A robust Bayesian latent position approach for community detection in networks with continuous attributes

Zhumengmeng Jin[1,*], Juan Sosa[2] , Brenda Betancourt[3]

[1] University of Florida, 116D Griffin-Floyd Hall, P.O. Box 118545, Gainesville, FL, 32611-8545, USA

[2] Universidad Nacional de Colombia, Carrera 45 # 26-85, Edificio Uriel Gutiérrez, Bogotá D.C., Colombia

[3] NORC at the University of Chicago, 4350 East-West Highway, 8th Floor, Bethesda, MD, 20814, USA

Abstract

The increasing prevalence of multiplex networks has spurred a critical need to take into account potential dependencies across different layers, especially when the goal is community detection, which is a fundamental learning task in network analysis. We propose a full Bayesian mixture model for community detection in both single-layer and multi-layer networks. A key feature of our model is the joint modeling of the nodal attributes that often come with the network data as a spatial process over the latent space. In addition, our model for multi-layer networks allows layers to have different strengths of dependency in the unique latent position structure and assumes that the probability of a relation between two actors (in a layer) depends on the distances between their latent positions (multiplied by a layer-specific factor) and the difference between their nodal attributes. Under our prior specifications, the actors' positions in the latent space arise from a finite mixture of Gaussian distributions, each corresponding to a cluster. Simulated examples show that our model outperforms existing benchmark models and exhibits significantly greater robustness when handling datasets with missing values. The model is also applied to a real-world three-layer network of employees in a law firm.

Keywords: community detection; latent position model; mixture model; missing data; spatial process; multiplex network; visualization.

1 Introduction

Network data conveniently describes the relationships between actors in complex systems and is ubiquitous in many statistical applications, including finance, social science, criminology, biology, epidemiology, and

[*] Corresponding author.

E-mail addresses: z.jin@ufl.edu (Z. Jin), jcsosam@unal.edu.co (J. Sosa), betancourt-brenda@norc.org (B. Betancourt).

computer science, among others. Understanding the relationships between actors can aid domain experts. For instance, in epidemiology, people in a certain area can be portrayed in a contact network that can be studied to detect infectious disease outbreaks. In criminology, communications between terrorists form a terrorist network, helping intelligence agencies to better counter terrorism.

Many models have been developed for the inference of networks over the past decades (e.g., Erdös and Rényi, 1959, Frank and Strauss, 1986), among which the broad class of latent space models is one of the most widely used (see, e.g., Sosa [2021] for an exhaustive review). Suppose the network under study has N actors, then under latent space models, there are N independent and identically distributed (i.i.d.) latent variables $z_1, \ldots, z_N$, one for each actor. Under a mild exchangeability assumption in Hoff [2007], results in Aldous [1985] and Hoover [1982] show that edge variables $y_{i,j}$, where $i, j \in \{1, \ldots, N\}$, depend on latent variables through a symmetric function $\gamma(z_i, z_j)$ that is meant to capture any pattern in the network beyond any known covariate information.

Many well-known models fall into the category of latent space models, which can be distinguished between two cases depending on whether latent variables are discrete or continuous [Matias and Robin, 2014]. For instance, stochastic block models [Nowicki and Snijders, 2001, Wang and Wong, 1987] – hereafter SBM – are special cases of latent space models with discrete latent variables $z_i \in \{1, 2, \ldots, K\}$. When latent variables are assumed to be continuous, another approach using latent variables is the class of latent position models (LPM) proposed by Hoff et al. [2002] which our model in the paper is built upon. In its basic formulation, LPMs model the edge variables $y_{i,j}$ as conditionally independent given the distance between latent variables $\gamma(z_i, z_j) = -\|z_i - z_j\|$, which naturally accounts for transitivity effects through the latent space (typically a Euclidean K-dimensional space for a predetermined K) where z_i lives. Later on, Handcock et al. [2007] proposed an extension on Hoff et al.'s LPM, namely the latent position cluster model (LPCM), by imposing a Gaussian mixture prior on the latent positions to perform clustering tasks. Krivitsky et al. [2009] further extended Handcock et al.'s model by adding the random sender and receiver effects proposed by Hoff [2005]. Other formulations of $\gamma(\cdot, \cdot)$ can be found in Schweinberger and Snijders [2003], Hoff [2005, 2009], Athreya et al. [2017], Minhas et al. [2019], among others.

Besides edge information of a network, extra information like node and edge attributes and different types of edges are often available, and should ideally be leveraged for inference. Typical ways to incorporate attributes in a network model include: (1) modeling the network as a function of the attributes (see, e.g., Hoff et al., 2002, Hoff, 2005); (2) modeling the attributes as a function of the network [Guha and Rodriguez, 2021]; (3) jointly modeling the network and attributes (Linkletter, 2007, Kim and Leskovec, 2012, Fosdick and Hoff, 2015, Ciminelli et al., 2019). We consider taking the joint modeling approach similar to the social network spatial model (SNSM) proposed by Ciminelli et al. [2019]. Denote the continuous nodal attribute for actor i as x_i, Ciminelli et al. modeled edges $y_{i,j}$ as conditionally independent given $\|z_i - z_j\|$ and

$\|x_i - x_j\|$, and further modeled the nodal attributes as a spatial process over the latent space. Note that joint modeling does not require the network or the attributes to be fully observed as the first two approaches do, hence one could predict missing network and attribute data (if there is any). In addition, it improves model fitting by capturing the dependence structure between latent variables and the attributes (when such dependency exists), as we will see in Section 3.

