Biometrika (2022), **xx**, x, *pp*. 1–15
Printed in Great Britain

Joint Latent Space Models for Network Data with High-dimensional Node Variables

BY XUEFEI ZHANG, GONGJUN XU, AND JI ZHU
Department of Statistics, University of Michigan
1085 South University Avenue, Ann Arbor, Michigan, 48109 U.S.A
{xfzhang, gongjun, jizhu}@umich.edu

SUMMARY

Network latent space models assume each node is associated with an unobserved latent position in a Euclidean space, and such latent variables determine the probability of two nodes connecting with each other. In many applications, nodes in the network are often observed along with high-dimensional node variables, and these node variables provide important information for understanding the network structure. However, the classical network latent space models have several limitations in incorporating the node variables. In this paper, we propose a joint latent space model where we assume that the latent variables not only explain the network structure, but also are informative for the multivariate node variables. We develop a projected gradient descent algorithm that estimates the latent positions using a criterion incorporating both network structure and node variables. We establish theoretical properties of the estimators and provide insights on how incorporating high-dimensional node variables could improve the estimation accuracy of the latent positions. We demonstrate the improvement in latent variable estimation and the improvements in associated downstream tasks, such as missing value imputation for node variables, by simulation studies and an application to a Facebook data example.

Some key words: Network analysis; latent space models; high-dimensional data

1. INTRODUCTION

Network data that describe the relations or interactions among individuals have been prevalent in many scientific and engineering fields, including but not limited to social media, world wide webs, and neuroscience (Newman, 2010; Kolaczyk & Csárdi, 2014). A network is usually represented by nodes and edges, where each node represents an individual and an edge represents the connectivity between two nodes. In recent years, a collection of statistical models have been proposed to analyze network data appearing in various domains, for example, see Goldenberg et al. (2010) for a review. Many of the existing models are based on the assumption that the formation of network links is driven by nodal latent variables, including stochastic block models (Holland et al., 1983), latent space models (Hoff et al., 2002), random dot product graph models (Young & Scheinerman, 2007; Athreya et al., 2017), etc. It is critical to estimate the node latent variables accurately, because the estimated latent representations of nodes not only provide insights on the structure of the network, but also can be further used as node features for subsequent tasks, such as node clustering, prediction for node response variables, and network link prediction.

In practice, the network link information is often collected along with additional high-dimensional node variables. For example, in an online social network, where nodes represent users and links represent friendship relationships, we also observe users' multiple personal in-

formation such as age, gender, and education institution (Leskovec & Mcauley, 2012); and in a citation network where nodes represent papers and links represent citation relationships, word frequencies over a large number of words for each paper are recorded as well (McCallum et al., 2000). The dimension of node variables in these applications can be large in the sense that it is comparable to the number of nodes. Existing studies have shown that node variables provide complementary information to network links and often play important roles for estimating the latent structure of the network (Zhang et al., 2016; Binkiewicz et al., 2017; Newman & Clauset, 2016). Thus, it is important to model the network and node variables jointly such that the node variable information can be utilized for improved understanding of the node latent variables.

This paper proposes a joint latent space model to model network links and high-dimensional node variables simultaneously using shared latent variables. On one hand, as mentioned above, many commonly used network models assume that the network links are determined through node latent variables. Among these models, *latent space models* are probably the most popular(Hoff et al., 2002) and have been shown to be powerful for capturing many commonly observed features of real-world networks (Ward & Hoff, 2007; Ward et al., 2007, 2011; Friel et al., 2016), such as node degree heterogeneity, homophily, and community structures (Ma et al., 2020). Taking advantage of these nice properties of the network latent space model, we also assume that each node can be represented by a latent vector in a (low) dimensional Euclidean space, and the connecting probability between two nodes depends on the corresponding pair of nodes' positions in the unobserved space. On the other hand, it is also commonly observed that high- or moderate-dimensional variables that are correlated can often be explained in terms of a few unobserved latent variables as well (Bai & Li, 2012; Wang et al., 2017; Hair et al., 2018). Further, for network data with node variables, the latent variables that explain the observed high-dimensional node variables could be correlated with the latent position variables that explain the network links. This motivates us to consider the joint latent space modeling framework, which uses the shared latent variables to model both parts of the observed information, with the goal of utilizing node variables effectively to help estimate the node latent positions.

Various latent variable based network models have been proposed for modeling network data with node variables, such as the latent space model (Hoff et al., 2002) and its variants (Hoff, 2003; Handcock et al., 2007; Hoff, 2005, 2008, 2009; Krivitsky et al., 2009; Sewell & Chen, 2015, 2016; Ma et al., 2020). When node covariates are present, existing latent space models usually incorporate such information by including pairwise node variable similarities to model link probabilities (Hoff et al., 2002; Ma et al., 2020). Such similarity-based approaches have several limitations when node variables are of high dimension. First, majority of existing latent space models adopt Bayesian estimation approaches. The high-dimensional similarity vector would introduce a large number of parameters and additional MCMC sampling, therefore, making the estimation much more computationally challenging. Second, the performance of the similarity-based method would be sensitive to the specific choice of the similarity measure; and it does not model the relationship between the observed node variables and the latent variables. In practice, node variables are often correlated with latent variables and this relationship can be utilized for better understanding of the network structure (Xu et al., 2012; Yang et al., 2013; Kim & Leskovec, 2012). Further, from the theoretical perspective, the existing literature has rarely studied the effects of high-dimensional node variables on estimating network latent representations, which is necessary in modern network data analysis with node variables.

The main contributions of this paper are summarized from the following perspectives. From the modeling perspective, the proposed framework has several advantages in comparison to the existing work. First, we model the relationship between node latent variables and node covariates by a set of shared latent variables, which provides a natural way of borrowing information from

node variables to improve the estimation of the latent variables. Second, the proposed model adopts the framework of generalized linear factor models to model the distribution of node variables and, therefore, can handle multiple types of node variables arising in practice (such as continuous, binary, and count variables, etc.). Further, we develop an efficient projected gradient descent algorithm to estimate the model parameters and latent representations, by treating the latent representations as fixed effects. Such an estimation method is computationally more efficient than the Bayesian approaches in the literature.

