s$ of each view can be consistently recovered under mild conditions. We now investigate the performance of the estimated shared correlation matrix $\hat{C} = \hat{U} \hat{U}^\top$. The following theorem shows that, as long as the weights are properly chosen, the consensus graph can be estimated more accurately when more views of data sources are available.

Theorem 2. Suppose the conditions of Theorem 1 hold, $h_{\min}^4 \geq \max_{s \in [m]} \sigma_s^2 \cdot C_0'(h_{\max}/h_{\min})^4 r^2 (r + m)$ and $C_1'(h_{\max}/h_{\min})^4 \sum_{s=1}^m h_s^{-4} \sigma_s^2 r^2 |\Psi_s| \log n \leq n$ for large but absolute constants $C_0', C_1' > 0$ depending only on $\kappa_1$, if $\alpha_s \asymp h_s^{-4}$, the following bound holds with probability at least $1 - mn^{-2}$,

$$\frac{\| \hat{C} - C \|_F^2}{n^2} \leq C_2 \frac{r^2}{nm} \cdot \left( \max_{s \in [m]} \alpha_s \sigma_s^2 \right) + \frac{C_3 r}{mn} \sum_{s=1}^m \alpha_s \sigma_s^2 |\Psi_s| \log n,$$

where $C_2, C_3 > 0$ are absolute constants depending on $\kappa_1$ only. Meanwhile, we have

$$\min_{O \in O_{r \times r}} \| \hat{U} - UO \|_F^2 \leq C_4 \frac{r^3}{m} \cdot \left( \max_{s \in [m]} \alpha_s \sigma_s^2 \right) + \frac{C_5 r^2}{mn} \sum_{s=1}^m \alpha_s \sigma_s^2 |\Psi_s| \log n,$$

where $C_4, C_5 > 0$ are absolute constants depending on $\kappa_1$ only.
In the case \( r, m = \mathcal{O}(1) \) and \( \frac{h_{\text{max}}}{h_{\text{min}}} = \mathcal{O}(1) \), the first condition of Theorem 2 becomes \( h_{\text{min}} \gtrsim \max_{s \in [m]} \sigma_s^2 \). This requires that the diagonal entries of \( W_s \) should be larger than the noise standard deviation. It is a very mild condition in EHR where the observed diagonal entries are often significantly dominating.

Recall that \( \|U\|_F \approx \frac{n}{1/2} \). For general \( r \) and \( m \) satisfying the conditions of Theorem 2, we have \( \|C\|_F \gtrsim \frac{n}{\sqrt{r}} \). Theorem 2, together with the conditions of \( h_{\text{min}} \), implies that

\[
\|C\|_2^2 = O\left( \frac{r^3}{nm} \cdot \max_{s \in [m]} \sigma_s^2 \right),
\]

which holds with probability at least \( 1 - mn^{-2} \). Interestingly, it suggests that the relative error decreases as either \( n \) or \( m \) or both increase. Thus, integrating more data sources can improve the estimation of the correlation matrix \( C \). We note that the sub-optimal term \( r^3 \) in the above bound is due to the technical difficulty in bounding the sup-norm error rate \( \|\Delta_s\|_{\ell_\infty} \), which is derived by exploiting the incoherence properties of \( L_s \)'s.

Under similar conditions, we can also get

\[
\min_{O \in \mathbb{O}_{r \times r}} \frac{\|\hat{U} - UO\|_F^2}{\|U\|_F^2} = O\left( \frac{r^3}{nm} \cdot \max_{s \in [m]} \sigma_s^2 \right)
\]

implying that the factor \( U \) can be consistently recovered if \( nm/r^3 \to \infty \). The rows of \( U \) provide the information of cluster memberships of vertices. Now we study the clustering error based on its empirical counterpart \( \hat{U} \).

We apply the K-means algorithm on \( \hat{U} \) to get the approximate \((1 + \epsilon)\) solution as in (22). Let \( \hat{Z} \in \mathcal{Z}_{n,K} \) denote the output membership matrix. In this section, we show that the proposed algorithm in Section 3 can consistently recover the latent membership matrix under the minimal SNR condition. For two membership matrices \( Z_1, Z_2 \in \mathcal{Z}_{n,K} \), define the mis-clustering number as

\[
e_1(Z_1, Z_2) = \min_{P \in \mathcal{P}_{K,K}} \frac{\|Z_1P - Z_2\|_{\ell_1}}{2}
\]

where \( \mathcal{P}_{K,K} \) denotes the set of all \( K \times K \) permutation matrices.

**Theorem 3.** Suppose the conditions of Theorem 2 holds and \( mn^2_{\text{min}} \sigma^2_{\Omega} \geq C_0 r (r + m) n \max_s (\alpha_s \sigma_s^2) + C_1 r \sum_{s=1}^{m} \alpha_s \sigma_s^2 |\Psi_s| \log n \) for large constants \( C_0, C_1 \) depending on \( \kappa_1 \) only, the following bound holds with probability at least \( 1 - mn^{-2} \),

\[
\frac{e_1(\hat{Z}, Z)}{n} \leq 4(2 + \epsilon)^2 \frac{n}{nm} \sigma^2_{\Omega} \left[ C_2 r^2 \left( \max_{s \in [m]} \alpha_s \sigma_s^2 \right) + C_3 r \sum_{s=1}^{m} \alpha_s \sigma_s^2 |\Psi_s| \log n \right],
\]

where \( C_2, C_3 > 0 \) depends only on \( \kappa_1 \).
In the case that \( n_{\min} \approx nK^{-1} \) and \( \varepsilon \leq 1 \), together with the condition of \( h_{\min} \) in Theorem 2, the mis-clustering relative error becomes

\[
e_1(\tilde{Z}, Z) = O\left( \frac{K}{mn\delta_{\Omega}^2} \right),
\]

which converges to 0 if \( mn\delta_{\Omega}^2/K \to \infty \) as \( nm \to \infty \). The number of clusters \( K \) in the motivating EHR application is large, e.g., around \( O\left( \frac{n}{2} \right) \) in as seen in Section 6. However, if \( K \) is small such that \( K = O(m) \), and moreover if \( \delta_{\Omega} \gtrsim 1 \), we get \( e_1(\tilde{Z}, Z) = O(Km^{-1}) \) implying that \( e_1(\tilde{Z}, Z) = O(1) \). This is interesting since it suggests that our algorithm can exactly recover nearly all of the vertices’ membership.