We propose a full hierarchical Bayesian model that builds on Ciminelli et al.'s SNSM. But instead of using a Gaussian distribution as the prior for latent positions as in Ciminelli et al. [2019], we impose a Gaussian mixture prior as in Handcock et al. [2007], so that our model could capture the group structure in the network. Detecting communities or clusters among actors in the network is an important task in network analysis and has spurred the development of many models and algorithms, among which the SBM has motivated an active line of research that deals with community detection (see, e.g., Lee and Wilkinson [2019] for a review). However, SBM suffers from poor model fitting when many actors fall between clusters [Hoff et al., 2002] as we will see in the simulation studies. We will compare our model with an SBM that incorporates covariates as fixed effects (i.e., model the edge variables as a function of latent classes and covariates [Leger, 2016]), and we call this model a covariate-assisted stochastic block model (CSBM). We will show that our model presents improved model fitting while producing similar clustering results as CSBM.

We also propose an extension of our model to multi-layer network settings. Multi-layer networks can generally be categorized into two cases: cross-sectional networks that have different types of connections (e.g., social networks of friendship, coworker-ship, etc.) and time-varying networks where the same type of connections are measured over time (e.g., a trade network that changes over time). We consider a type of cross-sectional multi-layer network where each layer has a common set of actors. Substantial work has been done on latent space models for cross-sectional multi-layer networks that take a Bayesian approach (see, e.g., Gollini and Murphy, 2016, Salter-Townshend and McCormick, 2017, D'Angelo et al., 2019, 2023, Sosa and Betancourt, 2022, Durante and Dunson, 2018, Wang et al., 2019, MacDonald et al., 2020). In extending our model to the multiple networks setting, we adopt the approach in Sosa and Betancourt [2022] in a parsimonious way, where latent positions are assumed to be the same for all layers, but the strength of borrowing such latent structure information is allowed to be different across different layers. Note that, the original model in Sosa and Betancourt [2022] assumed different latent positions for different layers and had an additional hierarchy on the hyperparameters. In D'Angelo et al., 2023, the authors adopted a similar approach where they also assume the same latent positions across all layers and their model is capable of clustering assignments by using a Dirichlet process mixture on latent positions instead of a finite Gaussian mixture prior. However, their model does not take into account attribute information. We propose a model for attributed network data sets that jointly models the network and attributes and performs clustering tasks. The focus of the work is to see how joint modeling improves network estimation and clustering accuracy.

The remainder of the paper is organized as follows. Section 2 contains general background on the spatial process and introduces the proposed model (in both single- and multi-layer network settings) which we call the latent position joint mixture model (LPJMM) in the rest of the paper. In addition, prior specification, identifiable problem, and inference will also be discussed in this section. Several simulation studies are conducted in Section 3, where LPJMM is compared with Handcock et al.'s LPCM, Ciminelli et al.'s SNSM and CSBM in single-layer settings. We will see that LPJMM outperforms these benchmark models under different scenarios, especially when networks have missing edges. The multi-layer model is also evaluated under a two-layer network scenario. Section 4 illustrates how to use the proposed model to provide insights from a given network and the model is applied to a real-world multi-layer network data set. Finally, we conclude with some discussion in Section 5.

2 Models

We first review the LPM introduced in Hoff et al. [2002], and then build upon it with a spatial process as in Ciminelli et al. [2019] to allow for joint modeling of the network and the nodal attributes, and with a finite Gaussian mixture distribution for latent positions to allow for clustering.

Consider a binary single-layer network with N actors. Denote its adjacency matrix as $\mathbf{Y} = (y_{i,j}) \in \{0,1\}^{N\times N}$, where $y_{i,j} = 1$ if actors i and j are connected, and $y_{i,j} = 0$ if they are not connected. Suppose the network data comes with a one-dimensional nodal attribute x_i for each actor, and denote the covariate as $\mathbf{x} = (x_i) \in \mathbb{R}^N$. The LPM assumes that each actor i has an observed latent position z_i in a K-dimensional Euclidean latent space, the so-called latent space, for some $K \in \mathbb{N}$. Let $\mathbf{z} = (z_i) \in \mathbb{R}^{N\times K}$, then LPM models $y_{i,j}$ as conditionally independent given distances between nodal attributes as well as distances between latent positions via logistic regression. But instead of the logistic link, we use the probit link in our model. The analysis of probit regression models can often be facilitated by a Gibbs sampler constructed using the data augmentation approach that introduces latent variables with truncated normal distributions [Albert and Chib, 1993]. (See also Sosa and Betancourt (2022) for a discussion on the choice of link functions.) Specifically, for $i, j \in \{1, \dots, N\}$ and $i \neq j$,

$$y_{i,j} \mid \mathbf{z}, \mathbf{x}, a, b, \theta \overset{\text{ind}}{\sim} \operatorname{Ber}\big(\Phi(a + b|x_i - x_j| - \theta\|z_i - z_j\|)\big)\,, \tag{1}$$

where $a, b \in \mathbb{R}$ and $\theta \in \mathbb{R}^+$, $\operatorname{Ber}(p)$ is a Bernoulli distribution that takes value 1 with some probability p, $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^K$ and $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. Note that we impose a factor θ for the distance between latent positions, which is different from Hoff et al. [2002] and Krivitsky et al. [2009]. Although θ is unidentifiable in single-layer networks, it plays a non-trivial role in multi-layer network settings (introduced in Section 2.1). We defer a detailed discussion of θ to Section 2.4.

