Link Prediction and Denoising in Netowrks

by

Yun-Jhong Wu

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in the University of Michigan
2017

Doctoral Committee:

Professor Elizaveta Levina, Co-Chair
Professor Ji Zhu, Co-Chair
Assistant Professor Ambuj Tewari
Associate Professor Clayton Scott

Yun-Jhong Wu

yjwu@umich.edu

ORCID iD: 0000-0002-3470-4837

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

ABSTRACT

Link Prediction and Denoising in Netowrks
by
Yun-Jhong Wu

Co-Chairs: Professor Elizaveta Levina and Professor Ji Zhu

Network data represent connections between units of interests, but are often noisy and/or include missing values. This thesis focuses on denoising network data via inferring underlying network structure from an observed noisy realization. The observed network data can be viewed as a single random realization of an unobserved latent structure, and our general approach to estimating this latent structure is based factorizing it into a product of interpretable components, with structural assumptions on the components determined by the nature of the problem.

We first study the problem of predicting links when edge features are available, or node features that can be converted into edge features. We propose a regression-type model to combine information from network structure and edge features. We show that estimating parameters in this model is straightforward and the estimator enjoys excellent theoretical performance guarantees.

Another direction we study is predicting links in time-stamped dynamic networks. A common approach to modeling networks observed over time is aggregating the networks to a few snapshots, which reduces computational complexity, but also loses information. We address this limitation through a dynamic network model based on tensor factorization, which simultaneously captures time trends and the graph structure of dynamic networks without aggregating over time. We develop an efficient algorithm to fit this model and demonstrate the method performs well numerically.

The last contribution of this thesis is link prediction for ego-networks. Ego-networks are constructed by recording all friends of a particular user, or several users, which is widely used in survey-based social data collection. There are many methods for filling in missing data in a matrix when entries are missing independently at random, but here it is more appropriate to assume that whole rows of the matrix are missing (corresponding to users), whereas other rows are observed completely. We develop an approach to estimate missing links in this scenario via subspace estimation, exploiting potential low-rank structure common in networks. We obtain theoretical bounds on the estimator's performance and demonstrate it significantly outperforms many widely used benchmarks in both simulated and real networks.

CHAPTER 1

Introduction

This thesis focuses on link prediction in statistical network analysis. Network data consists of sets of nodes or vertices linked in pairs by edges. Examples of network data include social networks, metabolic networks, gene regulatory networks, neural networks, geographic networks, the Internet. In friendship networks collected from social media, for example, people have real life friends outside of social media, but they can be linked to people they barely know. In biology, regulatory links between genes or proteins are often inferred from experimental data, which may result in both false positives and false negatives. In security applications, when tracking communications within a suspected criminal network, for example, discovering previously unknown connections is one of the main goals. Statistical network analysis has become an important tool for understanding social and epidemiological dynamics, designing efficient and robust architecture of distributed systems, road traffic networks. Formally, a network $G = (V, E)$ of size n consists of a node set $V = \{1, \ldots, n\}$ and an edge set $E = \{(i, j); i, j \text{ are linked}\}$. For undirected networks, $(i, j) \in E$ if and only if (j, i); for weighted networks, a link is represented by a triple (i, j, w_{ij}). Another useful representation of networks is adjacency matrices. An adjacency matrix of a network of size n is a $n \times n$ matrix $\mathbf{A} = [A_{ij}]_{n\times n}$.

$$A_{ij} = \begin{cases} 1, & \text{if } i \text{ and } j \text{ are linked,} \\ 0, & \text{otherwise.} \end{cases}$$

and, for weighted networks, $A_{ij} = w_{ij}$. This thesis investigated statistical link prediction for several types of network data.

Link prediction is a fundamental problem in network analysis. Network data are often corrupted by random noise, missing values, and sampling procedures. Corruption in network data may significantly change network structure (Burt, 1987; Kossinets, 2006; Wang et al., 2012). In simple networks, the link prediction problem can be cast as classifying pairs of nodes into linked/unlinked categories, which can be done by a score-based proce-

dure. Given pairs of nodes $\{(i,j)\}_{i\neq j}$, we estimate a score function $f : \{(i,j)\}_{i\neq j} \mapsto \mathbb{R}$ and choose a threshold value c. Then, the estimated class label of (i,j) is assigned to be $1(f(i,j) > c)$. With a score-based classification procedure, a natural score for this classification task is $\mathbb{E}[A_{ij}] = \mathbb{P}(A_{ij} = 1)$, where $A_{ij} = 1$ if nodes i and j are linked. Therefore, link prediction can be done by estimating a score that is strictly monotonic with respect to $\mathbb{E}[A_{ij}]$ since $1(f(i,j) > c) = 1(g(f(i,j)) > g(c))$ for any strictly monotonic function g. Moreover, the formulation can adapt to link prediction in weighted networks and further connects with research on matrix completion and graphon estimation.

We consider the framework of factorization models. The basic idea of factorization models is to decompose each effect as a function of several factors. The number of factors are usually much fewer than that of data points. In the context of typical network analysis, a network of n nodes consists of $n(n-1)/2$ pairs of nodes, and each pair receive a specific effect from underlying factors. By factorizing these effects to r latent factors $Z_i \in \mathbb{R}^r$ for $i = 1, dots, n$, the number of parameters of a factorization model is therefore of $O(nr)$. This makes estimating parameters more tractable. Moreover, since nodes in a network live in a space with very weak topological structure, it is useful to interpret each Z_i as a latent node feature of node i. Latent factors provide a natural embedding to transform nodes into a metric space. We emphasize that the embedding not only gives an interpretable data visualization but also makes most statistical and machine learning methods for data in an Euclidean space able to be adapted to network data. For example, spectral clustering (von Luxburg et al., 2008; Rohe et al., 2011) is an algorithm to detect community structure in network by combining spectral decomposition and the K-means algorithm. Clearly, finding a proper factorization, or equivalently transformation from nodes to data points in a metric space, is essential for statistical network analysis. In the context of link prediction, we look for a model of the form

$$\mathbb{E}[A_{ij} \mid Z_1, \ldots, Z_n] = f(Z_i, Z_j; \theta).$$

My work makes contribution to design and analyze performance of factorization models for network data in various contexts.

Link prediction via low-rank effects models

Most existing methods predict links based on observed unweighted links, but many networks in the real world include additional information such as edge weights and features such as coauthorship networks (Leskovec et al., 2007), rating networks (Dror et al., 2012), and metabolic networks (Duch and Arenas, 2005). Although network topological features

have been used for predicting links, a natural question arises how to properly combine information in network topology and additional features for further improving the accuracy of link prediction. Some similarity-based methods can incorporate additional features (e.g. Hasan et al., 2006; Murata and Moriyasu, 2007; Doppa et al., 2009; Zhao et al., 2017), but most of the existing methods are unable to properly rescale and rewight multiple features based on importance of each feature.

We proposed a generalized linear model for predicting links. To incorporate edge weights and features, we model the expected weights of a link as $\mathbb{E}[A_{ij} \mid X_{ij}] = L(\beta^\top X_{ij} + Z_i^\top \mathbf{\Lambda} Z_j)$, where L is a link function, X_{ij} is an edge feature, and Z_i is a latent factor from node i. The latent factors can be expressed as a low-rank effect on $\mathbb{E}[A_{ij} \mid X_{ij}]$, which can characterize homophily and structure equivalence of nodes. In our model, β plays a role to adjust the importance of each feature and may screen out nuisance features. Furthermore, the relative magnitude of β and $\mathbf{\Lambda}$'s can properly reweight the contribution of information from network structure and additional features. Exploiting the low-rank structure, we proposed a nuclear-norm regularized maximum likelihood estimator. We proved the consistency of our estimator under some mild regularity conditions and demonstrated its empirical performance on simulated and data examples.

Link prediction for time-stamped dynamic networks

While network data are usually modeled as a static structure, networks in the real world often change over-time. For example, people interact with each other via sending E-mails, tweeting, face-to-face communication. The actions are performed only at certain time-points and the frequency of the actions can be time-varying. Many models of dynamic networks have been proposed such as latent space models (Sarkar and Moore, 2005; Fu et al., 2009; Hanneke et al., 2010; Xing et al., 2010; Yang et al., 2011; Kim and Leskovec, 2013; Richard et al., 2014; Sewell and Chen, 2015; Durante and Dunson, 2017), Markov processes of binary status (Xu, 2015; Zhang et al., 2016), the Cox intensity models (Vu et al., 2011; Perry and Wolfe, 2013), semi-parametric models (Matias et al., 2015), and non-parametric/over-parametrized methods (Sarkar et al., 2013; Li et al., 2014). These methods show a trade-off among model flexibility, computational efficiency, and interpretability, and most of the methods suffer from sparsity of networks.

In this project, we proposed a model based on the canonical tensor decomposition. We view counts of interactions between each pair of nodes as an inhomogeneous Poisson process. We model dependency between models through latent factors of structure $\mathbf{z}_\ell$'s and the intensity of factor $\mathbf{u}_\ell^*(t)$'s. Thus, the expected intensity between nodes i and j at time t can be expressed as $\sum_{\ell=1}^r \mathbf{z}_{\ell i}^\top \mathbf{z}_{\ell j} \mathbf{u}_\ell^*(t)$. We showed that a sufficient fine partition induces an

identifiable model, which can be expressed as the canonical tensor decomposition and can be estimated by solving a standard tensor decomposition problem with roughness penalty on the time dimension. We designed an algorithm with the local convergence guarantee to simultaneously determine the number of latent factors and provide warm-starts for a proximal block coordinate descent algorithm. We applied our method to synthetic data and the Enron E-mail data demonstrated that our method can efficiently produce interpretable results.

Link prediction for ego-networks

Most methods often implicitly assume entry-wise missing at random or entry-wise sample at random. That is, the observed adjacency matrix can be expressed as $\mathbf{A}_{obs} = \mathbf{A} \odot \mathbf{M}$, where $M_{ij} \overset{i.i.d.}{\sim} \text{Bernoulli}(p)$ and $\odot$ denotes the Hadamard product. Although under this assumption the link prediction and matrix completion problem has been extensively studied (Candès and Tao, 2010; Candès and Plan, 2010; Keshavan et al., 2010; Lin et al., 2010; Chatterjee, 2015; Davenport et al., 2014, e.g), many social network data collected via specific design of social surveys violate the missing-at-random assumption. A widely-used survey method is egocentric sample, which consists of selected nodes and links attached to the selected nodes, since it is easy to integrate to standard social surveys. The ego-network is a sub-network constructed from an egocentric sample, and correspondingly, we can think this sampling procedure as selecting rows and columns from an adjacency matrix. Exploiting this special sampling procedure enables us to recover the population network and further improve accuracy of link prediction.

In this project, we proposed a subspace estimation method for predicting links in ego-networks. We directly estimate principal sub-row-space of the underlying probability matrix by the subspace spanned by in-sample rows of the adjacency matrix. The estimated subspace can be viewed as an embedding of nodes in a pseudo-Euclidean space, and the complete estimator can be factorized as latent positions and a scalar product, which determines structure of the space that nodes embed into. Our model include a wide range of latent space models such as stochastic block models, dot-product models (Young and Scheinerman, 2007), latent eigenmodels (Hoff, 2007), and hyperbolic models (Krioukov et al., 2010; Albert et al., 2014). Our method is computationally efficient and numerically robust. We also showed that our estimator can accurately predict links from egocentric sample and significantly outperforms matrix completion/graph estimation algorithms.

CHAPTER 2

Low-rank effects models for network estimation with edge attributes

2.1 Introduction

Networks are widely used to represent and analyze data in many domains, for example, for social, biological, and communication systems. Each network consists of nodes and edges. For example, in social networks, nodes may correspond to people and edges represent friendships; in biological networks, nodes may correspond to genes or proteins while edges represent regulatory relationships. Besides nodes and edges, other information is often available in the form of node and/or edge covariates, such as people's demographic information or the closeness of a friendship in social networks, and proteins' chemical components or the strength of the regulatory relationship in biological networks.

One fundamental problem in network analysis is to understand the mechanism that generates the edges by estimating the expectation of the adjacency matrix, sometimes referred to as network denoising. The expectation gives probabilities of links for every pair, which can be further used to perform link prediction; in fact for link prediction any monotone transformation of the link probabilities is sufficient. For binary networks, link prediction can be framed as a classification problem, which presence/absence of edge as the class label for each paper, and some sort of score for each pair of nodes (e.g. an estimated probability of link) used to predict the class.

Most approaches to the estimating the probabilities of edges (or more generally scores) use the information from node features when available, and/or network topology such as the number of common neighbors, etc. Many approaches are based on homophily, which means that the more "similar" two nodes are, the more likely they are to become connected. Homophily has been widely observed in social networks (McPherson et al., 2001) and other contexts (Zhou et al., 2009). If homophily is assumed, estimating adjacency matrices is closely related to the question of how to measure similarity between nodes. For

node features, any appropriate similarity measure for vectors can be used. Multiple measures based on network topology are also available; see e.g., Section 3 in Lü and Zhou (2011). Other proposals include aggregating several similarities such as the number of 2-path and 3-path between two nodes (Zhou et al., 2009) and using kernels to measure the similarity between node pairs and combining it with the support vector machine (SVM) for classification in the context of estimating protein-protein interactions (Ben-Hur and Noble, 2005).

Alternatively, one can embed nodes in an Euclidean space and measure the similarity between nodes according to the distance between the nodes' latent positions. This approach includes various probabilistic network models such as the latent space model (Hoff et al., 2002), the latent variable model (Hoff, 2007), the latent feature model (Miller et al., 2009), the latent factor model (Hoff, 2009), the latent variable models with Gaussian mixture positions Krivitsky et al. (2009), and the Dirichlet network modelWilliamson (2016). In all these models, the latent positions have to be estimated via Markov Chain Monte Carlo (MCMC), which is very time consuming. More computationally efficient approaches have been developed. For example, the leading eigenvectors of the graph Laplacian can be used to embed the nodes in a low-dimensional space (e.g. Kunegis and Lommatzsch, 2009) by spectral decomposition, and their embedding coordinates can be veiwed at the latent node positions. Other recent efforts have been devoted to fitting latent space models by stochastic variational inference (Zhu, 2012) and gradient descent algorithms (Ma and Ma, 2017). The latter paper was written simultaneously and independently of the current work, and while it uses a similar algorithm in optimizaiton, it fits a different model, focuses on the problem of latent position estimation rather than link prediction, and, unlike ours, does not cover the directed case.

In another related line of work, graphon estimation methods estimate the edge probability matrix under node exchangeability and various additional assumptions on the matrix (smoothness, low-rankness, etc) (e.g. Choi and Wolfe, 2014; Yang et al., 2014; Olhede and Wolfe, 2014; Gao et al., 2015; Zhang et al., 2015). However, when node or edge features are available, exchangeability does not apply. Instead, a common approach is to aggregate information on the features and multiple similarity indexes to create a single score for predicting links. For example, Kashima et al. (2009) and Menon and Elkan (2011) treat topology-based similarities as edge attributes and propose an SVM-based approach for edge estimation.

Assumptions other than homophily have also been considered, such as hierarchical network structure (Clauset et al., 2008), structural equivalence (Hoff, 2007). In another approach, Zhao et al. (2017) used pair similarity instead of node similarity for edge predic-

tion, arguing that edges between similar pairs of nodes should have similar probability of occurring.

The problem of link prediction is also related to the problem of matrix completion, which is commonly solved under low rank constraints (e.g. Candès and Recht, 2009). In fact if the network is undirected and binary without any covariates, our proposed method is equivalent to the 1-bit matrix completion algorithm of Davenport et al. (2014), who established consistency of the maximum likelihood estimator for this setting. However, the 1-bit matrix completion formulation is much narrower: it does not allow for covariates and, crucially, assumes that the links are missing completely at random with equal probability, which is not a realistic assumption for networks.

The model we propose here represents the probability of an edge through a small number of parameters, like the latent space models; but unlike previous work, all we assume is a general low rank structure, without requiring anything more specific. This makes our method easily applicable to many types of networks: directed and undirected, binary and weighted, with and without node/edge covariates. Unlike latent space models, we do not require computationally expensive MCMC; instead, we fit the proposed model through an efficient projected gradient algorithm. In addition to computational efficiency, our method also has attractive theoretical properties, such as consistency for network estimation under the low rank assumption on the true probability matrix.

The rest of this article is organized as follows. The proposed model and the estimation algorithm are presented in Section 2.2. In Section 2.3, we establish several theoretical guarantees including consistency. Numerical evaluation of the proposed method and comparisons to other network estimation approaches on simulated networks are presented in Section 2.4. In Section 2.5, we illustrate the proposed method on two real networks, the friendship network of users of the Last.fm music website and the C. Elegans neural network. Section 2.6 concludes the paper with discussion and future work. All proofs are given in the Appendix. S

2.2 Generalized linear models for network data with low rank effects

We start with setting up notation. The data consist of a single observed $n \times n$ adjacency matrix $\boldsymbol{A} = [A_{ij}]_{n \times n}$, where A_{ij} represents the edge from node i to node j, which can be either a value binary (0/1) or a weight. If additional information on nodes and/or edges is available, we represent it as an m-dimensional attribute vector for each pair of nodes i

and j, denoted by $\boldsymbol{x}_{ij} = (x_{ij1}, \ldots, x_{ijm})^\top$. If the attributes are attached to nodes rather than edges, we convert them to edge attributes using a similarity measure, discussed in more detail below. Our goal is to compute a score for each pair of nodes to represent the strength of an edge that may connect them. A natural score is the expected value $\boldsymbol{P} = [p_{ij}]_{n\times n} = \mathbb{E}[\boldsymbol{A}]$. Then we can view the problem as a generalized regression question, fitting the model

$$p_{ij} = g(\boldsymbol{x}_{ij}),$$

where g is a mean function.

2.2.1 Generalized linear models for network data

A natural way to connect covariates to the strength of network edges is to use the generalized linear model (GLM). For example, logistic regression and logit models have been used for fitting binary directed networks (Wasserman and Pattison, 1996). It is straightforward to generalize this approach to various types of networks by considering a generalized linear model

$$L(p_{ij}) = \theta_{ij} + \boldsymbol{x}_{ij}^\top \boldsymbol{\beta}, \tag{2.2.1}$$

where L is a link function to be specified and $\boldsymbol{\beta} \in \mathbb{R}^m$ is a vector of coefficients. As normally done in GLM, we assume that the distribution of A_{ij} only depends on covariates through their linear combination with an unknown coefficient vector $\boldsymbol{\beta}$, and that edges are independent conditional on covariates. The parameter θ_{ij} represents an interaction between nodes i and j for $i, j = 1, \ldots, n$. Further assuming an exponential family distribution, the conditional distribution of $\boldsymbol{A}$ with the mean matrix $\boldsymbol{P}$ takes the canonical form

$$f_{(\boldsymbol{\Theta},\boldsymbol{\beta})}(\boldsymbol{A} \mid \mathcal{X}) = \prod_{ij} f_{(\theta_{ij},\boldsymbol{\beta})}(A_{ij} \mid \boldsymbol{x}_{ij}) = \prod_{ij} c(A_{ij}) \exp\Big(\eta_{ij} A_{ij} - b(\eta_{ij})\Big), \tag{2.2.2}$$

where $\boldsymbol{\Theta} = [\theta_{ij}]_{[n\times n]}$, $\mathcal{X} = [\boldsymbol{X}_1, \ldots, \boldsymbol{X}_m] \in \mathbb{R}^{n\times n\times m}$, $\boldsymbol{X}_k = [x_{ijk}]_{n\times n}$, $k = 1, \ldots, m$, $\eta_{ij} = \theta_{ij} + \boldsymbol{x}_{ij}^\top \boldsymbol{\beta}$, and the corresponding canonical link function is given by $L^{-1} = g = b'$. This general setting includes, for example, the logistic model for fitting binary networks and binomial and Poisson models for integer-weighted networks. Extending it to multinomial logistic models for networks with signed or labeled edges is also straightforward.

Model (2.2.1) involves more parameters than can be fitted without regularization or additional assumptions on $\boldsymbol{\Theta}$. One possibility is to impose regularization through the commonly occurring dependency among edges in networks known as transitivity: if A and B

are friends, and B and C are friends, then A and C are more likely to be friends. This idea has been utilized by Hoff (2005), in which the random effects model was extended to the so-called bilinear mixed-effects model to model the joint distribution of adjacent edges. Here we take a different and perhaps more general approach by imposing a low rank constraint on the effects matrix, implicitly inducing sharing information among the edges; this allows us to both model individual node effects and share information, which seems to be more appropriate for network data than the random effects modeling assumption of random and identically distributed θ_{ij}'s.

2.2.2 The low rank effects model

In general, regularization can be applied to either $\boldsymbol{\Theta}$ or $\boldsymbol{\beta}$ or both; a sparsity constraint on $\boldsymbol{\beta}$ would be natural when the number of attributes m is large, but the more important parameter to constrain here is $\boldsymbol{\Theta}$, which contains n^2 parameters. A natural general constraint that imposes structure without parametric assumptions is constraining the rank of $\boldsymbol{\Theta}$, assuming

$$L(\boldsymbol{P}) = \boldsymbol{\Theta} + \mathcal{X} \otimes \boldsymbol{\beta},\ \mathrm{rank}(\boldsymbol{\Theta}) \leq r, \tag{2.2.3}$$

where $\mathcal{X} \otimes \boldsymbol{\beta} = \sum_{k=1}^{m} \beta_k \mathbf{X}_k$, and, in a slight abuse of notation, $L(\boldsymbol{P})$ is the link function applied element-wise to the matrix $\boldsymbol{P}$.

The rank constrained model (2.2.3) is related to latent space models, for example, the eigenmodel proposed by Hoff (2007) for undirected binary networks. The projection model assumes that the edge probability is given by

$$\mathrm{logit}(p_{ij}) = \alpha + \mathbf{z}_i^\top \boldsymbol{\Lambda} \mathbf{z}_j + \boldsymbol{x}_{ij}^\top \boldsymbol{\beta}, \tag{2.2.4}$$

where $\mathbf{z}_i \in \mathbb{R}^{(r-1)}$ represents the position of node i in a latent space. Note that the $n \times n$ matrix $\alpha \mathbf{1}\mathbf{1}^\top + \mathbf{Z}\boldsymbol{\Lambda}\mathbf{Z}^\top$, where $\mathbf{Z} = [\mathbf{z}_1 \cdots \mathbf{z}_n]^\top \in \mathbb{R}^{n \times (r-1)}$, is at most of rank r. By setting L to be the logit link, the eigenmodel can be obtained as a special case of the low rank effects model (2.2.3), although the fitting method proposed for the eigenmodel by Hoff (2007) is much more computationally intensive.

Full identifiability for (2.2.3) requires additional assumptions, even though the mean matrix $\boldsymbol{P}$ is always identifiable and so is $\boldsymbol{\Theta} + \mathcal{X} \otimes \boldsymbol{\beta}$. For $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$ to be individually identifiable, $\mathcal{X} \otimes \boldsymbol{\beta}$ cannot be of low rank, and $\mathbf{X}_k$'s cannot be collinear. Formally, we make the following assumptions:

A1. $\mathrm{rank}(\mathcal{X} \otimes \boldsymbol{\beta}) > r$ for all $\boldsymbol{\beta} \neq \mathbf{0}$;

A2. $\mathrm{vec}(\mathbf{X}_1), \ldots, \mathrm{vec}(\mathbf{X}_m)$ are linearly independent.

Assumption A1 implies that $\mathcal{X} \otimes \boldsymbol{\beta}$ is linearly independent of $\boldsymbol{\Theta}$, and assumption A2 ensures that $\boldsymbol{\beta}$ is identifiable.

2.2.3 Estimation

In principle, estimates of $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$ can be obtained by maximizing the constrained log-likelihood as follows

$$(\overline{\boldsymbol{\Theta}}, \overline{\boldsymbol{\beta}}) = \underset{(\boldsymbol{\Theta}, \boldsymbol{\beta}): \mathrm{rank}(\boldsymbol{\Theta}) \leq r}{\arg\max} \ell_{\mathbf{A}, \mathcal{X}}(\boldsymbol{\Theta}, \boldsymbol{\beta}), \tag{2.2.5}$$

where $\ell_{\mathbf{A}, \mathcal{X}}$ is the log-likelihood based on the distribution in (2.2.2). Note the distinction between directed and undirected networks is not crucial here because the estimators will automatically be symmetric when $\mathbf{X}_1, \ldots, \mathbf{X}_m$ and $\mathbf{A}$ are symmetric.

Although in practice certain algorithms such as the alternating direction method may be applied to solve (2.2.5), no computationally feasible algorithm is guaranteed to find the global maximum due to the non-convexity of the rank constraint $\mathrm{rank}(\boldsymbol{\Theta}) \leq r$. To circumvent this, the rank constraint is often replaced with a convex relaxation (e.g. Candès and Recht, 2009). Let $\mathrm{conv}(\mathcal{S})$ denote the convex hull of set $\mathcal{S}$, $\sigma_i(\boldsymbol{\Theta})$ the i-th largest singular value of $\boldsymbol{\Theta}$, and $\|\boldsymbol{\Theta}\|_*$ the nuclear norm of $\boldsymbol{\Theta}$. Then a common relaxation is

$$\begin{aligned}
&\mathrm{conv}\{\boldsymbol{\Theta} : \mathrm{rank}(\boldsymbol{\Theta}) \leq r, \|\boldsymbol{\Theta}\|_2 \leq 1\} \\
&= \mathrm{conv}\{\boldsymbol{\Theta} : \boldsymbol{\Theta} \text{ has at most } r \text{ non-zero singular values and } \sigma_i(\boldsymbol{\Theta}) \leq 1 \ \forall i\} \\
&= \{\boldsymbol{\Theta} : \sum_{i=1}^{n} \sigma_k(\boldsymbol{\Theta}) \leq r\} = \{\boldsymbol{\Theta} : \|\boldsymbol{\Theta}\|_* \leq r\}.
\end{aligned}$$

Using this relaxation, one can estimate $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$ by solving the problem

$$(\widetilde{\boldsymbol{\Theta}}, \widetilde{\boldsymbol{\beta}}) = \underset{(\boldsymbol{\Theta}, \boldsymbol{\beta}): \|\boldsymbol{\Theta}\|_* \leq R}{\arg\max} \ell_{\mathbf{A}, \mathcal{X}}(\boldsymbol{\Theta}, \boldsymbol{\beta}), \tag{2.2.6}$$

where R is a tuning parameter. The exponential family assumption and the use of the nuclear norm ensure the strict convexity of (2.2.6) as a function of θ_{ij}'s and therefore the uniqueness of the maximum. Finally, the mean matrix $\boldsymbol{P}$ can be estimated by $\widetilde{\boldsymbol{P}} = L^{-1}(\widetilde{\boldsymbol{\Theta}} + \mathcal{X} \otimes \widetilde{\boldsymbol{\beta}})$.

The optimization problem (2.2.6) can be solved by the standard projected gradient algorithm (Boyd and Vandenberghe, 2009). Specifically, the main (block-coordinate) updating formulas are

1. $\beta_k^{(t+1)} \leftarrow \beta_k^{(t)} + \gamma_t \nabla_{\beta_k} \ell_{\mathbf{A},\mathcal{X}}(\boldsymbol{\Theta}^{(t)}, \boldsymbol{\beta})|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}}$ for $k = 1, \ldots, m$
2. $\boldsymbol{\Theta}^{(t+1)} \leftarrow \mathcal{P}\Big(\boldsymbol{\Theta}^{(t)} + \gamma_t \nabla_{\boldsymbol{\Theta}} \ell_{\mathbf{A},\mathcal{X}}(\boldsymbol{\Theta}, \boldsymbol{\beta}^{(t+1)})|_{\boldsymbol{\Theta}=\boldsymbol{\Theta}^{(t)}}\Big)$

where γ_t is a step size and $\mathcal{P}$ is a projection operator onto the set $\{\boldsymbol{\Theta} : \|\boldsymbol{\Theta}\|_* \leq R\}$. The first updating formula is the same as the standard gradient ascent algorithm since there is no constraint on $\boldsymbol{\beta}$. The second formula consists of a gradient ascent step and a projection operation to ensure that the algorithm produces a solution in the feasible set. Thus, for solving (2.2.6), we have

$$\begin{aligned}
\boldsymbol{\beta}_k^{(t+1)} &\leftarrow \boldsymbol{\beta}_k^{(t)} + \gamma_t \Big(\operatorname{tr}\big(\mathbf{X}_k^\top (\mathbf{A} - L^{-1}(\boldsymbol{\Theta}^{(t)} + \mathcal{X} \otimes \boldsymbol{\beta}^{(t)}))\big)\Big) \text{ for } k = 1, \ldots, m \\
\boldsymbol{\Theta}^{(t+1)} &\leftarrow \mathcal{P}_{c_t}\Big(\boldsymbol{\Theta}^{(t)} + \gamma_t \big(\mathbf{A} - L^{-1}(\boldsymbol{\Theta}^{(t)} + \mathcal{X} \otimes \boldsymbol{\beta}^{(t+1)})\big)\Big),
\end{aligned} \tag{2.2.7}$$

where $\mathcal{P}_{c_t}(\boldsymbol{\Theta}) = \sum_{i=1}^n (\sigma_i - c_t)_+ \mathbf{u}_i \mathbf{v}_i^\top$ $\mathcal{P}_{c_t}$ is a soft-thresholding operator, $\boldsymbol{\Theta} = \sum_{i=1}^n \sigma_i \mathbf{u}_i \mathbf{v}_i^\top$ is the singular value decomposition (SVD) of $\boldsymbol{\Theta}$, and $c_t = \arg\min_c \{\sum_{i=1}^n (\sigma_i - c)_+ \leq R\}$. Since the log-likelihood is continuously differentiable, convergence of the algorithm is guaranteed by choosing $\gamma_t < K^{-1}$ when the gradient of the log-likelihood is K-Lipschitz continuous on the feasible set. For example, in the case of the logit link, $K = 1$, and for the logarithm link (when the edge weight follows a Poisson distribution), $K = \exp(\|\boldsymbol{\Theta}\|_{\max} + \max_{(i,j)} \boldsymbol{x}_{ij}^\top \boldsymbol{\beta})$, where $\|\boldsymbol{\Theta}\|_{\max}$ denotes the maximum absolute entry of $\boldsymbol{\Theta}$. See Boyd and Vandenberghe (2009) for theoretical details and a variety of accelerated projected gradient algorithms.

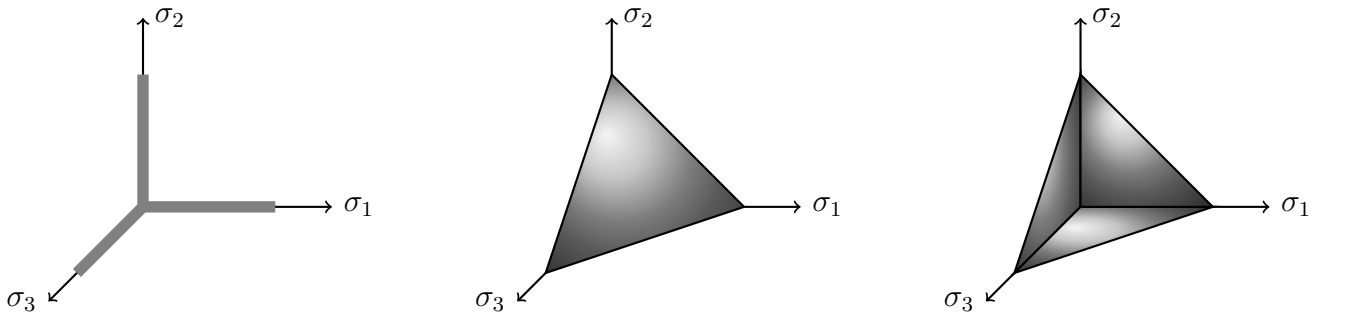
The updating formulas require solving a full SVD in each iteration, which can be computationally expensive, especially when n is large. In practice, if the matrix $\boldsymbol{\Theta}^{(t)} + \gamma_t\big(\mathbf{A} - L^{-1}(\boldsymbol{\Theta}^{(t)} + \mathcal{X} \otimes \boldsymbol{\beta}^{(t+1)})\big)$ is approximately low rank, solving the SVD truncated at rank s for some $s > r$ usually gives the same optimum. Thus, we consider an alternative criterion to (2.2.6) to estimate $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$, i.e.

$$(\widehat{\boldsymbol{\Theta}}, \widehat{\boldsymbol{\beta}}) = \underset{(\boldsymbol{\Theta},\boldsymbol{\beta}):\|\boldsymbol{\Theta}\|_* \leq R, \operatorname{rank}(\boldsymbol{\Theta}) \leq s}{\arg\max} \ell_{\mathbf{A},\mathcal{X}}(\boldsymbol{\Theta}, \boldsymbol{\beta}), \tag{2.2.8}$$

and solve the optimization problem by replacing the nuclear-norm projection operator in (2.2.7) with $\mathcal{P}_{(R,s)} = \sum_{i=1}^s (\sigma_i - c_t)_+ \mathbf{u}_i \mathbf{v}_i^\top$, with c_t as defined above. Finally, the mean matrix $\boldsymbol{P}$ is estimated by

$$\widehat{\boldsymbol{P}} = L^{-1}(\widehat{\boldsymbol{\Theta}} + \mathcal{X} \otimes \widehat{\boldsymbol{\beta}}).$$

Although the optimization problem in (2.2.8) is non-convex, as illustrated in Figure 2.1c, the algorithm is computationally efficient, and we will also show that the estimator enjoys theoretical guarantees similar to those of (2.2.6).