Moreover, from the theoretical perspective, we show that the proposed estimators of the (fixed effect) joint latent space model are error rate optimal under mild conditions. We also establish the corresponding non-asymptotic upper and lower error bounds. In addition, we provide new findings on how the information from both the network and node variables would balance with each other to affect the estimation of latent variables. In particular, we provide a theoretical guarantee that when the dimension of signal node variables is large enough, borrowing information from node variables would always achieve improvement in estimating node latent positions, in comparison with the results using network link information only. We also investigate how the sparsity level of the network would affect the necessity of including node variables for joint estimation. Our theoretical findings are further supported by extensive simulation studies.

2. Proposed Method

2.1. *Joint Latent Space Model*

We start by introducing notations. Assume we have n data points connected by a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, 2, \ldots, n\}$ denotes the node set, and $\mathcal{E}$ denotes the edge set. We consider undirected networks and write $(i, i') \in \mathcal{E}$ if there is a link between node i and node i'. The network can be represented by an adjacency matrix $A \in \{0, 1\}^{n \times n}$, where $A_{ii'} = A_{i'i} = 1$ if $(i, i') \in \mathcal{E}$ and $A_{ii'} = A_{i'i} = 0$ otherwise. Further, we assume for each node i, we also observe a vector of covariate variables, denoted by $Y_i \in \mathbb{R}^q$. The matrix of node variables is denoted by $Y = [Y_1, Y_2, \ldots, Y_n]^T \in \mathbb{R}^{n \times q}$. For a general matrix $M \in \mathbb{R}^{m \times n}$, we denote its ith row by $M_{i\cdot}$ and its jth column by $M_{\cdot j}$.

We consider a joint latent space model for modeling network data with high-dimensional node variables. Specifically, we assume each node $i \in \mathcal{V}$ can be represented by a low-dimensional vector $Z_i \in \mathbb{R}^k$ in an unobserved latent space. The latent variables of all the nodes are denoted by $Z = [Z_1, Z_2, \ldots, Z_n] \in \mathbb{R}^{n \times k}$. As commonly assumed in previous network latent space models, two nodes that are close in the latent space are more likely to be connected. Meanwhile, it is also proper to assume that when the two nodes are close in the latent space, they may display similarity regarding the observed traits. For instance, for two individuals who have close latent representations in a social network, they may choose similar jobs or hold similar political perspectives. This naturally leads to the consideration that the latent variables not only model the network connectivity, but also are informative for the node variables. In particular, we make the assumption that the distribution of network links and that of node variables are driven by the shared latent variables (Figure 1).

For the network A, we assume that for each pair of nodes (i, i'), given their latent positions Z_i and $Z_{i'}$, the presence or absence of an edge between them is determined by the corresponding pair of latent variables and is independent of any other edges. Specifically, for $i < i'$, we assume $A_{ii'} = A_{i'i} \overset{ind}{\sim} \text{Bernoulli}(P_{ii'})$, with $P_{ii'} = f(Z_i, Z_{i'})$ for some function f. Multiple choices for the function f are available, see Hoff et al. (2002), Hoff (2003, 2008), and Ma et al. (2020) for examples. In this paper, we consider using the inner-product latent space model (Hoff, 2003; Ma

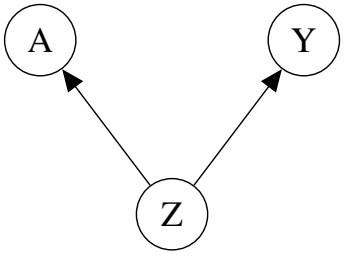


Fig. 1: Graphical representation of the joint latent space model

et al., 2020), i.e.,

$$\text{logit}P_{ii'} = \Theta^A_{ii'} = \alpha_i + \alpha_{i'} + Z_i^T Z_{i'}. \tag{1}$$

The model specification (1) can also be expressed in a matrix form:

$$\text{logit}P = \Theta^A = \alpha 1_n^T + 1_n \alpha^T + ZZ^T,$$

where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)^T$. We choose the inner-product latent space model to model the network part because of its flexibility to capture commonly observed network characteristics. For example, it allows node degree heterogeneity through the parameter α_i's, and in general the larger α_i, the more likely that node i connects with other nodes. It also allows for transitivity, i.e., nodes with common neighbors are more likely to connect since their latent positions are more likely to have larger inner product.

Further, we assume that the same latent variables Z are used to describe the multivariate node variables $Y \in \mathbb{R}^{n\times q}$. Specifically, we assume that given Z, the entries in Y are independent and Z models Y through generalized linear models (Dunn & Smyth, 2018), with

$$g(\mathbb{E}Y) = \Theta^Y = 1_n \gamma^T + ZB, \tag{2}$$

where $\gamma \in \mathbb{R}^q$ and $B \in \mathbb{R}^{k\times q}$ are the "regression" coefficients. For example, when entries in Y are continuous and $g(x) = x$ is the identity mapping, we assume

$$Y_{ij} \overset{ind}{\sim} \mathcal{N}\left((1_n \gamma^T + ZB)_{ij}, \sigma^2\right) \tag{3}$$

for some σ^2. When entries in Y are all binary and $g(x) = \log\{x/(1-x)\}$, we have

$$Y_{ij} \overset{ind}{\sim} \text{Bernoulli}\left(\frac{\exp\left(1_n \gamma^T + ZB\right)_{ij}}{1 + \exp\left(1_n \gamma^T + ZB\right)_{ij}}\right). \tag{4}$$

More generally, when there are more than one type of variables in Y, we could divide Y into R blocks, i.e., $Y = [Y_1|\ldots|Y_R]$, with each sub-block containing the same type of variables and having its own link function. For the following algorithms and theoretical results, we mainly focus on the case that there is only one type of variables in Y. The corresponding results for Y with multiple variable types can be naturally obtained.

It is worth noting that certain modeling approaches for the community detection problem can be considered as having a similar flavor to jointly modeling the distribution of A and Y via shared latent variables, where the discrete latent variable $Z_i \in \{0,1\}^k$ represents the unobserved community membership of node i. For instance, Xu et al. (2012), Yang et al. (2013), and Kim & Leskovec (2012) assumed that for nodes from the same community or cluster, their edge connections and node variables should follow the common distribution specific to that cluster. In other words, the node latent communities determine the distribution of both A and Y, and

therefore information from both A and Y could be used for community detection. Our joint latent space model considers a more general setting where the latent variables could take continuous values in the unobserved latent space.