To allow for joint modeling of the network and nodal attributes, we model the nodal attributes as a spatial process over the latent space $\mathbb{R}^K$. Hence, nodal attributes are treated as random variables indexed by their latent positions, and the distance between these random variables is found by the distance between their corresponding positions. As in Ciminelli et al. [2019], we specify the spatial process as a Gaussian process that is stationary with mean β and isotropic (see Banerjee et al., 2015 for definitions). In this case, the process is completely defined by its covariance function $\mathrm{Cov}(d)$, where d is the distance between two random variables in the Gaussian process. In particular, we specify $\mathrm{Cov}(d)$ with an exponential kernel, that is,

$$\mathrm{Cov}(d) = \begin{cases} \tau^2 + \sigma^2, & \text{if } d = 0; \\ \sigma^2 \exp(-\phi d), & \text{if } d > 0, \end{cases}$$

where $\tau \geq 0$, $\sigma > 0$ and $\phi > 0$. It is well-known that such a covariance structure is *valid*, i.e., the covariance matrix for any finite collection of random variables in the process is positive definite [Banerjee et al., 2015]. Let $M_{\mathbf{z}} = (m_{ij}) \in \mathbb{R}^{N \times N}$ where $m_{ij} = \exp(-\phi \| z_i - z_j \|)$ and denote I_N as the N-dimensional identity matrix, then the Gaussian process of the nodal attributes is constructed as follows,

$$\mathbf{x} \mid \mathbf{z}, \beta, \sigma, \tau, \phi \sim \mathrm{N}_N(\beta \mathbf{1}_N,\, \sigma^2 M(\mathbf{z}, \phi) + \tau^2 I_N), \tag{2}$$

where N_d is a d-dimensional multivariate normal distribution for some dimension $d \in \{2, 3, \dots\}$, and $\mathbf{1}_N$ is an N-dimensional vector with all 1s.

As in Krivitsky et al. [2009], we impose a Gaussian mixture distribution on latent positions, which allows us to cluster actors into different groups. Suppose there are $H < \infty$ predetermined number of components in the Gaussian mixture distribution, then

$$z_i \mid \boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\kappa} \overset{\text{i.i.d.}}{\sim} \sum_{h=1}^{H} \omega_h \mathrm{N}_K(\mu_h,\, \kappa_h^2 I_K)\,, \tag{3}$$

where $\boldsymbol{\omega} = \{\omega_1, \dots, \omega_H\}$, $\boldsymbol{\mu} = \{\mu_1, \dots, \mu_H\}$, $\boldsymbol{\kappa} = \{\kappa_1, \dots, \kappa_H\}$. Note that μ_h is a K-dimensional mean vector where $h \in \{1, \dots, H\}$, and ω_h is the probability that an actor belongs to the h-th group such that $\omega_h \in (0, 1)$ and $\sum_{h=1}^{H} \omega_h = 1$.

In single-layer network settings, the model is given by Eqs. (1) to (3). Under our model, nodal attributes of two actors whose latent positions are close are more likely to be similar according to the exponential covariance structure. If $b < 0$ ($b > 0$), actors with similar attributes are more (less) likely to be connected. When $b = 0$, nodal attributes do not affect the distribution of the network directly (but it still has an indirect impact on the network through latent positions by Eq. (2)).

2.1 An extension to multi-layer networks.

Our model can also be extended to multi-layer network settings in the following way. Suppose we have L layers $\mathbf{Y}_1, \ldots, \mathbf{Y}_L$ in the network, where all layers are defined over the same set of actors. We assume the same latent positions $\mathbf{z}$ for all layers but allow the strength of borrowing such latent structure information to be different by imposing layer-specific factors θ_ℓ for $\ell \in \{1, \ldots, L\}$. Our model in multi-layer settings is then presented as follows

$$y_{i,j,\ell} \mid \mathbf{z}, \mathbf{x}, a_\ell, b_\ell, \theta_\ell \overset{\text{ind}}{\sim} \mathrm{Ber}\big(\Phi(a_\ell + b_\ell|x_i - x_j| - \theta_\ell\|z_i - z_j\|)\big)\,, \tag{4}$$

$$\mathbf{x} \mid \mathbf{z}, \beta, \sigma, \tau, \phi \sim \mathrm{N}_N(\beta\mathbf{1}_N,\, \sigma^2 M(\mathbf{z}, \phi) + \tau^2 I_N)\,, \tag{5}$$

$$z_i \mid \boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\kappa} \overset{\text{i.i.d.}}{\sim} \sum_{h=1}^{H} \omega_h \mathrm{N}_K(\mu_h,\, \kappa_h^2 I_K)\,, \tag{6}$$

where $y_{i,j,\ell}$ is the edge variable between actors i and j in layer ℓ, and a_ℓ, b_ℓ and θ_ℓ are layer-specific parameters. Note that Eqs. (5) and (6) are the same as Eqs. (2) and (3). Fig. 1 shows a directed acyclic graph (DAG) representation of the model given by Eqs. (4) to (6).