(a) $\mathrm{rank}(\mathbf{\Theta}) \leq 1$ (b) $\|\mathbf{\Theta}\|_* \leq R$ (c) $\mathrm{rank}(\mathbf{\Theta}) \leq 2$ and $\|\mathbf{\Theta}\|_* \leq R$

Figure 2.1: Constraints in optimization problems (2.2.5), (2.2.6), and (2.2.8) in the space of singular values of $\mathbf{\Theta}$

2.3 Theoretical properties

In this section, we show asymptotic properties of our estimates for the low rank GLM, in Frobenius matrix norm. We make the following additional assumptions on the parameter space and covariates:

A3. $\|\mathbf{\Theta}\|_{\max} \leq K_\theta$ and $\mathrm{rank}(\mathbf{\Theta}) \leq r$

A4. $\|\boldsymbol{\beta}\|_2 \leq K_\beta$

A5. $\|\boldsymbol{x}_{ij}\|_2 \leq K_x$ for all i, j

Theorem 1. *Under assumptions A3-A5, we have*

$$n^{-1}\|\widetilde{\mathbf{P}} - \mathbf{P}\|_F \xrightarrow{p} 0,$$

where $\widetilde{\mathbf{P}} = L^{-1}(\widetilde{\mathbf{\Theta}} + \mathcal{X} \otimes \widetilde{\boldsymbol{\beta}})$*, and* $\widetilde{\mathbf{\Theta}}$ *and* $\widetilde{\boldsymbol{\beta}}$ *are obtained from (2.2.6).*

Similarly, consistency of $\widehat{\mathbf{P}}$ can also be established.

Corollary 2. *Under assumptions A3-A5, we have,*

$$n^{-1}\|\widehat{\mathbf{P}} - \mathbf{P}\|_F \xrightarrow{p} 0,$$

where $\widehat{\mathbf{P}} = L^{-1}(\widehat{\mathbf{\Theta}} + \mathcal{X} \otimes \widehat{\boldsymbol{\beta}})$*, and* $\widehat{\mathbf{\Theta}}$ *and* $\widehat{\boldsymbol{\beta}}$ *are obtained from (2.2.8).*

The tail probabilities of both $n^{-1}\|\widetilde{\mathbf{P}} - \mathbf{P}\|_F$ and $n^{-1}\|\widehat{\mathbf{P}} - \mathbf{P}\|_F$ have a polynomially-decaying rate. We can obtain a better probability bound for some widely-used models such as logit models as stated in the following corollary.

Corollary 3. *Under the assumptions of Theorem 1, if A_{ij}'s are uniformly bounded, then both $n^{-1}\|\widetilde{\mathbf{P}} - \mathbf{P}\|_F$ and $n^{-1}\|\widehat{\mathbf{P}} - \mathbf{P}\|_F$ have an exponentially-decaying tail probability.*

Beyond $\widehat{\mathbf{P}}$, asymptotic properties of $\widehat{\mathbf{\Theta}}$ and $\widehat{\boldsymbol{\beta}}$ are also often of interest. If they are identifiable, the following corollary gives consistency for the parameters.

Corollary 4. *If assumptions A1-A5 hold,* $\inf_{ij} \mathrm{Var}(A_{ij}) > 0$*, and there exists* $0 < \delta < 1$ *such that*

$$\sup_{\boldsymbol{\beta}} \frac{\sum_{i=1}^{r+s} \sigma_i^2(\mathcal{X} \otimes \boldsymbol{\beta})}{\sum_{i=1}^{n} \sigma_i^2(\mathcal{X} \otimes \boldsymbol{\beta})} \leq \delta < 1, \tag{2.3.1}$$

then

$$n^{-1}\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}\|_F \xrightarrow{P} 0$$
$$\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_F \xrightarrow{P} 0.$$

Note that we can drop the supermum in the condition (2.3.1) as $\frac{\sum_{i=1}^{r+s} \sigma_i^2(\mathcal{X})}{\sum_{i=1}^{n} \sigma_i^2(\mathcal{X})} \leq \delta < 1$ if $\boldsymbol{\beta}$ is univariate and correspondingly $\mathcal{X}$ is a matrix.

The convex relaxation in (2.2.6) changes the feasible set, and in the new parameter space, ($\boldsymbol{\beta}$ and $\mathbf{\Theta}$) may no longer be identifiable. Therefore consistency of $\widetilde{\boldsymbol{\beta}}$ and $\widetilde{\mathbf{\Theta}}$ is not guaranteed.

A case of practical interest is when $\mathbf{\Theta}$ is only approximately rather than exactly low rank (i.e., has a few large leading eigenvalues and the other eigenvalues are relatively small but not necessarily 0). We can then show the bias of $\widetilde{\mathbf{P}}$ and $\widehat{\mathbf{P}}$ caused by model misspecification can be bounded as follows.

Theorem 5. *Under the assumptions of Theorem 1, except that* $\mathrm{rank}(\mathbf{\Theta}) > r$*, we have*

$$\frac{\|\widetilde{\mathbf{P}} - \mathbf{P}\|_F^2}{\sum_{k=r+1}^{n} \sigma_k(\mathbf{\Theta})} = O_P(1),$$

and

$$\frac{\|\widehat{\mathbf{P}} - \mathbf{P}\|_F^2}{\sum_{k=r+1}^{n} \sigma_k(\mathbf{\Theta})} = O_P(1).$$

This result suggests that our proposed estimates enjoy robustness under model misspecification if the eigenvalues following the first r are small. This holds even if r grows with n as long as $r = o(n)$. As an application of Theorem 5, we present the error bound for the low rank effects model for binary networks as an example.

Example 2.3.1 (Bias of the low rank effects logistic model)**.** *For logistic models,* $b(\eta) = \log(1 + e^{\eta})$*. Thus, by* (A.1.6) *in the proof of Theorem 5, we have*

$$\sum_{ij}\big(b(\widehat{\theta}_{ij} + \mathbf{x}_{ij}^{\top}\boldsymbol{\beta}) - b(\theta_{ij} + \mathbf{x}_{ij}^{\top}\boldsymbol{\beta})\big)$$
$$= \sum_{ij} \log\left(\frac{1 + e^{\widehat{\theta}_{ij} + \mathbf{x}_{ij}^{\top}\beta}}{1 + e^{\theta_{ij} + \mathbf{x}_{ij}^{\top}\beta}}\right) \leq n \sum_{k=r+1}^{n} \sigma_k(\boldsymbol{\Theta})$$

and therefore

$$\mathbb{P}\left(n^{-1}\|\widehat{\mathbf{P}} - \mathbf{P}\|_F \leq \left(2n^{-1} \sum_{k=r+1}^{n} \sigma_k(\boldsymbol{\Theta})\right)^{\frac{1}{2}}\right) \to 1.$$

2.4 Results on synthetic networks

In this section, we present numerical results on simulated data to demonstrate the finite sample performance of the proposed low rank effects model and compare to benchmark methods. For the sake of computational efficiency, we focus on the estimate given by (2.2.8).

We consider a generative model similar to (2.2.4), with the mean function given by

$$L(\mathbf{P}) = \mathbf{Z}\mathbf{Z}^{\top} + \alpha\mathbf{1}\mathbf{1}^{\top} + \mathcal{X} \otimes \boldsymbol{\beta}, \tag{2.4.1}$$

where $\mathbf{Z} \sim [N(0,1)]_{n\times(r-1)}$ with independent entries. For the feature tensor $\mathcal{X} = [\mathbf{X}_1, \mathbf{X}_2]_{n\times n\times 2}$, we first generate $\widetilde{\mathbf{X}} \sim [N(0,1)]_{n\times n}$ with independent entries and then compute $\mathbf{X} = \mathbf{U}\mathbf{V}^{\top}$, where $\mathbf{U}$ and $\mathbf{V}$ are obtained from $\widetilde{\mathbf{X}} \overset{SVD}{=} \mathbf{U}\mathbf{D}\mathbf{V}^{\top}$. Therefore, all singular values of both $\mathbf{X}_1$ and $\mathbf{X}_2$ are equal to 1. Specifically, so they are full rank. We set $n = 200$ and $r = 2$, and $\boldsymbol{\beta} = (c, -c)$. Given the mean function L and $\mathcal{X}$ and $\mathbf{Z}$, we generate conditionally independent edges. We vary the parameters α and c to investigate the density of the network and the relative importance of low rank effects and covariates.

As benchmarks, we fit the classical GLMs and latent models, with details given below. The estimation for latent models is based on 500 burn-in and 10,000 MCMC iterations in each setting. Following the evaluation method for link prediction in Zhao et al. (2017), all tuning parameters for the low rank effects model and latent models are selected with subsampling validation. Specifically, we create training data networks by setting randomly selected 20% of all edges to 0, and calculate the predictive area under the ROC curve

(AUC), which is defined as

$$\mathrm{AUC}(\mathbf{A}, \widehat{\mathbf{P}}) = \frac{\sum_{(i,j),(i',j')\in\mathcal{I}} 1(A_{ij}=0, A_{i'j'}>0, \widehat{p}_{ij} < \widehat{p}_{i'j'})}{\sum_{(i,j),(i',j')\in\mathcal{I}} 1(A_{ij}=0, A_{i'j'}>0)},$$

where $\mathcal{I}$ is the index set of the "held-out" edges. With the selected tuning parameter, we fit the model to the entire network to obtain $\widehat{\mathbf{P}}$. We then generate test networks $\mathbf{A}_{test}$ and compute $\mathrm{AUC}(\mathbf{A}_{test}, \widehat{\mathbf{P}})$. In simulation studies, we have also computed the relative mean squared error for $\boldsymbol{P}$, defined as $\mathrm{RMSE}(\widehat{\mathbf{P}}) = \|\widehat{\mathbf{P}} - \mathbf{P}\|_F / \|\mathbf{P}\|_F$.

2.4.1 Binary networks

By setting $L(p) = \mathrm{logit}(p)$ in model (2.4.1), we generated directed binary networks, with edges conditional on parameters generated independent Bernoulli random variables. For each training network, we generated 10 test networks using the same parameters and covariates to evaluate the predictive AUC. For each setting, we also computed the RMSE. The logistic regression model and the latent factor model (Hoff, 2009) were used as benchmarks.

Average results over 100 replications are shown in Figures 2.2 and 2.3. Although the low rank effects model (LREM) has a somewhat larger parameter RMSE when the networks are sparse (small values of α), it outperforms both logistic regression and the latent factor model in terms of predictive AUC. When the value of c is large, most of the signal comes from the covariates rather than the low rank effects, and thus LREM behaves similarly to logistic regression. However, when the value of c is small, LREM outperforms logistic regression, especially on predictive AUC, by properly combining the information from both the network and the node covariates. We also observed that our algorithm produced much more numerically stable results than the latent factor model, with vastly lower computational cost. For example, in this simulation, for each setting our algorithm can converge in a few minutes on one single laptop.

2.4.2 Integer-weighted networks

An important advantage of the proposed low rank effects model is that it extends trivially to weighted networks. Using the link function $L(p) = \log p$, we generated networks based on (2.4.1) with edges conditionally independent Poisson random variables. All other aspects of the simulation remain the same. We consider the Poisson model and the fixed rank nomination model (Hoff et al., 2013) for integer-weighted networks as benchmarks. Note the

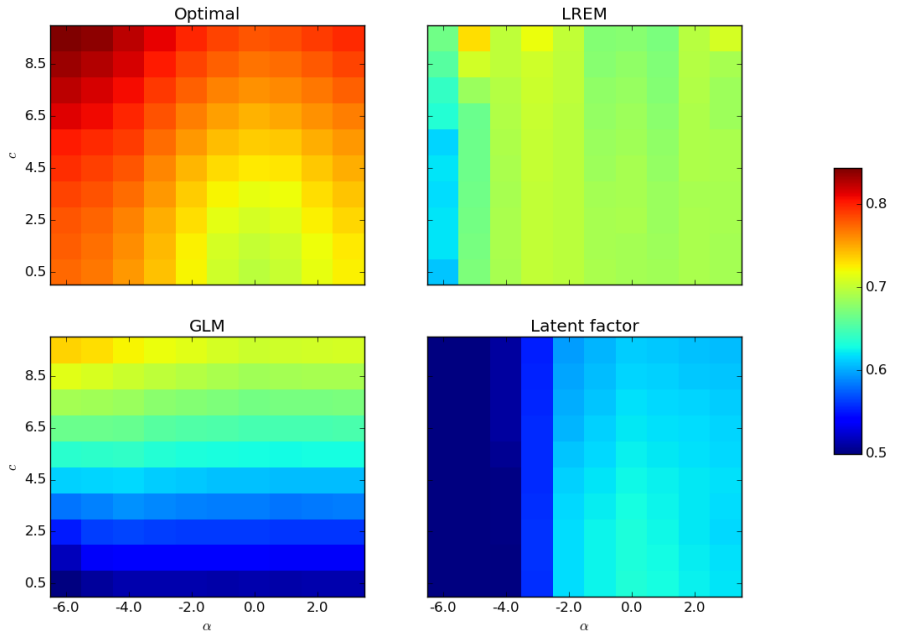


Figure 2.2: Predictive AUC for binary networks with various α and c. "Optimal" is the AUC based on the true mean matrix $\mathbf{P}$.

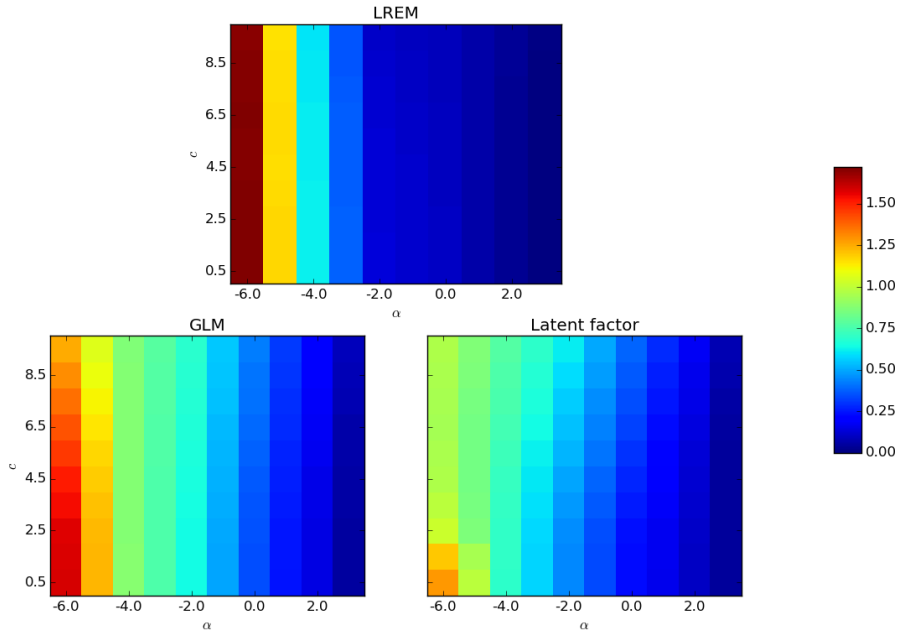


Figure 2.3: RMSE for $\boldsymbol{P}$ for binary networks with various α and c.

fixed rank nomination model was originally developed for networks with partial rank ordering relationships, but since integer weights can be viewed as the strength of relationships in this model, it is a natural benchmark for comparison. Since the AUC cannot be readily calculated on non-binary networks, we measure the performance based on "classifying" pairs of nodes that are connected ($A_{ij} > 0$) versus not connected ($A_{ij} = 0$).

Average results over 100 replications are shown in Figures 2.4 and 2.5. In terms of predictive AUC, which is more relevant in practice, the low rank effects model substantially outperforms the Poisson model and the fixed rank nomination model, except for the largest values of α where the fixed rank nomination model performs slightly better.

For the RMSE, the low rank effects model performs much better for all but the largest values of both c and α, which correspond to dense networks with high variation in node degrees. In this setting for integer-weighted networks, one needs a larger sample size in order to obtain an accurate estimate of $\mathbf{P}$, which is consistent with the theoretical results in Theorem 1 and Corollary 3.

2.5 Data examples

Next, we apply the proposed low rank effects model to two real-world datasets. To evaluate the performance, we randomly set 20% of the entries in the adjacency matrices to 0 and compute the predictive AUC on this "hold-out" set. This evaluation mechanism corresponds to the setting of partially observed networks discussed in Zhao et al. (2017). Reported results are averages over 20 repetitions.

2.5.1 The Last.fm friendship data

This dataset from the Last.fm music website friendships and 17,632 artists listened to or tagged by each user (Cantador et al., 2011). The friendship network contains 1,892 nodes (users) and 12,717 edges. We constructed two edge attributes $\mathbf{X}_{\text{lis(ten)}}$ and $\mathbf{X}_{\text{tag}}$ as follows: let $\widetilde{X}_{\text{lis},ij}$ and $\widetilde{X}_{\text{tag},ij}$ be the number of artists who are listened to and tagged by, respectively, both users i and j. These counts were then normalized, setting $X_{\text{lis},ij} = \widetilde{X}_{\text{lis},ij} / \max_{ij}\{\widetilde{X}_{\text{lis},ij}\}$ and $X_{\text{tag},ij} = \widetilde{X}_{\text{tag},ij} / \max_{ij}\{\widetilde{X}_{\text{tag},ij}\}$.

The prediction results are shown in Figure 2.6. The low rank effects model with covariates obtains the best AUC value of 0.876 at $r = 42$ and $R = 470$. Although this value of AUC is likely to be overly optimistic, note that the predictive AUC of the low rank effects model is larger than 0.75 over the entire range of parameters r and R, where as the logistic regression model only gives the AUC of 0.412. This suggests that modeling low rank

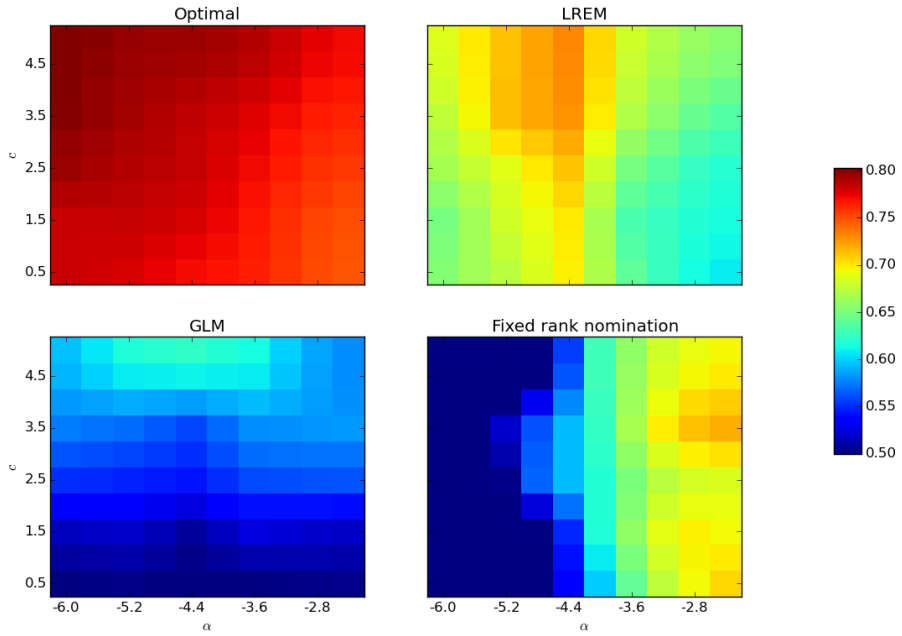


Figure 2.4: Predictive AUC for integer-weighted weighted networks with various α and c. "Optimal" refers to the AUC based on the true mean matrix $\mathbf{P}$.

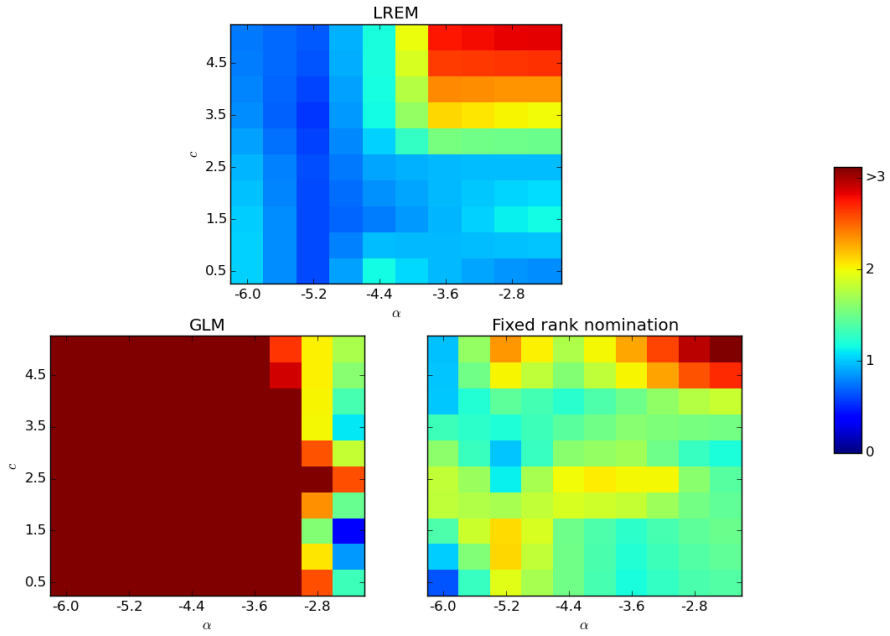


Figure 2.5: RMSE for $\boldsymbol{P}$ for integer-weighted weighted networks with various α and c.

pairwise effects is important for this dataset. The latent factor model (implemented via the package `amen` in R) failed to converge due to the size of the dataset.

In Figure 2.7, both $\widehat{\beta}_{\text{lis}}$ and $\widehat{\beta}_{\text{tag}}$ are positive and indicate that the Last.fm friendship network likely follows the principle of homophiliy. The rank constraint r has very little effect on the estimates of the coefficients, while the estimates shrink toward 0 as the nuclear-norm constraint R decreases due to the bias caused by a small R and the fact that $\|\widehat{\boldsymbol{\Theta}}\|_{\max} \leq \|\widehat{\boldsymbol{\Theta}}\|_* \leq R$.

2.5.2 The Elegans neural network data

This dataset contains the neural network of the nematode worm C. elegans, which is a directed integer-weighted network with 297 nodes. In this network, an edge represents a synapse or a gap junction between two neurons (Watts and Strogatz, 1998), and the weight between a pair of nodes is the number of edges between two neurons. The mean weight is 29.69 and 2.66% of pairs have non-zero weights. The original dataset does not contain any covariates. Therefore, we did not consider the classical GLM here, and the fixed rank nomination model was used as the benchmark model. Similar to the simulation studies for integer-weighted networks, we calculated the AUC based on "classifying" connected versus non-connected pairs.

Figure 2.8 shows the results from the low rank effects model. The AUC obtains the maximum 0.824 at $r = 26$ and $R = 85$, which is roughly the same as the best performance of the fixed rank nomination model (AUC=0.821, fitted by 1,000 burn-in and 20,000 MCMC iterations, which is vastly more expensive computationally). The relatively high value of AUC indicates that there might be a low-rank effect associated with the observed network.

2.6 Discussion

We proposed a generalized linear model with low-rank effects for network data with covariates, and an efficient projected gradient descent algorithm to fit this model. The model is more general than the various latent space models(Hoff et al., 2002; Hoff, 2007, 2009; Ma and Ma, 2017) because we do not require the effect matrix to be positive definite or symmetric, allowing for more general graph structures like bipartite graphs, and incorporating the directed case automatically. The simultaneous work of Ma and Ma (2017) is the only scalable algorithm we are aware of for fitting relatively general latent space models, but it is still less general than ours; and all previous work relied on MCMC and did not

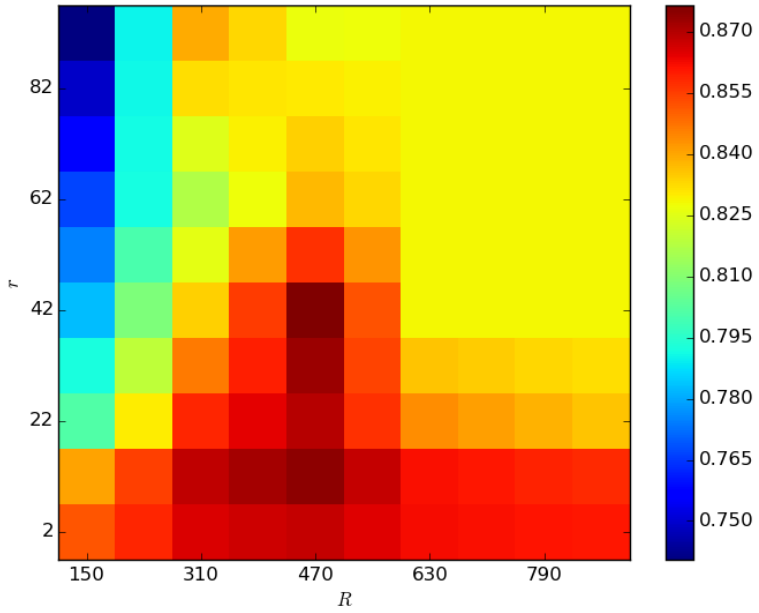
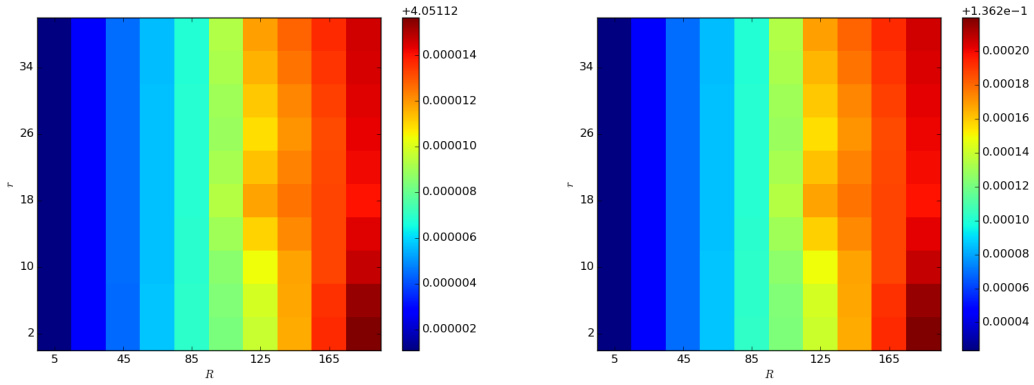


Figure 2.6: Predictive AUC for the Last.fm music dataset with various tuning parameters r and R.



(a) $\widehat{\beta}_{\text{lis}}$ (b) $\widehat{\beta}_{\text{tag}}$

Figure 2.7: Estimated coefficients for the Last.fm music dataset with various tuning parameters r and R.

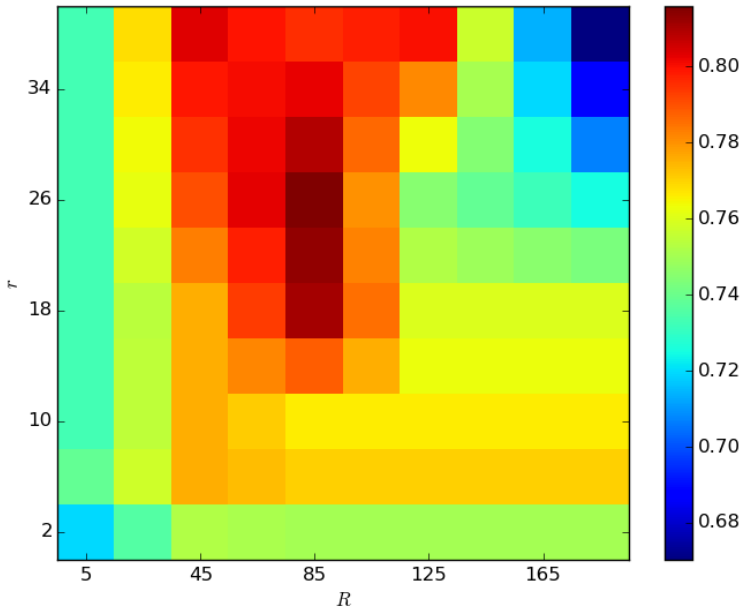


Figure 2.8: Predictive AUC for the neural dataset with various values of r and R.

scale well at all.

Figure 2.9 shows a simple comparison between the computational cost of our method and that of the latent factor model, for the simulation settings in this section. For both methods, we show the relative cost for fitting binary networks described in Section 2.4.1. Compared to the case of $n = 200$, it takes about 40 times of computational time for fitting the case of $n = 2000$ for our method and about 120 times for the latent factor model. The latent factor model becomes not feasible for networks with 10^5 or more nodes.

There are several directions of future work to explore. Any algorithm based on the SVD is in general considered not scalable to very large networks. Boosting the computational speed of SVD-based algorithms usually relies on the sparsity of decomposed matrices, which does not apply to the low rank effects model even if the data network is sparse. An alternative approach is the alternating direction method, which may find the global optimum when the estimator is obtained by minimizing the squared error loss under constraints. However, generalizing the algorithm to the GLM setting is not trivial. A stochastic gradient descent approach can also be applied to improve scalability.

An obvious extenstion in the setting of high-dimensional covariates is to incorporate variable selection via penalties on $\boldsymbol{\beta}$. It should also be relatively straightforward to adapt this framework to modeling dynamic networks, where different networks are observed at

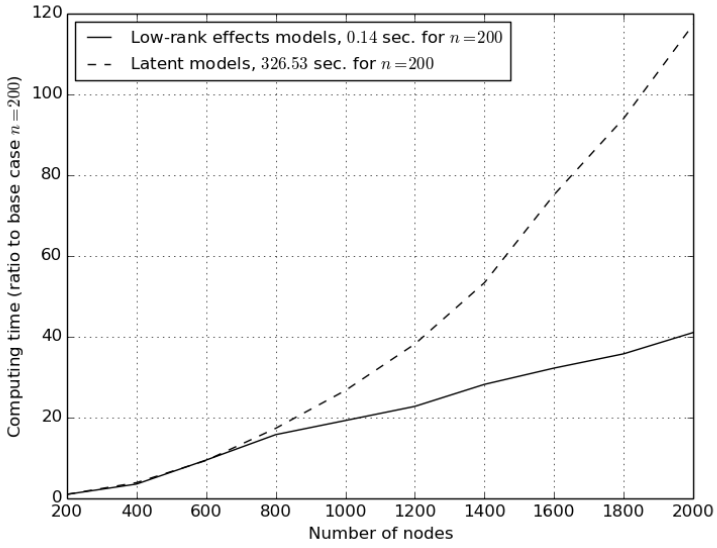


Figure 2.9: Comparison of the computing time of the low rank effects model (using Python) and that of the latent factor model (using the R package `amen`), relative to their computing time when $n = 200$.

different time points, with an underlying smoothly changing low rank probability matrix structure.

CHAPTER 3

Regularized tensor decomposition for link prediction in dynamic networks

3.1 Introduction

Developing realistic models for network data is a key challenge of network analysis. Considerable progress has been made in moving beyond simplistic random graph models for static networks that the field started with, such as the Erdos-Renyi graph or the planted partition model, towards more complex models that allows, for instance, mixed membership (Airoldi et al., 2008), overlapping communities (Latouche et al., 2011; Psorakis et al., 2011), and node covariates (Fellows and Handcock, 2012; Fosdick and Hoff, 2015). Most of this effort has been for static networks, but the networks we treat as static often result from interactions observed between individuals over time, such as sending emails or co-authoring papers. Many empirical analyses of such network data are based on either modeling global summary statistics, e.g., edge density, as a time series, or aggregating links at different time points into a single static network, or perhaps a small number of those (e.g. Chapanond et al., 2005; Keila and Skillicorn, 2005; Diehl et al., 2007). These approaches may be sufficient for answering some practical questions of interest but do not allow for modeling the dynamics of the whole network over time.

Statistical models aimed at modeling the entire network structure over time have been proposed more recently, often based on extending algorithms developed for static network problems, such as community detection or link prediction, to networks changing over time. One popular approach is through latent variables which often (but not always) represent community memberships; this line of work includes the mixed-membership model (Fu et al., 2009), the exponential random graph model (Hanneke et al., 2010), the state-space model (Xing et al., 2010), the mixed-group model (Yang et al., 2011; Kim and Leskovec, 2013), the autoregressive model (Richard et al., 2014), the latent space model (Sarkar and Moore, 2005; Sewell and Chen, 2015), locally adaptive dynamic networks (Durante and

Dunson, 2017), and smoothing-spectral-clustering procedure (Pensky and Zhang, 2017). These models represent nodes as points in a metric space that move over time and characterize network dynamics by a measure of distance between the nodes in that space, but identifiability is often a challenge. Moreover, these approach require MCMC or forward-backward MAP algorithms to fit hierarchical models and do not work well for sparse networks or data with more than a few dozen time points.

The temporal aspect of dynamic networks can also be modeled with Markov processes and characterized by transition probabilities or intensity functions. This approach includes the Cox intensity model (Perry and Wolfe, 2013) and semi-parametric regression models for longitudinal data (Matias et al., 2015), but time-varying covariates are required in these methods. Other approaches based on Markov processes focus on modeling persistent links by Markov chains of binary status (Xu, 2015; Zhang et al., 2016), but stationarity is crucial for success of these methods. Other more flexible approaches include predicting links via kernel density estimation (Sarkar et al., 2013) and restricted Boltzmann machine Li et al. (2014). These methods are fully nonparametric or over-parameterized and thus tend to lack interpretability, and are also computationally expensive.