### 4.3 Consensus Graph Estimation

We next establish the error of

\[
\Omega = (\tilde{Z}^\top\tilde{Z})^{-1}\tilde{Z}^\top(\tilde{U}\tilde{U}^\top)\tilde{Z}(\tilde{Z}^\top\tilde{Z})^{-1}
\]

Theorem 4. Suppose that the conditions of Theorem 3 hold such that \( e_1(\tilde{Z}, Z) \leq n_{\min}/2 \), there exist constants \( C_3, C_4 > 0 \) depending only on \( \kappa_1 \) such that

\[
\|\hat{\Omega} - \hat{P}\Omega\hat{P}^\top\|_F^2 \leq (2 + \varepsilon)^2\left\|\Omega\right\|^2 \cdot \frac{n_{\max}^3 n}{n_{\min}^5} \cdot \left[ C_3 r^2 m \cdot \left( \max_{s \in [m]} \alpha_s \sigma_s^2 \right) + \frac{C_4 r mn}{m} \sum_{s=1}^{m} \alpha_s \sigma_s^2 |\Psi_s| \log n \right]
\]

where \( \hat{P} \) is the \( K \times K \) permutation matrix realizing \( \min_{P \in \mathcal{P}_{K,K}} \|Z - \hat{Z}P\|_1 \).

The condition \( e_1(\tilde{Z}, Z) \leq n_{\min}/2 \) is mild. Indeed, by (25), this condition holds as long as \( nm\delta_{\Omega}^2 \gtrsim K^2 \) in the case \( n_{\min} \approx n_{\max} \approx nK^{-1} \). If the \( \delta_{\Omega} \) is bounded away from 0, then the number of clusters is allowed to grow as fast as \( (nm)^{1/2} \). Together with the condition on \( h_{\min} \) in Theorem 2, the bound in Theorem 4 implies that

\[
\frac{\|\hat{\Omega} - \hat{P}\Omega\hat{P}^\top\|_F^2}{\|\Omega\|^2} = O\left( \frac{K^2}{mn\delta_{\Omega}^2} \right),
\]

which converges to zero as long as \( mn\delta_{\Omega}^2 \to \infty \) as \( mn \to \infty \).

### 5 Simulations

In this section, we present simulation results to evaluate the finite sample performance of the proposed msLBM estimator obtained through Algorithm 3 and compare it to existing methods. Throughout we set \( m = 3, n = 500, r = 25 \). For simplicity, we considered a balanced underlying
clustering structure such that \( n_k \approx n/K, 1 \leq k \leq K \) under a range of \( K = 25, 50, 75, 100 \). For each configuration setting, we summarize results based on the average from 50 independent experiments.

To mimic a real-world sparse network, we first generate a sparse matrix \( A = [A(i, j)]_{K \times r} \) with normalized rows and then set \( \Omega = AA^\top \). Specifically, we generate \( A_{i,j} \overset{i.i.d.}{\sim} (1-\pi_0) \cdot 1_{(x=0)} + \pi_0 \cdot U(0, 1) \) for \( i \in [K] \) and \( j \in [r] \) where \( \pi_0 = 0.2 \) to be comparable to what we observed in the real data analysis. Then we normalize the rows of \( A \) to make all of its rows have unit \( \ell_2 \) norm. We then fix \( \Omega \) for all of the repetitions.

We consider two settings for generating \( H_s \) and \( \Theta_s \): setting (1) representing a heterogeneous view \( H_s \) and \( \Theta_s \neq 0 \) and setting (2) representing a more homogeneous scenario with \( H_s = \lambda \sqrt{s} I_n \) and \( \Theta_s = 0 \).

In setting (1), we generate \( \{\Theta_s\}_{s \in [m]} \) by sampling its entries independently from the distribution \((1 - \pi) 1_{(x=0)} + \pi \cdot N(0, \tau^2) \) with \( \pi = 0.05 \) and \( \tau = 5 \) here. To show that our algorithm is useful for a wide range of \( H_s \), we generate the diagonal entries of \( H_s \) from Uniform\([0, d_s]\) where \( (d_1, d_2, d_3) = \lambda(1, \sqrt{2}, \sqrt{3}) \) and \( \lambda \) is chosen between 1 and 2 to represent varying signal strengths.

In the homogeneous setting (2), the three views share a common eigenspace but have different relative signal strengths. We let \( \lambda \) vary from 0.25 to 1 to reflect different levels of signal strengths. Finally, given \( \Omega, H_s, \) and \( \Theta_s \), we generate the sparse error matrix \( E_s \) by sampling its entries independently from 0.5\( 1_{(x=0)} + 0.5 \cdot N(0, \tau^2) \). For setting (1), \( \sigma_s = 0.1 \) for \( s \in [m] \), and for setting (2), \( (\sigma_1, \sigma_2, \sigma_3) = (0.3, 0.2, 0.1) \).