Note that here we treat Z as fixed effects rather than random. This is due to two reasons. First, our method does not require specific assumptions on the distribution of Z and therefore is more flexible and general; while treating Z as random effects usually needs to make certain parametric assumptions on the distribution of Z. Second, by treating Z as fixed parameters, gradient descent methods can be adopted for parameter estimation and the computation is usually efficient and scalable. On the other hand, when Z is viewed as random effects, the popularly used Bayesian estimation approaches may be computationally expensive.

Remark 1. Although we assume that models (1) and (2) share the same set of latent variables Z, this assumption can be easily relaxed. For instance, consider that we have latent variables $Z_A \in \mathbb{R}^{n\times k}$ modeling A through (1). Meanwhile, there exists another set of latent variables $Z_Y \in \mathbb{R}^{n\times k'}$ that are specifically informative for Y:

$$g(\mathbb{E}Y) = 1_n\gamma^T + Z_Y B. \tag{5}$$

If there exists a matrix $W \in \mathbb{R}^{k\times k'}$ such that $Z_Y \approx Z_A W$, then model (5) could be rewritten as $g(\mathbb{E}Y) \approx 1_n\gamma^T + Z_A W B = 1_n\gamma^T + Z_A B'$ with $B' = WB$, which gives a good approximation to the model in (2). Therefore, even when the two sets of latent variables explaining network and node variables are not exactly the same, but if there is an approximate linear transformation relationship between them, our proposed joint latent space model is still valid, with Z_A being the shared latent variables that explain both parts of the observed information. Additionally, we could further consider a more general case where Z_A and Z_Y are different but have some overlapped latent variables. These two sets of parameters Z_A and Z_Y can still be jointly estimated, but it is beyond the scope of the current paper, and we leave the investigation for future work.

To ensure the joint model to be identifiable, we need to put additional structural constraints on the latent variables Z. First, note that we could add a constant term to both Z_i and Z_j and subtract the corresponding terms from α_i and α_j to keep the distribution of A invariant, so we require that the latent variables are centered, i.e., $JZ = Z$ where $J = I_n - 1_n 1_n^T/n$. This constraint makes Z identifiable up to an orthogonal transformation of its rows. Correspondingly, B is identifiable up to an orthogonal transformation of its columns. Therefore, we further require that the sample covariance of Z, i.e., $Z^T Z/n$, is a diagonal but non-identity matrix. Then the parameters α, Z, B and γ can be uniquely determined.

2.2. *Estimation*

The parameters that need to be estimated are Z, α, B and γ. For the network data A, we consider the loss function as its conditional negative log-likelihood:

$$L_A = -\log P(A|Z,\alpha) = -\sum_{1\le i,i'\le n} \left\{A_{ii'}\Theta^A_{ii'} - f_A(\Theta^A_{ii'})\right\}, \tag{6}$$

where $f_A(x) = \log\{1+\exp(x)\}$. For the node variables Y, we have the negative conditional log-likelihood as

$$L_Y = -\log P(Y|Z,B,\gamma) = -\sum_{1\le i\le n; 1\le j\le q} \left\{Y_{ij}\Theta^Y_{ij} - f_Y(\Theta^Y_{ij})\right\}, \tag{7}$$

where the terms that are irrelevant to Z and γ are omitted. The form of $f_Y(\cdot)$ depends on how the distribution of Y is specified. For example, when Y is continuous as in model (3), $f_Y(x) = x^2/2$; and when Y is binary as in model (4), then $f_Y(\cdot)$ takes the same form as $f_A(\cdot)$.

We define the objective function as

$$L(Z, \alpha, B, \gamma) = L_A + \lambda L_Y, \tag{8}$$

where λ is a weight parameter that controls the information contributed from each part. Our goal is to find the estimators $\widehat{Z}, \widehat{\alpha}, \widehat{B}$ and $\widehat{\gamma}$ that are the solutions to the following optimization problem:

$$\min_{Z\in\mathbb{R}^{n\times k}, JZ=Z, \alpha\in\mathbb{R}^n, B\in\mathbb{R}^{k\times q}, \gamma\in\mathbb{R}^q} L(Z, \alpha, B, \gamma). \tag{9}$$

To optimize (9), we consider using a projected gradient descent algorithm, which is commonly used for solving constrained optimization problems. The term 'projected' means that we project the parameter estimate into the space that satisfies the constraint. Specifically, at each iteration, given all the other parameter estimates, the $\widehat{B}$ and $\widehat{\gamma}$ that minimize the objective function (8) can be solved directly, and therefore, they are updated with those values. As for $\widehat{Z}$ and $\widehat{\alpha}$, they are updated along the direction of their negative gradients. After the updates, we include an additional projection step such that the parameter estimates satisfy the constraint in (9). The algorithm is summarized in Algorithm 1. As pointed out by Ma et al. (2020), where the convergence property of such algorithm has been studied, this algorithm is not guaranteed to converge to any global optimizer when the objective function is not convex (as in this work). In Section 4, we use simulation studies to demonstrate that the algorithm generally converges well and provides good estimation results.

Algorithm 1. Projected Gradient Descent Algorithm for Parameter Estimation

Input: network adjacency matrix $A \in \mathbb{R}^{n\times n}$; node variables $Y \in \mathbb{R}^{n\times q}$; latent space dimension $k \geq 1$; initial estimates: $Z^0, \alpha^0, B^0, \gamma^0$; step sizes η_z, η_α; number of iterations T
Parameters: Z, α, B, γ
For $t = 0, 1, \ldots, T-1$

$$Z^{t+1} = Z^t - \eta_z \nabla_Z L = Z^t + 2\eta_z\{A - f_A'(\Theta^t)\}(Z^t)^T + \lambda\eta_z\{Y - f_Y'(Z^tB^t)\}\left(B^t\right)^T$$
$$\alpha^{t+1} = \alpha^t - \eta_\alpha \nabla_\alpha L = \alpha^t + 2\eta_\alpha\{A - f_A'(\Theta^t)\}1_n$$
$$(\gamma_j^{t+1}, B_{\cdot j}^{t+1}) = \arg\max_{b\in\mathbb{R}^{k+1}} \textstyle\sum_i \{Y_{ij}[1, Z_{i\cdot}^t]b - f_Y([1, Z_{i\cdot}^t]b)\},\ j = 1, \ldots, q$$
$$Z^{t+1} = JZ^{t+1}$$

Output: $\widehat{Z} = Z^T, \widehat{\alpha} = \alpha^T, \widehat{B} = B^T, \widehat{\gamma} = \gamma^T$