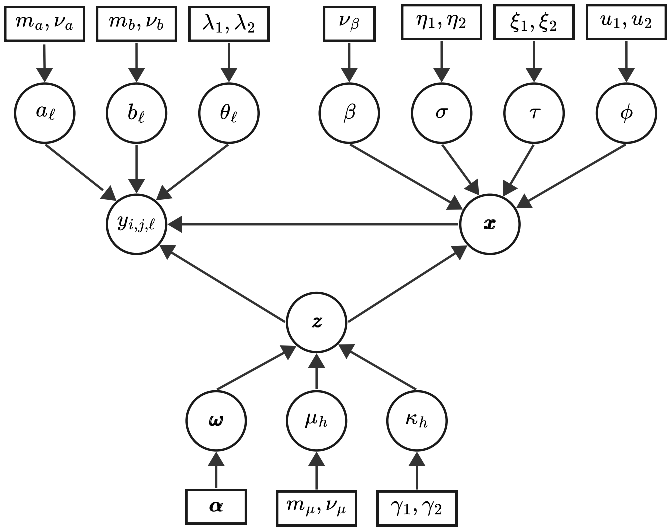


Figure 1: DAG representation of the LPJMM in multi-layer settings.

2.2 Prior specification

We take a Bayesian approach to estimate the model parameters. Without loss of generality, a Bayesian version of the model given by Eqs. (4) to (6) is formed by placing prior distributions on the unknown parameters $a_\ell, b_\ell, \theta_\ell$, β, σ, τ, ϕ, $\boldsymbol{\omega}$, $\boldsymbol{\mu}_h, \kappa_h$, for $\ell = \{1, \ldots, L\}$ and $h = \{1, \ldots, H\}$. In the model we consider, these parameters are assumed *a priori* independent. For parameters in the probit regression tier as specified by

Eq. (4), their priors are specified as follows:

$$a_\ell \overset{\text{i.i.d.}}{\sim} \mathrm{N}(m_a, \nu_a^2)\,, \qquad b_\ell \overset{\text{i.i.d.}}{\sim} \mathrm{N}(m_b, \nu_b^2)\,, \qquad \theta_\ell \overset{\text{i.i.d.}}{\sim} \mathrm{Gamma}(\lambda_1, \lambda_2)\,,$$

for $\ell \in \{1, \ldots, L\}$. The priors for the parameters in the spatial process tier as given in Eq. (5) are given as follows:

$$\beta \sim \mathrm{N}(0, \nu_\beta^2)\,, \qquad \sigma^2 \sim \mathrm{InvG}(\eta_1, \eta_2)\,, \qquad \tau^2 \sim \mathrm{InvG}(\xi_1, \xi_2)\,, \qquad \phi \sim \mathrm{U}(u_1, u_2)\,.$$

Finally, we put the following priors on the rest of the parameters:

$$\boldsymbol{\omega} \sim \mathrm{Dir}(\alpha)\,, \qquad \mu_h \overset{\text{i.i.d.}}{\sim} \mathrm{N}_K(m_\mu, \nu_\mu^2 I_K)\,, \qquad \kappa_h^2 \overset{\text{i.i.d.}}{\sim} \mathrm{InvG}(\gamma_1, \gamma_2)\,.$$

Note that, m_a, ν_a, m_b, ν_b, λ_1, λ_2, ν_β, η_1, η_2, ξ_1, ξ_2, u_1, u_2, α, m_μ, ν_μ, γ_1 and γ_2 are user-specified hyperparameters, and $\mathrm{Gamma}(\cdot,\cdot)$, $\mathrm{InvG}(\cdot,\cdot)$, $\mathrm{U}(\cdot,\cdot)$, $\mathrm{Dir}(\cdot)$ represents Gamma, Inverse-Gamma, uniform, and Dirichlet distributions respectively.

2.3 Posterior distribution and model estimation

As is standard in Bayesian estimation of mixture models (see, e.g., Diebolt and Robert [1994]), we define a new variable g_i that serves as the missing data of group membership of actor i whose distribution depends on $\boldsymbol{\omega}$. In particular, $g_i = h$ if actor i belongs to the h-th group. The joint density of (z_i, g_i) given $\boldsymbol{\omega}$, $\boldsymbol{\mu}$ and $\boldsymbol{\kappa}$ is then given by

$$\prod_{h=1}^{H} \left\{ \omega_h \frac{1}{\sqrt{2\pi\kappa_h^2}} \exp\Big(-\frac{1}{2\kappa_h^2} \|z_i - \mu_h\|^2 \Big) \right\}^{\mathrm{I}_{\{g_i = h\}}}\,,$$

where the indicator function $\mathrm{I}_{\{g_i=h\}} = 1$ if $g_i = h$, and $\mathrm{I}_{\{g_i=h\}} = 0$ otherwise. Let $\mathbf{g} = (g_i)_{i=1}^N$ be the group membership for all actors and $\mathcal{L}(\cdot)$ be the law of a random variable. Then the posterior distribution of $\mathbf{z}$, $\mathbf{g}$ and the parameters (whose priors are specified in Section 2.2) is given by

$$\begin{aligned}
&\Pi(\mathbf{z}, \mathbf{g}, a_1, \ldots, a_L, b_1, \ldots, b_L, \theta_1, \ldots, \theta_L, \beta, \tau^2, \sigma^2, \phi, \boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\kappa} \mid \mathbf{Y}_1, \ldots, \mathbf{Y}_L, \mathbf{x}) \\
&\propto \Big\{ \prod_{\ell=1}^{L} \mathcal{L}(\mathbf{Y}_\ell \mid \mathbf{z}, \mathbf{x}, a_\ell, b_\ell, \theta_\ell) \Big\} \mathcal{L}(\mathbf{x} \mid \mathbf{z}, \sigma, \tau, \phi) \mathcal{L}(\mathbf{z}, \mathbf{g} \mid \boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\kappa}) \Big\{ \prod_{\ell=1}^{L} \mathcal{L}(a_\ell) \mathcal{L}(b_\ell) \mathcal{L}(\theta_\ell) \Big\} \\
&\quad \times \mathcal{L}(\beta) \mathcal{L}(\sigma^2) \mathcal{L}(\tau^2) \mathcal{L}(\phi) \mathcal{L}(\boldsymbol{\omega}) \mathcal{L}(\boldsymbol{\mu}) \mathcal{L}(\boldsymbol{\kappa})\,.
\end{aligned}$$