To evaluate the performance of our proposed and benchmark methods, we consider the ability of the methods to recover \( Z, \Omega, \Theta_s, \) and the eigenspace of \( C \), respectively. More specifically, the mis-clustering error (MCE) for \( Z \) is defined as

\[
\text{MCE}(\hat{Z}, Z) = \min_{P \in \mathcal{P}_{K,K}} \frac{1}{2n} \|\hat{Z}P - Z\|_{\ell_1},
\]

where \( \mathcal{P}_{K,K} \) denotes the set of all \( K \times K \) permutation matrices. We consider the \( \ell_2 \) loss for \( \Omega \) and the \( \ell_0 \) loss for the sparse matrices \( \{\Theta_s\}_{s \in [m]} \), defined as

\[
L_2(\hat{S}, S) = \frac{\|\hat{S} - S\|_F^2}{\|S\|_F^2} \quad \text{and} \quad L_0(\hat{S}, S) = \frac{\|\text{supp}(\hat{S}) - \text{supp}(S)\|_{\ell_1}}{K^2},
\]

for any matrix \( \hat{S} \) and \( S \). In addition, given a pair of matrices with orthonormal columns \( V \) and \( \hat{V} \in \mathcal{O}_{n,r} \), we measure the distance between their invariant subspaces via the spectral norm of the difference between the projections, given by \( \|\hat{V}^\top - VV^\top\| \).

Since no existing methods consider the same model as ours, we compare to some relevant methods that can be used to identify the group structure \( Z \). Specifically, we compare to (i) the sum of adjacency matrices (SAM) approach (Bhattacharyya and Chatterjee, 2018) that estimates the common eigenvectors and the group structures \( Z \) of multi-view networks; and (ii) the MASE ap-
proach on multiple low-rank networks with common principal subspaces (Arroyo et al., 2021). After performing SVD on the sum of the adjacency matrices to obtain $\hat{U}$, the SAM algorithm identifies $Z$ by either performing a K-means clustering algorithm on the row vectors of $\hat{U}$ (SAM-mean) or K-median clustering algorithm on the normalized row vectors of $\hat{U}$ (SAM-median). We only keep $\hat{U}$ to be rank $r$ since the full rank version performs poorly in our settings. We include both MASE and the scaled version of MASE (MASE-scaled) as proposed in Arroyo et al. (2021). In addition, we compare the ASALM algorithm applied to a single source in recovering $\Omega$ and $\Theta_s$. To be specific, the estimated sparse matrix by ASALM is exactly the estimator of $\Theta_s$ in each view. We then input the estimated low-rank matrix by ASALM to Algorithm 4 to obtain the estimator of $\hat{U}$ for each view. Then we use the $\hat{U}$ to estimate $\hat{Z}$ by (22) and $\hat{\Omega}$ by (23).

For setting (1), we only compare the clustering performance of SAM-mean, SAM-median, MASE, and MASE-scaled to our algorithm since under setting (1), these views do not share a common eigenspace and these competing methods assume a common eigenspace across different views. In addition, we compare the $\ell_2$ loss for $\Omega$ and the $\ell_0$ loss for the sparse matrices $\{\Theta_s\}_{s \in [m]}$ of msLBM and ASALM, averaged over the three views. For setting (2), we compare the clustering performance of the four methods mentioned above as well as the eigenspace error of the leading $r$ eigenvectors from SAM, MASE, and MASE-scaled.

We choose the tuning parameters using the procedure detailed in Section 3.4 by randomly sampling 500 entries from $C$ plus normal noise $N(0, 0.01^2)$. This procedure mimics our real data example where we have several sets of human annotated similarity and relatedness of vertex pairs. See Section 6 for details. For the choice of the number of groups $K$, to validate the tuning strategy in Section 3.4, we focus on setting (1) with $\lambda = 1.25$ and $K = 100$, which is the most difficult task due to the lower signal to noise ratio and the large $K$. To be specific, we randomly sample 100 positive pairs (100 pairs of vertices within groups) and 1000 negative pairs with a correlation larger than 0.5 (1000 pairs of vertices between groups). Then for each $K$, we can get $\hat{Z}$ from (22) and finally choose the $K$ achieving the optimal composite score. The procedure is repeated 50 times, and the average optimal $K$ is 100.28 with a standard deviation of 7.83. Due to the effectiveness of the method, we decide to treat $K$ as known during the simulation.

We first compare the MCE of msLBM, SAM-mean, SAM-median, MASE and MASE-scaled. The result is shown in Figure 1. In general, the SAM-mean, MASE, and MASE-scaled perform the worst while msLBM wins across all scenarios. In addition, SAM-mean performs better than the other three comparable methods. It is reasonable that the SAM-mean is designed for degree corrected model which is more suitable for the current case. In addition, SAM and MASE require a common eigenspace across all views but our model does not necessarily satisfy the assumption. It is apparent that msLBM significantly outperforms other methods with even more advantages as $K$ becomes larger.

Then, we evaluate the performance of $\hat{\Omega}$ and $\{\hat{\Theta}_s\}_{s \in [m]}$ estimated from msLBM and ASALM.
Figure 1: Comparison on clustering performance under setting (1).

The result is shown in Figure 2. Again, the msLBM performs much better than ASALM. This is intuitive given that msLBM utilizes the information of all views while ASALM does not.

Finally, we present the mis-clustering error and the eigenspace of these methods under setting 2 in Figure 3, when the three views share a common eigenspace. Even though the current models also satisfy the assumption of the completing methods SAM and MASE, msLBM still performs much better than them because it can fully exploit the relative signal and noise strength of each view. On the contrary, SAM can not achieve an optimal weighted combination of all views by simple average. MASE-scaled can use the relative signal strength of each view by using the eigenvalues of each view. However, it is not robust when the signal strength is small compared to the noise level. So we observe that when $\lambda$ is small, it performs worst. But when $\lambda$ increases, it performs better than MASE. MASE performs worst when $\lambda$ is large because it is unweighted.