Algorithm 1 requires initial inputs of several hyperparameters. For the initialization value $Z^0, \alpha^0, B^0, \gamma^0$ of Z, α, B, γ and choice of step size η_Z and η_α, we adapt the initialization method and step size choice proposed in Ma et al. (2020), which proposed to optimize a regularized version of the negative log-likelihood of the network data as in (6) to obtain initial estimates Z^0 and α^0. Then we regress Y on Z^0 to get an initialization of B^0 and γ^0. For the step size choice, we let $\eta_Z = \eta/\|Z^0\|_F^2$ and $\eta_\alpha = \eta/(2n)$ for a small and fixed constant η. As for the latent space dimension k, it can be selected through cross validation. As we focus on estimating the latent variables that explain the network links, we consider performing cross validation on the adjacency matrix. In particular, we randomly remove entries from the adjacency matrix, then fit the joint latent space model and predict those missing links based on the fitted values. The

process can be repeated for several times and the k that gives the best link prediction performance is chosen from a set of candidate values.

3. THEORETICAL RESULTS

In this section we state our main theoretical results on the estimation of Z under the joint modeling framework. We first show the error bound of the estimators obtained from (9). Then we discuss about how the joint modeling framework could improve the estimation of Z.

To study the theoretical properties, we make the following assumptions on parameters.

Assumption 1. There exists $M_1 > 0$ such that $-M_1 < \Theta^A_{ii'} < M_1$, for $1 \le i, i' \le n$.

Assumption 2. There exists $M_2 > 0$ such that $-M_2 < \Theta^Y_{ij} < M_2$, for $1 \le i \le n, 1 \le j \le q$.

Moreover, we introduce a feasible parameter space as

$$\begin{aligned}\mathcal{F}(n,k,M_1,M_2) = \Big\{&\Theta \in \mathbb{R}^{n\times(n+q)} \mid \Theta = [\Theta^A, \Theta^Y],\\ &\Theta^A = \alpha 1_n^T + 1_n\alpha^T + ZZ^T, \Theta^Y = 1_n\gamma^T + ZB,\\ &\max_{1\le i,i'\le n} |\Theta^A_{ii'}| < M_1, \max_{1\le i\le n, 1\le j\le q} |\Theta^Y_{ij}| < M_2, JZ = Z\Big\}\end{aligned} \tag{10}$$

In the rest of the paper, we use $\mathcal{F}$ as an abbreviation of $\mathcal{F}(n,k,M_1,M_2)$ to denote the parameter space. We denote

$$\Theta^* = \left[\Theta^{*A}, \Theta^{*Y}\right] = \left[\alpha^* 1_n^T + 1_n(\alpha^*)^T + Z^*(Z^*)^T, 1_n(\gamma^*)^T + Z^*B^*\right] \in \mathcal{F}$$

as the ground truth parameter. We denote

$$\widehat{\Theta} = \left[\widehat{\Theta}^A, \widehat{\Theta}^Y\right] = \left[\widehat{\alpha} 1_n^T + 1_n^T\widehat{\alpha} + \widehat{Z}\widehat{Z}^T, 1_n\widehat{\gamma}^T + \widehat{Z}\widehat{B}\right],$$

where $\widehat{Z}$, $\widehat{\alpha}$, $\widehat{B}$ and $\widehat{\gamma}$ are the solutions obtained from the optimization problem: $\min_{\Theta\in\mathcal{F}} L(Z,\alpha,B,\gamma)$. Note that we constrain the true parameters and estimators in the feasible parameter space, mainly for the purpose of theoretical analysis. In practical implementation of Algorithm 1, we do not put additional constraints regarding $\max_{1\le i,i'\le n}|\widehat{\Theta}^A_{ii'}|$ and $\max_{1\le i\le n,1\le j\le q}|\widehat{\Theta}^Y_{ij}|$, and simulation studies indicate that this does not affect the results.

The next theorem provides upper and lower bounds on the estimation error of $\widehat{\Theta}$.

THEOREM 1. *Under Assumptions* 1 *and* 2*, we have*

$$\frac{1}{\{n(n+q)\}^{1/2}}\mathbb{E}\|\widehat{\Theta} - \Theta^*\|_F \le \frac{\kappa\max(\lambda,1)(2k+3)^{1/2}}{\min\left(\min_{|v|<M_1} f''_A(v), \lambda\min_{|v|<M_2} f''_Y(v)\right)} \cdot \frac{1}{n^{1/2}}, \tag{11}$$

where κ is an absolute constant and M_1, M_2 and k are allowed to change with n.

Moreover, denote $\overline{\Theta} \in \mathbb{R}^{n\times(n+q)}$ as an arbitrary estimator. When $q = \mathcal{O}(n)$, there exist $\Theta^0 \in \mathcal{F}$, $\epsilon_1 > 0$ and $n_0, q_0 > 0$ such that for $n > n_0$ and $q > q_0$,

$$P\left(\frac{1}{\{n(n+q)\}^{1/2}}\|\overline{\Theta} - \Theta^0\|_F \ge \frac{\epsilon_1}{n^{1/2}}\right) \ge \frac{1}{2}. \tag{12}$$

The proof is given in Section A of the Supplementary Material. When M_1, M_2 and k are fixed constants, the result in (11) implies that $\|\widehat{\Theta} - \Theta^*\|_F/\{n(n+q)\}^{1/2} = \mathcal{O}_p(1/n^{1/2})$. Together with the lower bound in (12), we can see that the rate of estimation error obtained in Theorem 1 is optimal. Moreover, the results also indicate that using the network itself, i.e., $q = 0$, we have

$\|\widehat{\Theta}^A - \Theta^{*A}\|_F/n = \mathcal{O}_p(e^{M_1}(k/n)^{1/2})$. Therefore, we achieve the same order of estimation error under the joint modeling framework, compared to that obtained with the network information only. In particular, the upper bound of $\|\widehat{\Theta}^A - \Theta^{*A}\|_F/n$ is consistent with the results in Ma et al. (2020), which considered the problem of estimating latent variables using network only, by minimizing L_A as defined in (6).