Many dynamic network datasets are stored as time-stamped links. Graph snapshots over fixed periods can be easily constructed from time-stamped links, and significantly reduce storage costs. Formally, we consider raw data recorded as time-stamped links $\{(i_k, j_k, t_k)\}_{k=1}^m$, meaning node i_k interacted with node j_k at time t_k, with $1 \leq k \leq m$, $1 \leq i_k, j_k \leq n$, and $t_1 \leq t_2 \leq \cdots \leq t_m$. For simplicity, we assume that observed links are undirected, but our model, algorithm, and theory can be easily extended to directed networks.

We model the interactions between nodes over time as an inhomogeneous Poisson process $A_{ij}(t) = \#\{\text{interations between } i \text{ and } j \text{ before time } t\}$ for $i, j = 1, \ldots, n$. Formally, the number of interactions between nodes i and j in the time interval $(t, t + \Delta t]$ follows the distribution

$$A_{ij}(t + \Delta t) - A_{ij}(t) \sim \text{Poisson}\bigg(\int_t^{t+\Delta t} \lambda_{ij}(s) ds \bigg), \tag{3.1.1}$$

where $\lambda_{ij}(m)$ is a smooth intensity function. We further parametrize the intensity function as

$$\lambda_{ij}(t) = \lambda(t; \mathbf{z}_i, \mathbf{z}_j) = \sum_{\ell=1}^{r} z_{\ell i} z_{\ell j} u_\ell^*(t),$$

where $\mathbf{z}_i \in \mathbb{R}_+^r$ are negative vectors for $i = 1, \ldots, n$ and $u_\ell^*(t)$'s are non-negative smooth functions. This parametrization represents intensity functions as a product of time trends

$u_\ell^*(t)$ and network structure components $\mathbf{z}_i$, which play the role of latent variables. We will see that this factorization can be described through a tensor decomposition.

An order-3 tensor $\mathcal{A} = [a_{ijk}]_{n_1 \times n_2 \times n_3}$ is an array with entries with three indexes; see Kolda and Bader (2009) for a thorough review on basic operations and properties of tensors. A canonical polyadic decomposition (CANDECOMP/PARAFAC), or a Kruskal decomposition of $\mathcal{A}$, represents each entry by $a_{ijk} = \sum_{\ell=1}^r x_{\ell i} y_{\ell j} z_{\ell k}$, which can also be written as $\mathcal{A} = \sum_{\ell=1}^r \mathbf{x}_\ell \otimes \mathbf{y}_\ell \otimes \mathbf{z}_\ell$. This decomposition has been widely used for modeling higher order arrays, such as movie recommendations, video, and fMRI data (e.g. Abdallah et al., 2007; Karatzoglou et al., 2010; Pang et al., 2010; Zhao et al., 2011), and has recently be applied to network community detection (Anandkumar et al., 2013). Given a data tensor, the factors are usually obtained by solving

$$\min_{\mathbf{x}_\ell, \mathbf{y}_\ell, \mathbf{z}_\ell} \|\mathcal{A} - \sum_{\ell=1}^r \mathbf{x}_\ell \otimes \mathbf{y}_\ell \otimes \mathbf{z}_\ell\|_F$$

for $\mathbf{x}_\ell \in \mathbb{R}^{n_1}$, $\mathbf{y}_\ell \in \mathbb{R}^{n_2}$, and $\mathbf{z}_\ell \in \mathbb{R}^{n_3}$. This decomposition is analogous to matrix factorization

$$\min_{\mathbf{x}_\ell, \mathbf{y}_\ell} \|\mathbf{A} - \sum_{\ell=1}^r \mathbf{x}_\ell \mathbf{y}_\ell^\top\|_F.$$

Although this optimization problem is non-convex and in general considered intractable, many heuristic algorithms have been proposed to solve it, such as alternating least squares (ALS) and its variants (Carroll and Chang, 1970; Harshman, 1970; Kolda and Bader, 2009), matrixizing and nuclear norm regularization (Gandy et al., 2011; Tomioka et al., 2011; Tomioka and Suzuki, 2013), and power methods (De Lathauwer et al., 2000). Algorithms have also been developed for solving tensor decompositions for factors with more structure, such as non-negativity and robustness (Xu and Yin, 2013), incoherence (Anandkumar et al., 2014), and sparsity (Sun et al., 2016). These advances in algorithms for tensor decompositions enable us to directly model time-stamped links without having to collapse the data in the time dimension.

Our main contribution is a new model for time-stamped dynamic networks based on tensor decomposition. This model represents network dynamics by decomposing it into several rank-1 network structure components, which have the same interpretation as in latent variable models, and their corresonding time trends, which allow for different structures to emerge and/or disappear at different times. We do not require stationarity overtime, instead only requiring a certain amount of smoothness in the component intensity over time, and identifyability conditions analogous to incoherence. This approach enables us to analyze dynamic networks with a large number of time points and/or very sparse network

Table 3.1: Notations for modeling dynamic networks.

Notation	Description
n	Number of nodes
n_T	Number of time intervals
$\mathbb{R}_+^n$	Space of n-dimensional non-negative vectors
$\mathcal{A}$	Data tensor in $\mathbb{R}_+^{n\times n\times n_T}$
$\mathbf{z}_\ell$	ℓ-th network structure component in $\mathbb{R}_+^n$
$E_{ij}(t)$	Count of interactions between nodes i and j before time t
u_ℓ^*	Intensity function of ℓ-th component $\mathbf{u}_\ell$
$\mathbf{u}_\ell$	ℓ-th time component in $\mathbb{R}_+^{n_T}$
σ_ℓ	Positive weight of the ℓ-th component
$\mathbf{\Omega}$	$n_T \times n_T$ roughness penalty matrix
$\otimes$	Cartesian product
$\otimes_i$	Tensor-vector contraction along i-th dimension
$\|\cdot\|_2$	Spectral norm of a matrix/tensor
$\|\cdot\|$	ℓ_2 norm of a vector
$\|\cdot\|_F$	Frobenius norm of a matrix/tensor

snapshots, both of which are very challenging for previously proposed models. We also develop an efficient and parallelizable algorithm to fit this model, based on regularized tensor decomposition, as well as a theoretical guarantee of local convergence for this non-convex problem. Experiments on synthetic networks demonstrate our method performs very well numerically. We also applied this method to the Enron email data set which has almost two million time points and obtained interpretable results.

The rest of the article is organized as follows. In Section 3.2, we describe an algorithm based on a high-order power method and block coordinate descent. Theoretical convergence results are presented in Section 3.3. Section 3.4 reports numerical results of applying our algorithm to synthetic networks and the Enron email dataset. Section 3.5 concludes with discussion.

3.2 Modeling time-stamped links

We model the interactions between nodes i and j as an inhomogeneous Poisson process $E_{ij}(t)$ as described in the previous section, which is a natural model for counts. Without loss of generality, assume that the time stamps span the time interval $[0, T]$. Further, we factorize the intensity into components representing the network structure and its changes

in time. Thus, the count of links between nodes i and j in the time interval $[t, t+\Delta t)$ is modeled as

$$E_{ij}(t+\Delta t) - E_{ij}(t) \sim \text{Poisson}\bigg(\sum_{\ell=1}^{r} z_{\ell i} z_{\ell j} \int_{t}^{t+\Delta t} u_{\ell}^{*}(s) ds \bigg). \tag{3.2.1}$$

The network structure is represented with a sum of r rank-1 components $\mathbf{z}_\ell \mathbf{z}_\ell^\top$, where $\mathbf{z}_\ell = (z_{\ell 1}, \ldots, z_{\ell n})$ and $\|\mathbf{z}_\ell\| = 1$ for all $\ell = 1, \ldots, r$. The intensity of the ℓ-th rank-1 structure may change over time and is determined by $u_\ell^*(t)$, which is assumed to be integrable over $[0, T]$ for all $0 \le T < \infty$.

3.2.1 Tensor representation and identifiability

Given a time interval partition $\{t_0 = 0 < t_1 < \cdots < t_{n_T} = T\}$ on $[0, T]$, we consider a collection of adjacency matrices $\{\mathbf{A}_k\}_{k=1}^{n_T}$, where each $\mathbf{A}_k \in \mathbb{Z}^{n \times n}$ represents all the links formed in the time interval $(t_k, t_{k+1}]$. This collection of $n \times n$ matrices forms a $n \times n \times T$ tensor $\mathcal{A} = [\mathbf{A}_k]_{k=1}^{n_T} = [A_{ijk}]_{n \times n \times n_T}$, where $A_{ijk} = E_{ij}(t_k) - E_{ij}(t_{k-1})$ is the count of time-stamped link between i and j in $(t_{k-1} - t_k]$. Then, with the inhomogeneous Poisson model, we can express our model as a Kruskal decomposition of tensors as follows.

$$\mathbb{E}[\mathcal{A}] = \sum_{\ell=1}^{r} \sigma_\ell \mathbf{z}_\ell \otimes \mathbf{z}_\ell \otimes \mathbf{u}_\ell = \bigg[\sum_{\ell=1}^{r} \sigma_\ell z_{\ell i} z_{\ell j} u_{\ell k} \bigg]_{n \times n \times n_T}, \tag{3.2.2}$$

where $u_{\ell k}$ is proportional to $\int_{t_{k-1}}^{t_k} \tilde{u}_\ell(t) dt$. To make these factors identifiable, we normalize the temporal factors as well, setting $\|\mathbf{u}_\ell\| = 1$, and introduce σ_ℓ to represent the "strength" of the ℓth component in model 3.2.2. We assume $r \le \min\{n, n_T\}$; in practice, r should be much smaller than n and T. We also assume that all factors are non-negative to ensure the non-negativity of the Poisson intensity functions.

The identifiability of model (3.2.2) can be formally verified by checking the uniqueness conditions for the Kruskal decomposition. For $r = 1$, the decomposition is unique (Kolda and Bader, 2009). For $r > 1$, a sufficient condition for uniqueness, known as the Kruskal condition (Kruskal, 1977), is given by

$$2 \cdot k\text{-rank}(\mathbf{Z}) + k\text{-rank}(\mathbf{U}) \ge 2r + 2, \tag{3.2.3}$$

where $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_r)$, $\mathbf{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_r\}$, and

$$k\text{-rank}(\mathbf{V}) = \arg\max_{m} \{\widetilde{V} \subset \text{columns of } \mathbf{V} \text{ are linearly independent for all } |\widetilde{V}| = m\}.$$

Hence model (3.2.2) is identifiable if we assume that both $\mathbf{Z}$ and $\mathbf{U}$ have linearly independent column vectors. Uniqueness of the decomposition follows from the fact that the k-rank of each set of vectors is r (Kolda and Bader, 2009). In fact, the columns of $\mathbf{U}$ are linearly independent if the time partition is sufficiently fine as long as the u_ℓ^*'s are continuous on $[0, T]$ and linearly independent, as stated in the following theorem.

Theorem 6. *Assume the intensity functions u_ℓ^*'s are continuous and linearly independent, i.e. $\sum_{\ell=1}^r \alpha_\ell u_\ell^*(t) \equiv 0$ for all $t \in [0, T]$ if and only if all $\alpha_\ell = 0$. Then there exists a partition $t_0 = 0 < t_1 < \cdots < t_{n_T} = T$ such that the column vectors of $\mathbf{U}$ are linearly independent.*

The proof is given in Appendix B.1.

Thus a coarse time partition not only loses information, but may also lead to unidentifiability, and a fine time partition is preferred. Note that we allow for some $\mathbf{A}_k$'s to be matrices of 0s, that is, there is no problem if no interactions take place in a given time interval. In practice, one also has to consider the computational burden added by a fine time partition; fortunately, the time and space cost for our algorithms is linear in the number of time points n_T, as discussed in the next section.

3.2.2 The optimization criterion

We fit the the tensor model (3.2.2) by minimizing the following penalized least squares loss,

$$g(\mathbf{z}_1, \ldots, \mathbf{z}_r, \mathbf{u}_1, \ldots, \mathbf{u}_r; \gamma, \mathbf{\Omega}) = \left\| \mathcal{A} - \sum_{\ell=1}^r \sigma_\ell \mathbf{z}_\ell \otimes \mathbf{z}_\ell \otimes \mathbf{u}_\ell \right\|_F^2 + \gamma \sum_{\ell=1}^r \mathbf{u}_\ell^\top \mathbf{\Omega} \mathbf{u}_\ell, \quad (3.2.4)$$

The penalty term encourages smoothness of $u_\ell(t)$'s, with γ a tuning parameter to emphasize the smoothness of u_ℓ's.The penalty is imposed via constructing an appropriate smoothing matrix $\mathbf{\Omega} \in \mathbb{R}^{n_T \times n_T}$; here we use

$$\mathbf{\Omega} = \mathbf{W}\mathbf{H}\mathbf{H}^\top\mathbf{W}^\top,$$

where $\mathbf{W} = \mathrm{diag}(\frac{1}{t_1 - t_0}, \frac{1}{t_2 - t_1}, \ldots, \frac{1}{t_{n_T} - t_{n_T - 1}}) \in \mathbb{R}^{n_T \times n_T}$ and $\mathbf{H} = [h_{ij}] \in \mathbb{R}^{n_T \times (n_T - 1)}$ with

$$h_{ij} = \begin{cases} -1, & \text{if } i = j, \\ 1 & \text{if } i - j = 1, \\ 0, & \text{otherwise.} \end{cases}$$

The matrix $\mathbf{\Omega}$ constructs a discretized version of penalizing the integral $\int (u_\ell'(t))^2 dt$, a standard smoothness penalty. Many choices of smoothing penalties are available; we use the tri-diagonal $\mathbf{\Omega}$ because of its great computational efficiency and good results in numerical experiments.

3.3 The algorithm

Recall that fitting our dynamic network model requires solving the optimization problem (3.2.4),

$$\begin{aligned} &\underset{\mathbf{Z},\mathbf{U}}{\text{minimize}} \left\| \mathcal{A} - \sum_{\ell=1}^{r} \sigma_\ell \mathbf{z}_\ell \otimes \mathbf{z}_\ell \otimes \mathbf{u}_\ell \right\|_F^2 + \gamma \sum_{\ell=1}^{r} \mathbf{u}_\ell^\top \mathbf{\Omega} \mathbf{u}_\ell \\ &\text{subject to } \mathbf{z}_\ell \in \mathbb{R}_+^n, \mathbf{u}_\ell \in \mathbb{R}_+^{n_T}, \|\mathbf{z}_\ell\| = 1, \|\mathbf{u}_\ell\| = 1. \end{aligned} \tag{3.3.1}$$

This objective is multi-convex since the projection of the objective function onto each $\mathbf{z}_\ell$ and $\mathbf{u}_\ell$ is a convex function, but it is not convex. Therefore convergence of efficient algorithms to the global optimum of (3.3.1) is not guaranteed; however, many existing algorithms can efficiently find local minima of multi-convex functions under mild conditions, discussed next.

3.3.1 Proximal coordinate descent algorithm

Xu and Yin (2013) proposed a proximal block coordinate descent (proximal BCD) algorithm to find a critical point for multi-convex optimization problems. Applying the proximal BCD to (3.3.1) requires solving the following two optimization problems:

$$\begin{aligned} &\min_{\mathbf{z}_i} g(\mathbf{z}_1^{(k)}, \ldots, \mathbf{z}_{i-1}^{(k)}, \mathbf{z}_i, \mathbf{z}_{i+1}^{(k-1)}, \ldots, \mathbf{z}_r^{(k-1)}, \mathbf{u}_1^{(k-1)}, \ldots, \mathbf{u}_r^{(k-1)}; \gamma, \mathbf{\Omega}) \\ &\qquad + L\|\mathbf{z}_i - \mathbf{z}_i^{(k-1)}\|^2 \\ &\text{subject to } \mathbf{z}_i \in \mathbb{R}_+^n, \|\mathbf{z}_i\| = 1 \end{aligned} \tag{3.3.2}$$

and

$$\begin{aligned} &\min_{\mathbf{u}_i} g(\mathbf{z}_1^{(k)}, \ldots, \mathbf{z}_r^{(k-1)}, \mathbf{u}_1^{(k)}, \ldots, \mathbf{u}_{i-1}^{(k)}, \mathbf{u}_i, \mathbf{u}_{i+1}^{(k-1)}, \ldots, \mathbf{u}_r^{(k-1)}; \gamma, \mathbf{\Omega}) \\ &\qquad + L\|\mathbf{u}_i - \mathbf{u}_i^{(k-1)}\|^2 \\ &\text{subject to } \mathbf{u}_i \in \mathbb{R}_+^{n_T}, \|\mathbf{u}_i\| = 1. \end{aligned} \tag{3.3.3}$$

cyclical for $i = 1, \ldots, r$. The proximal BCD guarantees finding a critical point of g, with a polynomial rate of convergence. The number of components r has to be pre-determined, and in practice the proximal BCD may be sensitive to the choice of initial values.

3.3.2 Initial values and estimation of rank

To run the proximal BCD, we need to choose reasonably good initial values generates as well as estimate r. To do this, we will take advantage of a distinct property of canonical tensor decompositions that matrices do not have. Consider a rank-r canonical decomposition of a tensor defined as a solution to the problem

$$\min \left\| \mathcal{A} - \sum_{\ell=1}^{r} \sigma_\ell \mathbf{y}_\ell \otimes \mathbf{z}_\ell \otimes \mathbf{u}_\ell \right\|_F^2. \tag{3.3.4}$$

Assuming enough incoherence between components, Anandkumar et al. (2014) proposed an approach based on rank-1 updates to solve the problem (3.3.4) . They first find several critical points of the rank-1 approximation problem

$$\min_{\mathbf{y},\mathbf{z},\mathbf{u}} \|\mathcal{A} - \sigma \mathbf{y} \otimes \mathbf{z} \otimes \mathbf{u}\|_F^2 \tag{3.3.5}$$

and use these points as initial values for solving (3.3.4) by a block coordinate descent algorithm. Solving (3.3.5) by the alternating least squares method is equivalent to applying the updating equations

$$\begin{cases} \mathbf{y}^{(m)} = \phi(\mathcal{A} \otimes_2 \mathbf{z}^{(m-1)} \otimes_3 \mathbf{u}^{(m-1)}), \\ \mathbf{z}^{(m)} = \phi(\mathcal{A} \otimes_1 \mathbf{y}^{(m)} \otimes_3 \mathbf{u}^{(m-1)}), \\ \mathbf{u}^{(m)} = \phi(\mathcal{A} \otimes_1 \mathbf{y}^{(m)} \otimes_2 \mathbf{z}^{(m)}), \end{cases} \tag{3.3.6}$$

which is an analogue of the matrix power iteration algorithm for tensors. Most operations of the power iterations can be executed in-place in memory, so the algorithm has great efficiency and scalabiliy.

Notably, solving the rank-1 problem by the proximal algorithm in (3.3.2) and (3.3.3), we obtain the following updating equations:

$$\begin{aligned} \mathbf{z}^{(m)} &\leftarrow \phi(\rho\mathcal{A} \otimes_1 \mathbf{z}^{(m-1)} \otimes_3 \mathbf{u}^{(m-1)} + (1-\rho)\mathbf{z}^{(m-1)}) \,, \\ \mathbf{u}^{(m)} &\leftarrow \phi((\mathbf{I} + \gamma\boldsymbol{\Omega})^{-1}(\rho\mathcal{A} \otimes_1 \mathbf{z}^{(m)} \otimes_2 \mathbf{z}^{(m)} + (1-\rho)\mathbf{u}^{(m-1)})) \,, \end{aligned}$$

where $\rho = (L+1)^{-1}$. The resulting algorithm is an analogue of shifted power iterations

for matrices and enjoys a convergence guarantee inherited from the proximal BCD. Since multiple threads of power iterations are embarrassingly parallel, it is possible to search for multiple critical points simultaneously. Further, the number of distinct critical points detected is a natural estimate of the number of components r. The shifted power iterations are therefore a great option for computing warm starts for (3.3.1) by the proximal BCD. Algorithm 1 summarizes the entire procedure.

Algorithm 1 (Regularized tensor decomposition)

Input: data tensor $\mathcal{A} \in \mathbb{R}^{n\times n\times n_T}$, $\epsilon > 0$, $\rho \in (0,1)$, $\eta > 0$, $\mathbf{\Omega} \in \mathbb{R}^{n_T\times n_T}$, $\gamma > 0$.
for each $I = 1, \ldots, N$ **do** (in parallel)
    Randomly generate $\mathbf{z}_I^{(0)} \in \mathbb{R}_+^n$ and $\mathbf{u}_I^{(0)} \in \mathbb{R}_+^n$, where $\|\mathbf{z}_I\| = 1$ and $\|\mathbf{u}_I\| = 1$.
    while $\|\mathbf{z}_I^{(m-1)} - \mathbf{z}_I^{(m)}\| > \epsilon$ or $\|\mathbf{u}_I^{(m-1)} - \mathbf{u}_I^{(m)}\| > \epsilon$ or **do**
        $\mathbf{z}_I^{(m+1)} \leftarrow \phi(\rho\mathcal{A} \otimes \mathbf{z}_I^{(m)} \otimes \mathbf{u}_I^{(m)} + (1-\rho)\mathbf{z}_I^{(m)})$
        $\mathbf{u}^{(m+1)} \leftarrow \phi((\mathbf{I} + \gamma\mathbf{\Omega})^{-1}(\rho\mathcal{A} \otimes \mathbf{z}_I^{(m+1)} \otimes \mathbf{z}_I^{(m+1)} + (1-\rho)\mathbf{u}_I^{(m)}))$,
    end while
end for
$\mathcal{I} = \{1, \ldots, N\}$.
while $\mathcal{I} \neq \emptyset$ **do**
    Randomly choose $I \in \mathcal{I}$.
    $\widetilde{\mathcal{I}} = \widetilde{\mathcal{I}} \cup I$.
    Remove I' from $\mathcal{I}$ for all $\|\mathbf{z}_{I'} - \mathbf{z}_I\| < \eta$ and $\|\mathbf{u}_{I'} - \mathbf{u}_I\| < \eta$.
end while
$\widehat{r} \leftarrow |\widetilde{\mathcal{I}}|$.
Apply proximal BCD to solve (3.3.1) for $\mathbf{Z}, \mathbf{U}$, with $r = \widehat{r}$ and initial values $\mathbf{Z}^{(0)} = [\mathbf{z}_{I_1} \cdots \mathbf{z}_{I_r}]_{I\in\widetilde{\mathcal{I}}}$ and $\mathbf{U}^{(0)} = [\mathbf{u}_{I_1} \cdots \mathbf{u}_{I_r}]_{I\in\widetilde{\mathcal{I}}}$.
return $\mathbf{Z}, \mathbf{U}$

3.3.3 Convergence analysis

Since the problem is not convex and the solution depends on initial values, we focus on the local convergence properties of the proposed algorithm. We will show that the power iterations provide good initial values with high probability under certain regularity conditions. To state the conditions, we first quantify certain properties of the model.

Incoherence. Intuitively, each network component $\mathbf{z}_k$ should characterize a different aspect of the network structure, and each $\mathbf{u}_k^*$ should be a unique time trend to make the model identifiable. To quantify this, define

$$\eta_z = \max_{k,\ell=1,\ldots,r,k\neq\ell} \mathbf{z}_k^\top \mathbf{z}_\ell,$$

and, given a partition $0 = t_0 < t_1 < \cdots < t_{n_T}$, let

$$\begin{aligned}\eta_u &= \max_{k,\ell=1,\ldots,r,k\neq\ell} \mathbf{u}_k^\top \mathbf{u}_\ell \\ &= \max_{k,\ell=1,\ldots,r,k\neq\ell} \frac{\sum_{i=1}^{n_T} \int_{t_{i-1}}^{t_i} u_k^*(t)dt \int_{t_{i-1}}^{t_i} u_\ell^*(t)dt}{\sqrt{\sum_{i=1}^{n_T} (\int_{t_{i-1}}^{t_i} u_k^*(t)dt)^2 \sum_{i=1}^{n_T} (\int_{t_{i-1}}^{t_i} u_\ell^*(t)dt)^2}}\end{aligned}$$

Note that $\max\{\eta_z, \eta_u\} < 1$ automatically ensures identifiability by the Kruskal condition (3.2.3).

For example, let

$$z_{ki} \propto \begin{cases} \alpha & \text{if } \lfloor r(i-1)/n \rfloor = k, \\ \beta & \text{if } \lfloor r(i-1)/n \rfloor \neq k, \end{cases} \tag{3.3.7}$$

which represents a block structure. Then, $\eta_z = \mathbf{z}_k^\top \mathbf{z}_\ell = \frac{2n\alpha\beta + \beta^2 r(n-2)}{n\alpha^2 + r(n-1)\beta^2}$. When the number of factors r is fixed, $\eta_z \to 0$ as $\beta/\alpha \to 0$. On the temporal factors, for example, let $u_1^*(t) = (t+1)^{-\alpha}$ and $u_2^*(t) = (t+1)^{-\beta}$. Then, given a partition $\{t_i = i\}_{i=1}^{n_T}$ on $[0, T]$, η_u can be bounded as follows:

$$\frac{\int_1^{T+1} (t+1)^{-(\alpha+\beta)} dt}{\sqrt{\int_0^T (t+1)^{-2\alpha} dt \int_0^T (t+1)^{-2\beta} dt}} \leq \eta_u \leq \frac{\int_0^T (t+1)^{-(\alpha+\beta)} dt}{\sqrt{\int_1^{T+1} (t+1)^{-2\alpha} dt \int_1^{T+1} (t+1)^{-2\beta} dt}}.$$

Then, $\eta_u \to 0$ as $T \to \infty$ if $\alpha > 0.5 \geq \beta$, and $\eta_u \geq c > 0$ for some c if $\alpha, \beta \geq 0.5$ or $\alpha, \beta < 0.5$.

Relative magnitude of signals. Let $\sigma_{\min} = \min_\ell \sigma_\ell$, $\sigma_{\max} = \max_\ell \sigma_\ell$, and $\omega = \frac{\sigma_{\min}}{\sigma_{\max}}$. The quantity ω represents detectability of weak signals corresponding to small σ_k's. A small ω leads to a hardness to find the factors corresponding to $\sigma_{\min}$.

The following theorem can be viewed as detectability of true factors. A factor is detectable if there exists initial values that converge to a neighborhood of some factor through the power iteration. A factor is more likely to be detected if ω is large or η_z are η_u are sufficient small.

Theorem 7 (Local convergence of power iterations)**.** *Given a contraction rate $\nu \in (0, 1)$, with probability $p(\delta) = 1 - \exp(-\frac{\sigma_{\max}^2 \delta^2}{8+4\delta} + (2n + n_T)\log 15)$, where δ depends only on $r, \omega, \eta_z, \eta_u, \gamma, \rho$, and ν (see equation* (B.2.11)*), there exists $0 < s_z^- < s_z^+ < 1$ and $0 < s_u^- < s_u^+ < 1$ such that*

1. *If $\sqrt{1 - (\mathbf{z}_k^\top \mathbf{z}^{(m)})^2} = \epsilon_z \in (s_z^-, s_z^+)$ and $\sqrt{1 - (\mathbf{u}_k^\top \mathbf{u}^{(m)})^2} = \epsilon_u \in (s_u^-, s_u^+)$, the*

power iterations always improve accuracy of estimates $\sqrt{1-(\mathbf{z}_k^\top \mathbf{z}^{(m+1)})^2} < \nu\epsilon_z$ *and* $\sqrt{1-(\mathbf{u}_k^\top \mathbf{u}^{(m+1)})^2} < \nu\epsilon_u$ *for all* m *satisfying* $\nu\epsilon_z > s_z^-$ *and* $\nu\epsilon_u > s_u^-$.

2. *The power iterations guarantee that* $\sqrt{1-(\mathbf{z}_k^\top \mathbf{z}^{(m+1)})^2} < s_z^-$ *and* $\sqrt{1-(\mathbf{u}_k^\top \mathbf{u}^{(m+1)})^2} < s_u^-$ *for all* $m \in \mathbb{N}$ *if* $\sqrt{1-(\mathbf{z}_k^\top \mathbf{z}^{(m)})^2} < s_z^-$ *and* $\sqrt{1-(\mathbf{u}_k^\top \mathbf{u}^{(m)})^2} < s_u^-$.

Furthermore, s_z^+ *and* s_u^+ *is bounded below by a constant that depends only on* $r, \omega, \eta_z, \eta_u, \gamma,$ *and* ν.

Proof. See Appendix B.2. □

Remark 1. *Assume that* $\sigma_{\max} = \Theta(n\sqrt{n_T})$. *The probability* $p(\delta) \to 0$ *for any given* $\delta > 0$ *if* $\frac{1}{\sqrt{n_T}} + \frac{\sqrt{n_T}}{n} \to 0$.

An intuitive explanation of this theorem is shown in Figure 3.1. If the power iterations start within the inner (yellow) ball, they never leave the ball. Furthermore, since the shifted power iteration is also a proximal BCD and the objective function in (3.3.1) is a polynomial function, Theorem 2.8 in Xu and Yin (2013) ensures that the estimates eventually converge to the critical point. If they start in the outer (purple) ring, they will eventually move into the inner ball. If they start outside the purple ring, we cannot guarantee that they will converge to the critical point $\mathbf{z}_k$, but they will converge to a critical point by Theorem 2.8 in Xu and Yin (2013).

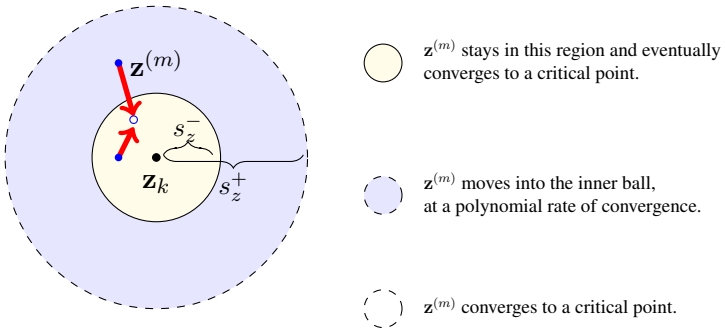


Figure 3.1: An illustration of local convergence of the power iteration described in Theorem 7: Blue points $\mathbf{z}^{(m)}$ converge to a critical point closed to some $\mathbf{z}_k$.

Sketch of proof of Theorem 7. The full proof is given in Appendix B.2.3. It contains three steps:

1. Prove a concentration inequality for the spectral norm of centered Poisson tensors in Lemma 10. Then the algorithm applied to the noisy data tensor $\mathcal{A}$ should behave similarly to how it would behave on the model-based population version $\mathbb{E}[\mathcal{A}]$.

2. Establish a bound on the cosine of the angle between one true component and its estimate after one power iteration, i.e., on $\sqrt{1-(\widehat{\mathbf{z}}^\top \mathbf{z}_k)^2}$ (Lemma 11) and $\sqrt{1-(\widehat{\mathbf{u}}^\top \mathbf{u}_k)^2}$ (Lemma 12), where $\widehat{\mathbf{z}}$ and $\widehat{\mathbf{u}}$ are produced by one step of the power iteration.

3. Conclude Theorem 7 by showing that the power iteration improves the estimate if the starting value is good enough.

Example 3.3.1 (Non-overlapping components)**.** *Let* $D = 4(\nu\omega + r)\sigma_{\max}^{-1}\|\mathcal{E}\|_2$. *When* $\eta_u = \eta_z = 0$ *and* $\rho = 0$*, i.e. the factors are orthogonal, by Theorem 7, with probability at least* $1 - \exp(-\frac{\sigma_{\max}\delta^2}{8+4\delta} + (2n + n_T)\log 15)$*, where* $\delta = \frac{\nu^2\omega^2}{4(\nu\omega + C(\gamma)r)}$*, there exists intervals* (s_z^-, s_z^+) *and* (s_u^-, s_u^+) *such that both initial values* $\mathbf{z}^{(0)}$ *and* $\mathbf{u}^{(0)}$ *can be improved by the power iterations if* $\sqrt{1-(\mathbf{z}_k^\top \mathbf{z}^{(0)})^2} \in (s_z^-, s_z^+)$ *and* $\sqrt{1-(\mathbf{u}_k^\top \mathbf{u}^{(0)})^2} \in (s_u^-, s_u^+)$ *for some* $k = 1, \ldots, r$*, where*

$$
\begin{aligned}
s_z^+ &= \frac{\nu\omega + \sqrt{\nu^2\omega^2 - D}}{2(\nu\omega + r)}, \\
s_z^- &= \frac{\nu\omega - \sqrt{\nu^2\omega^2 - D}}{2(\nu\omega + r)}, \\
s_u^+ &= \frac{\nu\omega + \sqrt{\nu^2\omega^2 - C(\gamma)D}}{2(\nu\omega + r)}, \\
s_u^- &= \frac{\nu\omega - \sqrt{\nu^2\omega^2 - C(\gamma)D}}{2(\nu\omega + r)}.
\end{aligned}
$$

The above quantities are obtained by solving the quadratic equations

$$
h_z(\epsilon_z) = \sigma_{\max}^{-1}\|\mathcal{E}\|_2 - (\nu\omega)\epsilon_z + (\nu\omega + r)\epsilon_z^2 = 0
$$

and

$$
h_u(\epsilon_u) = \sigma_{\max}^{-1}\|\mathcal{E}\|_2 - \left(\frac{\nu\omega}{C(\gamma)}\right)\epsilon_u + \left(\frac{\nu\omega}{C(\gamma)} + r\right)\epsilon_u^2 = 0,
$$

which are defined in (B.2.6) and (B.2.9).