6 Applications to learning clinical knowledge graph

We next apply the proposed msLBM method to learn both embeddings for medical concepts and a consensus clinical knowledge graph by synthesizing a few sources of medical text data.

6.1 Data summary

The input data ensemble consists of three similarity matrices of $n = 7,217$ clinical concepts, independently derived from three heterogeneous data sources: (i) 10 million clinical narrative notes of 62K patients at Partners Healthcare System (PHS); (ii) 20 million clinical notes from a Stanford hospital (Finlayson et al., 2014); (iii) clinical notes from the MIMIC-III (Medical Information Mart for Intensive Care) database (Johnson et al., 2016). The clinical concepts were extracted from textual data via natural language processing by mapping clinical terms to Concept Unique
Figure 2: Comparison on recovering $\Omega$ (top panel) and $\Theta_s$ (bottom panel) under setting (1).

Figure 3: Comparison on recovering $Z$ (top panel) and the eigenspace of $C$ (bottom panel) under setting (2).
Identifiers (CUIs) from UMLS. Heterogeneity inherently plays a role due to the different natures of data sources. For instance, MIMIC-III data only involves intensive care unit (ICU) patients. The frequency of disease conditions seen in ICU settings differs from those in the other two views. These datasets also have very different sample sizes. As a result, the marginal frequencies $H_s$ and the noise level $\sigma_s$ can vary greatly across views. The management of disease conditions, such as the treatment used for a given condition, could also differ across health systems and hospital settings (e.g., ICU vs. outpatient), necessitating $\Theta_s$ to account for this type of population-level bias.

For each source $s$, we construct $W_s$ as the shifted-positive pointwise mutual information (SPPMI) matrix for all $n$ concepts. For a pair of concepts $x$ and $y$, the pointwise mutual information (PMI) is a well-known information-theoretic association measure between $x$ and $y$, defined as

$$\text{PMI}(x, y) = \log \frac{P(x, y)}{P(x)P(y)}$$

where $P(x, y)$ is the probability of $x$ and $y$ co-occurring and $P(x), P(y)$ are respective marginal occurrence probabilities. Using PMI as a measure of association in NLP was introduced by Church and Hanks (1990) and has been widely adopted for word similarity tasks (Dagan et al., 1994; Turney and Pantel, 2010, e.g.). However, the empirical PMI matrix is not computationally or statistically feasible to use since the PMI estimate would be $-\infty$ for a pair that never co-occurs and the matrix is also dense. The SPPMI matrix with $\text{SPPMI}(x, y) = \max\{\text{PMI}(x, y), 0\}$ is a sparse and consistent alternative estimate of PMI widely used in the NLP literature (Levy and Goldberg, 2014).

Previous studies (Levy and Goldberg, 2014; Arora et al., 2016, e.g.) have noted that the SPPMI matrix often exhibits low-rank characteristics, which can be approximated by the inner product of word embeddings under the dynamic log-linear topic model proposed by Arora et al. (2016). Our model assumption stems from the use of cosine similarities (instead of inner products) of word embeddings. Cosine similarities between two features should remain consistent across different views or sources. However, the norms of the embeddings, represented by $H_s$, can vary to reflect the heterogeneity in the marginal occurrence probabilities of these concepts in the generative model of Arora et al. (2016). Taking into account that various healthcare systems might display distinct frequencies of the same concept and have unique patterns inherent to each system, it is logical to consider different $H_s$ for different views. Additionally, incorporating the sparse matrix $\Theta_s$ is beneficial to accommodate the view-specific patterns that the low-rank component does not explain and may differ across sites, thereby enhancing the robustness of the model.

We demonstrate below how Algorithm 3 can be used to optimally combine information from the SPPMI matrices from the three data sources to both improve the estimation of embeddings and construct a sparse knowledge graph about coronary artery disease network. We aim to group the 7,217 CUIs into $K$ subgroups such that CUIs within the same subgroups are considered synonyms. We anticipate $K$ to be large in this particular application since CUIs were manually curated to
represent distinct clinical concepts.

### 6.2 Clinical concept Embedding and Network

With the three SPPMI matrices as input, our Algorithm 3 will output the clinical concept network \( \mathbf{\hat{C}} \). \( \mathbf{\hat{C}} \) has the expression \( \mathbf{\hat{C}} = \mathbf{\hat{U}}\mathbf{\hat{U}}^\top \) where \( \mathbf{\hat{U}} \in \mathbb{R}^{n \times r} \) is the spherical embedding of the \( n \) clinical concepts. We will present that \( \mathbf{\hat{C}} \) (or equivalently \( \mathbf{\hat{U}} \)) have better quality than using single source only or other comparable methods.

At first, we can compare the result of Algorithm 3 to the clinical concept learned from a single source. Given \( \mathbf{W}_s \), we can apply Algorithm 4 to it to get the clinical concept network. In addition, we can compare our method to SAM and MASE. For SAM, we get the \( \mathbf{\hat{V}}_{\text{SAM}} \in \mathbb{R}^{n \times r} \) consisting of the leading \( r \) eigenvectors of the sum of the three SPPMI matrices. Then we normalize its row to get \( \mathbf{\hat{V}}_{\text{SAM}} \), the spherical embedding and the clinical concept network \( \mathbf{\hat{C}}_{\text{SAM}} = \mathbf{\hat{V}}_{\text{SAM}}\mathbf{\hat{V}}_{\text{SAM}}^\top \). For MASE (or MASE-scaled), when we get \( \mathbf{\hat{V}}_{\text{MASE}} \), we also normalize its rows to get \( \mathbf{\hat{V}}_{\text{MASE}} \) and get \( \mathbf{\hat{C}}_{\text{MASE}} = \mathbf{\hat{V}}_{\text{MASE}}\mathbf{\hat{V}}_{\text{MASE}}^\top \). To fully highlight msLBM’s ability to exploit the relative signal and noise strengths of each view, we also include the collapsing method (Collapse). This method estimates each of the three probabilities in PMI by utilizing all views and then applies Algorithm 4 to the estimated SPPMI matrix.