Note that the dimension of the posterior distribution has dimension $NK + N + 3L + 3H + 4$ and the corresponding posterior density is presented as follows,

$$
\begin{aligned}
&\pi(\mathbf{z}, \mathbf{g}, a_1, \ldots, a_L, b_1, \ldots, b_L, \theta_1, \ldots, \theta_L, \beta, \tau^2, \sigma^2, \phi, \boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\kappa} \mid \mathbf{Y}_1, \ldots, \mathbf{Y}_L, \mathbf{x}) \\
&\propto \prod_{\substack{i,j=1 \\ i \neq j}}^{N} \prod_{\ell=1}^{L} \big[\Phi(a_\ell + b_\ell |x_i - x_j| - \theta_\ell \|z_i - z_j\|)\big]^{y_{i,j,\ell}} \big[1 - \Phi(a_\ell + b_\ell |x_i - x_j| - \theta_\ell \|z_i - z_j\|)\big]^{1 - y_{i,j,\ell}} \\
&\times |\sigma^2 M(\mathbf{z}, \phi) + \tau^2 I_N|^{-\frac{1}{2}} \exp\Big(- \frac{1}{2} (\mathbf{x} - \beta \mathbf{1})^\intercal \big(\sigma^2 M(\mathbf{z}, \phi) + \tau^2 I_N\big)^{-1} (\mathbf{x} - \beta \mathbf{1})\Big) \\
&\times \prod_{i=1}^{N} \prod_{h=1}^{H} \Big\{ \frac{\omega_h}{\sqrt{\kappa_h^2}} \exp\Big(- \frac{1}{2\kappa_h^2} \|z_i - \mu_h\|^2\Big)\Big\}^{\mathrm{I}_{\{g_i = h\}}} \\
&\times \exp\Big(\frac{1}{2\nu_a^2} \sum_{\ell=1}^{L} (a_\ell - m_a)^2 + \frac{1}{2\nu_b^2} \sum_{\ell=1}^{L} (b_\ell - m_b)^2 \Big) \prod_{\ell=1}^{L} \theta_\ell^{\lambda_1 - 1} \exp(-\lambda_2 \theta_\ell) \\
&\times \exp\Big(\frac{\beta^2}{2\nu_\beta^2}\Big) (\sigma^2)^{-\eta_1 - 1} (\tau^2)^{-\xi_1 - 1} \exp\Big(- \frac{\eta_2}{\sigma^2} - \frac{\xi_2}{\tau^2}\Big) \mathrm{I}_{\{\phi \in [u_1, u_2]\}} \\
&\times \prod_{h=1}^{H} \Big\{ \omega_h^{\alpha_h - 1} \mathrm{I}_{\{\sum_{h=1}^{H} \omega_h = 1\}} \exp\Big(- \frac{1}{2\nu_\mu^2} \|\mu_h - m_\mu\|^2\Big) (\kappa_h^2)^{-\gamma_1 - 1} \exp\Big(- \frac{\gamma_2}{\kappa_h^2}\Big)\Big\}.
\end{aligned}
$$

2.4 Inference and identifiability of parameters

Note that the posterior distribution is highly intractable, hence we must resort to Markov chain Monte Carlo (MCMC) methods for inferences on model parameters. A Markov chain of the parameters is generated via the program "Just Another Gibbs Sampler" (JAGS) which is implemented in R [R Core Team, 2021] using the rjags package [Plummer, 2022].

Several parameters are not identifiable in our model. Firstly, due to factors θ_ℓ and ϕ, and the fact that latent positions are incorporated in the posterior only through their distances, the posterior is, therefore, invariant to θ_ℓs and ϕ, and is invariant to scaling, reflection, rotation, and translation of the latent positions $\mathbf{z}$. (Note that, Hoff et al., 2002 and Krivitsky et al., 2009 did not have θ_ℓs, hence their posterior is not invariant to the scaling of latent positions.) Although θ_ℓs are not identifiable and do not affect the model fitting, their ratios $\theta_{\ell 1}/\theta_{\ell 2}$ still provide valid information on layer's relative strength of borrowing information from the latent space in multi-layer settings.