3.4 Numerical results

We demonstrate the proposed method on simulated dynamic networks and the Enron email dataset. Although a number of methods have been developed for dynamic network data, hierarchical or hidden Markov models often have to estimate the posterior distribution of all entries in a tensor representation. Therefore, these algorithms are not scalable to dynamic

networks with thousands of (or more) time-points although they may be useful to characterize very complicate dynamics of small networks. Therefore, we compare the performance of our method with the latant space model proposed by Sewell and Chen (2015), which models network dynamics as a stochastic process of node positions in a latent Euclidean space, for fitting graph snapshots.

To illustrate the importance of high-order representation for finely-partitioned dynamic networks, we compare the performance of our method with the following two scores:

1. Low-rank approximation of the aggregated adjacency matrix $\mathbf{A} = [\sum_{k=1}^{n_t} A_{ijk}]_{n\times n}$, denoted by $\widehat{\mathbf{P}} = [\widehat{p}_{ij}]$. We use $\widehat{p}_{ij}$ as a score of A_{ijk}. This estimator does not use information of network dynamics and is equivalent to setting $u_\ell^* \equiv 0$.

2. Second-order B-spline regression on $\sum_{i=1}^{n}\sum_{j=1}^{n} \mathcal{A}_{ij\cdot}$. This method estimates an overall intensity function and ignores inhomogeneity among pairs of nodes.

3.4.1 Simulated dynamic networks

We generate dynamic networks from the proposed model (3.2.1), with 300 nodes and three network factors defined by, for $\ell = 1, 2, 3$,

$$z_{\ell i} \propto \begin{cases} 1 & \text{if } \lfloor (i-1)/100 \rfloor = \ell, \\ \rho_z & \text{if } \lfloor (i-1)/100 \rfloor \neq \ell. \end{cases} \tag{3.4.1}$$

The corresponding time trends are modelled as

$$u_\ell^*(t) \propto \rho_u + (1-\rho_u)\max\{0, 1 - |0.006t - (2k-1)|\} \tag{3.4.2}$$

for $t \in [0, 1000]$. We set $\sigma_1 = \sigma_2 = \sigma_3$ and chose them so that the expected total number of edges in the dynamic network over all time points is 3000. The time interval is partitioned into 1000 equal segments to convert the time-stamped links to a sparse tensor of size $300 \times 300 \times 1000$. Thus the networks are extremely sparse, with about 3 interactions per time period on average, reflecting the sparsity levels often found in real data.

3.4.1.1 Tuning parameter selection

In this section, we demonstrate that tuning parameter selection can be done by random subsample validation. We first generate a training and a test dataset. Then, we use random data splitting for tuning parameter selection as follows. We first randomly select 1% of entries

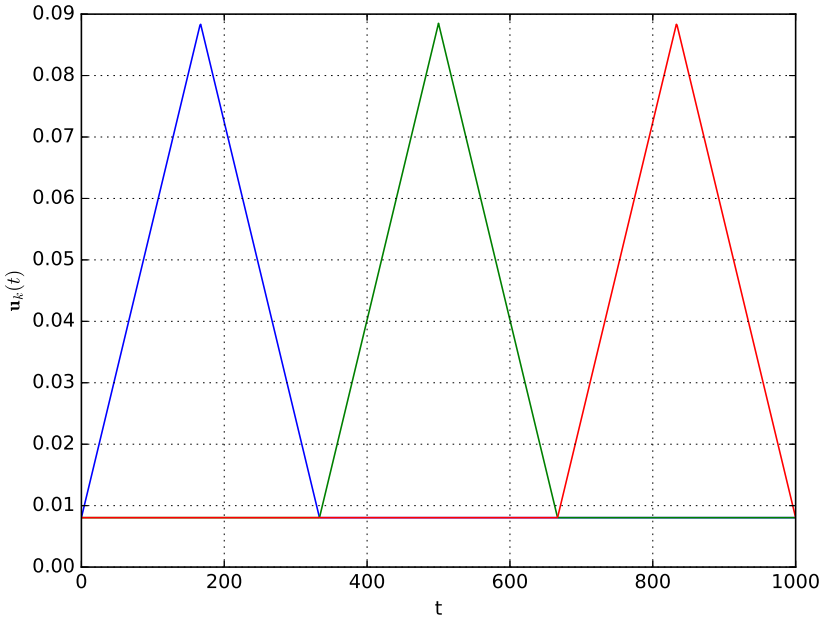


Figure 3.2: Time trend functions $\mathbf{u}_\ell(t)$ for simulated networks.

of the data tensor and set the selected entries to be 0 to obtain a subsampled dataset. Applying the proposed algorithm to this subsampled dataset and the test dataset, we evaluate predictive power of the resulting fit in terms of the predictive AUC on the test dataset and removed links defined by the average matching angle

$$M(\mathbf{X}, \mathbf{Y}) = r^{-1} \max_{\sigma:\{1,\ldots,r\}\mapsto\{1,\ldots,r\}} \left\{ \sum_{k=1}^{r} \mathbf{x}_k^\top \mathbf{y}_{\sigma(k)} \right\}.$$

The reported values are averages over 100 replications.

Table 3.2 shows that the maximum predictive AUC on removed entries is obtained at $\gamma = 1$, which also achieves good predictive AUC (0.713) on the test data and is close to the optimal value (0.720 at $\gamma = 10$) for the test data. The algorithm fails to find all factors when γ is very large, but the large canonical angle (above 0.94) indicates that the detected factor is still in the subspace spanned by the true factors.

3.4.1.2 Testing robustness to the incoherence assumption

We also conducted numerical experiments with various levels of parameters ρ_z and ρ_u in (3.4.1) and (3.4.2) respectively, which measure the incoherence between different components. The tuning parameters were by random subsplitting described in the previous

Table 3.2: Performance for various tuning parameters (on validation and test data) for $\rho_z = 0.075$ and $\rho_u = 0.2$

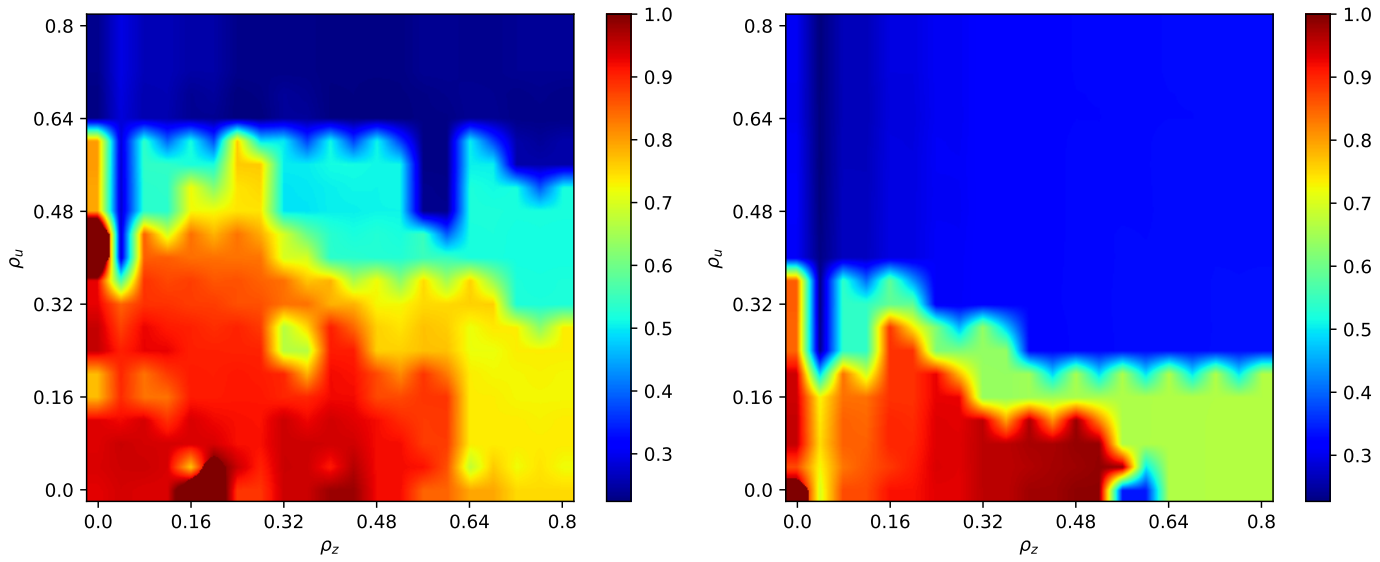
		Predictive AUC		Matching	
γ	$\widehat{r}$	Test data	Removal	$M(\mathbf{u}, \widehat{\mathbf{u}})$	$M(\mathbf{z}, \widehat{\mathbf{z}})$
10^{-4}	3	0.512	0.555	0.396	0.257
10^{-3}	3	0.624	0.667	0.896	0.765
10^{-2}	3	0.688	0.708	**0.975**	0.937
10^{-1}	3	0.715	0.734	0.920	0.946
10^{0}	3	0.713	**0.746**	0.790	0.952
10^{1}	3	**0.720**	0.743	0.674	**0.955**
10^{2}	1	0.564	0.571	0.224	0.233

subsection. Reported results are averages over 20 replications for each combination of ρ_z and ρ_u.

Figures 3.3 and 3.4a shows the matching scores and predictive AUC on test data as a function of incoherence parameters for structure (ρ_z) and time trends (ρ_u). As expected, larger values of ρ_z or ρ_u make the components more difficult to distinguish, and the predictive AUC decreases as ρ_z and ρ_u increase. Figure 3.3 further indicates that increasing either ρ_z or ρ_u can affect the accuracy of both $\widehat{\mathbf{Z}}$ and $\widehat{\mathbf{U}}$ in terms of the matching scores. The low-rank approximation (Figure 3.4c) and the B-spline regression (Figure 3.4b) only work when $\mathbf{z}_\ell$'s and $\mathbf{u}_\ell$'s are strongly incoherent, respectively. This shows that jointly modeling time trends and network structure increases power to detect both relative to modeling just one of them at a time.

3.4.2 Performance on graph snapshots

We convert the above simulated training and test datasets of time-stamped links to graph snapshots partitioning the time dimension into 50 subintervals of equal length and truncate each entry to 1. For fitting the latent space model, we ran 1000 burn-in iterations and 20,000 MCMC iterations for each setting. For the datasets of 50 graph snapshots, our method can produce results in a few seconds and achieved comparative predictive AUC with the latent space model fitted by a MCMC algorithm.



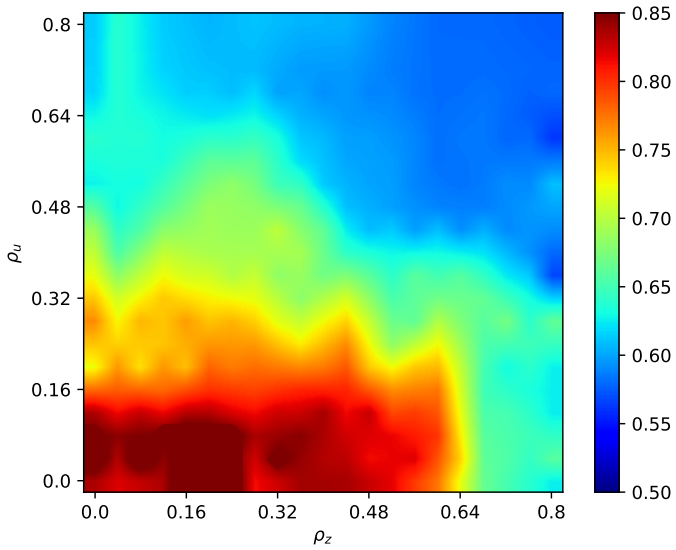
(a) $M(\widehat{\mathbf{U}}, \mathbf{U})$ (b) $M(\widehat{\mathbf{Z}}, \mathbf{Z})$

Figure 3.3: Matching cosine angle on test data as a function of incoherence parameters.

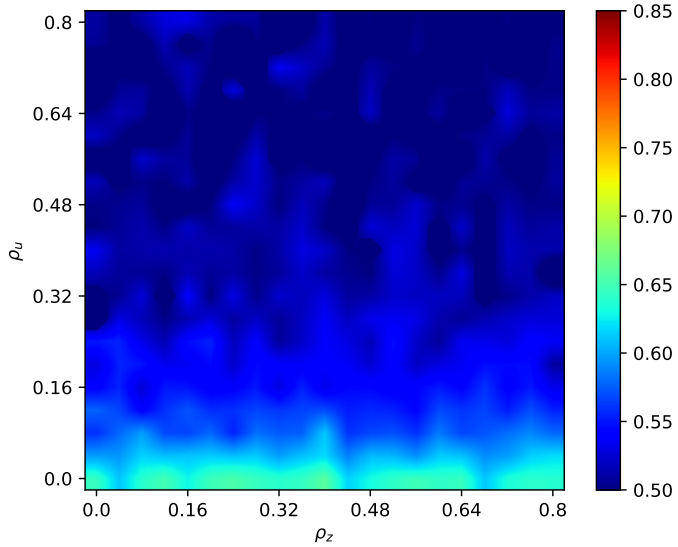
3.4.3 Data: Enron emails

We also applied the proposed method to the Enron email dataset (May 7, 2015 Version, retrived from `https://www.cs.cmu.edu/~enron/`; see Klimt and Yang (2004) for a full description), which contains emails between 156 employees of Enron sent between October 1998 and June 2002. We removed e-mails that had unparseable time stamps or were sent to more than 10 people, which resulted in 25,830 e-mails included in the analysis. To illustrate the scalability of our method, we partitioned the time interval into hours, resulting in a $156 \times 156 \times 31416$ sparse tensor. We then randomly selected 5% of entries from the tensor and set them to zero. We use these selected entries as a test dataset and the remaining entries as a training dataset. The tuning parameter is selected by randomly splitting the training data as described in Section 3.4.1.1. The number of factors r is estimated by parallelly running 10,000 threads of the power iteration.

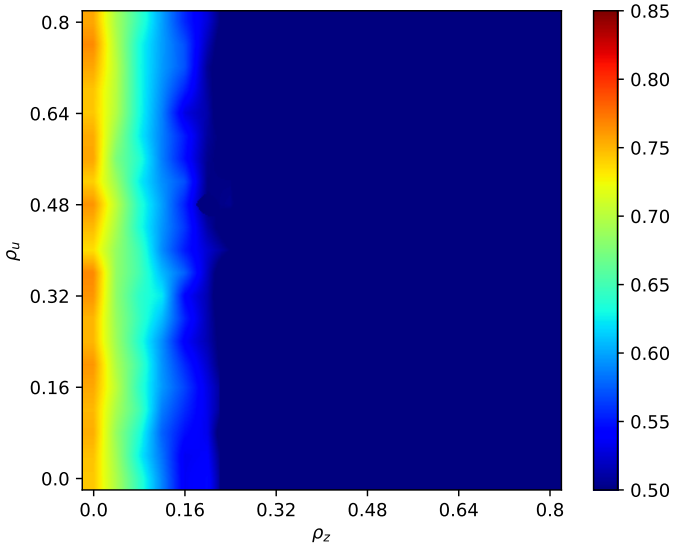
Our method achieved predictive AUC of 0.855, compared to 0.761 and 0.575 achieved by the low-rank approximation and the B-spline regression, respectively. Furthermore, when we partition the Enron email dataset into 50 time intervals of equal length, our method obtained $\mathrm{AUC} = 0.847$ compared to $\mathrm{AUC} = 0.852$ given by the latent space model with 1,000 burn-in and 20,000 MCMC iterations. We detected two network factors in this dataset, shown in Figures 3.6 and 3.7. The node pie charts in Figure 3.6 show the values of $\frac{z_{1i}}{z_{1i}+z_{2i}}$ and $\frac{z_{2i}}{z_{1i}+z_{2i}}$, which can be interpreted as a mixed community membership vector for the two overlapping structures. The time trends indicate that the structure shown



(a) Regularized tensor decomposition.

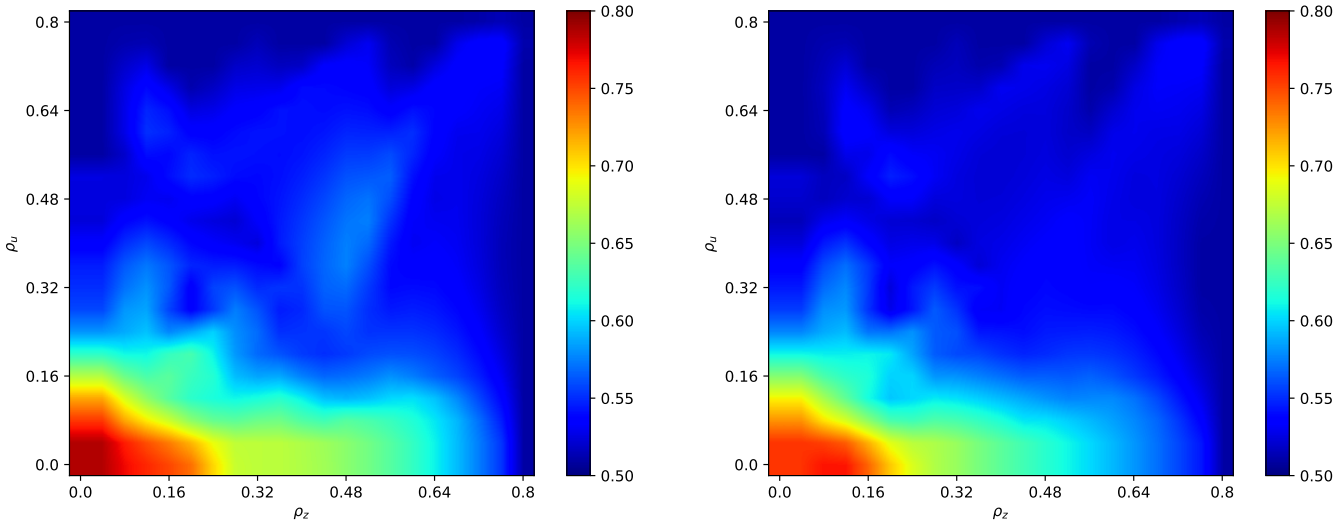


(b) Second-order B-spline regression.



(c) Low rank approximation of the aggregrated adjacency matrix.

Figure 3.4: Predictive AUC on test data as a function of incoherence parameters.



(a) Regularized tensor decomposition.

(b) Latent space model.

Figure 3.5: Predictive AUC on test data of graph snapshots.

in red becomes very active around the time the scandal broke, with its estimated time trend function reaching its highest value right after the CEO was replaced. The structure shown in blue, on the other hand, is mainly present before the scandal. This simple analysis already reveals some interesting dynamics in the network structure, and could be enhanced by making use of the emails' contents.

3.5 Discussion

In this paper, we proposed a tensor decomposition approach and a very scalable algorithm for modeling dynamic networks represented by time-stamped links data. The algorithm can easily handle thousands of nodes and up to millions of time-points on one single computer, compared to hundreds of nodes and tens of time-points handled by existing methods. The low rank representation is useful for producing interpretable results, as demonstrated by our data analysis. The model can be trivially extended to directed networks by breaking symmetry in (3.2.2).

Other extensions include treating the time trends as continuous time processes instead of relying on a discrete time interval partition. The core of our algorithm is the power iteration. Ignoring the shift, we can rewrite the updating equation for a structure factor as

$$\mathcal{A} \otimes_1 \mathbf{z}^{(m)} \otimes_3 \mathbf{u}^{(m)} = (\mathcal{A} \otimes_3 \mathbf{u}^{(m)})\mathbf{z}^{(m)},$$

where $\mathcal{A} \otimes_3 \mathbf{u}^{(m)}$ is a weighted sum of slices of $\mathcal{A}$. We can think of this update as running

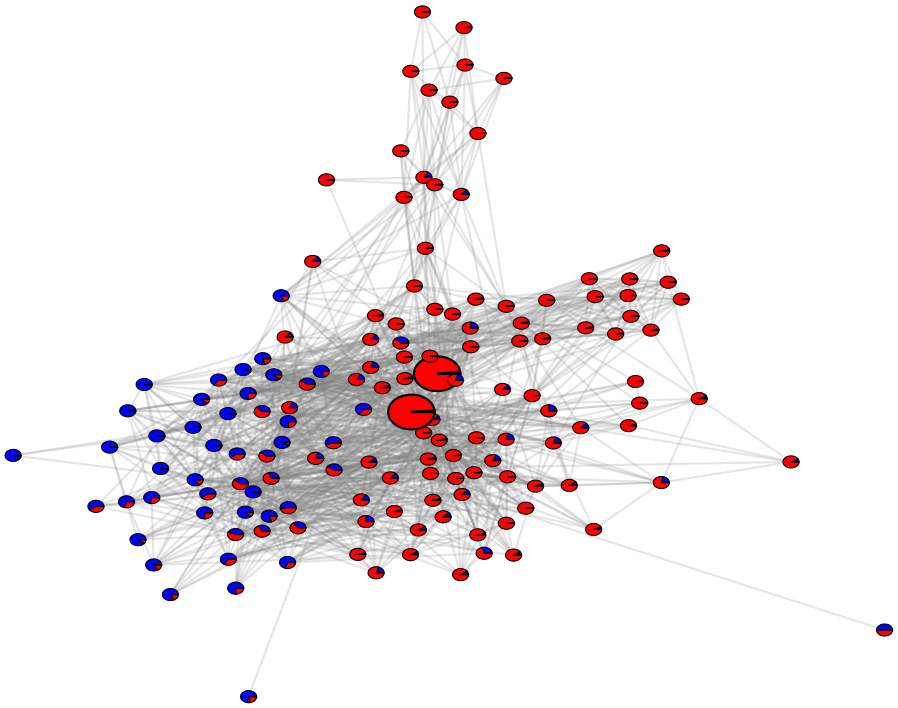


Figure 3.6: Structural factors $\widehat{\mathbf{z}}_\ell$ represented by red and blue, proportional to their weights for each node. The largest nodes are the founder and the CEO of Enron at the time of the scandal.

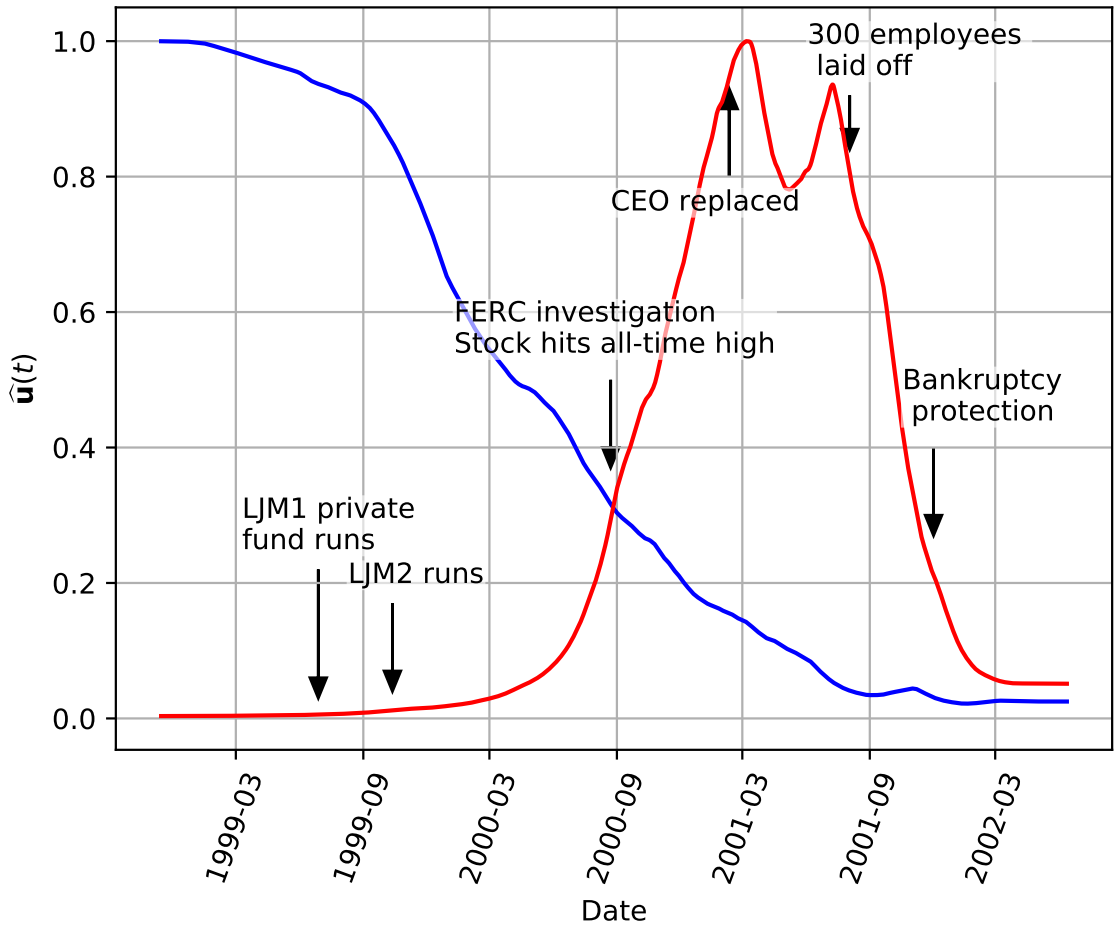


Figure 3.7: Time trends $\widehat{\mathbf{u}}_\ell(t)$ normalized to have maximum value 1. Colors match the structure colors in Figure 3.6.

one power iteration on matrix $\mathcal{A} \otimes_3 \mathbf{u}^{(m)}$. On the other hand, the updating equation of a temporal factor is $(\mathbf{I} + \gamma\mathbf{\Omega})^{-1}(\mathcal{A} \otimes_1 \mathbf{z}^{(m)} \otimes_2 \mathbf{z}^{(m)})$. The operator $(\mathbf{I} + \gamma\mathbf{\Omega})^{-1}$ enhances smoothings of the temporal factor $\mathcal{A}\otimes_1 \mathbf{z}^{(m)} \otimes_2 \mathbf{z}^{(m)}$, which is a weighted sum of interaction processes. The power iteration is essentially alternating between a spectral embedding of the network structure and a smoothing step along the temporal dimension. One can always replace this smoothing step with other techniques, for example, parametric models for longitudinal data, or non-parametric kernel smoothing, and handle continuous time processes without partitioning the time domain.

Another possible extension is considering a decomposition into factors of rank higher than 1. Our model is based on the Kruskal decomposition into rank-1 factors, and if two or more structure factors follow the same time trend, the Kruskal condition (3.2.3) may not hold and thus the model is not identifiable. We also empirically observed numerical instability in this situation. This suggests that some alternative constraints would need to be imposed to maintain identifiability of the model if higher rank factors are allowed, or else the focus would need to shift to only identifying structural "subspaces". Finally, incorporating other information into the model that may be recorded along with the interactions, such as covariates on the nodes and edges (e.g., in the Enron email example, this could be information about the sender's and receiver's positions and email contents) would allow for much richer models for dynamic network data.

CHAPTER 4

Subspace estimation for link prediction in ego-networks

4.1 Introduction

Social networks consist of nodes and links that represent individuals and relations, and a large body of work has been devoted to their quantitative analysis. Social network data are often collected via surveys and almost always includes noise and missing values. The problem of link prediction is the task of removing noise and imputing missing links, for which many techniques have been delveloped; see Liben-Nowell and Kleinberg (2007) and Lü and Zhou (2011) for reviews.

Undirected network on N nodes can be represented with a symmetric adjacency matrix $\mathbf{A} \in \{0,1\}^{N\times N}$ with $A_{ij} = A_{ji} = 1$ if nodes i and j are linked. In statistical network analysis, we usually assume that the adjacency $\mathbf{A}$ is generated from an underlying probability matrix $\mathbf{P}$, with A_{ij}'s generated as independent Bernoulli(p_{ij}) random variables, where p_{ij} is the probability that nodes i and j are connected with each other. The assumption of independence is not always realistic in practice, but so far the vast majority of probabilistic models for networks rely on it, and it has been found to produce useful algorithms.

The link prediction problem can be thought of as classifying pairs of nodes as "linked" and "unlinked", which is frequently done on the basis of a score for each pair, with an estimate of p_{ij}'s providing a natural score function. Thus link prediction naturally leads to the problem of estimating $\mathbf{P}$ or, alternatively, a monotone function of the probabilities if only a relative ranking of links is important. This is closely related to the problem of matrix completion.

4.1.1 Matrix completion

Matrix completion techniques pursue the goal of estimating underlying matrix structure from an noisy or incomplete data matrix, usually based on low-rank approximation. Formally, the problem is formulated as an optimization problem as follows:

$$\begin{aligned} \underset{\mathbf{X}}{\text{minimize}} \quad & \ell(\Omega(\mathbf{X}), \Omega(\mathbf{P})) \\ \text{subject to} \quad & \text{rank}(\mathbf{X}) \leq r, \end{aligned}$$

where Ω is an entry-wise mask operator such that $\Omega(X_{ij}) = X_{ij}$ if the entry p_{ij} is observable otherwise $\Omega(X_{ij}) = 0$ and ℓ is a loss function. In the context of link prediction, we try to estimate $\mathbf{P}$ based on $\mathbf{A}$, which is corresponding to the inexact matrix completion problem. The value of p_{ij} is noisy even if the entry is observable. In this case, we solve an optimization based on the empirical loss function:

$$\begin{aligned} \underset{\mathbf{X}}{\text{minimize}} \quad & \ell(\Omega(\mathbf{X}), \Omega(\mathbf{A})) \\ \text{subject to} \quad & \text{rank}(\mathbf{X}) \leq r, \end{aligned}$$

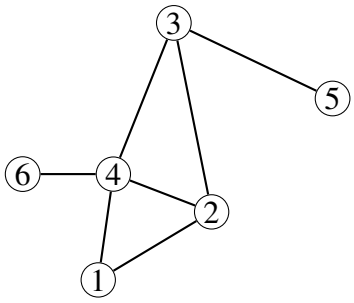
where $A_{ij} = p_{ij} + e_{ij}$ with $\mathbb{E}[e_{ij}] = 0$ and e_{ij}'s being independent. A series of theoretical results has been developed for either formalization(Candès and Tao, 2010; Candès and Plan, 2010; Keshavan et al., 2010; Davenport et al., 2014). In related work, Chatterjee (2015) proposed the universal singular value thresholding approach for general matrix estimation. However, most works assume the observed adjacency matrix are missing at random with a missing rate $1 - \rho$. In this case, entries of the observed adjacency matrix can be denoted by $A_{ij}^{obs} = M_{ij}A_{ij}$, where M_{ij}'s$\sim$ Bernoulli(ρ) indicate whether links are observed. Thus, one can estimate $\mathbb{E}[A_{ij}^{obs}] = \rho\mathbf{P}$ and a score-based classification method is still valid for predicting links even if ρ is unknown. Although this simple assumption allows researchers to gain theoretical insights, it may be violated in datasets collected via many practical survey methods.

4.1.2 Egocentric networks

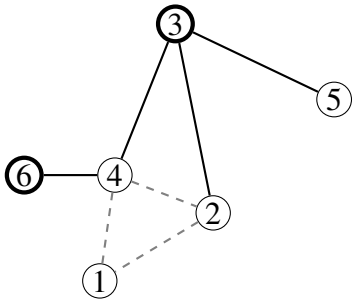
In this paper, we focus on the task of predicting links for networks constructed by egocentric sampling. Egocentric networks have been studied in the quantitative social sciences for several decades (Freeman, 1982; Marsden, 2002; Kogovšek and Ferligoj, 2005; Almquist, 2012) and more recently in physics and computer science (e.g. Newman, 2003; Mcauley and Leskovec, 2012). It has been pointed out that summary statistics of egocentric net-

works can be dramatically different from those of a randomly sampled population network, due to the different noise structure introduced by the sampling mechanism. It is reasonable to expect that this different noise structure will also affect many of existing link prediction algorithms.

Social network data are often collected through surveys that ask a sample of subjects to name the people they are connected to (the definition of connected varies with the purpose of the study). We model this process as sampling n people without replacement from a group of size N, and asking them to name all their connections, without any upper bound on the number. This results in an *egocentric sample* or *ego-network* consisting of a random sample of n rows from the full $N \times N$ adjacency matrix.



(a) Network G



(b) Solid: Ego network G_{ego} constructed from the egocentric sample $\{3, 6\}$; dashed: Unobserved edges.

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

(c) Adjacency matrix of (a)

$$\mathbf{A}_{ego} = \begin{pmatrix} ? & ? & 0 & ? & ? & 0 \\ ? & ? & 1 & ? & ? & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ ? & ? & 1 & ? & ? & 1 \\ ? & ? & 1 & ? & ? & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

(d) Observed adjacency matrix

Figure 4.1: An illustration of egocentric sampling.

Formally, suppose that our target network $G = (V, E)$ has the node set $V = \{1, \ldots, N\}$ and an edge set E with $|E| = m$. We sample nodes $\mathcal{I} = \{i_1, \ldots, i_n\} \subset V$, and the observed subsampled network $G_{ego} = (V_{ego}, E_{ego})$ has the same set of nodes $V_{ego} = V$, and $E_{ego} = \cup\{(u, v) \in E : u \in I\}$. See Figure 4.1 for an illustration. Equivalently, when the i node is sampled, we observe the i-th row and column of the adjacency matrix $\mathbf{A}$.