To assess the quality of the estimated concept networks, we refer to two sets of human evaluations released by Pakhomov et al. (2010), which focus on the semantic similarity and relatedness between clinical concepts. Semantic similarity measures the extent of semantic overlap between concepts based on psycholinguistic definitions (for example, ‘arthritis’ versus ‘joint pain’), while relatedness pertains to the likelihood of one concept evoking thoughts of another (like ‘diabetes’ versus ‘metformin’). We evaluate the alignment between human annotations and the estimated clinical concept networks by calculating the Spearman rank correlation between human assessments of similarity/relatedness among CUI pairs and the correlations between CUI pairs in the estimated networks. These human annotations provide a valuable benchmark for assessing the quality of our generated network.

Furthermore, the effectiveness of the concept networks can be gauged through their ability to identify known relational pairs. Specifically, we utilize five sets of CUI relation pairs extracted from a medical database, which include May Cause (MayCause), May Be Caused By (Causedby), Differential Diagnosis (Ddx), Belong(s) to the Category of (Bco), and May Treat (MayTreat). Following the methodology of Beam et al. (2019), we report the Area Under the Curve (AUC) and the true positive rate (TPR), setting the false positive rate (FPR) at 1%, 5%, and 10%, respectively, to measure performance.

To select the appropriate rank \( r \), we examine the eigenvalue decay of \( \{\mathbf{W}_s\}_{s=1}^m \). This technique is commonly employed for identifying the rank of low-rank matrices, as seen in various applications such as principal component analysis (Jolliffe, 2005), word embedding (Hong et al., 2021), and
network analysis (Arroyo et al., 2021). We assess the eigenvalue decay of \( \{W_s\}_{s=1}^m \), opting for the rank \( r \) that results in the cumulative eigenvalue percentage exceeding 95% for all matrices. This criterion leads us to choose \( r = 250 \). Consequently, we apply \( r = 250 \) across all methods in our analysis. For fine-tuning the remaining parameters, we use the strategy in Section 3.4 using the relatedness set as the noisy labels.

In addition, we perform a sensitivity analysis by experimenting with various values of \( r \) in these methods and assessing their performance. The outcomes of this analysis are depicted in Figure 4. This figure indicates that our method maintains a similar level of performance across a broad range of \( r \) values and performs adequately when \( r \geq 150 \). Furthermore, our method consistently outperforms all other methods for all \( r \)'s. Therefore, in subsequent analyses, we will primarily focus on results obtained with \( r = 250 \).

![Figure 4: The Spearman rank correlation is depicted for the relatedness scores (the left panel) and the similarity scores (the middle panel). Additionally, the average AUC is presented for the five types of relation pairs (the right panel).](image-url)

Table 1 summarizes the Spearman rank correlation using each single data source and the output given by Algorithm 3 with \( r = 250 \). Tables 2, 3, and 4 show the AUC and TPR given FPR = 0.05 and 0.1 of relation detection, respectively. Synthesizing information from three data sources via Algorithm 3 yields higher quality embeddings compared to a single source concept network as evidenced by the performance of all of these tasks. In addition, our method is better than SAM, MASE, and MASE-scaled. The three methods are even worse than a single source in some cases. A possible reason may be that they can not impose suitable weights on different sources. The collapsing method is similar to SAM in the sense that they try to estimate the common network by aggregating the data from all views. However, since it cannot fully exploit the relative signal and noise strength of each view, its performance is not as good as msLBM. The collapsing method is similar to SAM in that it tries to estimate the common network by aggregating data from all views. However, it fails to fully exploit the relative signal and noise strengths of each view, resulting in
inferior performance compared to msLBM.

| Method (Data Source) | Relatedness | Similarity |
|----------------------|-------------|------------|
| MIMIC                | 0.533       | 0.547      |
| Biobank              | 0.539       | 0.605      |
| Stanford             | 0.567       | 0.660      |
| msLBM                | **0.635**   | **0.686**  |
| SAM                  | 0.588       | 0.640      |
| MASE                 | 0.577       | 0.590      |
| MASE-scaled          | 0.517       | 0.577      |
| Collapse             | 0.601       | 0.662      |

Table 1: Spearman rank correlation under \( r = 250 \).

6.3 Coronary artery disease network

We next construct a disease network for Coronary Artery Disease (CAD), a leading cause of death involving multiple progression states. We set \( K = 1000 \) and Figure 5 suggests the disease network related to CAD is sparse since the magnitude of \( \tilde{\Omega} \) associated with the CAD CUI decays very fast. To further visualize the network, we focused on a subset of 371 CUIs that can have been previously identified as potentially related to CAD from 5 publicly available knowledge sources – including Mayo, Medline, Medscape, Merck Manuals, and Wikipedia – as in Yu et al. (2016). Algorithm 3 grouped these CUIs into 86 groups. We present in Figure 6 the CUIs groups that are most important for CAD as measured by the magnitude of \( \tilde{\Omega}_{kl} \) and in Figure 7 the CUIs included in each of the CUI groups. Our method can yield a very insightful network that unearths the progression states: Hyperlipidemia \( \gg \) Atherosclerosis \( \gg \) Angina \( \gg \) Myocardial Infarction (MI) \( \gg \) Congestive Heart Failure (CHF). Associated symptoms such as chest pain are also identified. In addition, our network successfully identifies medications important for CAD including Nitrate: for Angina and Myocardial Infarction; Beta-Blocker for MI and CHF; Anti-platelet for CAD, Angina, and MI. In addition to