Despite being unidentifiable, one can still make inferences on the latent positions and find a reasonable estimate for $\mathbf{z}$ through a post-process which we now describe. Similar to the definition in [Hoff et al., 2002], we define the equivalence class of $\mathbf{z} \in \mathbb{R}^{N \times K}$, denoted as $[\mathbf{z}]$, to be the set of positions that are equivalent to $\mathbf{z}$ under scaling, reflection, rotation, and translation. Given a fixed reference position $\mathbf{z}_{ref}$, a position $\mathbf{z}_*$ is found in $[\mathbf{z}]$ such that $\mathbf{z}_* = \arg\min_{\mathbf{z}' \in [\mathbf{z}]} \operatorname{tr}(\mathbf{z}_{ref} - \mathbf{z}')^\intercal (\mathbf{z}_{ref} - \mathbf{z}')$, which is the so-called Procrustes transformation. In simulation studies, $\mathbf{z}_{ref}$ is naturally chosen to be the true latent position, while in practical applications, we could use the last iteration of the Markov chain of latent positions as the reference. The Procrustes transformation is performed for each iteration of the Markov chain of the latent positions $\{\mathbf{z}_n\}$, and an estimate for $\mathbf{z}$ is taken as the mean of the Procrustes transformations of $\{\mathbf{z}_n\}$.

As occurs in Bayesian mixture models, the label-switching problem for the group membership $\mathbf{g}$ is another source of non-identifiability. That is, the posterior is invariant under permutations of clustering labels. Many algorithms have been proposed to obtain a single clustering estimate based on the MCMC sample of the group membership $\{\mathbf{g}_n\}$, including an optimization method (which we call "MaxPEAR" hereafter) in Fritsch and Ickstadt [2009] that finds a clustering that maximizes posterior expected adjusted Rand index (ARI), an optimization method ("MinBinder") in Lau and Green [2007] that minimizes Binder's loss function, and a greedy algorithm ("GreedyEPL") in Rastelli and Friel [2018] that aims to minimize the variation of information, among others. These approaches may generate different clustering estimates, and to get a better understanding of the model performance, all aforementioned algorithms (MaxPEAR, MinBinder, and GreedyEPL) are used to assess the model. Estimates based on these approaches are found using the packages `GreedyEPL` [Rastelli, 2021] and `mcclust` [Fritsch, 2022].

3 Simulation

Four network scenarios are considered in this section to evaluate our model. The first three simulations consider single-layer networks generated from LPJMM, but the underlying latent positions display varying degrees of clustering. In each of the three simulations, LPJMM is compared with three other models designed only for single-layer networks, namely LPCM in Handcock et al. [2007], SNSM in Ciminelli et al. [2019], and CSBM in Leger [2016]. Model assessments include how well a model could recover the group membership and the latent position configuration, and a goodness-of-fit test using summaries of networks including density, transitivity, and assortative coefficient based on the estimated group membership $\mathbf{g}$ (see Kolaczyk and Csárdi, 2020 for definitions). Furthermore, LPJMM is also evaluated by how accurately it estimates certain parameters. In the last simulation, we consider a two-layer network and the performance of LPJMM could be further evaluated by the estimated ratio θ_1/θ_2, which reflects differences in each layer's dependency on the latent position structure.

LPJMM and SNSM are implemented using the `rjags` package, and LPCM and CSBM are implemented using the `latentnet` [Krivitsky and Handcock, 2022] and `sbm` [Chiquet et al., 2022] packages respectively. Model specifications of these models can be found in Appendix A.

3.1 A single-layer network

Consider a single-layer network (i.e., $L = 1$) with $N = 100$ actors generated as follows. Firstly, generate latent positions $\mathbf{z}$ from a mixture of $H = 5$ multivariate normal distributions, and then generate attributes $\mathbf{x}$ jointly from a multivariate normal distribution with mean $\beta \mathbf{1}_N = \mathbf{0}$ and covariance matrix given by $\text{Cov}(d)$ in Section 2 where $\phi = 0.5$, $\tau^2 = 0.3$, $\sigma^2 = 1$. Finally, the network data is generated according to Eq. (1) with $a = 5$, $b = -2$, and $\theta = 2.72$. See Fig. 2 for a visualization of the simulated network. The network is fairly sparse with the density equal to 0.1531 and shows strong transitivity and assortative mixing with coefficients 0.5049 and 0.5512 respectively. Note that the clustering pattern in the true latent space $\mathbf{z}$ is discernible with little overlap between different groups. Of course, the estimation of $\mathbf{z}$, hence $\mathbf{g}$, will be more accurate when clusters are far away from each other. On the other hand, when members are thoroughly blended, the probabilities of connections within clusters are comparable to those across clusters, making it more difficult to estimate true group membership. We explore these two extreme scenarios (one with a very distinctive clustering pattern and another with well-blended members) in Sections 3.2 and 3.3.

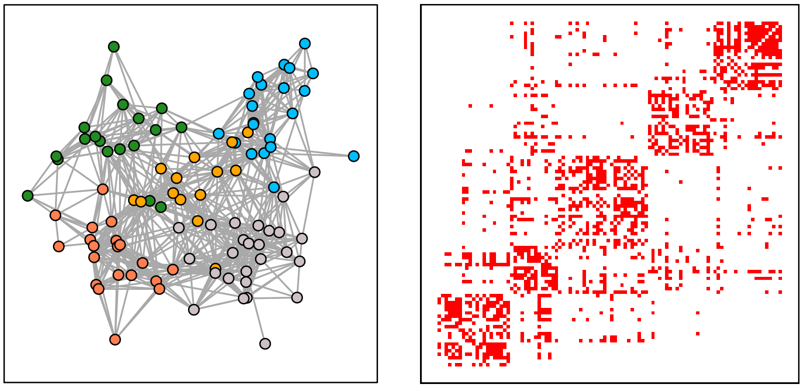


Figure 2: Left: A visualization of the network based on the true latent position and color indicates group membership $\mathbf{g}$. Right: Heatmap of the adjacency matrix (where actors are reordered according to $\mathbf{g}$).