Related work on low rank approximations for egocentrically sampled networks include the CUR decomposition Mahoney and Drineas (2009). The purpose of CUR is to find a

matrix $\mathbf{U}$ such that $\mathbf{P} \approx \mathbf{CUR}$, where $\mathbf{C}$ and $\mathbf{R}$ are exactly the columns and rows sampled from $\mathbf{P}$. This approach greatly reduces the time and space complexity for compressing a matrix. To obtain $\mathbf{U}$, the CUR algorithm solves a least squares problem, letting

$$\mathbf{U} = \arg\min_{\mathbf{X}} \|\Omega(\mathbf{A}) - \Omega(\mathbf{C}^\top \mathbf{X}\mathbf{R})\|_F^2,$$

where $\Omega(A_{ij}) = 1(i \text{ or } j \text{ is selected})$. The theoretical foundation for the CUR decomposition (Drineas et al., 2006b, 2008) assumes the matrix to be noiseless, which for us would mean observing the probability matrix $\mathbf{P}$ directly instead of the adjacency matrix $\mathbf{A}$. For link prediction, the CUR decomposition suffers from overfitting since we can always achieve $\min_{\mathbf{X}} \|\Omega(\mathbf{A}) - \Omega(\mathbf{C}^\top \mathbf{X}\mathbf{R})\|_F^2 = 0$ by choosing $\mathbf{U}$ to be the pseudo-inverse of the intersection of $\mathbf{C}$ and $\mathbf{R}$. Moreover, to improve the accuracy of the CUR approximation, it is essential to use importance sampling to sample $\mathbf{C}$ and $\mathbf{R}$ based on a probability distribution that is computed from the entire data matrix, but importance sampling is generally not feasible in social network data without additional information about subjects, and it is more feasible to treat egocentric networks as if the rows are uniformly sampled without replacement.

In this paper, we propose a computationally efficient method for link prediction for egocentrically sampled networks based on a low rank approximation. The key idea is subspace estimation, which obtains the approximate row space of the probability matrix $\mathbf{P}$ and allows us to solve the link prediction problem in the context of egocentric sampling. In Section 4.2, we describe our method and provide some theoretical results. We conduct a numerical evaluation of on subspace estimation method on both synthetic and real networks and compare to benchmark link prediction methods in Section 4.3. Section 4.4 concludes with discussion and future work.

4.2 Link prediction via subspace estimation

Without loss of generality, we can assume that the first n nodes out of N were selected, and the observed adjacency matrix can be partitioned into blocks $\mathbf{A}_{ij}$ for $i, jin\{1, 2\}$, where $\mathbf{A}_{11} \in \{0, 1\}^{n\times n}$, $\mathbf{A}_{12} \in \{0, 1\}^{n\times(N-n)}$, and $\mathbf{A}_{21} = \mathbf{A}_{21}^\top$, and the block $\mathbf{A}_{22} \in \{0, 1\}^{(N-n)\times(N-n)}$ is not observed; see Figure 4.2. The corresponding submatrices of $\mathbf{P}$ are defined as $\mathbf{P}_{ij}$ for $i, j = 1, 2$. We also define the sampled rows $\mathbf{A}_{in} = [\mathbf{A}_{11}\ \mathbf{A}_{12}]_{n\times N}$ and the corresponding probability sub-matrix $\mathbf{P}_{in} = [\mathbf{P}_{11}\ \mathbf{P}_{12}]_{n\times N}$.

$$\mathbf{A} = \begin{array}{|c|c|} \hline \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} \\ \hline \end{array}$$

Figure 4.2: Grey: observed blocks of the adjacency matrix. White: unobserved block.

4.2.1 Estimation

Our goal is to predict links between nodes that were not sampled, or equivalently to estimate $\mathbf{P}_{22}$. Our goal is to approximate the probability matrix $\mathbf{P}$ with a rank r symmetric matrix $\mathbf{P}_r$. Suppose that we could directly sample rows from matrix $\mathbf{P}$. The CUR decomposition compute $\mathbf{X}$ by the following risk function:

$$\ell(\mathbf{X}; \mathbf{P}) = \|\Omega(\mathbf{P}) - \Omega(\mathbf{P}_{in}^\top \mathbf{X} \mathbf{P}_{in})\|_F.$$

The corresponding estimator is

$$\mathbf{U} = \underset{\mathbf{X} \in \mathbb{R}^{n \times n}}{\arg\min} \|\Omega(\mathbf{A}) - \Omega(\mathbf{A}_{in}^\top \mathbf{X} \mathbf{A}_{in})\|_F.$$

The solution to this optimization problem is $\mathbf{U} = \mathbf{A}_{11}^+$, where $\mathbf{A}_{11}^+$ is the pseudo-inverse of $\mathbf{A}_{11}$, which is the $\mathbf{U}$ matrix in the standard CUR decomposition (Mahoney and Drineas, 2009). However, the in-sample error of $\mathbf{U}$ is $\|\Omega(\mathbf{A}) - \Omega(\mathbf{A}_{in}^\top \mathbf{U} \mathbf{A}_{in})\|_F = 0$, and the estimator $\mathbf{A}_{in}^\top \mathbf{A}_{11}^+ \mathbf{A}_{in}$ often gives poor predictions in numerical experiments. This suggests that the estimator $\mathbf{U}$ may suffer from overfitting, a problem that can be solved with various forms of regularization. A natural regularization to consider is constraining the rank of X, computing instead

$$\widetilde{\mathbf{X}} = \underset{\mathrm{rank}(\mathbf{X}) \leq r}{\arg\min} \|\Omega(\mathbf{A}_{obs}) - \Omega(\mathbf{A}_{in}^\top \mathbf{X} \mathbf{A}_{in})\|_F.$$

The resulting estimator $\mathbf{A}_{in}^\top \tilde{\mathbf{X}} \mathbf{A}_{in}$ is a rank r approximation to $\mathbf{P}$. We can interpret this as an estimator of the row space of P $\mathrm{row}(\mathbf{P}_r)$ based on a subspace of $\mathrm{row}(\mathbf{A}_{in})$. However, solving this non-convex optimization problem directly is highly non-trivial. Instead, we

$$\widehat{\mathbf{P}} = \begin{array}{|c|c|} \hline \frac{1}{2}(\widetilde{\mathbf{P}}_{11}^{+} + \widetilde{\mathbf{P}}_{11}^{\top +}) & \frac{1}{2}(\widetilde{\mathbf{P}}_{11}^{+\top}\widetilde{\mathbf{P}}_{11}^{+} + \mathbf{I}_{n\times n})\widetilde{\mathbf{P}}_{in,2} \\ \hline \widehat{\mathbf{P}}_{12}^{\top} & \frac{1}{2}\widetilde{\mathbf{P}}_{in,2}^{\top}(\widetilde{\mathbf{P}}_{11}^{+} + \widetilde{\mathbf{P}}_{11}^{\top +})\widetilde{\mathbf{P}}_{in,2} \\ \hline \end{array}$$

Figure 4.3: Blocks of $\widehat{\mathbf{P}}$ in (4.2.1), where $\widetilde{\mathbf{P}}_{in} = [\widetilde{\mathbf{P}}_{in,1}\ \widetilde{\mathbf{P}}_{in,2}]$, where $\widetilde{\mathbf{P}}_{in,1} \in \mathbb{R}^{n\times n}$ and $\widetilde{\mathbf{P}}_{in,2} \in \mathbb{R}^{n\times(N-n)}$

propose the following two-stage estimation procedure:

1. Estimate $\mathrm{row}(\mathbf{P}_r)$, which can be viewed as a principal subspace of $\mathrm{row}(\mathbf{P})$. We first use the best rank r approximation of $\mathbf{A}_{in}$, denoted by $\widetilde{\mathbf{P}}_{in}$, as a estimator of $\mathbf{P}_{in}$. Thus, the resulting estimate is a rank-r matrix and satisfies $\mathrm{row}(\widetilde{\mathbf{P}}_{in}) = \mathrm{row}(\widehat{\mathbf{P}})$.

2. Construct an estimator of the form $\widehat{\mathbf{P}} = \widetilde{\mathbf{P}}_{in}^{\top}\widehat{\mathbf{X}}\widetilde{\mathbf{P}}_{in}$, where $\widehat{\mathbf{X}} = \frac{1}{2}(\widehat{\mathbf{P}}_{11}^{+} + \widehat{\mathbf{P}}_{11}^{\top +})$ is a $n \times n$ symmetric matrix. The idea for selecting $\widehat{\mathbf{X}}$ for our estimator is considering the loss function
$$\ell(\mathbf{X}) = \|\Omega(\mathbf{P}) - \Omega(\mathbf{P}_{in}^{\top}\mathbf{X}\mathbf{P}_{in})\|_F.$$
The minimizer of the loss function is $\mathbf{X} = \mathbf{P}_{11}^{+}$, which gives $\mathbf{P}_{in}^{\top}\mathbf{P}_{11}^{+}\mathbf{P}_{in}$ as an approximation of $\mathbf{P}$. Let $\widetilde{\mathbf{P}}_{in}^{+} = [\widetilde{\mathbf{P}}_{11}^{+}\ \widetilde{\mathbf{P}}_{12}^{+}]$. We estimate $\mathbf{P}_{11}$ by symmetrized $\widetilde{\mathbf{P}}_{11}^{+}$.

Thus, we obtained a plug-in estimator

$$\widehat{\mathbf{P}} = \frac{1}{2}\widetilde{\mathbf{P}}_{in}^{\top}(\widetilde{\mathbf{P}}_{11}^{+} + \widetilde{\mathbf{P}}_{11}^{\top +})\widetilde{\mathbf{P}}_{in}. \tag{4.2.1}$$

The block-wise estimator is illustrated in Figure 4.3.

4.2.2 Interpretations of $\widehat{\mathbf{P}}$

The low-rank approximation is not only a general approach for estimating probability matrices, but also provides an interpretable parametrization for network data. The rank-r approximation of $\mathbf{P}$ can be always decomposed as $\mathbf{P}_r = \mathbf{R}^{\top}\mathbf{Z}\mathbf{R}$, where $\mathbf{R} \in \mathbb{R}^{r\times N}$ and

$\mathbf{Z} \in \mathbb{R}^{r\times r}$. Let $\mathbf{A}_{in} \overset{SVD}{=} \mathbf{U}\mathbf{D}\mathbf{V}^\top$ and $\widetilde{\mathbf{P}}_{in} \overset{SVD}{=} \mathbf{U}_r\mathbf{D}_r\mathbf{V}_r^\top$. Correspondingly, we can rewrite (4.2.1) as follows

$$\begin{aligned}\widehat{\mathbf{P}} &= \mathbf{V}_r\mathbf{D}_r\mathbf{U}_r^\top\widehat{\mathbf{X}}\mathbf{U}_r\mathbf{D}_r\mathbf{V}_r^\top \\ &= \mathbf{V}_r(\mathbf{D}_r\mathbf{U}_r^\top\widehat{\mathbf{X}}\mathbf{U}_r\mathbf{D}_r)\mathbf{V}_r^\top \\ &:= \widehat{\mathbf{R}}^\top\widehat{\mathbf{Z}}\widehat{\mathbf{R}}.\end{aligned}$$

Thus, $\mathbf{V}_r$ gives an estimated embedding of the network in a space equipped with scalar product $\widehat{\mathbf{Z}}$.

Similarly to adjacency spectral embedding methods (e.g. Sussman et al., 2012; Tang et al., 2014), one can think that the columns of $\mathbf{R}$ describe the coordinates of nodes in a pseudo-Euclidean space, and p_{ij} is determined by a scalar product between points corresponding to nodes i and j. The space is equipped with a scalar product characterized by $\mathbf{Z}$. Without imposing any constraint on $\mathbf{Z}$, the candidate models of our estimator include stochastic block models, dot-product models (Young and Scheinerman, 2007), latent eigenmodels (Hoff, 2007), and hyperbolic models (Krioukov et al., 2010; Albert et al., 2014). Thus, one can impose more constraints on $\widehat{\mathbf{Z}}$ to further restrict the collection of the candidate models.

4.2.3 Theoretical justification

In this section, we provide a theoretical justification for our estimator. Let $\rho_N = \frac{n}{N}$ be a sampling rate. We will derive an error bound of $\widehat{\mathbf{P}}$ in terms of $\|\mathbf{P} - \widehat{\mathbf{P}}\|_2$. We investigate theoretical properties of $\widehat{\mathbf{P}}$ under the following assumptions.

(A1) **Dense graphs.** The concentration for adjacency matrices of dense graphs has been well-studied (see Section 1.1 in Le et al. (2015) for a brief review). We will directly apply the result $\|\mathbf{A} - \mathbf{P}\|_2 = O(\sqrt{d})$ by assuming that maximum expected degree $d = O(\log N)$, where $d = \max_i \sum_{j=1}^N p_{ij}$.

(A2) **Random sample.** We assume that nodes are sampled without replacement, and sampling is independent of the network structure.

Theorem 8. *Assume (A1) and (A2). Thus,*

$$\|\mathbf{P}-\widehat{\mathbf{P}}\|_2 = 2\sigma_{r+1}(\mathbf{P})+O_p\left(\sqrt{d}+N^{\frac{1}{2}}\left(\frac{\log n}{\rho_N}\right)^{\frac{1}{4}}+\|\mathbf{P}\|_2\left(\sqrt{1-\rho_N}+\sqrt{\log N(1-\rho_N)}\right)\right),$$

where $\sigma_{r+1}(\mathbf{P})$ is the $(r+1)$th singular value of $\mathbf{P}$.

Proof. See Appendix C.1. □

Roughly speaking, in the above error bound, the difference between network data and the expected adjacency matrix, i.e. $\mathbf{A} - \mathbf{P}$, gives $O_p(\sqrt{d})$; the quantity $\sigma_{r+1}(\mathbf{P})$ reflects error from model mis-specification. The last term comes from randomness of egocentric sampling.

4.3 Numerical study

4.3.1 Tuning parameter selection

We need to choose tuning parameter r to construct a rank-r approximation of $\mathbf{P}$. In this section, we will conduct sub-sampling validation to select r. We repeatedly sample $k \in \{1, \ldots, n\}$ and set $\mathbf{A}_{sub}$ be a submatrix of $\mathbf{A}_{in}$ by deleting the k-th row from $\mathbf{A}_{in}$. Applying the proposed algorithm to $\mathbf{A}_{sub}$, we can estimate predictive accuracy by computing the area under the ROC curve (AUC) on the entries $\{A_{ki} : i \notin I \cup \{k\}\}$. Alternatively, one may use some self-tuned or tuning-free methods to obtain $\widehat{\mathbf{P}}_{in}$. This approach will further reduce computational cost.

4.3.2 Comparison with benchmarks

We compare the numerical performance of several widely used algorithms for link prediction. We included the standard CUR decomposition (CUR) (Mahoney and Drineas, 2009) to show the importance of the subspace estimation step for link prediction. From matrix completion methods with independently and identically sampled entries, we chose universal singular value thresholding (USVT) (Chatterjee, 2015) and nuclear norm regularization with inexact augmented Lagrange multiplier method (MC-IALM) (Lin et al., 2010). For these two methods, we also applied them to incomplete adjacency matrices with i.i.d. missing entries to show the effect of the sampling scheme of ego-networks on the performance of standard matrix completion methods. We also included neighborhood smoothing method (NS) (Zhang et al., 2015), which is a graphon estimation method and has demonstrated performance on solving link prediction problems. The method requests a similarity measurement between nodes. We used $\mathbf{A}_{in}^{\top}\mathbf{A}_{in}$, instead of $\mathbf{A}^2$ proposed in (Zhang et al., 2015), as a similarity measurement between pairs of nodes for neighborhood selection.

Table 4.1: Generative models for systhetic networks

Model	X_i	$p_{ij} \propto f(X_i, X_j)$	Rank of $\mathbf{P}$	Numerical rank of $\mathbf{A}$*
Distance	$\text{Normal}_5(0, 1)$	$(1 + \exp(\|X_i - X_j\|))^{-1}$	Full rank	1.85–14.70
Product	$\text{Beta}_5(0.5, 1)$	$X_i^\top X_j$	5	1.65–14.16
SBM	$\text{Uniform}(1, 2, \ldots, 5)$	$0.05 + \frac{i-0.3}{6} 1(i = j)$	5	2.63–12.61

*Changes as average degrees increase.

4.3.3 Numerical experiments

First, we evaluate the performance of our method on simulated datasets. We generate the networks from the models described in Table 4.1. For all networks, we first generate i.i.d. X_i's for $i = 1, \ldots, 500$ and then generate $A_{ij} \sim \text{Bernoulli}(\phi f(X_i, X_j))$, where ϕ is a coefficient that controls average degrees of simulated networks. We tested our method as well as benchmark methods under different sampling rate $\rho = n/500$ with average degree $d = 100$. In addition, we draw entry-wise samples and fed them to both USVT and MC, which assume entry-wise random sample, to see how sampling scheme may affect their performance. See Table 4.1 for details and descriptive statistics of synthetic networks. To further address how network structure affects predictive accuracy, we generate networks with $d = 10, 20, \ldots, 200$ with fixed $\rho = 0.2$. To assess performance of the methods, we compute predictive AUC

$$\text{AUC}(\widehat{\mathbf{P}}, \mathbf{A}) = \frac{\sum_{i,j,i',j' \notin I} 1(A_{ij} = 1, A_{i'j'} = 0, \widehat{p}_{ij} > \widehat{p}_{i'j'})}{\sum_{i,j,i',j' \notin I} 1(A_{ij} = 1, A_{i'j'} = 0)}$$

and predictive Kendall's tau

$$\tau(\widehat{\mathbf{P}}, \mathbf{P}) = \frac{2 \sum_{i,j,i',j' \notin I} 1(p_{ij} > p_{i'j'}, \widehat{p}_{ij} > \widehat{p}_{i'j'})}{\sum_{i,j,i',j' \notin I} 1(p_{ij} > p_{i'j'})} - 1$$

in the unobserved sub-adjacency matrix $[A_{ij}]_{i,j \in \mathcal{I}}$. The numerical results of each scenario are averaged over 100 repetitions.

In Figures 4.4-4.6, the predictive errors of our method, CUR, and NS reduce as networks become denser. Our method uniformly out perform all the benchmark methods in terms of predictive AUC and Kendall's tau in ego-network data generated from distance and product models. NS also gives comparable accuracy when networks are sufficiently dense since block structure ensures NS able to select sufficient numbers of nodes as neighbors, but slightly under-performed our method for $\rho = 0.05$. Figures 4.7-4.9 show the performance of all the methods under different sampling rates. The effect of sampling rates on

Table 4.2: Descriptive statistics of datasets

Dataset	N	m	Avg. deg.	Numerical rank
Residence hall	217	2672	24.6	7.91
Adolescent health	2539	12969	10.2	119.44
Wikipedia elections	7118	103675	28.3	10.57

predictive accuracy is similar to the effect of average degrees. The more nodes are selected for constructing ego-networks, the better performance we can achieve.

Finally, we evaluated the performance of our method and the benchmark methods on the residence hall network (Freeman et al., 1998), the adolescent health network (Moody, 2001), and the Wikipedia election network (Leskovec et al., 2010). Descriptive statistics of these three social networks are summarized in Table 4.2. We sampled 5% to 50% of nodes to construct ego-network samples and evaluated predictive AUC on unobserved pairs of nodes. Since the underlying true $\mathbf{P}$ is unavailable, we only report predictive AUC as the performance assessment. Note that some benchmark methods were not tested for all datasets due to high space complexity of the methods. As shown in Figure 4.10, the trends of the performance of the methods are similar to that in 4.7-4.9. Again, our method performs better than the benchmark methods particularly for small ρ.

To sum up, our method achieved great accuracy for predicting links, and its computational cost is only slightly more expensive than CUR-decomposition. Remarkably, although in some scenarios NS produces comparable results, our method always outperforms the benchmark methods for relatively sparse networks or small sampling rates ρ, which are often encountered in social surveys. This demonstrates the usefulness of our method for reconstructing underlying social networks from ego-networks in practice.

4.4 Discussion

In this work, we proposed a computationally efficient method to predict links based on ego-network data. By exploiting low-rank structure of the probability matrix and employing subspace estimation, our method achieves good performance compared to existing methods for matrix completion/link prediction with entry-wise random sample. Furthermore, our method is essentially first removing noise from $\mathbf{P}_{in}$ and achieve an estimated subspace. That is, we actually conduct matrix completion for estimating $\mathbf{P}_{in}$. This suggests our method can tolerate some missing values appearing in $\mathbf{A}_{in}$.

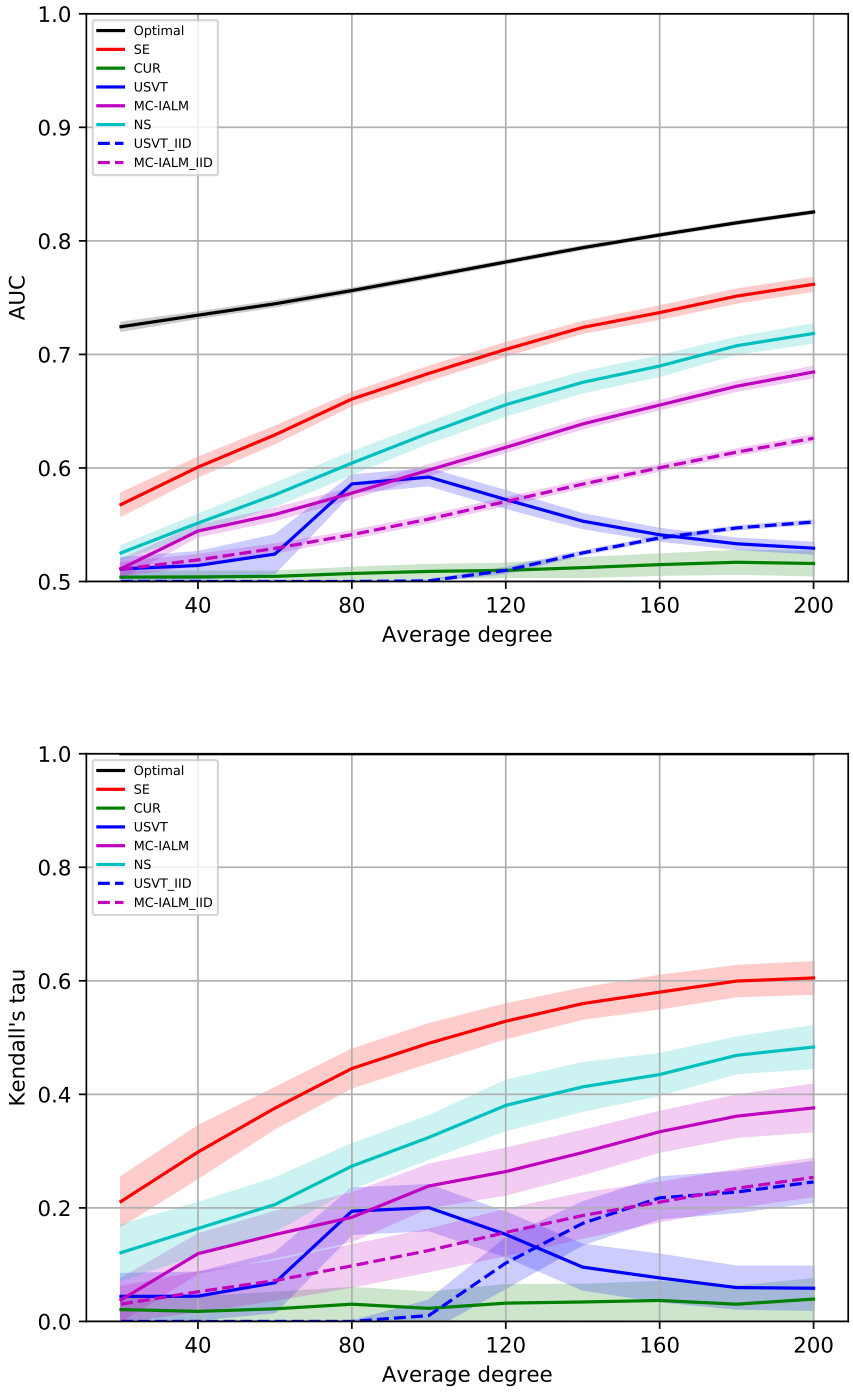


Figure 4.4: Predictive AUC and Kendall's tau for distance models with various average degrees with confidence bands of 1 standard errors

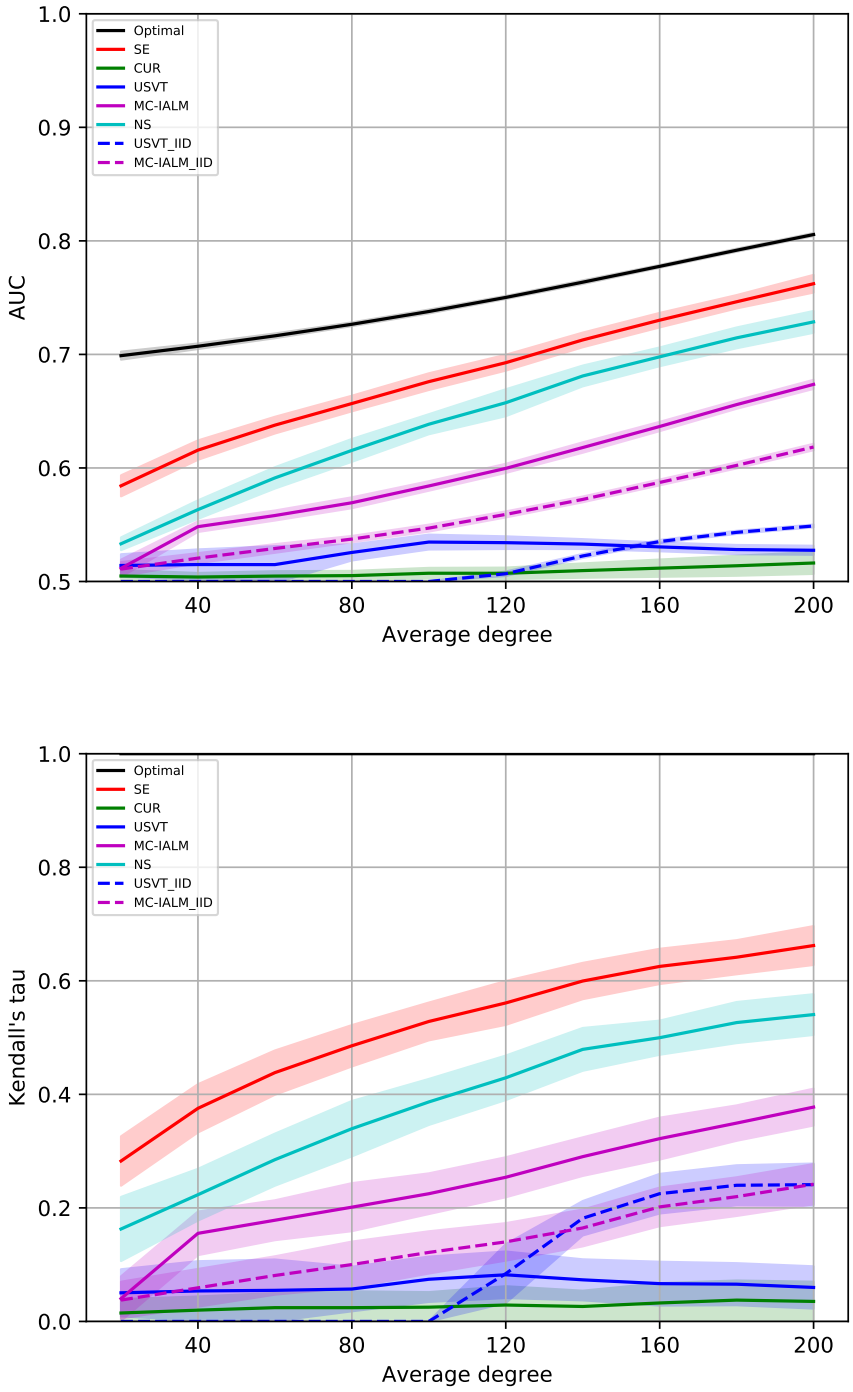


Figure 4.5: Predictive AUC and Kendall's tau for product models with various average degrees with confidence bands of 1 standard errors

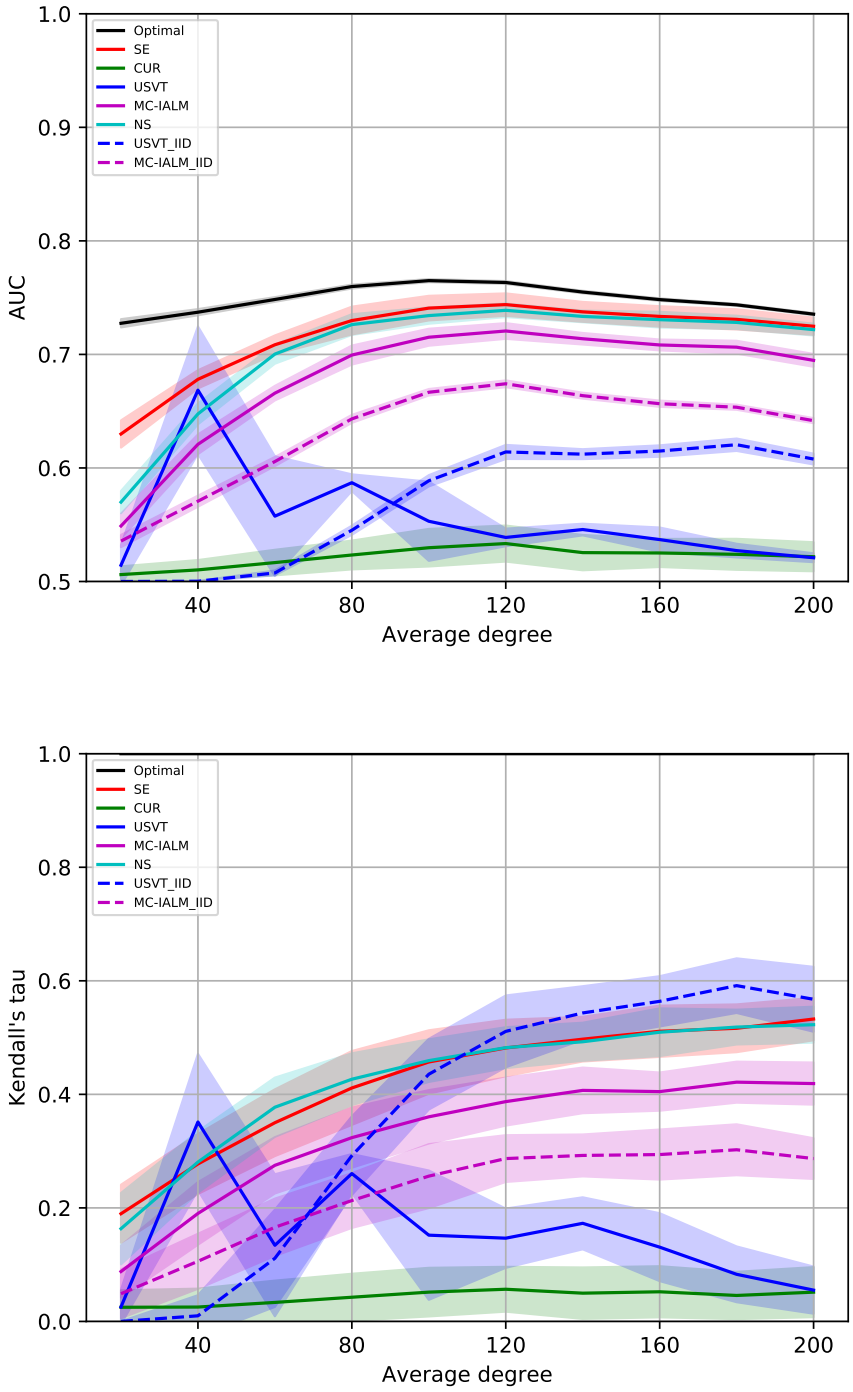


Figure 4.6: Predictive AUC and Kendall's tau for SBM with various average degrees with confidence bands of 1 standard errors

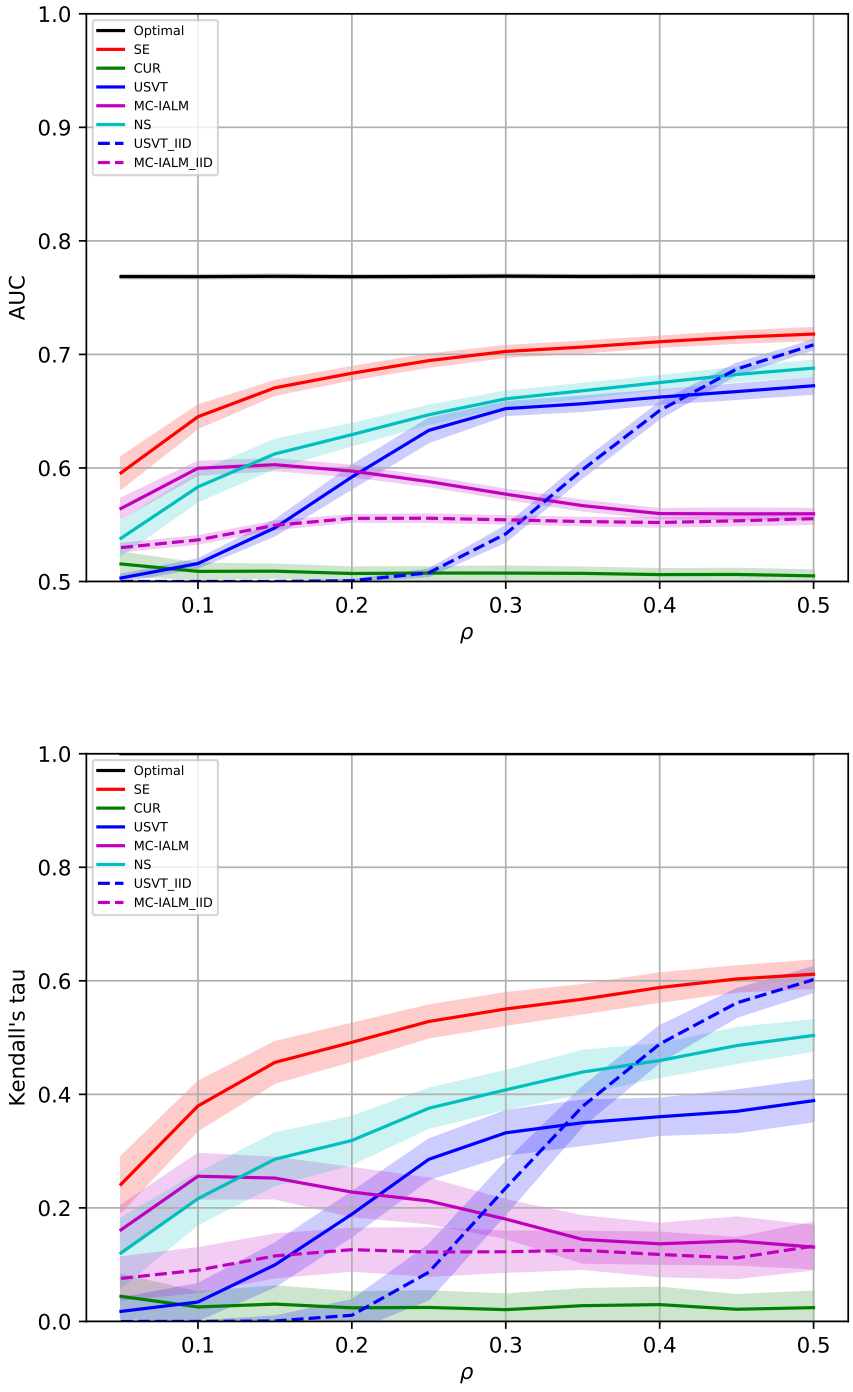


Figure 4.7: Predictive AUC and Kendall's tau for distance models with various sampling rate ρ with confidence bands of 1 standard errors