| Method (Data Source) | MayCause | Causedby | Ddx | Bco | MayTreat |
|----------------------|----------|----------|-----|-----|----------|
| MIMIC                | 0.752    | 0.782    | 0.789 | 0.720 | 0.777    |
| Biobank              | 0.731    | 0.764    | 0.780 | 0.698 | 0.787    |
| Stanford             | 0.757    | 0.786    | 0.818 | 0.741 | 0.819    |
| msLBM                | **0.802**| **0.831**| **0.864**| **0.804**| **0.850**|
| SAM                  | 0.722    | 0.763    | 0.770 | 0.650 | 0.761    |
| MASE                 | 0.709    | 0.750    | 0.773 | 0.637 | 0.752    |
| MASE-scaled          | 0.700    | 0.739    | 0.762 | 0.647 | 0.751    |
| Collapse             | 0.770    | 0.806    | 0.820 | 0.795 | 0.821    |

Table 2: AUC of Clinical Relation Detection
### Table 3: TPR of Clinical Relation Detection with fixed FPR=0.05

| Method (Data Source) | MayCause | Causedby | Ddx | Bco | MayTreat |
|----------------------|----------|----------|-----|-----|----------|
| MIMIC                | 0.339    | 0.398    | 0.419| 0.238| 0.392    |
| Biobank              | 0.332    | 0.403    | 0.399| 0.238| 0.456    |
| Stanford             | 0.319    | 0.385    | 0.427| 0.233| 0.465    |
| msLBM                | **0.429**| **0.495**| **0.562**| **0.374**| **0.559**|
| SAM                  | 0.371    | 0.439    | 0.474| 0.239| 0.472    |
| MASE                 | 0.333    | 0.398    | 0.450| 0.205| 0.432    |
| MASE-scaled          | 0.336    | 0.396    | 0.448| 0.221| 0.444    |
| Collapse             | 0.364    | 0.434    | 0.436| 0.292| 0.477    |

### Table 4: TPR of Clinical Relation Detection with fixed FPR=0.1

| Method (Data Source) | MayCause | Causedby | Ddx | Bco | MayTreat |
|----------------------|----------|----------|-----|-----|----------|
| MIMIC                | 0.451    | 0.508    | 0.523| 0.359| 0.500    |
| Biobank              | 0.443    | 0.511    | 0.515| 0.356| 0.558    |
| Stanford             | 0.458    | 0.529    | 0.565| 0.420| 0.577    |
| msLBM                | **0.540**| **0.604**| **0.672**| **0.516**| **0.661**|
| SAM                  | 0.451    | 0.515    | 0.542| 0.330| 0.542    |
| MASE                 | 0.417    | 0.480    | 0.538| 0.275| 0.514    |
| MASE-scaled          | 0.409    | 0.473    | 0.531| 0.309| 0.514    |
| Collapse             | 0.490    | 0.554    | 0.568| 0.480| 0.595    |
recovering the disease network, our method also successfully grouped near identical concepts into meaningful concept groups as shown in Figure 7. For example, the CAD group consists of multiple synonymous concepts including “C0010068” for coronary heart disease, “C0010054” for coronary arteriosclerosis, “C0151744” for myocardial ischemia as well as “C0264694” for chronic myocardial ischemia. All these concepts are frequently used in clinical notes to describe CAD. We observe that CUIs indicative of chest pain have been split into two groups named “Chest Pain” and “Chest Discomfort” respectively by our method. While it might be ideal from a clinical perspective to merge them into a single chest pain concept group, such little defect is acceptable due to data quality and more importantly its data-driven nature that will not affect the quality of the overall learned network.

![Graph showing |Ω_{kl}| decay.](image)

Figure 5: |Ω_{kl}| decay.

7 Discussion

In this paper, we proposed an msLBM model to synthesize information to learn a consensus graph from multiple sources. Under the msLBM, we developed an alternating minimization algorithm to estimate the unknown parameters associated with the graph and provided convergence properties for the algorithm. Our model, methodologies, and theories are established under the assumption that $\mathbf{C} = \mathbf{U}\mathbf{U}^\top$ is positive semi-definite. This assumption is made for technical convenience. However, we can easily adapt our model, methodologies, and theories to accommodate an asymmetric shared correlation structure. If $\mathbf{C}$ is asymmetric, we can estimate its balanced factorization $\mathbf{C} = \mathbf{U}\mathbf{V}^\top$, 

33
Figure 6: CAD Disease network.

Figure 7: Consensus CAD network.
where $\mathbf{U}^\top \mathbf{U} = \mathbf{V}^\top \mathbf{V}$. The estimating procedures and algorithms can be easily adapted, and the theoretical guarantees will be almost the same as in the positive semi-definite case.

8 Acknowledgement

Dong Xia’s research was partially supported by Hong Kong RGC Grant GRF 16302020. Doudou Zhou’s research was partially supported by NUS Start-up Grant A-0009985-00-00.

References

Abbe, E. (2018). Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research* 18(177), 1–86.

Agency for Healthcare Research and Quality (2019). Clinical classifications software.

Agterberg, J., Z. Lubberts, and J. Arroyo (2022). Joint spectral clustering in multilayer degree-corrected stochastic blockmodels. *arXiv preprint arXiv:2212.05053*.

Arora, S., Y. Li, Y. Liang, T. Ma, and A. Risteski (2016). A latent variable model approach to pmi-based word embeddings. *Transactions of the Association for Computational Linguistics* 4, 385–399.