As for the prior specifications, we set $m_a = m_b = 0$, and $\nu_a^2 = \nu_b^2 = 9$ to allow a wide range of values for a and b. Let $\theta \sim \text{Gamma}(1, 1)$ so that θ has mean 1. An almost flat prior is imposed on β by setting $\nu_\beta = 10^4$. The same uniform prior $\text{U}(0, 1)$ as in Ciminelli et al. [2019] is specified for ϕ. We suggest the sum of the prior means of τ^2 and σ^2 to be on the same scale as the sample variance of $\mathbf{x}$, and here we use $\sigma^2 \sim \text{InvG}(2, 1)$ and $\tau^2 \sim \text{InvG}(2, 1)$. Let $\alpha = \mathbf{1}$ so that the prior on $\boldsymbol{\omega}$ is a flat Dirichlet distribution. Following the heuristics in Sosa and Betancourt [2022], we specify $\mu_h \overset{\text{i.i.d.}}{\sim} \text{N}_K(\mathbf{0}, 2/3 I_K)$ and $\kappa_h^2 \overset{\text{i.i.d.}}{\sim} \text{InvG}(3, 2/3)$ so that $\text{var}(z_{ij}|g_i) = 1$.

Note that the latent space dimension K and the number of clusters H in the model need to be prespecified along with the priors. We take K to be the true dimensions of the latent space (i.e., $K = 2$) since this facilitates model assessment by allowing visualizations of the estimated latent positions. One could also use the Watanabe-Akaike Information Criterion (WAIC) to select a K with the smallest WAIC as in Sosa and

Betancourt [2022]. However, WAIC and other information criteria like the Deviance Information Criterion (DIC) do not help choose the number of clusters H. A comparison of the model assessment for different specified H is given in Appendix B. We noticed that model performances are significantly worse when H is chosen to be smaller than the truth. However, model performances are similar among models whose H is at least as large as the truth. Therefore, we suggest choosing H to be the largest number of groups that one is willing to accept, and in this example, we choose H to be 5.

We then fit LPJMM using MCMC sampling with $20\,000$ burn-in iterations and a further $10\,000$ iterations kept for posterior analysis. The Markov chain mixes reasonably well and shows no signs of lack of convergence (see Appendix C for the traceplot of the log-likelihood chain).

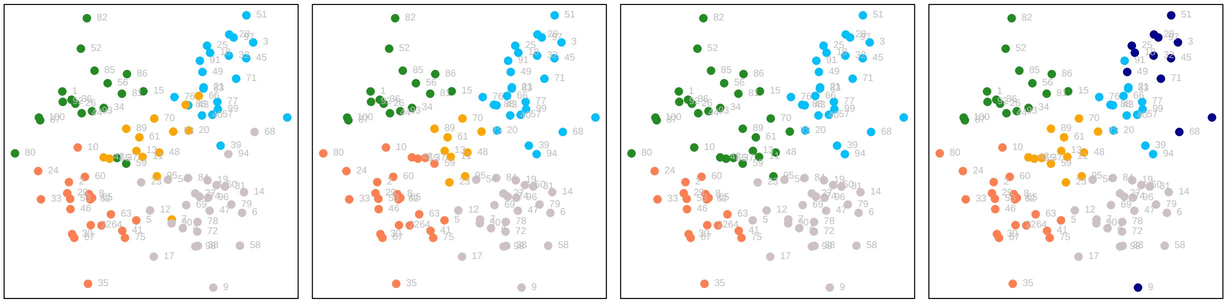
To evaluate a model's ability to recover the group membership, we first find estimates of clustering using the optimization algorithms (i.e., MaxPEAR, MinBinder, and GreedyEPL) mentioned in Section 2.4 and find ARI for each clustering estimate. Note that clusters are not defined in SNSM, therefore we only compare the ARI between LPJMM, LPCM, and CSBM. Since the `sbm` package takes a non-Bayesian approach that uses a Variational-EM algorithm to find a point estimator for the group membership $\mathbf{g}$, optimization methods like MaxPEAR are not necessary to analyze results from CSBM. Results in Table 1 suggest that these three models have a similar ability to recover group membership $\mathbf{g}$, with ARI of LPJMM using the MaxPEAR and MinBinder algorithms being higher than ARI (0.707) under the CSBM model. Although the highest ARI is given by LPCM using the MinBinder method, the estimated number of groups is 10 which significantly exceeds the true number of groups. Therefore, the overall best estimation in terms of both ARI and $\hat{\mathbf{g}}$ is given by LPJMM. A visualization of the estimated clusters based on the true latent positions is given in Fig. 3. Also, notice that the MinBinder algorithm tends to overestimate the number of clusters in the network in all models.

	LPJMM	LPCM	CSBM
MaxPEAR	0.737 (5)	0.707 (4)	–
MinBinder	0.712 (11)	0.748 (10)	–
GreedyEPL	0.664 (4)	0.688 (4)	–
Variational-EM	–	–	0.707 (6)

Table 1: ARI that corresponds to different estimation methods for group membership. Numbers in the parentheses represent numbers of estimated groups.

To further compare the ability to recover latent position configuration between LPJMM and LPCM, we find an estimate of the latent positions $\hat{\mathbf{z}}$ using the method of Procrustes transformation given in Section 2.4. Plots of $\hat{\mathbf{z}}$s of LPJMM and LPCM can be found in Appendix D, which suggest similar estimated configurations of $\mathbf{z}$.