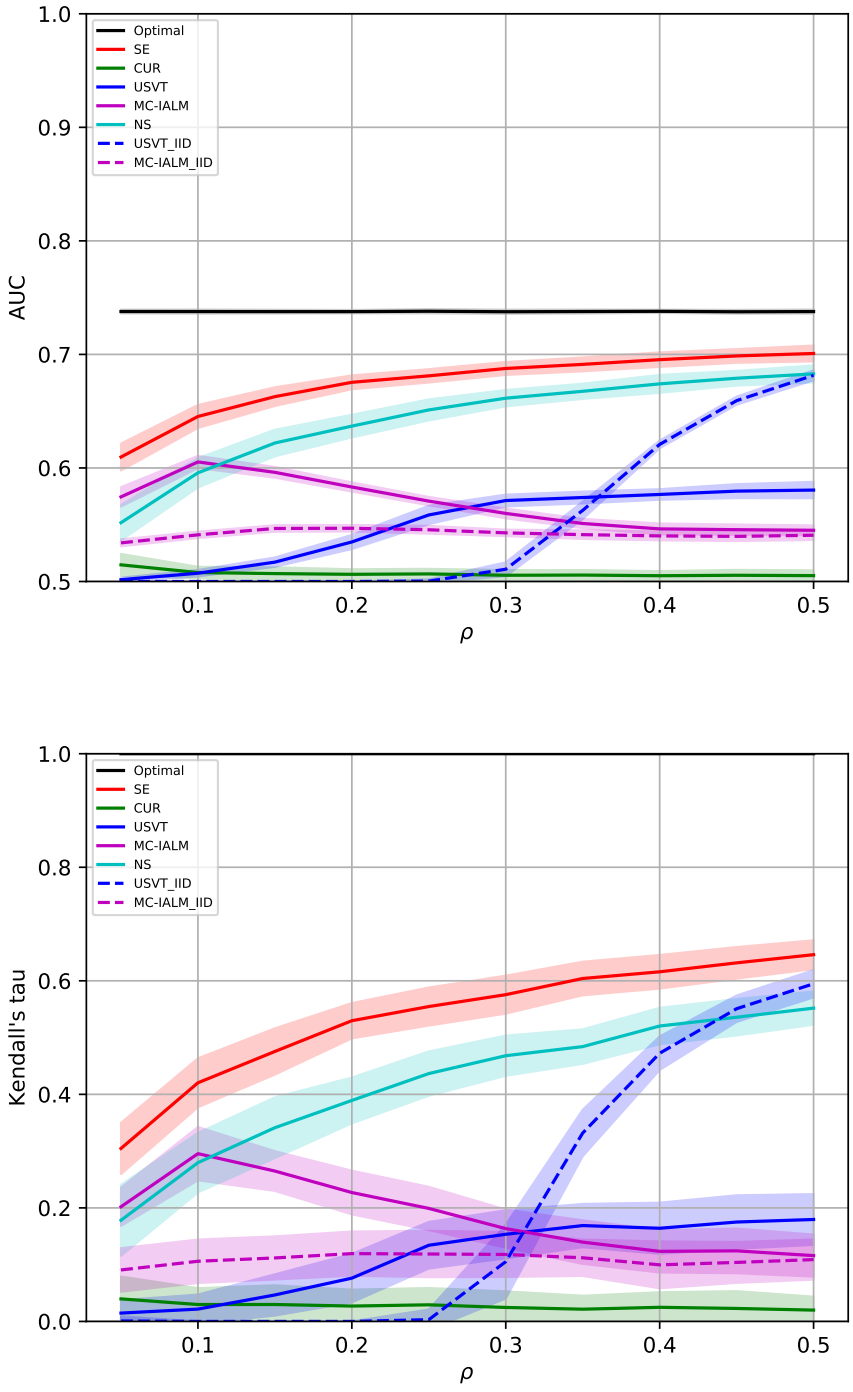


Figure 4.8: Predictive AUC and Kendall's tau for product models with various sampling rate ρ with confidence bands of 1 standard errors

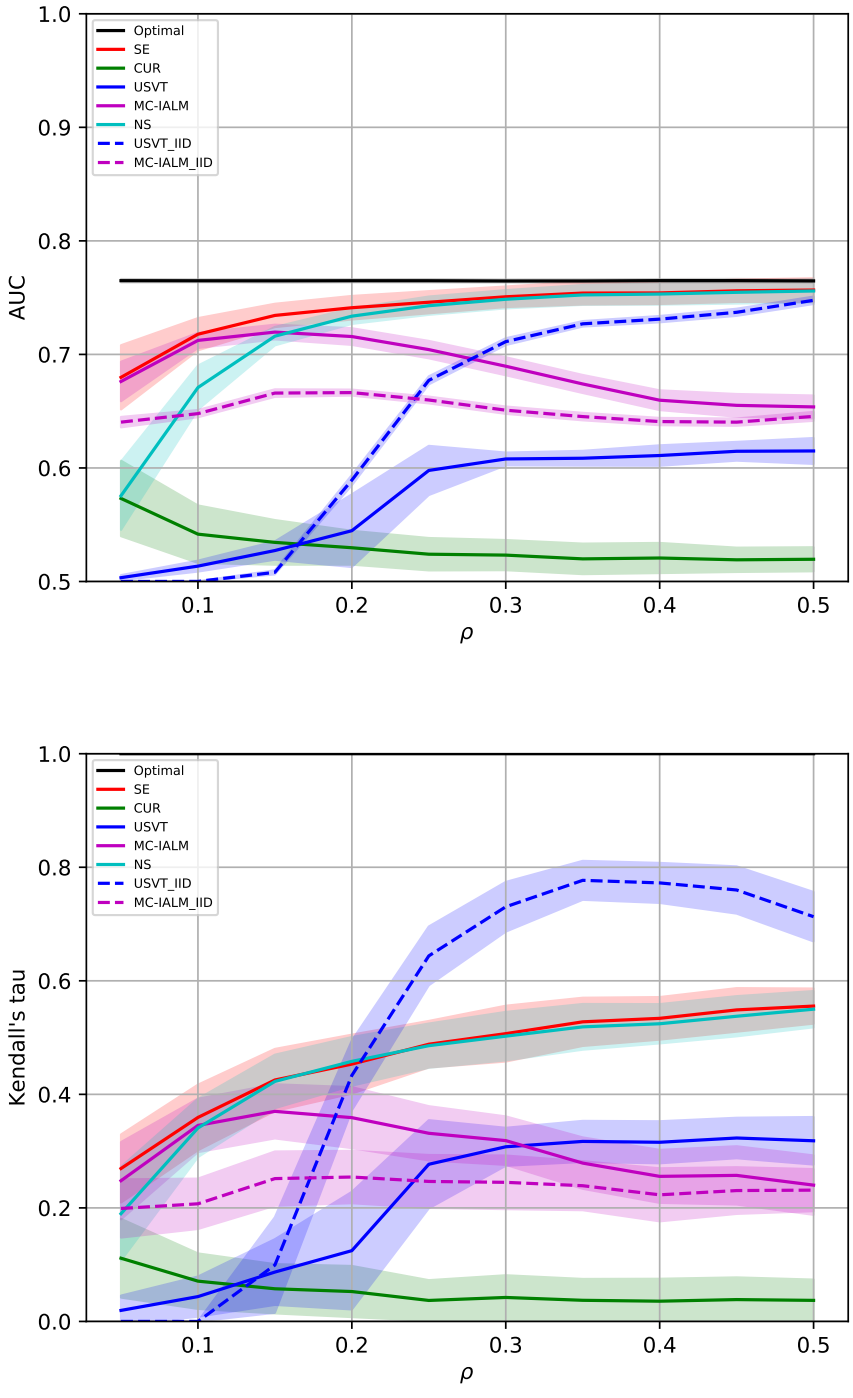


Figure 4.9: Predictive AUC and Kendall's tau for SBM with various sampling rate ρ with confidence bands of 1 standard errors

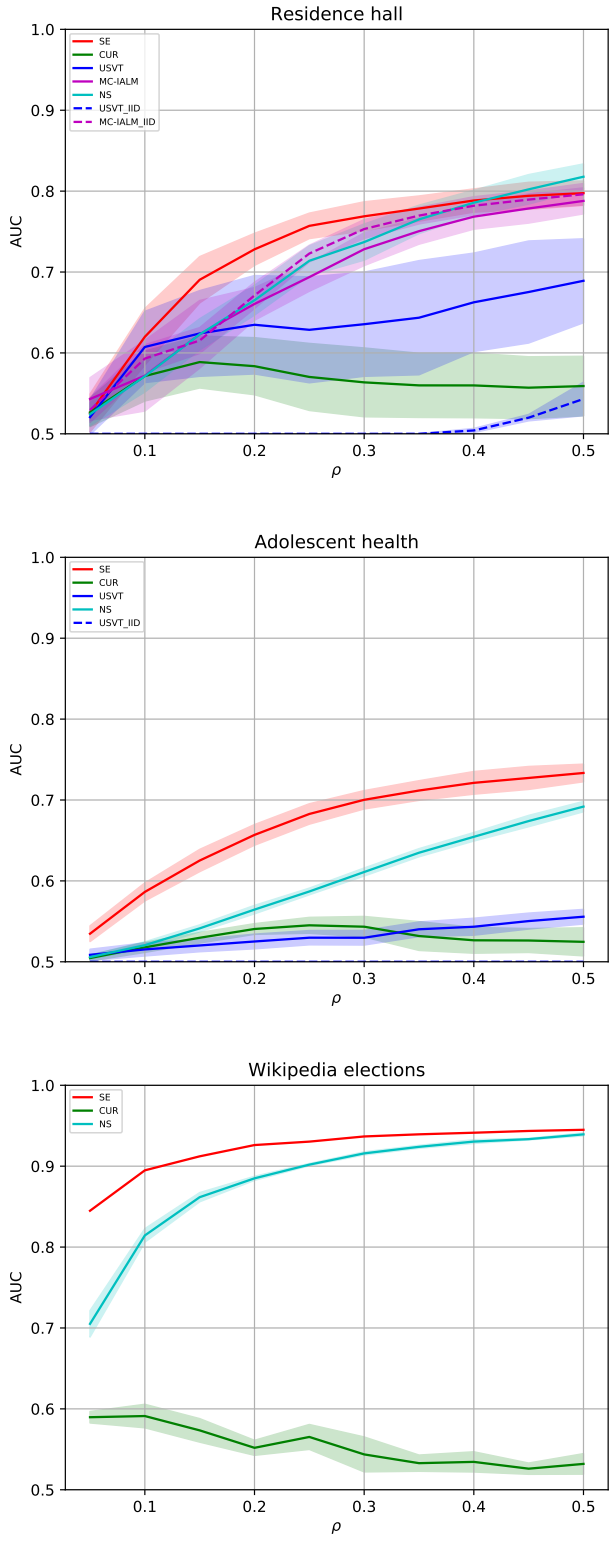


Figure 4.10: Predictive AUC for real datasets with various average degrees with confidence bands of 1 standard errors

We propose an algorithm by solving

$$\widehat{\mathbf{X}} = \arg\min_{\mathbf{X}} \|\Omega(\mathbf{A}_{obs}) - \Omega(\widehat{\mathbf{R}}^\top \mathbf{X} \widehat{\mathbf{R}})\|_F, \tag{4.4.1}$$

where the rows of $\widehat{\mathbf{R}}$ is formed by the r leading right singular vectors of $\mathbf{A}_{in}$. The above optimization problem is trying to find an approximation of the observed adjacency matrix and can be solved analytically as follows:

$$\begin{aligned}
\frac{\partial g(\mathbf{X})}{2\partial \mathbf{X}} &= \frac{\partial}{\partial \mathbf{X}}(\frac{1}{2}\|\mathbf{A}_{11} - \widehat{\mathbf{R}}_1^\top \mathbf{X} \widehat{\mathbf{R}}_1\|_F^2 + \|\mathbf{A}_{12} - \widehat{\mathbf{R}}_1^\top \mathbf{X} \widehat{\mathbf{R}}_2\|_F^2) \\
&= -\widehat{\mathbf{R}}_1^\top \mathbf{A}_{11} \widehat{\mathbf{R}}_1 + \widehat{\mathbf{R}}_1 \widehat{\mathbf{R}}_1^\top \mathbf{X} \widehat{\mathbf{R}}_1 \widehat{\mathbf{R}}_1^\top \\
&\quad - \widehat{\mathbf{R}}_1 \mathbf{A}_{12} \widehat{\mathbf{R}}_2^\top - \widehat{\mathbf{R}}_2 \mathbf{A}_{21} \widehat{\mathbf{R}}_1^\top + \widehat{\mathbf{R}}_1 \widehat{\mathbf{R}}_1^\top \mathbf{X} \widehat{\mathbf{R}}_2 \widehat{\mathbf{R}}_2^\top + \widehat{\mathbf{R}}_2 \widehat{\mathbf{R}}_2^\top \mathbf{X} \widehat{\mathbf{R}}_1 \widehat{\mathbf{R}}_1^\top \\
&= -\widehat{\mathbf{R}}_1^\top \mathbf{A}_{11} \widehat{\mathbf{R}}_1 - \widehat{\mathbf{R}}_1 \mathbf{A}_{12} \widehat{\mathbf{R}}_2^\top - \widehat{\mathbf{R}}_2 \mathbf{A}_{21} \widehat{\mathbf{R}}_1^\top + (\mathbf{X} - \widehat{\mathbf{R}}_2 \widehat{\mathbf{R}}_2^\top \mathbf{X} \widehat{\mathbf{R}}_2 \widehat{\mathbf{R}}_2^\top).
\end{aligned}$$

By solving $\partial g(\mathbf{X})/\partial \mathbf{X} = \mathbf{0}_{r \times r}$, we obtain the minimizer of g as

$$\mathrm{vec}(\overline{\mathbf{X}}) = (\mathbf{I}_{r^2} - (\widehat{\mathbf{R}}_2 \widehat{\mathbf{R}}_2^\top) \otimes (\widehat{\mathbf{R}}_2 \widehat{\mathbf{R}}_2^\top))^+ \mathrm{vec}(\widehat{\mathbf{R}}_1^\top \mathbf{A}_{11} \widehat{\mathbf{R}}_1 + \widehat{\mathbf{R}}_1 \mathbf{A}_{12} \widehat{\mathbf{R}}_2^\top + \widehat{\mathbf{R}}_2 \mathbf{A}_{21} \widehat{\mathbf{R}}_1^\top).$$

Then, the resulting estimator of $\mathbf{P}$ is $\overline{\mathbf{P}} = \widehat{\mathbf{R}}^\top \overline{\mathbf{X}} \widehat{\mathbf{R}}$. The two-stage estimation procedure gives a close form solution of $\widehat{\mathbf{X}}$ and also provides considerable computational efficiency. In our numerical experiments, this estimator slightly outperforms the method we proposed in this work particularly when ρ_N is small. The analysis of its theoretical behavior seems to be much less trivial.

This work is motivated by popular procedure of social surveys, and heavily relies on the theoretical groundwork on CUR-decomposition and sampling from matrices. As the standard CUR-decomposition and related algorithms, we often first obtain a data matrix for computing a probability distribution for conducting importance sampling to achieve a better theoretical guarantee. Interesting directions for future work on link prediction for ego-networks include design of survey procedures with importance sampling and analysis based on other sampling schemes such as snowball sampling.

CHAPTER 5

Summary and Future work

In this dissertation, we considered issues about link prediction in various types of network data and tackle them with matrix/tensor decomposition in the framework of factorization models. In Chapter 2, we proposed a model that can combine network topology and additional features. In Chapter 3, we take network dynamics into account as an additional latent factor. In Chapter 4, we investigated the link prediction incorporating with a specific generative mechanism of networks. We provided models, algorithms, theoretical analysis, and numerical study for each issue.

The framework of factorization models can provide a simple and interpretable parametrization for analyzing network data. There are a number of possible future directions to be further explored to the research of link prediction via factorization models. One direction is exploring other functions that characterize interaction between factors z_i and z_j. Currently, we describe the interaction by using inner product or scalar product of the form $f(z_i, z_j) = z_i \mathbf{\Lambda} z_j$ so that we can borrow strength from the topology structure of a low-dimensional inner/scalar product space. Although some latent space models have been built based on different interaction functions, such as distance models of the form $\|z_i - z_j\|$, solving the corresponding estimation problems are usually less tractable. It would be interesting to investigate theoretical behaviors of other interaction functions and design feasible algorithms for solving corresponding factorization problems. In addition, analysis of hypergraph data and dynamic networks also request more extensive research on functions of higher-order interaction.

Another direction to be explored is to develop methods for generating network data. Currently, we often exclude pre-processing of network data from our methods and simply assume observed links are conditionally independent given a probability matrix or latent factors, and this enables us to view network as sampling from the probability matrix and apply theoretical results from random matrix theory to analyze behaviors of our methods. However, in practice, the generative mechanism of links can seriously affect observed

topology structure of networks, which may not preserve information of interests. For example, we often computed correlation matrices of multiple time series in fMRI data and equity price data and convert them to adjacency matrices by thresholding values of the entries of the correlation matrices. The choice of the correlation measurement and threshold values in practice seems to be subject or even arbitrary, but clearly there is a gap between developing methods under the assumption of conditional independence and real generative mechanism of real network data. The gap is needed to be filled to build end-to-end network data analysis for solving practical problems.

APPENDIX A

Appendix for "Low-rank effects models for network estimation with edge attributes"

A.1 Proof of theorems

To establish consistency in Frobenius norm, we first state an inequality connecting the Frobenius norm to the Kullback-Leibler (KL) divergence, defined as

$$D_{KL}(f_{\mathbf{Q}_1} \| f_{\mathbf{Q}_2}) = n^{-2} \sum_{ij} \int_{-\infty}^{\infty} f_{q_{1,ij}}(a) \log \frac{f_{q_{1,ij}}(a)}{f_{q_{2,ij}}(a)} da,$$

where $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are $n \times n$ matrices and $f_{\mathbf{Q}_1}$ and $f_{\mathbf{Q}_2}$ are the probability distributions of random matrices with mean $\mathbf{Q}_1$ and $\mathbf{Q}_2$ as defined in (2.2.2).

Note that as a consequence of A3-A5, the ξ-th moment of $|A_{ij}|$ is uniformly bounded by some constant for each ξ, denoted by M_ξ, which does not depend on n. Then using the uniform integrability given by the bounded parameter space, we have the following lemma.

Lemma 9. *Under assumptions A3-A5, we have*

$$n^{-1} \|\mathbf{Q}_1 - \mathbf{Q}_2\|_F \leq \sqrt{2} M_{1+\delta}^{\frac{1}{1+\delta}} D_{KL}^{\frac{\delta}{2+2\delta}} (f_{\mathbf{Q}_1} \| f_{\mathbf{Q}_2})$$

for some $\delta > 0$.

Proof of Lemma 9. Let

$$\|f_{q_{1,ij}} - f_{q_{2,ij}}\|_{TV} = \sup_{g_{ij}:\mathbb{R}\to[-1,1]} \int g_{ij}(a)(f_{p_{ij}}(a) - f_{q_{ij}}(a)) d\mu(a),$$

where μ is the Lebesgue or counting measure. Then,

$$
\begin{aligned}
&\|\mathbf{Q}_1 - \mathbf{Q}_2\|_F^2 \\
&\leq \sum_{ij} \left(\int_0^\infty |a| |f_{q_{1,ij}}(a) - f_{q_{2,ij}}(a)| d\mu(a) \right)^2 \\
&\leq \sum_{ij} \left(u_{ij} \int_0^{u_{ij}} |f_{q_{1,ij}}(a) - f_{q_{2,ij}}(a)| d\mu(a) + u_{ij}^{-t} \int_{u_{ij}}^\infty |a|^{1+\delta} (f_{q_{1,ij}}(a) + f_{q_{2,ij}}(a)) d\mu(a) \right)^2 \\
&\leq \sum_{ij} \left(2u_{ij} \|f_{q_{1,ij}} - f_{q_{2,ij}}\|_{TV} + 2u_{ij}^{-t} M_{1+\delta} \right)^2
\end{aligned}
$$

As a function of u_{ij}, the minimum of $u_{ij}\|f_{q_{1,ij}} - f_{q_{2,ij}}\|_{TV} + u_{ij}^{-\delta} M_{1+\delta}$ is obtained by choosing $u_{ij} = \delta^{\frac{1}{1+\delta}} M_{1+\delta}^{\frac{1}{1+\delta}} \|f_{q_{1,ij}} - f_{q_{2,ij}}\|_{TV}^{-\frac{\delta}{1+\delta}}$ and so

$$
\begin{aligned}
n^{-2}\|\mathbf{Q}_1 - \mathbf{Q}_2\|_F^2 &\leq n^{-2} \sum_{ij} (\delta^{\frac{1}{1+\delta}} + \delta^{-\frac{\delta}{1+\delta}})^2 M_{1+\delta}^{\frac{2}{1+\delta}} \|f_{q_{1,ij}} - f_{q_{2,ij}}\|_{TV}^{\frac{2\delta}{1+\delta}} \\
&\leq 4n^{-2} M_{1+\delta}^{\frac{2}{1+\delta}} \sum_{ij} \|f_{q_{1,ij}} - f_{q_{2,ij}}\|_{TV}^{\frac{2\delta}{1+\delta}} \\
&\leq 2M_{1+\delta}^{\frac{2}{1+\delta}} D_{KL}^{\frac{\delta}{1+\delta}} (f_{\mathbf{Q}_1} \| f_{\mathbf{Q}_2})
\end{aligned}
$$

for any $\delta > 0$. The last inequality is given by Pinsker's inequality. □

Proof of Theorem 1. We define a feasible set of $(\boldsymbol{\Theta}, \boldsymbol{\beta})$ as

$$
\mathcal{T} = \{(\boldsymbol{\Theta}, \boldsymbol{\beta}) : \|\boldsymbol{\Theta}\|_* \leq \sqrt{r} n K_\theta, \|\boldsymbol{\beta}\|_2 \leq K_\beta\},
$$

and a corresponding estimator as

$$
(\widetilde{\boldsymbol{\Theta}}, \widetilde{\boldsymbol{\beta}}) = \underset{(\boldsymbol{\Theta}, \boldsymbol{\beta}) \in \mathcal{T}}{\arg\max}\, \ell_{\mathbf{A}, \mathcal{X}}(\boldsymbol{\Theta}, \boldsymbol{\beta}). \tag{A.1.1}
$$

Note that when $R = \sqrt{r} n K_\theta$ and K_β is large enough, the solution for (A.1.1) is the same as that for (2.2.6). Let $h(\mathbf{B}, \mathbf{c}) := \mathbb{E}[\ell_{\mathbf{A}, \mathbf{X}}(\mathbf{B}, \mathbf{c})]$. Note that the maximum likelihood criterion

in (2.2.8) ensures that $\ell_{\mathbf{A},\mathbf{X}}(\widehat{\mathbf{\Theta}}, \widehat{\boldsymbol{\beta}}) \geq \ell_{\mathbf{A},\mathbf{X}}(\mathbf{\Theta}, \boldsymbol{\beta})$. Hence, we have

$$\begin{aligned} n^2 D_{KL}(f_{\mathbf{P}} \| f_{\widehat{\mathbf{P}}}) &= h(\mathbf{\Theta}, \boldsymbol{\beta}) - h(\widehat{\mathbf{\Theta}}, \widehat{\boldsymbol{\beta}}) \\ &\leq \ell_{\mathbf{A},\mathbf{X}}(\widehat{\mathbf{\Theta}}, \widehat{\boldsymbol{\beta}}) - \ell_{\mathbf{A},\mathbf{X}}(\mathbf{\Theta}, \boldsymbol{\beta}) + h(\mathbf{\Theta}, \boldsymbol{\beta}) - h(\widehat{\mathbf{\Theta}}, \widehat{\boldsymbol{\beta}}) \\ &= \mathrm{tr}((\mathbf{A} - \mathbf{P})^\top (\widehat{\mathbf{\Theta}} - \mathbf{\Theta})) \\ &\quad + \sum_{k=1}^{m} (\widehat{\beta}_k - \beta_k) \mathrm{tr}((\mathbf{A} - \mathbf{P})^\top \mathbf{X}_k). \end{aligned} \tag{A.1.2}$$

To see the vanishing of the first term as n goes to infinity, one can derive that

$$\begin{aligned} \mathrm{tr}((\mathbf{A} - \mathbf{P})^\top (\widehat{\mathbf{\Theta}} - \mathbf{\Theta})) &\leq 2 \sup_{\mathbf{\Xi} \in \mathcal{T}} |\mathrm{tr}((\mathbf{A} - \mathbf{P})^\top \mathbf{\Xi})| \\ &\leq 2\sigma_1(\mathbf{A} - \mathbf{P}) \sup_{\mathbf{\Xi} \in \mathcal{T}} \|\mathbf{\Xi}\|_* \\ &\leq 2\sqrt{r} n R^* \sigma_1(\mathbf{A} - \mathbf{P}) \end{aligned}$$

by matrix norm inequalities $|\mathrm{tr}(\mathbf{B}^\top \mathbf{C})| \leq \|\mathbf{B}\|_2 \|\mathbf{C}\|_*$ and $\|\mathbf{C}\|_* \leq \sqrt{r}\|\mathbf{C}\|_F \leq \sqrt{r} n \|\mathbf{C}\|_{\max}$ for $\mathrm{rank}\mathbf{C} \leq r$. Together with Markov's inequality and the fact that

$$\begin{aligned} \mathbb{E}[\sigma_1(\mathbf{A} - \mathbf{P})] &\leq C_0 \bigg(\Big(\max_i \sum_j \mathbb{E}[A_{ij}^2] \Big)^{\frac{1}{2}} + \Big(\max_j \sum_i \mathbb{E}[A_{ij}^2] \Big)^{\frac{1}{2}} + \sum_{ij} \mathbb{E}[A_{ij}^4] \Big)^{\frac{1}{4}} \bigg) \\ &\leq C_0 \sqrt{n} (2\sqrt{M_2} + \sqrt[4]{M_4}) \end{aligned}$$

by Latala's theorem (Latala, 2005) where C_0 is some universal constant, we have

$$\begin{aligned} \mathbb{P}\Big(2 \sup_{\mathbf{\Xi} \in \mathcal{T}} |\mathrm{tr}((\mathbf{A} - \mathbf{P})^\top \mathbf{\Xi})| \geq n^2 \delta \Big) &\leq \mathbb{P}(2\sqrt{r} n R^* \sigma_1(\mathbf{A} - \mathbf{P}) \geq n^2 t) \\ &\leq \frac{2\sqrt{r} R \mathbb{E}[\sigma_1(\mathbf{A} - \mathbf{P})]}{nt} \\ &\leq \frac{2\sqrt{r} R C_0 (2\sqrt{M_2} + \sqrt[4]{M_4})}{\sqrt{n} t}. \end{aligned} \tag{A.1.3}$$

For the second term in (A.1.2),

$$\begin{aligned}
&\mathbb{P}\Big(\Big|\sum_{k=1}^{m}(\widehat{\beta}_k-\beta_k)\mathrm{tr}((\mathbf{A}-\mathbf{P})^\top\mathbf{X}_k)\Big|\geq n^2t\Big)\\
&\leq \mathbb{P}\Big(2\sup_{\|\boldsymbol{\beta}\|_{\max}\leq K_\beta}\|\boldsymbol{\beta}\|_{\max}\Big|\sum_{k=1}^{m}\mathrm{tr}((\mathbf{A}-\mathbf{P})^\top\mathbf{X}_k)\Big|\geq n^2t\Big)\\
&\leq \frac{4K_\beta^2\,\mathrm{Var}\left(\sum_{k=1}^{m}\mathrm{tr}(\mathbf{A}^\top\mathbf{X}_k)\right)}{n^4t^2}\\
&\leq \frac{4K_\beta^2K_x^2M_2}{n^2t^2}. \qquad \text{(A.1.4)}
\end{aligned}$$

Thus, the desired result follows from (A.1.3), (A.1.4), and Lemma 9. □

Proof of Corollary 3. The result is obtained by replacing (A.1.3) with Talagrand's inequality

$$\begin{aligned}
&\mathbb{P}(2\sqrt{r}nR^*\sigma_1(\mathbf{A}-\mathbf{P})\geq n^2t)\\
&\leq \mathbb{P}\Big(|\sigma_1(\mathbf{A}-\mathbf{P})-\mathbb{E}[\sigma_1(\mathbf{A}-\mathbf{P})]|\geq \frac{nt}{2\sqrt{r}R}-C_0(2\sqrt{M_2}+\sqrt[4]{M_4})\sqrt{n}\Big)\\
&\leq C_1\exp\Big(-C_2\Big(\frac{nt}{2\sqrt{r}R}-C_0(2\sqrt{M_2}+\sqrt[4]{M_4})\sqrt{n}\Big)_+^2\Big),
\end{aligned}$$

where C_1 and C_2 are some universal constants, and (A.1.4) with Hoeffiding's inequality

$$\mathbb{P}\Big(\Big|\sum_{k=1}^{m}(\widehat{\beta}_k-\beta_k)\mathrm{tr}((\mathbf{A}-\mathbf{P})^\top\mathbf{X}_k)\Big|\geq n^2t\Big)\leq 2\exp\Big(-\frac{n^2t^2}{4K_\beta^2K_x^2}\Big).$$

□

Proof of Corollary 4. By Taylor's expansion, for some η_{ij} between $\widehat{p}_{ij}$ and p_{ij} for $i,j=1,\ldots,n$,

$$\begin{aligned}
\|\widehat{\boldsymbol{\Theta}}-\boldsymbol{\Theta}+\mathcal{X}\otimes(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})\|_F&=\Big(\sum_{ij}\big(L(\widehat{p}_{ij})-L(p_{ij})\big)^2\Big)^{\frac{1}{2}}\\
&\leq \sup_{ij}L'(\eta_{ij})\|\widehat{\mathbf{P}}-\mathbf{P}\|_F\\
&\leq \frac{1}{\inf_{ij}b''(L(\eta_{ij}))}\|\widehat{\mathbf{P}}-\mathbf{P}\|_F\\
&\leq \frac{1}{\inf_{ij}\mathrm{Var}(A_{ij})}\|\widehat{\mathbf{P}}-\mathbf{P}\|_F.
\end{aligned}$$

Hence, the convergence of the linear predictor $\widehat{\boldsymbol{\Theta}} + \mathbf{X} \otimes \widehat{\boldsymbol{\beta}}$ follows from $\inf_{ij} \mathrm{Var}(A_{ij})$ being bounded away from 0. Since

$$\begin{aligned}
\frac{\left|\mathrm{tr}\Big((\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta})^\top \big(\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\big)\Big)\right|}{\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F \|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F} &\leq \frac{\sum_{i=1}^{n} \sigma_i(\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}) \sigma_i\big(\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\big)}{\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F \|\mathbf{X} \otimes (\widehat{\boldsymbol{\beta}}^* - \boldsymbol{\beta})\|_F} \\
&= \frac{\sum_{i=1}^{2r} \sigma_i(\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}) \sigma_i\big(\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\big)}{\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F \|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F} \\
&\leq \frac{\Big(\sum_{i=1}^{2r} \sigma_i^2\big(\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\big)\Big)^{\frac{1}{2}}}{\|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F} \\
&\leq \sqrt{\delta}
\end{aligned}$$

by the condition on the spectral distribution of $\mathcal{X} \otimes \boldsymbol{\beta}$, we see that

$$\begin{aligned}
&\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta} + \mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F^2 \\
&= \|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F^2 + \|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F^2 + 2\mathrm{tr}\Big((\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta})^\top \big(\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\big)\Big) \\
&\geq \|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F^2 + \|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F^2 - 2\sqrt{\delta}\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F \|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F \\
&\geq (1 - \sqrt{\delta})(\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F^2 + \|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F^2).
\end{aligned}$$