Arroyo, J., A. Athreya, J. Cape, G. Chen, C. E. Priebe, and J. T. Vogelstein (2021). Inference for multiple heterogeneous networks with a common invariant subspace. *Journal of Machine Learning Research* 22(142), 1–49.

Asgari, E. and M. R. Mofrad (2015). Continuous distributed representation of biological sequences for deep proteomics and genomics. *PloS one* 10(11), e0141287.

Athreya, A., D. E. Fishkind, M. Tang, C. E. Priebe, Y. Park, J. T. Vogelstein, K. Levin, V. Lyzinski, Y. Qin, and D. L. Sussman (2018). Statistical inference on random dot product graphs: a survey. *Journal of Machine Learning Research* 18(226), 1–92.

Awasthi, P., M. Charikar, R. Krishnaswamy, and A. K. Sinop (2015). The hardness of approximation of euclidean k-means. *arXiv preprint arXiv:1502.03316*.

Beam, A. L., B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, and I. S. Kohane (2019). Clinical concept embeddings learned from massive sources of multimodal medical data. In *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2020*, pp. 295–306. World Scientific.
Bhattacharyya, S. and S. Chatterjee (2018). Spectral clustering for multiple sparse networks: I. arXiv preprint arXiv:1805.10594.

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

Bodenreider, O. (2004). The unified medical language system (UMLS): integrating biomedical terminology. Nucleic acids research 32(suppl), D267–D270.

Cai, J.-F., J. Li, and D. Xia (2022). Generalized low-rank plus sparse tensor estimation by fast riemannian optimization. Journal of the American Statistical Association, 1–17.

Candès, E. J., X. Li, Y. Ma, and J. Wright (2011). Robust principal component analysis? Journal of the ACM (JACM) 58(3), 11.

Chen, J., Y. C. Leong, C. J. Honey, C. H. Yong, K. A. Norman, and U. Hasson (2017). Shared memories reveal shared structure in neural activity across individuals. Nature neuroscience 20(1), 115.

Choi, Y., C. Y.-I. Chiu, and D. Sontag (2016). Learning low-dimensional representations of medical concepts. AMIA Summits on Translational Science Proceedings 2016, 41.

Church, K. W. and P. Hanks (1990). Word association norms, mutual information, and lexicography. Computational linguistics 16(1), 22–29.

Dagan, I., F. Pereira, and L. Lee (1994). Similarity-based estimation of word cooccurrence probabilities. In Proceedings of the 32nd annual meeting on Association for Computational Linguistics, pp. 272–278. Association for Computational Linguistics.

Davis, C. and W. M. Kahan (1970). The rotation of eigenvectors by a perturbation. iii. SIAM Journal on Numerical Analysis 7(1), 1–46.

Finlayson, S. G., P. LePendu, and N. H. Shah (2014). Building the graph of medicine from millions of clinical narratives. Scientific Data 1, 140032.

Gao, C., Z. Ma, A. Y. Zhang, and H. H. Zhou (2017). Achieving optimal misclassification proportion in stochastic block models. Journal of Machine Learning Research 18(1), 1980–2024.

Goh, K.-I., M. E. Cusick, D. Valle, B. Childs, M. Vidal, and A.-L. Barabási (2007). The human disease network. Proceedings of the National Academy of Sciences 104(21), 8685–8690.

Holland, P. W., K. B. Laskey, and S. Leinhardt (1983). Stochastic blockmodels: First steps. Social Networks 5(2), 109–137.
Hong, C., E. Rush, M. Liu, D. Zhou, J. Sun, A. Sonabend, V. M. Castro, P. Schubert, V. A. Panickan, T. Cai, et al. (2021). Clinical knowledge extraction via sparse embedding regression (keser) with multi-center large scale electronic health record data. *NPJ digital medicine* 4(1), 151.

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

Johnson, A. E., T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark (2016, May). MIMIC-III, a freely accessible critical care database. *Scientific Data* 3(1).

Jolliffe, I. (2005). *Principal component analysis*. Wiley Online Library.

Jones, A. and P. Rubin-Delanchy (2020). The multilayer random dot product graph. *arXiv preprint arXiv:2007.10455*.

Karrer, B. and M. E. Newman (2011). Stochastic blockmodels and community structure in networks. *Physical Review E* 83(1), 016107.

Kelley, R. and T. Ideker (2005). Systematic interpretation of genetic interactions using protein networks. *Nature Biotechnology* 23(5), 561.

Koltchinskii, V. and D. Xia (2016). Perturbation of linear forms of singular vectors under gaussian noise. In *High Dimensional Probability VII*, pp. 397–423. Springer.

Kumar, A., Y. Sabharwal, and S. Sen (2004). A simple linear time (1+ epsilon)-approximation algorithm for k-means clustering in any dimensions. In *Annual Symposium on Foundations of Computer Science*, Volume 45, pp. 454–462. IEEE COMPUTER SOCIETY PRESS.

Le, C. M., K. Levin, E. Levina, et al. (2018). Estimating a network from multiple noisy realizations. *Electronic Journal of Statistics* 12(2), 4697–4740.

Lei, J., K. Chen, and B. Lynch (2020). Consistent community detection in multi-layer network data. *Biometrika* 107(1), 61–73.

Lei, J. and A. Rinaldo (2015). Consistency of spectral clustering in stochastic block models. *The Annals of Statistics* 43(1), 215–237.

Levin, K., A. Athreya, M. Tang, V. Lyzinski, and C. E. Priebe (2017). A central limit theorem for an omnibus embedding of multiple random dot product graphs. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 964–967. IEEE.
Levin, K., A. Lodhia, and E. Levina (2022). Recovering shared structure from multiple networks with unknown edge distributions. *Journal of Machine Learning Research* 23, 3–1.