While Theorem 1 indicates that the error bounds of estimators obtained under the joint modeling framework have the same order as that obtained from the network data (without Y), we are also interested in how the additional node variables Y can help the estimation of latent variables. We evaluate the effect of Y in terms of the one-step update analysis. In particular, assuming we have an estimated $\widetilde{Z}$ through the network, for example, from the algorithm proposed in Ma et al. (2020). Suppose with node variables Y, we update $\widetilde{Z}$ for one more step based on Algorithm 1,

$$\widehat{Z} = \widetilde{Z} + (1-\widetilde{\lambda})\eta_z(A - f_A'(\widetilde{\Theta}^A))\widetilde{Z} + \widetilde{\lambda}\eta_z(Y - f_Y'(\widetilde{\Theta}^Y))(\widetilde{B})^T, \tag{13}$$

where $\widetilde{\lambda} = \lambda/(\lambda+2)$ and $\widetilde{B}$ is a reasonable estimates of B. To investigate the properties of $\widehat{Z}$, we focus on (3) where Y is continuous and make the following additional assumptions.

Assumption 3. The dimension of node variables q satisfies the condition that $q = \mathcal{O}(n)$.

Assumption 4. $cov(Z) = Z^TZ/n = diag(\lambda_1, \ldots, \lambda_k) \neq I_k$ is diagonal and the diagonal elements are of constant order $\Theta(1)$.

Assumption 5. Denote the eigen-decomposition of BB^T/q as $U\Lambda U^T$, where $\Lambda = diag(\sigma_1, \ldots, \sigma_k)$. The eigenvalues are of constant order $\Theta(1)$.

Assumption 3 allows q to grow on a slower or the same order of n. Assumptions 4 and 5 are standard since $Z \in \mathbb{R}^{n\times k}$ and $B \in \mathbb{R}^{k\times q}$. Note that Assumption 5 implies that node covariates could not be pure noises but are signals. The following theorem shows that we can achieve more accurate estimation of Z, as long as the dimension of signal node variables is high enough.

THEOREM 2. *Suppose $\widetilde{Z}$ and $\widetilde{\alpha}$ are estimated from Algorithm 1 in Ma et al. (2020), and we have a fixed $\widetilde{B}$ satisfying $\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$. Then under Assumptions* 1–5*, there exist positive constants C and $\bar{\lambda}$ such that when $q > Cn$, we have $\mathbb{E}\|\widehat{Z} - Z\|_F^2 < \mathbb{E}\|\widetilde{Z} - Z\|_F^2$, for any $\widetilde{\lambda} \in (0, \bar{\lambda})$, where $\widehat{Z}$ is obtained from* (13)*. Moreover, we can show that under proper choice of $\widetilde{\lambda}$ (whose form is specified in* (S14) *in the Supplementary Material), the improvement of estimating Z, i.e., $\mathbb{E}\|\widetilde{Z} - Z\|_F^2 - \mathbb{E}\|\widehat{Z} - Z\|_F^2$, is at least*

$$\frac{\left\{\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z} - Z\|_F^2 - \|\widetilde{Z} - Z\|_F(nq^{-1}\widetilde{\sigma}_1\lambda_1)^{1/2}\|B - \widetilde{B}\|_F\right\}^2}{\rho_2\left[\left\{\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z} - Z\|_F - (nq^{-1}\widetilde{\sigma}_1\lambda_1)^{1/2}\|B - \widetilde{B}\|_F\right\}^2 + nq^{-1}\sigma^2\widetilde{\sigma}_1\right]}, \tag{14}$$

where $\rho_2 = \eta_z q$, and $\widetilde{\sigma}_1$ and $\widetilde{\sigma}_k$ are the maximum and minimum eigenvalues of $\widetilde{B}\widetilde{B}^T/q$ respectively.

From Theorem 2, we see that with additional high-dimensional node variables, we can achieve more accurate estimation of latent variables, in terms of $\mathbb{E}\|\widehat{Z} - Z\|_F^2$. The improvement in (14) depends on the signal and noise contained in the node covariates. Specifically, $\widetilde{\sigma}_k$, the minimum eigenvalue of $\widetilde{B}\widetilde{B}^T/q$, can be approximately viewed as the signal in node covariates, and the quantity in (14) would increase as $\widetilde{\sigma}_k$ increases. Moreover, the quantity monotonically decreases

depending on σ^2, the noise in the node covariates. In practice, a $\widetilde{B}$ that satisfies the requirement $\|\widetilde{B} - B\|_F^2 = O(1)$ can be obtained by regressing Y on $\widetilde{Z}$.

Theorem 2 relies on specific $\widetilde{Z}$ and $\widetilde{\alpha}$. More generally, we consider the scenario where we are given initial estimates of Z, α, B, γ, denoted by $\widetilde{Z}, \widetilde{\alpha}, \widetilde{B}, \widetilde{\gamma}$, respectively, satisfying the conditions that $\|\widetilde{Z} - Z\|_F^2 = \mathcal{O}(1)$, $\|\widetilde{\alpha}1_n^T - \alpha 1_n^T\|_F^2 = \mathcal{O}(n)$, $\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$, and $\|\widetilde{\gamma} - \gamma\|_2^2 = \mathcal{O}(1)$. The following proposition provides implications on under what scenarios the joint modeling framework can help better estimate the latent variables Z.

PROPOSITION 1. *Given $\widetilde{Z}$, $\widetilde{\alpha}$, $\widetilde{B}$, $\widetilde{\gamma}$ that satisfy the above required conditions, we consider updating $\widetilde{Z}$ one step further by* (13) *and obtaining a $\widehat{Z}$. Under Assumptions* 1–5*, there exists an optimal $\widetilde{\lambda}_{opt}$ such that $\mathbb{E}\|\widehat{Z} - Z\|_F^2$ is minimized, and $\widetilde{\lambda}_{opt}$ is given by* (S8)*. When $\widetilde{\lambda}_{opt}$ is strictly positive, the joint modeling framework achieves a mean square error of $\widehat{Z}$ that is better than the results when using information from A or Y only.*