Thus, by Theorem 1,

$$n^{-1}\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F \xrightarrow{p} 0$$

and

$$(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})^\top \Big(n^{-2} \sum_{ij} \mathbf{x}_{ij} \mathbf{x}_{ij}^\top\Big)(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) = n^{-2}\|\mathcal{X} \otimes (\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_F^2 \xrightarrow{p} 0.$$

□

Proof of Theorem 5. Let $\widehat{\boldsymbol{\Theta}}^* = \arg\min_{\boldsymbol{\Xi} \in \mathcal{T}} \|\boldsymbol{\Xi} - \boldsymbol{\Theta}\|_F$.

$$\begin{aligned}
n^2 D_{KL}(f_{\mathbf{P}} \| f_{\widehat{\mathbf{P}}}) &= h(\boldsymbol{\Theta}, \boldsymbol{\beta}) - h(\widehat{\boldsymbol{\Theta}}, \widehat{\boldsymbol{\beta}}) \\
&\leq \ell_{\mathbf{A},\mathbf{X}}(\widehat{\boldsymbol{\Theta}}, \widehat{\boldsymbol{\beta}}) - h(\widehat{\boldsymbol{\Theta}}, \widehat{\boldsymbol{\beta}}) - \ell_{\mathbf{A},\mathbf{X}}(\widehat{\boldsymbol{\Theta}}^*, \boldsymbol{\beta}) + h(\widehat{\boldsymbol{\Theta}}^*, \boldsymbol{\beta}) \qquad \text{(A.1.5)} \\
&\quad - h(\widehat{\boldsymbol{\Theta}}^*, \boldsymbol{\beta}) + h(\boldsymbol{\Theta}, \boldsymbol{\beta}) \\
&= \mathrm{tr}((\mathbf{A} - \mathbf{P})^\top (\widehat{\boldsymbol{\Theta}} - \widehat{\boldsymbol{\Theta}}^*)) + \sum_{k=1}^{m} (\widehat{\beta}_k - \beta_k) \mathrm{tr}((\mathbf{A} - \mathbf{P})^\top \mathbf{X}_k) \\
&\quad + \mathrm{tr}(\mathbf{P}^\top (\boldsymbol{\Theta} - \widehat{\boldsymbol{\Theta}}^*)) + \sum_{ij} \big(b(\widehat{\theta}_{ij}^* + \mathbf{x}_{ij}^\top \boldsymbol{\beta}) - b(\theta_{ij} + \mathbf{x}_{ij}^\top \boldsymbol{\beta})\big) \qquad \text{(A.1.6)}
\end{aligned}$$

The first two terms above converge to 0 in probability by a similar argument in the proof of Theorem 1. Note that

$$\mathrm{tr}(\mathbf{P}^\top(\mathbf{\Theta} - \widehat{\mathbf{\Theta}}^*)) \leq \sigma_1(\mathbf{P})\|\mathbf{\Theta} - \widehat{\mathbf{\Theta}}^*\|_* \leq n \sum_{k=r+1}^{n} \sigma_k(\mathbf{\Theta})$$

and that, by Taylor's expansion, for some ξ_{ij} between $\widehat{\theta}^*_{ij} + \mathbf{x}_{ij}^\top\boldsymbol{\beta}$ and $\theta_{ij} + \mathbf{x}_{ij}^\top\boldsymbol{\beta}$,

$$\begin{aligned}
\sum_{ij} \big(b(\widehat{\theta}^*_{ij} + \mathbf{x}_{ij}^\top\boldsymbol{\beta}) - b(\theta_{ij} + \mathbf{x}_{ij}^\top\boldsymbol{\beta})\big) &= \sum_{ij} b'(\xi_{ij})(\widehat{\theta}^*_{ij} - \theta_{ij}) \\
&\leq K_p \sum_{ij} |\widehat{\theta}^*_{ij} - \theta_{ij}| \\
&\leq nK_p \|\widehat{\mathbf{\Theta}}^* - \mathbf{\Theta}\|_F \\
&\leq nK_p \|\widehat{\mathbf{\Theta}}^* - \mathbf{\Theta}\|_* \\
&= nK_p \sum_{k=r+1}^{n} \sigma_k(\mathbf{\Theta}),
\end{aligned}$$

Therefore,

$$D_{KL}(f_{\mathbf{P}} \| f_{\widehat{\mathbf{P}}}) = O_p\bigg(n^{-1} \sum_{k=r+1}^{n} \sigma_k(\mathbf{\Theta})\bigg)$$

and by Lemma 9, for $\delta > 0$,

$$n^{-1}\|\widehat{\mathbf{P}} - \mathbf{P}\|_F = O_p\bigg(M_{1+\delta}^{\frac{1}{1+\delta}} \Big(n^{-1} \sum_{k=r+1}^{n} \sigma_k(\mathbf{\Theta})\Big)^{\frac{\delta}{2+2\delta}}\bigg).$$

□

APPENDIX B

Appendix for "Regularized tensor decomposition for link prediction in dynamic networks"

B.1 Proof of Theorem 6

Proof of Theorem 6. We prove this theorem by contradiction. Suppose that, for any partition of $[0, T]$, there exists a non-zero vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_\ell)^\top$ such that not all β_ℓ's are zero and $\mathbf{U}\boldsymbol{\beta} = \mathbf{0}$.

By the continuity of u_ℓ^*'s, the function $\sum_{\ell=1}^r \alpha_\ell u_\ell^*$ is uniformly continuous for any $\{\alpha_\ell\}$. That is, given $\epsilon > 0$, there exists $\delta > 0$ such that $|\sum_{\ell=1}^r \alpha_\ell u_\ell^*(t) - \sum_{\ell=1}^r \alpha_\ell u_\ell^*(s)| < \epsilon$ for all $|s - t| < \delta$. Consider a partition $t_0 = 0 < t_1 < \cdots < t_{n_T} = T$ on $[0, T]$ satisfying $t_k - t_{k-1} < \delta$ for all $k = 1, \ldots, n_T$. Since there exists $s_{\ell k} \in (t_{k-1}, t_k)$ such that $\int_{t_{k-1}}^{t_k} u_\ell^*(t)dt = (t_k - t_{k-1})u_\ell^*(s_{\ell k})$ for all $k = 1 \ldots, n_T$ and $\ell = 1, \ldots, r$, we have

$$u_{\ell k} = \frac{\int_{t_{k-1}}^{t_k} u_\ell^*(t)dt}{(\sum_{k'=1}^{n_T}(\int_{t_{k'-1}}^{t_{k'}} u_\ell^*(t)dt)^2)^{\frac{1}{2}}} = \frac{(t_k - t_{k-1})u_\ell^*(s_{\ell k})}{(\sum_{k=1}^{n_T}(t_{k'} - t_{k'-1})^2 u_\ell^{*2}(s_{\ell k'}))^{\frac{1}{2}}}.$$

Let $\alpha_\ell = (\sum_{k'=1}^{n_T}(t_{k'} - t_{k'-1})^2 u_\ell^{*2}(s_{\ell k'}))^{-\frac{1}{2}}\beta_\ell$. For $t \in (t_{k-1}, t_k]$,

$$\begin{aligned}
\left|\sum_{\ell=1}^r \alpha_\ell u_\ell^*(t)\right| &\le \left|\sum_{\ell=1}^r \alpha_\ell u_\ell^*(s_{\ell k})\right| + \left|\sum_{\ell=1}^r \alpha_\ell (u_\ell^*(s_{\ell k}) - u_\ell^*(t))\right| \\
&\le \left|(t_k - t_{k-1})^{-1}\sum_{\ell=1}^r \beta_\ell u_{\ell k}\right| + \sum_{\ell=1}^r |\alpha_\ell||u_\ell^*(s_{\ell k}) - u_\ell^*(t)| \\
&\le 0 + \epsilon\sum_{\ell=1}^r |\alpha_\ell|
\end{aligned}$$

Since $|\alpha_\ell|$ are not all zero and ϵ can be arbitrarily small, $\sum_{\ell=1}^r \alpha_\ell u_\ell^* \equiv 0$ and hence u_ℓ^*'s are not linearly independent. □

B.2 Proof of Theorem 7

B.2.1 Concentration in spectral norm

The spectral norm of $\mathcal{E}$ is defined as

$$\|\mathcal{E}\|_2 := \max_{\mathbf{y}\in\mathcal{S}^{n-1},\mathbf{z}\in\mathcal{S}^{n-1},\mathbf{u}\in\mathcal{S}^{n_T-1}} \mathcal{E}\otimes_1\mathbf{y}\otimes_2\mathbf{z}\otimes_3\mathbf{u}.$$

The following lemma gives a concentration inequality for $\mathcal{A}$ in spectral norm.

Lemma 10 (Spectral norm of $\mathcal{A}-\mathbb{E}[\mathcal{A}]$)**.** *Suppose that $\mathcal{A}_{ijk}\sim\mathrm{Poisson}(\mu_{ijk})$. Then,*

$$\mathbb{P}(\sigma_{\max}^{-1}\|\mathcal{A}-\mathbb{E}[\mathcal{A}]\|_2\geq\delta)\leq\exp\bigg(-\frac{\sigma_{\max}\delta^2}{8+4\delta}+(2n+n_T)\log 15\bigg).$$

Proof of Lemma 10. Let $\mathcal{E}=\mathcal{A}-\mathbb{E}[\mathcal{A}]$. Let $(\mathbf{y}^*,\mathbf{z}^*,\mathbf{u}^*)$ be the maximizer of $g(\mathbf{y},\mathbf{z},\mathbf{u})=\mathcal{E}\otimes_1\mathbf{y}\otimes_2\mathbf{z}\otimes_3\mathbf{u}$. Given a ϵ-net $\mathcal{N}_\epsilon$, which is a finite points set, on $\mathcal{S}^{n-1}\times S^{n-1}\times\mathcal{S}^{n_T-1}$, i.e. for all $\mathbf{x}\in\mathcal{S}^{n-1}\times S^{n-1}\times\mathcal{S}^{n_T-1}$, there exists $(\widetilde{\mathbf{y}},\widetilde{\mathbf{z}},\widetilde{\mathbf{u}})\in\mathcal{N}_\epsilon$ such that $\|\widetilde{\mathbf{y}}-\mathbf{y}^*\|<\epsilon$, $\|\widetilde{\mathbf{z}}-\mathbf{z}^*\|<\epsilon$, and $\|\widetilde{\mathbf{u}}-\mathbf{u}^*\|<\epsilon$, we have

$$\begin{aligned}
\|\mathcal{E}\|_2 &= \mathcal{E}\otimes_1\mathbf{y}^*\otimes_2\mathbf{z}^*\otimes_3\mathbf{u}^*\\
&= \mathcal{E}\otimes_1(\widetilde{\mathbf{y}}+(\mathbf{y}^*-\widetilde{\mathbf{y}}))\otimes_2(\widetilde{\mathbf{z}}+(\mathbf{z}^*-\widetilde{\mathbf{z}}))\otimes_3(\widetilde{\mathbf{u}}+(\mathbf{u}^*-\widetilde{\mathbf{u}}))\\
&\leq \mathcal{E}\otimes_1\widetilde{\mathbf{y}}\otimes_2\widetilde{\mathbf{z}}\otimes_3\widetilde{\mathbf{u}}+(3\epsilon+3\epsilon^2+\epsilon^3)\|\mathcal{E}\|_2\\
&\leq \max_{(\mathbf{y},\mathbf{z},\mathbf{u})\in\mathcal{N}_\epsilon}\mathcal{E}\otimes_1\mathbf{y}\otimes_2\mathbf{z}\otimes_3\mathbf{u}+(3\epsilon+3\epsilon^2+\epsilon^3)\|\mathcal{E}\|_2
\end{aligned}$$

Let $X\overset{d}{=}\sum_i c_i(Y_i-\mu_i)$, where Y_i's are independently sampled from $Poisson(\mu_i)$ and c_i's are some constants. Bernstein's inequality for sub-exponential distributions gives

$$\mathbb{P}(X\geq\theta)\leq\exp\bigg(-\frac{\theta^2}{2(\mu+\theta)}\bigg),$$

where $\mu=\sum_i c_i\mu_i$. Since the covering number of ϵ-net on $\mathcal{S}^{n-1}$ is bounded by $(\frac{2}{\epsilon})^n$, by

choosing $\epsilon = 0.14$, which gives $(1 - 3\epsilon - 3\epsilon^2 - \epsilon^3) > 0.5$ and $\frac{2}{\epsilon} < 15$, we obtain

$$\begin{aligned}
\mathbb{P}(\|\mathcal{E}\|_2 \geq \theta) &\leq \mathbb{P}((1 - 3\epsilon - 3\epsilon^2 - \epsilon^3)^{-1} \max_{(\mathbf{y},\mathbf{z},\mathbf{u}) \in \mathcal{N}_\epsilon} \mathcal{E} \otimes_1 \mathbf{y} \otimes_2 \mathbf{z} \otimes_3 \mathbf{u} \geq \theta) \\
&\leq \sum_{(\mathbf{y},\mathbf{z},\mathbf{u}) \in \mathcal{N}_\epsilon} \mathbb{P}(\mathcal{E} \otimes_1 \mathbf{y} \otimes_2 \mathbf{z} \otimes_3 \mathbf{u} \geq (1 - 3\epsilon - 3\epsilon^2 - \epsilon^3)\theta) \\
&\leq \sum_{(\mathbf{y},\mathbf{z},\mathbf{u}) \in \mathcal{N}_\epsilon} \exp\bigg(- \frac{(1 - 3\epsilon - 3\epsilon^2 - \epsilon^3)^2 \theta^2}{2(\sigma_{\max} + (1 - 3\epsilon - 3\epsilon^2 - \epsilon^3)\theta)} \bigg) \\
&\leq \bigg(\frac{2}{\epsilon}\bigg)^{2n + n_T} \exp\bigg(- \frac{(1 - 3\epsilon - 3\epsilon^2 - \epsilon^3)^2 \theta^2}{2(\sigma_{\max} + (1 - 3\epsilon - 3\epsilon^2 - \epsilon^3)\theta)} \bigg) \\
&\leq 15^{2n + n_T} \exp\bigg(- \frac{\theta^2}{4(2\sigma_{\max} + \theta)} \bigg).
\end{aligned}$$

The proof is completed by setting $\theta = \sigma_{\max}\delta$. □

B.2.2 One update of the power iteration

Lemma 11 (One update of $\mathbf{z}$)**.** *Assume that* $\mathbf{z}^\top \mathbf{z}_k > \sqrt{1 - \epsilon^2}$, $\mathbf{u}^\top \mathbf{u}_k > \sqrt{1 - \epsilon^2}$. *Then, for* $\widehat{\mathbf{z}} := \phi(\rho \mathcal{A} \otimes_2 \mathbf{z} \otimes_3 \mathbf{u} + (1 - \rho)\mathbf{z})$, *we have*

$$\sqrt{1 - (\widehat{\mathbf{z}}^\top \mathbf{z}_k)^2} \leq \frac{\rho(J_z(\epsilon) + \|\mathcal{E}\|_2) + (1 - \rho)\epsilon}{\rho(\sigma_{\min}(1 - \epsilon^2) - J_z(\epsilon) - \|\mathcal{E}\|_2) - (1 - \rho)}$$

if $\sigma_{\min}(1 - \epsilon^2) > J_z(\epsilon) + \|\mathcal{E}\|_2$, *where*

$$J_z(\epsilon) := \sigma_{\max}(\eta_z \eta_u \sqrt{r(1 + \eta_z r)} + \epsilon(\eta_z + \eta_u + 2r\eta_z\eta_u) + \epsilon^2 r).$$

Proof of Lemma 11. Let $\mathcal{P}_{\mathbf{v}}$ and $\mathcal{P}_{\mathbf{v}}^\perp$ be the orthogonal projections onto $\mathbf{v}$ and its orthogonal complement, respectively. Let $\mathbf{Z}_{[k]} = [\mathbf{z}_1, \ldots, \mathbf{z}_{k-1}, \mathbf{z}_{k+1}, \ldots, \mathbf{z}_r]$. To show that $\widehat{\mathbf{z}}^\top \mathbf{z}_k$ is close to 1, we will show that, for any $\mathbf{x} \perp \mathbf{z}_k$ with $\|\mathbf{x}\|_2 = 1$, $I_z(\epsilon) = \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathbf{z} \otimes_3 \mathbf{u}$ can be bounded by a small value, which is a function of ϵ.

$$\begin{aligned}
I_z(\epsilon) &= \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathbf{z} \otimes_3 \mathbf{u} \\
&= \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 (\mathcal{P}_{\mathbf{z}_k} + \mathcal{P}_{\mathbf{z}_k}^\perp)(\mathbf{z}) \otimes_3 (\mathcal{P}_{\mathbf{u}_k} + \mathcal{P}_{\mathbf{u}_k}^\perp)(\mathbf{u}) \\
&= \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}(\mathbf{z}) \otimes_3 \mathcal{P}_{\mathbf{u}_k}(\mathbf{u}) + \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}^\perp(\mathbf{z}) \otimes_3 \mathcal{P}_{\mathbf{u}_k}(\mathbf{u}) \\
&\quad + \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}(\mathbf{z}) \otimes_3 \mathcal{P}_{\mathbf{u}_k}^\perp(\mathbf{u}) + \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}(\mathbf{z})^\perp \otimes_3 \mathcal{P}_{\mathbf{u}_k}^\perp(\mathbf{u}) \\
&= I_1(\epsilon) + I_2(\epsilon) + I_3(\epsilon) + I_4(\epsilon)
\end{aligned}$$

We can bound each term as follows. Since $\mathcal{P}_{\mathbf{z}_k}(\mathbf{z}) = \mathbf{z}_k^\top \mathbf{z} \mathbf{z}_k$ and $\mathcal{P}_{\mathbf{u}_k}(\mathbf{u}) = \mathbf{u}_k^\top \mathbf{u} \mathbf{u}_k$, we

have

$$
\begin{aligned}
I_1(\epsilon) &= \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}(\mathbf{z}) \otimes_3 \mathcal{P}_{\mathbf{u}_k}(\mathbf{u}) \\
&= \sum_{\ell \neq k} \sigma_\ell \mathbf{z}^\top \mathbf{z}_k \mathbf{z}_k^\top \mathbf{z}_\ell \mathbf{u}^\top \mathbf{u}_k \mathbf{u}_k^\top \mathbf{u}_\ell \mathbf{z}_\ell^\top \mathbf{x} \\
&\leq \sigma_{\max} \eta_z \eta_u \mathbf{z}^\top \mathbf{z}_k \mathbf{u}^\top \mathbf{u}_k \sum_{\ell \neq k} \mathbf{z}_\ell^\top \mathbf{x} \qquad \text{(by } \mathbf{z}_k^\top \mathbf{z}_\ell \leq \eta_z \text{ and } \mathbf{u}_k^\top \mathbf{u}_\ell \leq \eta_u\text{)} \\
&\leq \sigma_{\max} \eta_z \eta_u |\mathbf{x}^\top \mathbf{Z}_{[k]} \mathbf{1}| \\
&\leq \sigma_{\max} \eta_z \eta_u \|\mathbf{Z}_{[k]} \mathbf{1}\| \\
&\leq \sigma_{\max} \eta_z \eta_u \sqrt{r(1 + \eta_z r)}.
\end{aligned}
$$

The last inequality is obtained by $\|\mathbf{Z}_{[k]}\|^2 \leq (1 + \eta_z(r-2))$, which is given by the Gershgorin circle theorem and the fact that the off-diagonal terms of $\mathbf{Z}_{[k]}^\top \mathbf{Z}_{[k]}$ are less than or equal to η_z. By the facts that $\|\mathcal{P}_{\mathbf{z}_k}^\perp(\mathbf{z})\| < \epsilon$ and $\|\mathcal{P}_{\mathbf{u}_k}^\perp(\mathbf{u})\| < \epsilon$, we have

$$
\begin{aligned}
I_2(\epsilon) &= \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}^\perp(\mathbf{z}) \otimes_3 \mathcal{P}_{\mathbf{u}_k}(\mathbf{u}) \\
&= \sum_{\ell \neq k} \sigma_\ell \mathbf{z}_\ell^\top \mathcal{P}_{\mathbf{z}_k}^\perp(\mathbf{z}) \mathbf{u}^\top \mathbf{u}_k \mathbf{u}_k^\top \mathbf{u}_\ell \mathbf{z}_\ell^\top \mathbf{x} \\
&\leq \sigma_{\max} \eta_u \mathcal{P}_{\mathbf{z}_k}^\perp(\mathbf{z})^\top \sum_{\ell \neq k} \mathbf{z}_\ell \mathbf{z}_\ell^\top \mathbf{x} \qquad \text{(by } \mathbf{u}_k^\top \mathbf{u}_\ell < \eta_u\text{)} \\
&\leq \sigma_{\max} \epsilon \eta_u \|\mathbf{Z}_{[k]}\|^2 \\
&\leq \sigma_{\max} \epsilon \eta_u (1 + \eta_z r).
\end{aligned}
$$

Similarly, $I_3(\epsilon) \leq \sigma_{\max} \epsilon \eta_z (1 + \eta_u r)$. Finally,

$$
\begin{aligned}
I_4(\epsilon) &= \mathbb{E}[\mathcal{A}] \otimes_1 \mathbf{x} \otimes_2 \mathcal{P}_{\mathbf{z}_k}(\mathbf{z})^\perp \otimes_3 \mathcal{P}_{\mathbf{u}_k}^\perp(\mathbf{u}) \\
&= \sum_{\ell \neq k} \sigma_\ell \mathbf{z}_\ell^\top \mathcal{P}_{\mathbf{z}_k}^\perp(\mathbf{z}) \mathbf{u}_\ell^\top \mathcal{P}_{\mathbf{u}_k}^\perp(\mathbf{u}) \mathbf{z}_\ell^\top \mathbf{x} \\
&\leq \epsilon^2 \sigma_{\max} r
\end{aligned}
$$

Then,

$$
\begin{aligned}
I(\epsilon) &= I_1(\epsilon) + I_2(\epsilon) + I_3(\epsilon) + I_4(\epsilon) \\
&\leq \sigma_{\max}(\eta_z \eta_u \sqrt{r(1 + \eta_z r)} + \epsilon(\eta_z + \eta_u + 2r\eta_z \eta_u) + \epsilon^2 r) \\
&= J_z(\epsilon)
\end{aligned}
$$

and hence

$$\rho\mathcal{A}\otimes_1\mathbf{x}\otimes_2\mathbf{z}\otimes_3\mathbf{u}+(1-\rho)\mathbf{x}^\top\mathbf{z}\le\rho(J_z(\epsilon)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon.$$

An upper bound on $\sum_{\ell\neq k}\sigma_\ell\mathbf{x}^\top\mathbf{z}_\ell\mathbf{z}^\top\mathbf{z}_\ell\mathbf{u}^\top\mathbf{u}_\ell$ for any $\|\mathbf{x}\|=1$ can be obtained by the same argument. Hence, $\|\sum_{\ell\neq k}\sigma_\ell\mathbf{z}_\ell\mathbf{z}^\top\mathbf{z}_\ell\mathbf{u}^\top\mathbf{u}_\ell\|\le J_z(\epsilon)$. By triangle inequality,

$$\begin{aligned}
&\|\rho\mathcal{A}\otimes_1\mathbf{x}\otimes_2\mathbf{z}\otimes_3\mathbf{u}+(1-\rho)\mathbf{z}\|\\
&\ge\rho(\|\sigma_k\mathbf{z}_k\mathbf{z}^\top\mathbf{z}_k\mathbf{u}^\top\mathbf{u}_k\|-\|\sum_{\ell\neq k}\sigma_\ell\mathbf{z}_\ell\mathbf{z}^\top\mathbf{z}_\ell\mathbf{u}^\top\mathbf{u}_\ell\|-\|\mathcal{E}\|_2)-(1-\rho)\\
&\ge\rho(\sigma_{\min}(1-\epsilon^2)-J_z(\epsilon)-\|\mathcal{E}\|_2)-(1-\rho),
\end{aligned}$$

Therefore, an upper bound of $\sqrt{1-(\widehat{\mathbf{z}}^\top\mathbf{z}_k)^2}$ follows,

$$\begin{aligned}
\sqrt{1-(\widehat{\mathbf{z}}^\top\mathbf{z}_k)^2}&=\min_{\|\mathbf{x}\|_2=1,\mathbf{x}\perp\mathbf{z}_k}\mathbf{x}^\top\widehat{\mathbf{z}}\\
&=\min_{\|\mathbf{x}\|=1,\mathbf{x}\perp\mathbf{z}_k}\frac{\rho\mathcal{A}\otimes_1\mathbf{x}\otimes_2\mathbf{z}\otimes_3\mathbf{u}+(1-\rho)\mathbf{x}^\top\mathbf{z}}{\|\rho\mathcal{A}\otimes_2\mathbf{z}\otimes_3\mathbf{u}+(1-\rho)\mathbf{z}\|}\\
&\le\frac{\rho(J_z(\epsilon)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon}{\rho(\sigma_{\min}(1-\epsilon^2)-J_z(\epsilon)-\|\mathcal{E}\|_2)-(1-\rho)}.
\end{aligned}\tag{B.2.1}$$

□

Lemma 12 (One update of $\mathbf{u}$)**.** *Assume that* $\mathbf{z}^\top\mathbf{z}_k>\sqrt{1-\epsilon^2}$, $\mathbf{u}^\top\mathbf{u}_k>\sqrt{1-\epsilon^2}$. *Then, for* $\widehat{\mathbf{u}}=\phi(\rho(\mathbf{I}+\gamma\mathbf{\Omega})^{-1}\mathcal{A}\otimes_1\mathbf{z}\otimes_2\mathbf{z}+(1-\rho)\mathbf{u})$, *we have*

$$\sqrt{1-(\widehat{\mathbf{u}}^\top\mathbf{u}_k)^2}\le C(\gamma)\frac{\rho(J_u(\epsilon)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon}{\rho(\sigma_k(1-\epsilon^2)-J_u(\epsilon)-\|\mathcal{E}\|_2)-(1-\rho)}$$

if $\sigma_{\min}(1-\epsilon^2)>J_u(\epsilon)+\|\mathcal{E}\|$, *where*

$$J_u(\epsilon):=\sigma_{\max}(\eta_z^2\sqrt{r(1+\eta_u r)}+2\epsilon(\eta_z+2r\eta_z^2)+\epsilon^2 r)$$

and $C(\gamma)=1+\frac{4\gamma}{\min_i(t_i-t_{i-1})^2}$.

Proof of Lemma 12. Similarly, let

$$I_u(\epsilon)=\mathbb{E}[\mathcal{A}]\otimes_1\mathbf{z}\otimes_2\mathbf{z}\otimes_3((\mathbf{I}+\gamma\mathbf{\Omega})^{-1}\mathbf{x}),$$

where $\mathbf{x}$ satisfies $\mathbf{x}\perp\mathbf{u}_k$ and $\|\mathbf{x}\|=1$. Note that $\|\mathbf{I}+\gamma\mathbf{\Omega}\|_2\ge1$ because of the semi-

positive-definiteness of $\mathbf{\Omega}$ and hence

$$\|\mathcal{E} \otimes_3 ((\mathbf{I} + \gamma\mathbf{\Omega})^{-1}\mathbf{x})\|_2 \leq \|\mathcal{E}\|_2.$$

Thus, a similar decomposition of $I_u(\epsilon)$ gives

$$I_u(\epsilon) \leq \sigma_{\max}(\eta_z^2\sqrt{r(1+\eta_u r)} + 2\epsilon(1+2r\eta_z)\eta_z + \epsilon^2 r) := J_u(\epsilon).$$

Since $\|(\mathbf{I} + \gamma\mathbf{\Omega})^{-1}\| \leq 1$ and therefore $\mathbb{P}(\|(\mathbf{I} + \gamma\mathbf{\Omega})^{-1}\mathcal{E}\| \geq \delta) \leq \mathbb{P}(\|\mathcal{E}\| \geq \delta)$, one can derive

$$\begin{aligned}
\sqrt{1-(\widehat{\mathbf{u}}^\top\mathbf{u}_k)^2} &= \min_{\|\mathbf{x}\|=1, \mathbf{x}\perp\mathbf{u}_k} \mathbf{x}^\top\widehat{\mathbf{u}} \\
&= \min_{\|\mathbf{x}\|=1, \mathbf{x}\perp\mathbf{u}_k} \frac{\mathbf{x}^\top(\mathbf{I}+\gamma\mathbf{\Omega})^{-1}(\rho\mathcal{A}\otimes_1\mathbf{y}\otimes_2\mathbf{z}+(1-\rho)\mathbf{u})}{\|(\mathbf{I}+\gamma\mathbf{\Omega})^{-1}(\rho\mathcal{A}\otimes_1\mathbf{y}\otimes_2\mathbf{z}+(1-\rho)\mathbf{u})\|} \\
&\leq \frac{\rho(J_u(\epsilon)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon}{\sigma_{\min}((\mathbf{I}+\gamma\mathbf{\Omega})^{-1})(\rho(\sigma_k(1-\epsilon^2)-J_u(\epsilon)-\|\mathcal{E}\|_2)-(1-\rho))} \\
&\leq C(\gamma)\frac{\rho(J_u(\epsilon)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon}{\rho(\sigma_k(1-\epsilon^2)-J_u(\epsilon)-\|\mathcal{E}\|_2)-(1-\rho)}.
\end{aligned}$$

The last inequality follows from the fact that the largest eigenvalue of $\mathbf{\Omega}$ is bounded by $\frac{4}{\min_i(t_i-t_{i-1})^2}$. □

B.2.3 Proof of the main theorem

Proof of Theorem 7. We will prove the theorem by induction. Assume that

$$\sqrt{1-(\widehat{\mathbf{z}}^{(m)\top}\mathbf{z}_k)^2} < \epsilon_z \text{ and } \sqrt{1-(\widehat{\mathbf{u}}^{(m)\top}\mathbf{u}_k)^2} < \epsilon_u.$$

Given $\nu \in (0,1)$, by Lemmas 11 and 12,

$$\sqrt{1-(\widehat{\mathbf{z}}^{(m+1)\top}\mathbf{z}_k)^2} < \nu\epsilon_z \text{ and } \sqrt{1-(\widehat{\mathbf{u}}^{(m+1)\top}\mathbf{u}_k)^2} < \nu\epsilon_u$$

if

$$\frac{\rho(J_z(\epsilon_z)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon_z}{\rho(\sigma_{\min}(1-\epsilon_z^2)-J_z(\epsilon_z)-\|\mathcal{E}\|_2)-(1-\rho)} < \nu\epsilon_z \tag{B.2.2}$$

and

$$\frac{\rho(J_u(\epsilon_u) + \|\mathcal{E}\|_2) + (1-\rho)\epsilon_u}{\rho(\sigma_k(1-\epsilon_u^2) - J_u(\epsilon_u) - \|\mathcal{E}\|_2) - (1-\rho)} < \frac{\nu}{C(\gamma)}\epsilon_u. \tag{B.2.3}$$

When

$$\|\mathcal{E}\|_2 < \min\{\sigma_{\min}(1-\epsilon_z^2) - J_u(\epsilon_z), \sigma_{\min}(1-\epsilon_u^2) - J_u(\epsilon_u)\} - L,$$

where $L = \frac{1-\rho}{\rho}$ is the same L as in (3.3.3), rearranging the inequalities (B.2.2) and (B.2.3), we have

$$\frac{1+\nu\epsilon_z}{1+\epsilon_z}(J_z(\epsilon_z) + \|\mathcal{E}\|_2) + \frac{1-\rho}{\rho}\frac{\epsilon}{1+\epsilon}(1+\nu) < \nu\sigma_{\min}(1-\epsilon_z)\epsilon_z$$

and

$$\frac{1+\frac{\nu}{C(\gamma)}\epsilon_u}{1+\epsilon_u}(J_u(\epsilon_u) + \|\mathcal{E}\|_2) + L\frac{\epsilon}{1+\epsilon}\left(1 + \frac{\nu}{C(\gamma)}\right) < \frac{\nu}{C(\gamma)}\sigma_{\min}(1-\epsilon_u)\epsilon_u.$$

The above inequalities are implied by

$$\|\mathcal{E}\|_2 < \nu\sigma_{\min}(1-\epsilon_z)\epsilon_z - J_z(\epsilon_z) - L(1+\nu)\epsilon_z$$

and

$$\|\mathcal{E}\|_2 < \frac{\nu}{C(\gamma)}\sigma_{\min}(1-\epsilon_u)\epsilon_u - J_u(\epsilon_u) - L\left(1 + \frac{\nu}{C(\gamma)}\right)\epsilon_u.$$

Let

$$h_z(\epsilon_z) := \sigma_{\max}^{-1}(J_z(\epsilon_z) + \|\mathcal{E}\|_2 - \nu\sigma_{\min}(1-\epsilon_z)\epsilon_z + L(1+\nu)\epsilon_z) \tag{B.2.4}$$

$$= \eta_z\eta_u\sqrt{r(1+\eta_z r)} + \sigma_{\max}^{-1}\|\mathcal{E}\|_2 - (\nu\omega - (\eta_z + \eta_u + 2r\eta_z\eta_u) - L(1+\nu))\epsilon_z \tag{B.2.5}$$

$$+ (\nu\omega + r)\epsilon_z^2 \tag{B.2.6}$$

and

$$h_u(\epsilon_u) := \sigma_{\max}^{-1}(J_u(\epsilon_u) + \|\mathcal{E}\|_2 - \frac{\nu}{C(\gamma)}\sigma_{\min}(1-\epsilon_u)\epsilon_u) \tag{B.2.7}$$

$$= \eta_z^2\sqrt{r(1+\eta_u r)} + \sigma_{\max}^{-1}\|\mathcal{E}\|_2 \tag{B.2.8}$$

$$- \left(\frac{\nu\omega}{C(\gamma)} - 2\eta_z(1+2r\eta_z) - L\left(1 + \frac{\nu}{C(\gamma)}\right)\right)\epsilon_u + \left(\frac{\nu\omega}{C(\gamma)} + r\right)\epsilon_u^2. \tag{B.2.9}$$

The inequalities (B.2.2) and (B.2.3) hold if $\epsilon_z, \epsilon_u \in (0,1)$ satisfy $h_z(\epsilon_z) < 0$ and $h_u(\epsilon_u) <$

0. Since both h_z and h_u are quadratic functions, each of $h_z(\epsilon_z) = 0$ and $h_u(\epsilon_u) = 0$ has two distinct real roots, denoted by $s_z^- < s_z^+$, respectively, such that ϵ_z and ϵ_u exist only if the quantity $\|\mathcal{E}\|_2$ is not too large. Specifically, (B.2.2) and (B.2.3) hold for some $\epsilon_z, \epsilon_u \in (0, 1)$ if

$$\frac{(\nu\omega - (\eta_z + \eta_u + 2r\eta_z\eta_u))^2 - L(1+\nu)}{4(\nu\omega + r)} - \eta_z\eta_u\sqrt{r(1+\eta_z r)} \geq \sigma_{\max}^{-1}\|\mathcal{E}\|_2$$

and

$$\frac{(\nu\omega - 2\eta_z(1+2r\eta_z)C(\gamma) - L(C(\gamma)+\nu))^2}{4(\nu\omega + rC(\gamma))} - \eta_z^2\sqrt{r(1+\eta_u z)} \geq \sigma_{\max}^{-1}\|\mathcal{E}\|_2,$$

respectively.