Levy, O. and Y. Goldberg (2014). Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems*, pp. 2177–2185.

Lu, J., M. Kolar, and H. Liu (2017). Post-regularization inference for time-varying nonparanormal graphical models. *The Journal of Machine Learning Research* 18(1), 7401–7478.

Luscombe, N. M., M. M. Babu, H. Yu, M. Snyder, S. A. Teichmann, and M. Gerstein (2004). Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature* 431(7006), 308.

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

Mahajan, M., P. Nimbhorkar, and K. Varadarajan (2009). The planar k-means problem is np-hard. In *International Workshop on Algorithms and Computation*, pp. 274–285. Springer.

Nabieva, E., K. Jim, A. Agarwal, B. Chazelle, and M. Singh (2005). Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. *Bioinformatics* 21(suppl_1), i302–i310.

Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103(23), 8577–8582.

Ng, A. Y., M. I. Jordan, and Y. Weiss (2002). On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pp. 849–856.

Ng, P. (2017). dna2vec: Consistent vector representations of variable-length k-mers. *arXiv preprint arXiv:1701.06279*.

Nguyen, D., W. Luo, D. Phung, and S. Venkatesh (2016). Control matching via discharge code sequences. *arXiv preprint arXiv:1612.01812*.

Nielsen, A. M. and D. Witten (2018). The multiple random dot product graph model. *arXiv preprint arXiv:1811.12172*.

Pakhomov, S., B. McInnes, T. Adam, Y. Liu, T. Pedersen, and G. B. Melton (2010). Semantic similarity and relatedness between clinical terms: an experimental study. In *AMIA annual symposium proceedings*, Volume 2010, pp. 572. American Medical Informatics Association.

Paul, S., Y. Chen, et al. (2020). Spectral and matrix factorization methods for consistent community detection in multi-layer networks. *The Annals of Statistics* 48(1), 230–250.
Rohe, K., S. Chatterjee, B. Yu, et al. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics 39*(4), 1878–1915.

Scott, J. (1988). Social network analysis. *Sociology 22*(1), 109–127.

Shaw, P., K. Eckstrand, W. Sharp, J. Blumenthal, J. Lerch, D. Greenstein, L. Clasen, A. Evans, J. Giedd, and J. Rapoport (2007). Attention-deficit/hyperactivity disorder is characterized by a delay in cortical maturation. *Proceedings of the National Academy of Sciences 104*(49), 19649–19654.

Shi, J. and J. Malik (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*(8), 888–905.

Srebro, N. and T. Jaakkola (2003). Weighted low-rank approximations. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 720–727.

Steinhaus, H. (1956). Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci 1*(804), 801.

Tang, M., J. Cape, and C. E. Priebe (2022). Asymptotically efficient estimators for stochastic blockmodels: The naive mle, the rank-constrained mle, and the spectral estimator. *Bernoulli 28*(2), 1049–1073.

Tang, R., M. Tang, J. T. Vogelstein, and C. E. Priebe (2017). Robust estimation from multiple graphs under gross error contamination. *arXiv preprint arXiv:1707.03487*.

Tao, M. and X. Yuan (2011). Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM Journal on Optimization 21*(1), 57–81.

Tu, S., R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht (2016). Low-rank solutions of linear matrix equations via procrustes flow. In *International Conference on Machine Learning*, pp. 964–973. PMLR.

Turney, P. D. and P. Pantel (2010). From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research 37*, 141–188.

Vodrahalli, K., P.-H. Chen, Y. Liang, C. Baldassano, J. Chen, E. Yong, C. Honey, U. Hasson, P. Ramadge, K. A. Norman, et al. (2018). Mapping between fMRI responses to movies and their natural language annotations. *Neuroimage 180*, 223–231.

Wang, L., Z. Zhang, and D. Dunson (2019). Common and individual structure of brain networks. *The Annals of Applied Statistics 13*(1), 85–112.
Wang, S., J. Arroyo, J. T. Vogelstein, and C. E. Priebe (2019). Joint embedding of graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43(4), 1324–1336.

Wedin, P.-Å. (1972). Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics* 12(1), 99–111.

Xia, D. (2021). Normal approximation and confidence region of singular subspaces. *Electronic Journal of Statistics* 15(2), 3798–3851.

Xia, D. and M. Yuan (2017). On polynomial time methods for exact low-rank tensor completion. *Foundations of Computational Mathematics*, 1–49.

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

Xia, D. and F. Zhou (2019). The sup-norm perturbation of hosvd and low rank tensor denoising. *Journal of Machine Learning Research* 20(61), 1–42.

Xia, Y., T. Cai, and T. T. Cai (2018). Multiple testing of submatrices of a precision matrix with applications to identification of between pathway interactions. *Journal of the American Statistical Association* 113(521), 328–339.

Young, S. J. and E. R. Scheinerman (2007). Random dot product graph models for social networks. In *International Workshop on Algorithms and Models for the Web-Graph*, pp. 138–149. Springer.

Yu, S., A. Chakrabortty, K. P. Liao, T. Cai, A. N. Ananthakrishnan, V. S. Gainer, S. E. Churchill, P. Szolovits, S. N. Murphy, I. S. Kohane, et al. (2016). Surrogate-assisted feature extraction for high-throughput phenotyping. *Journal of the American Medical Informatics Association* 24(e1), e143–e149.

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

Zhao, Y., E. Levina, J. Zhu, et al. (2012). Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics* 40(4), 2266–2292.

Zhou, T. and D. Tao (2011). Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *International Conference on Machine Learning*. Omnipress.