Proofs of Theorem 2 and Proposition 1 are given in Section B of the Supplementary Material. For Proposition 1, the explicit expression of the optimal $\widetilde{\lambda}_{opt}$ is given in (S8). It depends on the signal and noise contained in both networks and node covariates, which provides useful implications on how the information from both parts balance with each other. First, note that a strictly positive $\widetilde{\lambda}_{opt}$ suggests incorporating information from node variables is preferred. Based on the calculation in the Supplementary Material, we can see that an overall sparser network would more likely lead to a positive $\widetilde{\lambda}_{opt}$. Therefore, when the information from the network part is relatively limited, borrowing information from node variables would be preferred and helpful. Second, when controlling all other parameters and increasing q, we would also obtain a larger $\widetilde{\lambda}_{opt}$. This suggests that when node variables are of higher dimension or contain richer information, more weight should be put on the node variables part to improve the estimation of Z. Finally, suppose we do not use the optimal $\widetilde{\lambda}$ but fix it in the one step estimation. Section B of the Supplementary Material also calculates the difference between taking a non-zero $\widetilde{\lambda}$ and $\widetilde{\lambda} = 0$, and the result also suggests that when the dimension of node variables increases or the network gets sparser, the benefit of incorporating node variables becomes more significant in terms of estimating Z. In Section 4 and Section C of the Supplementary Material, we use simulation studies to demonstrate how network information and node variables information affect the estimation of Z. Specifically, we examine the influence of the node variables dimension and the network density in terms of estimating Z as well as the relationship between the dimension of node variables and the optimal weight.

Remark 2. Our work is closely related to the problem of low rank matrix estimation and completion (Chatterjee et al., 2015; Candès & Tao, 2010; Bhaskar & Javanmard, 2015), due to the assumption on the mean parameter $\Theta \in \mathbb{R}^{n\times(n+q)}$ defined in (10). Specifically, if we treat node covariates and the network equally, we could view the problem as recovering the low-rank matrix Θ with an observed matrix $[A, Y] \in \mathbb{R}^{n\times(n+q)}$. In fact, the upper bound in (11) depends on k and n through the rate $(k/n)^{1/2}$, which is consistent with that in low rank matrix estimation literature. However, we could not directly leverage the existing matrix estimation and completion method for estimating Z, since the matrix to be estimated is not an arbitrary low-rank matrix, but has a specific structure due to the share latent factors Z. This is the distinct part of our model assumption in comparison to the general low-rank matrix estimation problem.

4. SIMULATION STUDIES

4.1. *Effect of the Dimension of Node Variables*

To study how information borrowed from Y can affect estimating latent variables, we compare the estimation of Z using the network latent space model and the joint latent space model. For network latent space model, we consider the version without covariates to demonstrate how node variables can be useful in improving the estimation of Z.

We first study the effect of node variable dimension. We set $n = 200$ and $k = 2$ or 4. We vary q from 2 to 100 to study how the dimension of node variables affects the estimation of Z. The model parameters are specified as follows:

- Generate the degree heterogeneity parameters $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, where $\alpha_1, \alpha_2, \ldots, \alpha_n \overset{iid}{\sim} U[-0.25, -0.75]$;
- Generate k latent vector centers $\mu_1, \ldots, \mu_k \in \mathbb{R}^k$ with coordinates i.i.d. from $U[-1, 1]$;
- Generate latent variables $Z \in \mathbb{R}^{n\times k}$: first generate a matrix $Z_0 \in \mathbb{R}^{n\times k}$ such that each entry is i.i.d. $\mathcal{N}(0, 1)$. Then we divide n data points equally into k subsets, and for points in each subset, add $\mu_1, \ldots, \mu_k$ to them respectively. Lastly we transform Z by 1) setting $Z = JZ$, 2) normalizing Z such that $\|ZZ^T\|_F = n$, and 3) rotating $Z = ZR$ for some rotation matrix R such that the covariance of Z is a diagonal matrix;
- Generate the coefficients $B \in \mathbb{R}^{k\times q}$, with each entry i.i.d from $\mathcal{N}(0, 1)$.

After setting the parameters, we generate A based on model (1) and generate Y based on either model (3) or model (4). Under the considered parameter settings, the average density of the networks is about 0.30. Each setting is repeated 30 times. For each replication, we fit both the network inner-product latent space model and the joint latent space model to obtain estimations of Z, denoted by Z_{net} and Z_{joint} respectively. We evaluate the performance of each method using the criterion $\Delta_Z = \|\widehat{Z}\widehat{Z}^T - ZZ^T\|_F^2 / \|ZZ^T\|_F^2$. Figure 2 shows the average results of the 30 replications, and we can see that as the dimension of Y increases, the estimation of Z improves, and the joint latent space model starts to outperform the network latent space model even when q is relatively small, indicating the constant C in Theorem 2 is of small value. Figure 2 also demonstrates that overall the improvement of the joint latent space model over the network latent space model is robust to the choice of λ, though the specific value of λ may affect how much improvement we could obtain by incorporating node variables.

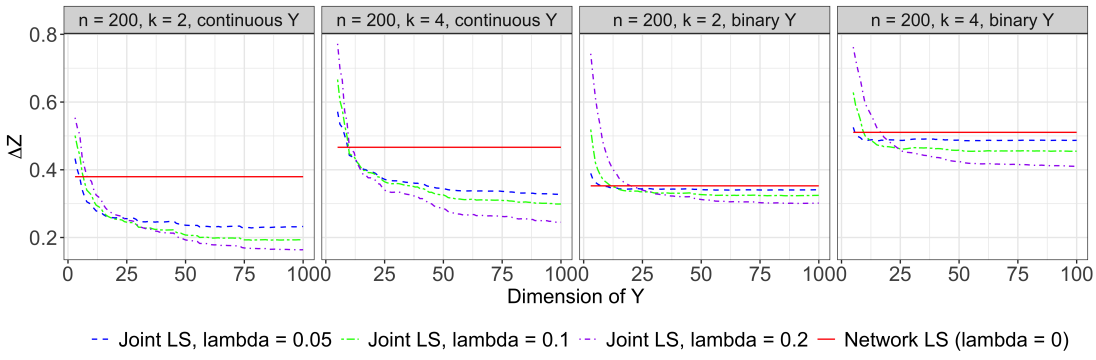


Fig. 2: Estimation of Z versus Dimension of Y

4.2. *Effect of Network Density*

Section 4.1 shows that the more information provided from Y, the more improvement we could obtain in the estimation of Z. As a counterpart, in this subsection we demonstrate the effect of network density on estimating Z when the information from Y is fixed. We fixed $q = 100$ and change the parameter settings in the network. The density of the network is controlled by varying the range of the node degree heterogeneity parameters α, specifically varying from $\alpha_1, \ldots, \alpha_n \sim U[-0.375, -0.125]$ to $\alpha_1, \ldots, \alpha_n \sim U[-2.25, -0.75]$. Z and B are set in the same manner as in Section 4.1. Now under different settings of α, the network density ranges from 0.08 to 0.39. We again repeat the simulation 30 times under each setting.