By Lemma 10, with probability

$$\mathbb{P}(\sigma_{\max}^{-1}\|\mathcal{E}\|_2 \geq \delta) \leq \exp\bigg(-\frac{\sigma_{\max}\delta^2}{8+4\delta} + (2n+n_T)\log 15\bigg),$$

where

$$\delta = \min\bigg\{ \frac{(\nu\omega - (\eta_z + \eta_u + 2r\eta_z\eta_u) - L(1+\nu))^2}{4(\nu\omega + r)} - \eta_z\eta_u\sqrt{r(1+\eta_z r)}, \tag{B.2.10}$$

$$\frac{(\nu\omega - 2\eta_z(1+2r\eta_z)C(\gamma) - L(C(\gamma)+\nu))^2}{4(\nu\omega + rC(\gamma))} - \eta_z^2\sqrt{r(1+\eta_u z)}\bigg\}, \tag{B.2.11}$$

there exists $s_z^- < s_z^+$ and s_u^-, s_u^+ such that, for $\epsilon_z \in (s_z^-, s_z^+)$ and $\epsilon_u \in (s_u^-, s_u^+)$, the inequalities (B.2.2) and (B.2.3) hold. By induction, the contraction holds throughout all iterations.

Remarkably, $s_z^+ > \frac{\nu\omega - (\eta_z+\eta_u+2r\eta_z\eta_u) - L(1+\nu)}{2(\omega+r)}$ and $s_u^+ > \frac{\nu\omega - 2\eta_z(1+2r\eta_z)C(\gamma) - L(C(\gamma)+\nu)}{2(\omega + C(\gamma)r)}$, which are the minimum of the midpoints of the intervals (s_z^-, s_z^+) and (s_u^-, s_u^+). That is, s_z^+ and s_u^+ are bounded away from 0 if η_z and η_u are sufficiently small.

Finally, we analyze the behavior of the power iterations in the case $\sqrt{1 - (\widehat{\mathbf{z}}^{(m)\top}\mathbf{z}_k)^2} < s_z^-$ and $\sqrt{1 - (\widehat{\mathbf{u}}^{(m)\top}\mathbf{u}_k)^2} < s_u^-$. Note that both J_z and J_u are increasing functions on $(0, 1)$ and hence the left-hand sides of (B.2.2) and (B.2.3) are strictly increasing functions of ϵ_z and ϵ_u, respectively, for $\epsilon \in (0, 1)$. Therefore,

$$\begin{aligned}\sqrt{1 - (\widehat{\mathbf{z}}^{(m+1)\top}\mathbf{z}_k)^2} &< \frac{\rho(J_z(\epsilon_z) + \|\mathcal{E}\|_2) + (1-\rho)\epsilon_z}{\rho(\sigma_{\min}(1-\epsilon_z^2) - J_z(\epsilon_z) - \|\mathcal{E}\|_2) - (1-\rho)} \\ &< \frac{\rho(J_z(s_z^-) + \|\mathcal{E}\|_2) + (1-\rho)s_z^-}{\rho(\sigma_{\min}(1-(s_z^-)^2) - J_z(s_z^-) - \|\mathcal{E}\|_2) - (1-\rho)}\end{aligned}$$

and

$$\sqrt{1-(\widehat{\mathbf{u}}^{(m+1)\top}\mathbf{u}_k)^2} < C(\gamma)\frac{\rho(J_u(\epsilon_u)+\|\mathcal{E}\|_2)+(1-\rho)\epsilon_u}{\rho(\sigma_k(1-\epsilon_u^2)-J_u(\epsilon_u)-\|\mathcal{E}\|_2)-(1-\rho)} < C(\gamma)\frac{\rho(J_u(s_u^-)+\|\mathcal{E}\|_2)+(1-\rho)s_u^-}{\rho(\sigma_k(1-(s_u^-)^2)-J_u(s_u^-)-\|\mathcal{E}\|_2)-(1-\rho)}.$$

Since $\epsilon_z \in (s_z^-, s_z^+)$ and $\epsilon_u \in (s_u^-, s_u^+)$ imply (B.2.2) and (B.2.3), respectively, by the continuity of h_z and h_u, $\epsilon_z \in [s_z^-, s_z^+]$ and $\epsilon_z \in [s_u^-, s_u^+]$ imply

$$\frac{\rho(J_z(s_z^-)+\|\mathcal{E}\|_2)+(1-\rho)s_z^-}{\rho(\sigma_{\min}(1-(s_z^-)^2)-J_z(s_z^-)-\|\mathcal{E}\|_2)-(1-\rho)} \leq \nu s_z^-$$

and

$$C(\gamma)\frac{\rho(J_u(s_u^-)+\|\mathcal{E}\|_2)+(1-\rho)s_u^-}{\rho(\sigma_k(1-(s_u^-)^2)-J_u(s_u^-)-\|\mathcal{E}\|_2)-(1-\rho)} \leq \nu s_u^-.$$

This ensures that

$$\sqrt{1-(\widehat{\mathbf{z}}^{(m+1)\top}\mathbf{z}_k)^2} < s_z^- \text{ and } \sqrt{1-(\widehat{\mathbf{u}}^{(m+1)\top}\mathbf{u}_k)^2} < s_u^-.$$

□

APPENDIX C

Appendix for "Subspace estimation for link prediction in ego-networks"

C.1 Proof of Theorem 8

We will begin the proof with decomposing $\|\mathbf{P} - \widehat{\mathbf{P}}\|_2$. The following lemma enables us to bound some terms of the decomposition as if we analyze the error bound in the scenario of sampling with replacement.

Lemma 13. *Let $\{\mathbf{y}_i\}_{i=1}^n$'s be a random sample without replacement from the column vectors of $\mathbf{Z} = [\mathbf{z}_1 \cdots \mathbf{z}_N]$, where $\mathbf{z}_i \in \mathbb{R}^N$'s satisfy $\|\mathbf{z}_i\| \leq 1$ for all $i = 1, \ldots, N$. Then,*

$$\mathbb{E}\left[\left\| n^{-1} \sum_{i=1}^{n} \mathbf{y}_i \mathbf{y}_i^\top - N^{-1} \mathbf{Z}\mathbf{Z}^\top \right\|_2\right] \leq C N^{-1} \rho_N^{-1} \sqrt{n \log n},$$

where C is a universal constant.

Proof. Let $\mathcal{D} = \{\mathbf{D}_1, \ldots, \mathbf{D}_N\}$, where $\mathbf{D}_k = [d_{k,ij}]_{N \times N}$ with $s_{k,ij} = 1$ if $i = j = k$ otherwise $d_{k,ij} = 0$. We define a function $g : \mathbf{R}^{N \times N} \mapsto \mathbb{R}$ as

$$g(\mathbf{D}) = \|\mathbf{Z}(\mathbf{D} - \rho_N \mathbf{I}_N)\mathbf{Z}^\top\|_2.$$

Therefore, it suffices to derive a bound for the case of sampling with replacement. Let $\{\mathbf{T}_k\}_{k=1}^n$ and $\{\widetilde{\mathbf{T}}_k\}_{k=1}^n$ denote random samples without and with replacement from $\mathcal{D}$, respectively. Since g is a convex function, following the proof as in Theorem 4 in Hoeffding (1963), one can show the result in the theorem applies to matrix-valued samples and obtain

$$\mathbb{E}\left[g\left(\sum_{i=1}^{n} \mathbf{T}_i\right)\right] \leq \mathbb{E}\left[g\left(\sum_{i=1}^{n} \widetilde{\mathbf{T}}_i\right)\right].$$

Since $\|\mathbb{E}[\widetilde{\mathbf{y}}_i\widetilde{\mathbf{y}}_i^\top]\|_2 \le \mathbb{E}[\|\widetilde{\mathbf{y}}_i\widetilde{\mathbf{y}}_i^\top\|_2] \le 1$ and $\mathbb{E}[\widetilde{\mathbf{y}}_i\widetilde{\mathbf{y}}_i^\top] = N^{-1}\mathbf{Z}\mathbf{Z}^\top$, we can apply Theorem 3.1 in Rudelson and Vershynin (2007) to achieve

$$\begin{aligned} n^{-1}\mathbb{E}\Big[g\Big(\sum_{i=1}^n \widetilde{\mathbf{T}}_i\Big)\Big] &= n^{-1}\mathbb{E}\Big[\Big\|\mathbf{Z}\Big(\sum_{i=1}^n \widetilde{\mathbf{T}}_i - \rho_N\mathbf{I}_N\Big)\mathbf{Z}^\top\Big\|_2\Big] \\ &= \mathbb{E}\Big[\Big\|n^{-1}\sum_{i=1}^n \mathbf{Z}\widetilde{\mathbf{T}}_i\mathbf{Z}^\top - N^{-1}\mathbf{Z}\mathbf{Z}^\top\Big\|_2\Big] \\ &= \mathbb{E}\Big[\Big\|n^{-1}\sum_{i=1}^n \widetilde{\mathbf{y}}_i\widetilde{\mathbf{y}}_i^\top - \mathbb{E}[\widetilde{\mathbf{y}}_i\widetilde{\mathbf{y}}_i^\top]\Big\|_2\Big] \\ &\le CN^{-1}\rho_N^{-1}\sqrt{n\log n}, \end{aligned}$$

where C is a universal constant. □

We are now ready to bound the tail probability of $\|\mathbf{P} - \widehat{\mathbf{P}}\|_2$.

Proof of Theorem 8. Since

$$\begin{aligned} \|\mathbf{P} - \widehat{\mathbf{P}}\|_2 &\le \frac{1}{2}\|\mathbf{P} - \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^+\widetilde{\mathbf{P}}_{in}\|_2 + \frac{1}{2}\|\mathbf{P} - \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^{+\top}\widetilde{\mathbf{P}}_{in}\|_2 \\ &= \|\mathbf{P} - \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^+\widetilde{\mathbf{P}}_{in}\|_2, \end{aligned}$$

it suffices to study the rate of growth of $\|\mathbf{P} - \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^+\widetilde{\mathbf{P}}_{in}\|_2$. Let $\mathbf{S} = [s_{ij}]_{N\times N}$ be a diagonal matrix with $s_{ij} = 1(\text{row } i \text{ or column } j \text{ is selected})$ and $\widetilde{\mathbf{S}}$ be a submatrix of $\mathbf{S}$ obtained by removing zero rows from $\mathbf{S}$. Then, $\mathbf{A}_{in} = \widetilde{\mathbf{S}}\mathbf{A}$, $\widetilde{\mathbf{P}}_{in} = \widetilde{\mathbf{S}}\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top$, and

$$\begin{aligned} \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^+\widetilde{\mathbf{P}}_{in} &= \widetilde{\mathbf{P}}_{in}^\top(\widetilde{\mathbf{P}}_{in}\widetilde{\mathbf{S}}^\top)^+\widetilde{\mathbf{P}}_{in} \\ &= \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{S}}\widetilde{\mathbf{P}}_{in}^+\widetilde{\mathbf{P}}_{in} \\ &= \mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\widetilde{\mathbf{S}}^\top\widetilde{\mathbf{S}}\mathbf{V}_r\mathbf{V}_r^\top \\ &= \mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{S}\mathbf{V}_r\mathbf{V}_r^\top. \end{aligned}$$

Therefore, we can bound $\|\mathbf{P} - \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^+\widetilde{\mathbf{P}}_{in}\|_2$ by applying the triangle inequality as follows:

$$\begin{aligned} &\|\mathbf{P} - \widetilde{\mathbf{P}}_{in}^\top\widetilde{\mathbf{P}}_{11}^+\widetilde{\mathbf{P}}_{in}\|_2 \\ &\le \|\mathbf{P} - \mathbf{A}\|_2 + \|\mathbf{A} - \mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top\|_2 \\ &\quad + \|\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top - \mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{S}\mathbf{V}_r\mathbf{V}_r^\top\|_2 \end{aligned} \tag{C.1.1}$$

Throughout the proof, we always have $\|\mathbf{A} - \mathbf{P}\|_2 = O(\sqrt{d})$ by the dense graph assumption.

A bound of the second term in (C.1.1) follows from

$$\begin{aligned}\|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top\|_2 &\le \|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\|_2+\|\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top\|_2\\ &\le \|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\|_2+\|\mathbf{V}_r\mathbf{V}_r^\top\|_2\|\mathbf{A}-\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top\|_2\\ &\le 2\|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\|_2.\end{aligned}$$

The derivation of a bound of $\|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\|_2$ is similar to the proof of Theorem 1.1 in Rudelson and Vershynin (2007) except that we consider uniformly sampling without replacement and random $\mathbf{A}$. First, by applying Theorem 3 in Drineas et al. (2006a), we obtain

$$\|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\|_2 \le \sigma_{r+1}(\mathbf{A})+\sqrt{2}\|\mathbf{A}^2-\rho_N^{-1}\mathbf{A}_{in}^\top\mathbf{A}_{in}\|_2^{\frac{1}{2}}. \tag{C.1.2}$$

By Weyl's inquality, we can bound $\sigma_{r+1}(\mathbf{A})$ by

$$\sigma_{r+1}(\mathbf{A}) \le \sigma_{r+1}(\mathbf{P})+\|\mathbf{P}-\mathbf{A}\|_2.$$

To derive a bound of the second term in (C.1.2), let $[\mathbf{y}_1\cdots\mathbf{y}_n]=N^{-\frac{1}{2}}\mathbf{A}_{in}^\top$ and $\mathbf{Z}=N^{-\frac{1}{2}}\mathbf{A}$. By Markov's inequality, we have a moment bound as follows

$$\begin{aligned}&\mathbb{P}(\|\rho_N^{-1}\mathbf{A}_{in}^\top\mathbf{A}_{in}-\mathbf{A}^2\|_2>t)\\ &=\mathbb{E}[\mathbb{P}(\|\rho_N^{-1}\mathbf{A}_{in}^\top\mathbf{A}_{in}-\mathbf{A}^2\|_2>t\mid\mathbf{A})]\\ &\le t^{-1}\mathbb{E}[\mathbb{E}[\|\rho_N^{-1}\mathbf{A}_{in}^\top\mathbf{A}_{in}-\mathbf{A}^2\|_2\mid\mathbf{A}]]\\ &=t^{-1}N^2\mathbb{E}\left[\mathbb{E}\left[\left\|n^{-1}\sum_{i=1}^n\mathbf{y}_i\mathbf{y}_i^\top-N^{-1}\mathbf{Z}\mathbf{Z}^\top\right\|_2>t\;\middle|\;\mathbf{Z}\right]\right].\end{aligned}$$

Hence, by Lemma 13,

$$\mathbb{P}(\|\rho_N^{-1}\mathbf{A}_{in}^\top\mathbf{A}_{in}-\mathbf{A}^2\|_2>t)\le t^{-1}CN\rho_N^{-1}\sqrt{n\log n},$$

where C is a universal constant. Thus, $\|\mathbf{A}^2-\rho_N^{-1}\mathbf{A}_{in}^\top\mathbf{A}_{in}\|_2=O_p(N\rho_N^{-1}\sqrt{n\log n})$, and therefore

$$\begin{aligned}\|\mathbf{A}-\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top\|_2 &\le 2\|\mathbf{P}-\mathbf{A}\|_2+2\sigma_{r+1}(\mathbf{P})+O_p\left(\left(\frac{N}{\rho_N}\right)^{\frac{1}{2}}(n\log n)^{\frac{1}{4}}\right)\\ &\le O(\sqrt{d})+2\sigma_{r+1}(\mathbf{P})+O_p\left(\left(\frac{N}{\rho_N}\right)^{\frac{1}{2}}(n\log n)^{\frac{1}{4}}\right).\end{aligned} \tag{C.1.3}$$

For the last term in (C.1.1), we consider

$$\|\mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{V}_r\mathbf{V}_r^\top - \mathbf{V}_r\mathbf{V}_r^\top\mathbf{A}\mathbf{S}\mathbf{V}_r\mathbf{V}_r^\top\|_2 \leq \|\mathbf{P}(\mathbf{I}-\mathbf{S})\|_2 + \|\mathbf{P}-\mathbf{A}\|_2.$$

By Theorem 1.8 in Rudelson and Vershynin (2007), we have

$$\|\mathbf{P}(\mathbf{I}-\mathbf{S})\|_2 = O(\|\mathbf{P}\|_2(\sqrt{1-\rho_N} + \sqrt{\log(N-n)})). \tag{C.1.4}$$

The desired result follows by combining (C.1.3), (C.1.4), and the fact that $\|\mathbf{A}-\mathbf{P}\|_2 = O(\sqrt{d})$. □

BIBLIOGRAPHY

Abdallah, E., Hamza, A., and Bhattacharya, P. (2007). MPEG video watermarking using tensor singular value decomposition. *Image Analysis and Recognition*, pages 772–783.

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008). Mixed membership stochastic blockmodels. *The Journal of Machine Learning Research*, 9:1981–2014.

Albert, R., DasGupta, B., and Mobasheri, N. (2014). Topological implications of negative curvature for biological and social networks. *Physical Review E*, 89(3):32811.

Almquist, Z. W. (2012). Random errors in egocentric networks. *Social Networks*, 34(4):493–505.

Anandkumar, A., Ge, R., Hsu, D., and Kakade, S. (2013). A tensor spectral approach to learning mixed membership community models. In *Conference on Learning Theory*, pages 867–881.

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

Ben-Hur, A. and Noble, W. S. (2005). Kernel methods for predicting protein-protein interactions. *Bioinformatics*, 21 Suppl 1:i38–46.

Boyd, S. P. and Vandenberghe, L. (2009). *Convex optimization*. Cambridge university press.

Burt, R. S. (1987). A note on missing network data in the general social survey. *Social Networks*, 9(1):63–73.

Candès, E. J. and Plan, Y. (2010). Matrix completion with noise. *Proceedings of the IEEE*.

Candès, E. J. and Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772.

Candès, E. J. and Tao, T. (2010). The power of convex relaxation: Near-optimal matrix completion. *Information Theory, IEEE Transactions on*, 56(5):2053–2080.

Cantador, I., Brusilovsky, P., and Kuflik, T. (2011). Second workshop on information heterogeneity and fusion in recommender systems (HetRec2011). In *RecSys*, pages 387–388.

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

Chapanond, A., Krishnamoorthy, M. S., and Yener, B. (2005). Graph theoretic and spectral analysis of Enron email data. *Computational and Mathematical Organization Theory*, 11(2004):265–281.

Chatterjee, S. (2015). Matrix estimation by universal singular value thresholding. *The Annals of Statistics*, 43(1):177–214.

Choi, D. S. and Wolfe, P. J. (2014). Co-clustering separately exchangeable network data. *The Annals of Statistics*, 42(1):29–63.

Clauset, A., Moore, C., and Newman, M. E. J. (2008). Hierarchical structure and the prediction of missing links in networks. *Nature*, 453(7191):98–101.

Davenport, M. A., Plan, Y., Berg, E. V. D., Wootters, M., van den Berg, E., and Wootters, M. (2014). 1-bit matrix completion. *Information and Inference*, 3(3):189–223.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000). On the best rank-1 and rank-(r_1, r_2,..., r_n) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342.

Diehl, C. P., Namata, G., and Getoor, L. (2007). Relationship identification for social network discovery. In *AAAI*, volume 22, pages 546–552.

Doppa, J. R., Yu, J., Tadepalli, P., and Getoor, L. (2009). Chance-constrained programs for link prediction. In *NIPS Workshop on Analyzing Networks and Learning with Graphs*.

Drineas, P., Kannan, R., and Mahoney, M. (2006a). Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix. *SIAM Journal on Computing*, 36(1):158–183.

Drineas, P., Kannan, R., and Mahoney, M. (2006b). Fast Monte Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition. *SIAM Journal on Computing*, 36(1):184–206.

Drineas, P., Mahoney, M., and Muthukrishnan, S. (2008). Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881.

Dror, G., Koenigstein, N., Koren, Y., and Weimer, M. (2012). The Yahoo! Music Dataset and KDD-Cup'11. In *KDD Cup*, pages 8–18.

Duch, J. and Arenas, A. (2005). Community detection in complex networks using extremal optimization. *Physical Review E*, 72(2):27104.

Durante, D. and Dunson, D. B. (2017). Locally adaptive dynamic networks. *The Annals of Applied Statistics*, 10(4):2203–2232.

Fellows, I. and Handcock, M. S. (2012). Exponential-family random network models. *arXiv preprint arXiv:1208.0121*.

Fosdick, B. K. and Hoff, P. D. (2015). Testing and modeling dependencies between a network and nodal attributes. *Journal of the American Statistical Association*, 110(511):1047–1056.

Freeman, L. C. (1982). Centered graphs and the structure of ego networks. *Mathematical Social Sciences*, 3(3):291–304.

Freeman, L. C., Webster, C. M., and Kirke, D. M. (1998). Exploring social structure using dynamic three-dimensional color images. *Social networks*, 20(2):109–118.

Fu, W., Song, L., and Xing, E. E. P. (2009). Dynamic mixed membership blockmodel for evolving networks. In *Proceedings of the 26th annual international conference on machine learning*, pages 329–336. ACM.

Gandy, S., Recht, B., and Yamada, I. (2011). Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2):025010.

Gao, C., Vaart, A. W. V. D., and Zhou, H. H. (2015). A General Framework for Bayes Structured Linear Models. *arXiv preprint arXiv:1506.02174*.

Hanneke, S., Fu, W., and Xing, E. P. (2010). Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605.

Harshman, R. A. (1970). Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. In *UCLA Working Papers in Phonetics*, volume 16, pages 1–84. University of California at Los Angeles Los Angeles, CA.

Hasan, M. A., Chaoji, V., Salem, S., and Zaki, M. (2006). Link prediction using supervised learning. In *Workshop on Link Analysis, Counter-terrorism and Security (at SIAM Data Mining Conference)*.

Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American statistical association*, 58(301):13–30.

Hoff, P. D. (2005). Bilinear mixed-effects models for dyadic data. *Journal of the American Statistical Association*, 100(469):286–295.

Hoff, P. D. (2007). Modeling homophily and stochastic equivalence in symmetric relational data. *Neural Information Processing Systems*, pages 1–8.

Hoff, P. D. (2009). Multiplicative latent factor models for description and prediction of social networks. *Computational and Mathematical Organization Theory*, 15(4):261–272.

Hoff, P. D., Fosdick, B. K., Volfovsky, A., and Stovel, K. (2013). Likelihoods for fixed rank nomination networks. *Network Science*, pages 253–277.

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

Karatzoglou, A., Amatriain, X., Baltrunas, L., and Oliver, N. (2010). Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 79–86. ACM.

Kashima, H., Kato, T., and Yamanishi, Y. (2009). Link propagation: A fast semi-supervised learning algorithm for link prediction. In *Proceedings of the 2009 SIAM Conference on Data Mining*.

Keila, P. S. and Skillicorn, D. B. (2005). Structure in the Enron email dataset. *Computational and Mathematical Organization Theory*, 11(3):183–199.

Keshavan, R., Montanari, A., and Oh, S. (2010). Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11(Jul):2057–2078.

Kim, M. and Leskovec, J. (2013). Nonparametric multi-group membership model for dynamic networks. In *Advances in Neural Information Processing Systems*, pages 1385–1393.

Klimt, B. and Yang, Y. (2004). The enron corpus: A new dataset for email classification research. In *European Conference on Machine Learning*, pages 217–226. Springer.

Kogovšek, T. and Ferligoj, A. (2005). Effects on reliability and validity of egocentered network measurements. *Social networks*, 27(3):205–229.

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

Kossinets, G. (2006). Effects of missing data in social networks. *Social Networks*, 28(3):247–268.

Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A., and Boguñá, M. (2010). Hyperbolic geometry of complex networks. *Physical Review E*, 82(3):036106.

Krivitsky, P., Handcock, M., A., R., and Hoff, P. (2009). Representing Degree Distributions, Clustering, and Homophily in Social Networks with Latent Cluster Random Effects Models. *Social Networks*, 31:204–213.

Kruskal, J. B. (1977). Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear algebra and its applications*, 18(2):95–138.

Kunegis, J. and Lommatzsch, A. (2009). Learning spectral graph transformations for link prediction. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, number D, pages 1–8, New York, New York, USA. ACM Press.

Latala, R. (2005). Some estimates of norms of random matrices. *Proceedings of the American Mathematical Society*, 133(5):1273–1282.

Latouche, P., Birmelé, E., and Ambroise, C. (2011). Overlapping stochastic block models with application to the french political blogosphere. *The Annals of Applied Statistics*, 5(1):309–336.

Le, C. M., Levina, E., and Vershynin, R. (2015). Concentration and regularization of random graphs. *arXiv preprint arXiv:1506.00669*.

Leskovec, J., Huttenlocher, D. P., and Kleinberg, J. M. (2010). Governance in social media: A case study of the wikipedia promotion process. In *ICWSM*.

Leskovec, J., Kleinberg, J., and Faloutsos, C. (2007). Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2.

Li, X., Du, N., Li, H., Li, K., Gao, J., and Zhang, A. (2014). A deep learning approach to link prediction in dynamic networks. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 289–297. SIAM.

Liben-Nowell, D. and Kleinberg, J. (2007). The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58(7):1019–1031.

Lin, Z., Chen, M., and Ma, Y. (2010). The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*.

Lü, L. and Zhou, T. (2011). Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170.

Ma, Z. and Ma, Z. (2017). Exploration of large networks via fast and universal latent space model fitting. pages 1–55.

Mahoney, M. W. and Drineas, P. (2009). CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 106(3):697–702.

Marsden, P. V. (2002). Egocentric and sociocentric measures of network centrality. *Social networks*, 24(4):407–422.

Matias, C., Rebafka, T., and Villers, F. (2015). Estimation and clustering in a semiparametric Poisson process stochastic block model for longitudinal networks. *arXiv preprint arXiv:1512.07075*.

Mcauley, J. and Leskovec, J. (2012). Learning to discover social circles in ego networks. *Advances in Neural Information Processing Systems*, 25:539–547.

McPherson, M., Smith-Lovin, L., and Cook, J. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27:415–444.

Menon, A. and Elkan, C. (2011). Link prediction via matrix factorization. *Machine Learning and Knowledge Discovery in Databases*, 6912:437–452.

Miller, K. T., Griffiths, T. L., and Jordan, M. I. (2009). Nonparametric latent feature models for link prediction. *Advances in Neural Information Processing Systems*, 22:1276–1284.

Moody, J. (2001). Peer influence groups: identifying dense clusters in large networks. *Social Networks*, 23(4):261–283.

Murata, T. and Moriyasu, S. (2007). Link prediction of social networks based on weighted proximity measures. *IEEE/WIC/ACM International Conference on Web Intelligence (WI'07)*, pages 85–88.

Newman, M. E. J. (2003). Ego-centered networks and the ripple effect. *Social Networks*, 25(1):83–95.

Olhede, S. C. and Wolfe, P. J. (2014). Network histograms and universality of blockmodel approximation. *Proceedings of the National Academy of Sciences*, 111(41):14722–14727.

Pang, Y., Li, X., and Yuan, Y. (2010). Robust tensor analysis with L1-norm. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(2):172–178.

Pensky, M. and Zhang, T. (2017). Spectral clustering in the dynamic stochastic block model. *arXiv preprint arXiv:1705.01204*.

Perry, P. O. and Wolfe, P. J. (2013). Point process modelling for directed interaction networks. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 75(5):821–849.

Psorakis, I., Roberts, S. J., Ebden, M., and Sheldon, B. C. (2011). Overlapping community detection using Bayesian non-negative matrix factorization. *Physical Review E*, 83(6):66114.

Richard, E., Gaïffas, S., and Vayatis, N. (2014). Link prediction in graphs with autoregressive features. *Journal of Machine Learning Research*, 15:565–593.

Rohe, K., Chatterjee, S., and Yu, B. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4):1878–1915.

Rudelson, M. and Vershynin, R. (2007). Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM*, 54(4):21.

Sarkar, P., Chakrabarti, D., and Jordan, M. (2013). Nonparametric link prediction in large scale dynamic networks. *arXiv preprint arXiv: 1206.6394*.

Sarkar, P. and Moore, A. W. (2005). Dynamic Social Network Analysis using Latent Space Models. *ACM SIGKDD Explorations Newsletter*, 7(2):31–40.

Sewell, D. K. and Chen, Y. (2015). Latent space models for dynamic networks. *Journal of the American Statistical Association*, 110(512):1646–1657.

Sun, W. W. W., Lu, J., Liu, H., and Cheng, G. (2016). Provable sparse tensor decomposition. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3):899–916.

Sussman, D. L., Tang, M., Fishkind, D. E., and Priebe, C. E. (2012). A consistent adjacency spectral embedding for stochastic blockmodel graphs. *Journal of the American Statistical Association*, 107(499):1119–1128.

Tang, M., Athreya, A., Sussman, D. L., Lyzinski, V., and Priebe, C. E. (2014). Two-sample hypothesis testing for random Dot product graphs via adjacency spectral embedding. *arXiv preprint arXiv:1403.7249*, page 23.

Tomioka, R. and Suzuki, T. (2013). Convex tensor decomposition via structured Schatten norm regularization. In *Advances in neural information processing systems*, pages 1331–1339.

Tomioka, R., Suzuki, T., Hayashi, K., and Kashima, H. (2011). Statistical performance of convex tensor decomposition. In *Advances in Neural Information Processing Systems*, pages 972–980.

von Luxburg, U., Belkin, M., and Bousquet, O. (2008). Consistency of spectral clustering. *The Annals of Statistics*, 36(2):555–586.

Vu, D. Q., Hunter, D., Smyth, P., and Asuncion, A. U. (2011). Continuous-time regression models for longitudinal networks. *Advances in Neural Information Processing Systems*, pages 2492–2500.

Wang, D. J., Shi, X., McFarland, D. A., and Leskovec, J. (2012). Measurement error in network data: A re-classification. *Social Networks*, 34(4):396–409.

Wasserman, S. and Pattison, P. (1996). Logit models and logistic regressions for social networks. I. An introduction to Markov graphs and p*. *Psychometrika*, 61(3):401–425.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393:440.

Williamson, S. A. (2016). Nonparametric network models for link prediction. *Journal of Machine Learning Research*, 17:1–21.

Xing, E. P., Fu, W., and Song, L. (2010). A state-space mixed membership blockmodel for dynamic network tomography. *The Annals of Applied Statistics*, 4(2):535–566.

Xu, K. (2015). Stochastic block transition models for dynamic networks. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, pages 1079–1087.

Xu, Y. and Yin, W. (2013). A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences*, 6(3):1758–1789.

Yang, J., Han, C., and Airoldi, E. (2014). Nonparametric estimation and testing of exchangeable graph models. In *Artificial Intelligence and Statistics*, volume 33, pages 1060–1067.

Yang, T., Chi, Y., Zhu, S., Gong, Y., and Jin, R. (2011). Detecting communities and their evolutions in dynamic social networks—a Bayesian approach. *Machine Learning*, 82(2):157–189.

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

Zhang, X., Moore, C., and Newman, M. E. J. (2016). Random graph models for dynamic networks. *arXiv preprint arXiv:1607.07570*.

Zhang, Y., Levina, E., and Zhu, J. (2015). Estimating network edge probabilities by neighborhood smoothing. *arXiv preprint arXiv:1509.08588*.

Zhao, Q., Caiafa, C. F., Mandic, D. P., Zhang, L., Ball, T., Schulze-Bonhage, A., and Cichocki, A. (2011). Multilinear Subspace Regression: An Orthogonal Tensor Decomposition Approach. In *Advances in Neural Information Processing Systems*, pages 1269–1277.

Zhao, Y., Wu, Y.-J., Levina, E., and Zhu, J. (2017). Link prediction for partially observed networks. *Journal of Computational and Graphical Statistics*, (just-accepted).

Zhou, T., Lü, L., and Zhang, Y.-C. (2009). Predicting missing links via local information. *The European Physical Journal B*, 71(4):623–630.

Zhu, J. (2012). Max-margin nonparametric latent feature models for link prediction. *arXiv preprint*.