Figure 3 shows the estimation of Z based on Z_{net} and Z_{joint} under different levels of the network density. As the network gets sparser, the result of Z estimation using A only gets worse, while the performance of Z_{joint} is relatively stable. This especially suggests the benefit of incorporating node variables in estimating Z, when the network is relatively sparse and may not provide enough information.

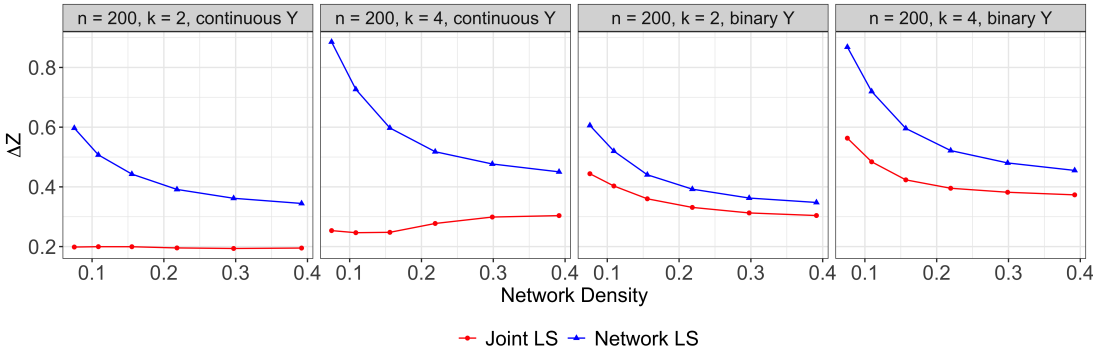


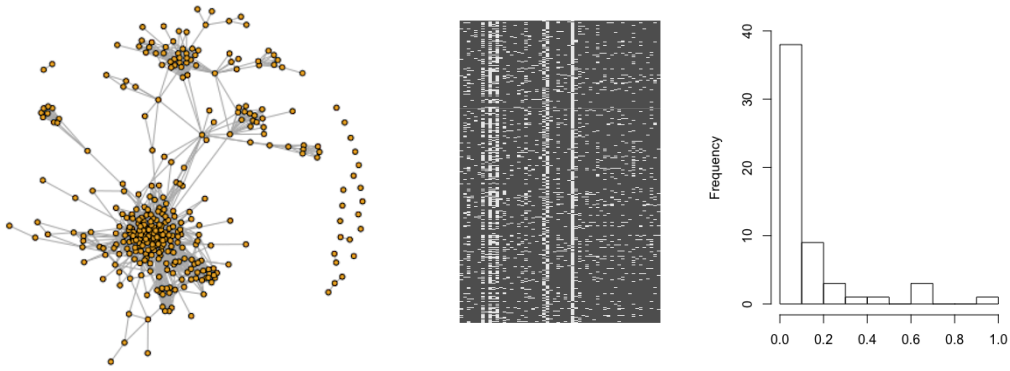
Fig. 3: Estimation of Z versus Network Density

More supplementary simulation studies that support our theorectical findings are provided in Section C of the Supplementary Material.

5. Real Data Example

In this section, we demonstrate the proposed model on a Facebook social circle data for the task of node variable missing value imputation. The dataset consists of 10 different networks, each representing an ego network of a selected user, where the ego network is defined as the network between all the user's friends. See Figure 4 for an example of an ego network. For each ego network, the data also records the users' anonymized variables. For example, the original dataset contains a variable 'political = Democratic Party', and the curated dataset would transform this into 'political = anonymized feature 1'. The variables associated with each node in this data example are all binary. We analyzed 8 out these 10 networks as two of the networks have relatively few nodes and several variable columns associated with them contain only one or a few 1s. For each network, we also remove the variables that have too many or too few 1s in that variable. If a variable of a user is missing, it is of interest to impute the missing value based on the user's own profile as well as his/her social connections.

We randomly sample 5% of the entries in the node variable matrix Y and set them as missing. Then we fit the network latent space model and the joint latent space model respectively to obtain



(a) (b) (c)

Fig. 4: Example of an ego-network: (a) Circle 1 network; (b) node variable matrix; (c) number of variables vs mean of the variable.

	n	q	density	AUC		
				network	joint	MICE
circle 1	347	56	0.042	0.840	0.884	0.707
circle 2	755	66	0.105	0.865	0.873	0.739
circle 3	792	100	0.045	0.856	0.901	0.780
circle 4	1045	153	0.049	0.867	0.871	0.693
circle 5	547	58	0.03	0.852	0.905	0.738
circle 6	227	87	0.124	0.832	0.850	0.586
circle 7	159	55	0.134	0.809	0.836	0.621
circle 8	170	37	0.115	0.812	0.854	0.698

Table 1: Node variable missing value imputation results for Facebook data.

estimated $\widehat{Z}_{net}$ and $\widehat{Z}_{joint}$. After obtaining the estimated $\widehat{Z}$, for the jth column of Y, $1 \leq j \leq q$, we regress those observed Y_j's on $\widehat{Z}_{obs}$ to obtain an estimated $\widehat{B}_j$, then we predict those missing entries in Y_j's by $\widehat{Z}_{miss}\widehat{B}_j$. Moreover, we also apply MICE (Azur et al., 2011), a commonly used method for missing value imputation which utilizes the node variables part only, as a benchmark.

The average AUROC over 30 replications are summarized in Table 1. The results indicate that both network-based methods outperform MICE, and the joint latent space model consistently achieves better performance than the network latent space model across all 8 networks. Note that the AUROC obtained by fitting the network latent space model is already promising, which suggests that the network itself contains substantial information. However, we can still achieve noticeable improvement after incorporating the node variable information. This implies that the latent position variables are associated with the node variables and the joint estimation can help achieve more accurate estimation and imputation results.

6. DISCUSSION

In our simulation and real data examples, we use all n data points with observed information for model fitting, and we obtain estimated model parameters $\widehat{Z} \in \mathbb{R}^{n\times k}$, $\widehat{\alpha} \in \mathbb{R}^n$, $\widehat{B} \in \mathbb{R}^{q\times k}$ and $\widehat{\gamma} \in \mathbb{R}^q$. The prediction for those missing node variables are also performed on these n data points. One possible extension is to consider an inductive setting, where we make prediction on a new node, which is not present during the training stage, with partially observed information. For example, considering the cold-start scenario where only the new node's variables Y_{n+1} are observed but its link information to all the other n nodes are unknown. This is a case commonly seen in social networks, where newly registered users only provide their personal information but have not connect with any other users. To predict links between the $(n+1)$th node and the previous n nodes, we can first estimate $\widehat{Z}_{n+1}$ by regressing Y_{n+1} on the fitted $\widehat{B}$, then compare $\widehat{\alpha}_i + \widehat{Z}_i^T \widehat{Z}_{n+1}$, for $i = 1, \ldots, n$, to rank which nodes have higher probabilities to connect with the new node. Correspondingly, we can also make predictions on the $(n+1)$th node variables, when we only know its link information with other nodes.

ACKNOWLEDGEMENT

The authors are grateful to the editor, Professor Paul Fearnhead, an associate editor, and three referees for their valuable comments and suggestions. This research was partially supported by the U.S. National Science Foundation.

REFERENCES

ATHREYA, A., FISHKIND, D. E., TANG, M., PRIEBE, C. E., PARK, Y., VOGELSTEIN, J. T., LEVIN, K., LYZINSKI, V. & QIN, Y. (2017). Statistical inference on random dot product graphs: a survey. *Journal of Machine Learning Research* **18**, 8393–8484.

AZUR, M. J., STUART, E. A., FRANGAKIS, C. & LEAF, P. J. (2011). Multiple imputation by chained equations: what is it and how does it work? *International journal of methods in psychiatric research* **20**, 40–49.

BAI, J. & LI, K. (2012). Statistical analysis of factor models of high dimension. *The Annals of Statistics* **40**, 436–465.

BHASKAR, S. A. & JAVANMARD, A. (2015). 1-bit matrix completion under exact low-rank constraint. In *2015 49th Annual Conference on Information Sciences and Systems (CISS)*. IEEE.

BINKIEWICZ, N., VOGELSTEIN, J. T. & ROHE, K. (2017). Covariate-assisted spectral clustering. *Biometrika* **104**, 361–377.

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

CHATTERJEE, S. et al. (2015). Matrix estimation by universal singular value thresholding. *The Annals of Statistics* **43**, 177–214.

DUNN, P. K. & SMYTH, G. K. (2018). *Generalized Linear Models with Examples in R*. Springer.

FRIEL, N., RASTELLI, R., WYSE, J. & RAFTERY, A. E. (2016). Interlocking directorates in irish companies using a latent space model for bipartite networks. *Proceedings of the National Academy of Sciences* **113**, 6629–6634.

GOLDENBERG, A., ZHENG, A. X., FIENBERG, S. E. & AIROLDI, E. M. (2010). A survey of statistical network models. *Foundations and Trends in Machine Learning* **2**, 129–233.

HAIR, J., BLACK, W., BABIN, B. & ANDERSON, R. (2018). *Multivariate Data Analysis*. Cengage Learning EMEA.

HANDCOCK, M. S., RAFTERY, A. E. & TANTRUM, J. M. (2007). Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **170**, 301–354.

HOFF, P. (2008). Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in Neural Information Processing Systems*.

HOFF, P. D. (2003). Random effects models for network data. *Technical Report* .

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

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

HOFF, P. D., RAFTERY, A. E. & HANDCOCK, M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association* **97**, 1090–1098.

HOLLAND, P. W., LASKEY, K. B. & LEINHARDT, S. (1983). Stochastic blockmodels: First steps. *Social Networks* **5**, 109–137.

KIM, M. & LESKOVEC, J. (2012). Latent multi-group membership graph model. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*. Omnipress.

KOLACZYK, E. D. & CSÁRDI, G. (2014). *Statistical Analysis of Network Data with R*, vol. 65. Springer.

KRIVITSKY, P. N., HANDCOCK, M. S., RAFTERY, A. E. & HOFF, P. D. (2009). Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social Networks* **31**, 204–213.

LESKOVEC, J. & MCAULEY, J. J. (2012). Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems*.

MA, Z., MA, Z. & YUAN, H. (2020). Universal latent space model fitting for large networks with edge covariates. *Journal of Machine Learning Research* **21**, 1–67.

MCCALLUM, A. K., NIGAM, K., RENNIE, J. & SEYMORE, K. (2000). Automating the construction of internet portals with machine learning. *Information Retrieval* **3**, 127–163.

NEWMAN, M. (2010). *Networks: An Introduction*. OUP Oxford.

NEWMAN, M. E. & CLAUSET, A. (2016). Structure and inference in annotated networks. *Nature Communications* **7**, 11863.

SEWELL, D. K. & CHEN, Y. (2015). Latent space models for dynamic networks. *Journal of the American Statistical Association* **110**, 1646–1657.

SEWELL, D. K. & CHEN, Y. (2016). Latent space models for dynamic networks with weighted edges. *Social Networks* **44**, 105–116.

WANG, J., ZHAO, Q., HASTIE, T., OWEN, A. B. et al. (2017). Confounder adjustment in multiple hypothesis testing. *The Annals of Statistics* **45**, 1863–1894.

WARD, M. D. & HOFF, P. D. (2007). Persistent patterns of international commerce. *Journal of Peace Research* **44**, 157–175.

WARD, M. D., SIVERSON, R. M. & CAO, X. (2007). Disputes, democracies, and dependencies: A reexamination of the kantian peace. *American Journal of Political Science* **51**, 583–601.

WARD, M. D., STOVEL, K. & SACKS, A. (2011). Network analysis and political science. *Annual Review of Political Science* **14**, 245–264.

XU, Z., KE, Y., WANG, Y., CHENG, H. & CHENG, J. (2012). A model-based approach to attributed graph clustering. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*. ACM.

YANG, J., MCAULEY, J. & LESKOVEC, J. (2013). Community detection in networks with node attributes. In *2013 IEEE 13th International Conference on Data Mining*. IEEE.

YOUNG, S. J. & SCHEINERMAN, E. R. (2007). Random dot product graph models for social networks. In *International Workshop on Algorithms and Models for the Web-Graph*. Springer.

ZHANG, Y., LEVINA, E. & ZHU, J. (2016). Community detection in networks with node features. *Electronic Journal of Statistics* **10**, 3153–3178.