Following Sosa and Betancourt [2022], we assess if models have a good fit in the sense of good reproduction



(a) Truth (b) LPJMM (c) LPCM (d) CSBM

Figure 3: (A): Color indicates the true group membership $\mathbf{g}$. (B)-(D): Color indicates the estimated group memberships $\hat{\mathbf{g}}$ of LPJMM, LPCM and CSBM respectively. Positions of the points in all plots are true latent positions $\mathbf{z}$.

of a variety of summary statistics. These statistics are calculated based on the networks generated from the model estimations. For LPJMM and SNSM, networks are simulated based on the parameters at every 10-th iteration of the generated Markov chain. For LPCM and CSBM, 1000 networks are simulated using their respective packages. Then for each model, we calculate the density, transitivity, and assortative coefficient (if applicable) calculated according to the true group membership for each simulated network. Boxplots and the averages of these summary statistics are given in Fig. 4 and Table 2 respectively. In general, the summary statistics results are not too disparate for all four models. However, LPJMM better captures these structural features of the network data, while LPCM tends to overestimate transitivity in the network, and both SNSM and CSBM tend to underestimate transitivity and assortativity in the network.

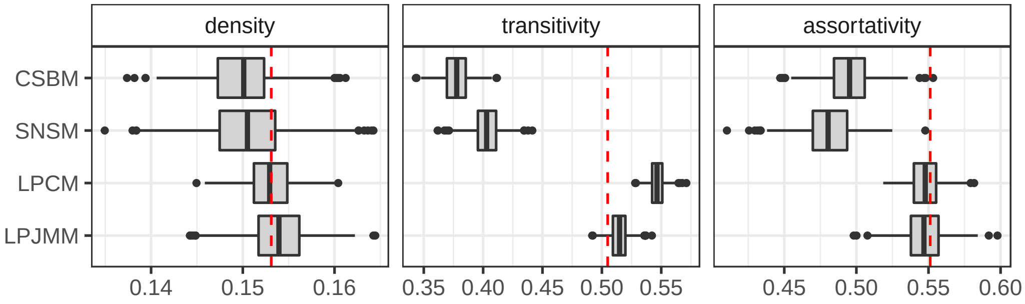


Figure 4: Boxplots of summary statistics for each model. Red dotted lines indicate the true values for network characteristics respectively.

3.2 A network with missing edges

In this simulation scenario, a network with $N = 100$, $H = 5$ is generated, which shows highly discernible clustering patterns (as shown in Fig. 5). The model parameters used to generate the network are given by $a = 1$, $b = 3.5$, $\theta = 1.25$, $\phi = 0.3$, $\tau^2 = 0.2$, $\sigma^2 = 1$ and $\beta = 0$. Model specifications are determined

	true value	LPJMM	LPCM	SNSM	CSBM
density	0.1531	0.1539	0.1530	0.1504	0.1499
transitivity	0.5049	0.5144	0.5467	0.4027	0.3776
assortativity	0.5512	0.5468	0.5475	0.4811	0.4954

Table 2: Means of the summary statistics of the simulated networks for each model.

similarly as in Section 3.1. We will show that LPJMM clearly outperforms other models when the network exhibits missing edges.

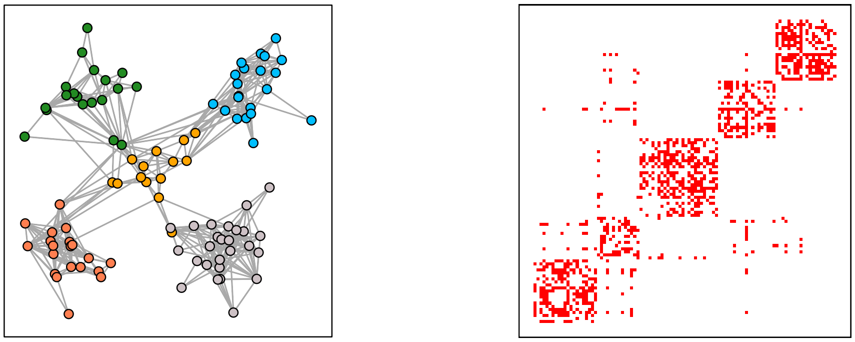


Figure 5: Network visualization and heatmap of the adjacency matrix.

Table 3 shows that, when edge information is fully observed, all models can accurately estimate the group membership (not considering SNSM), with LPJMM slightly outperforming the rest of the two models. Results from Table 4 are consistent with what we have seen in Section 3.1, where LPJMM and LPCM can reproduce the summary statistics quite close to the true values, while SNSM and CSBM tend to underestimate transitivity and assortativity.

	LPJMM	LPCM	CSBM
MaxPEAR	0.956 (5)	0.918 (5)	–
MinBinder	0.956 (5)	0.933 (6)	–
GreedyEPL	0.938 (5)	0.918 (5)	–
Variational-EM	–	–	0.900 (5)

Table 3: ARI and numbers of estimated groups (in parentheses).

	true value	LPJMM	LPCM	SNSM	CSBM
density	0.1008	0.1011	0.1008	0.0956	0.0965
transitivity	0.4822	0.4775	0.5453	0.2758	0.3563
assortativity	0.8643	0.8510	0.8594	0.6540	0.7725

Table 4: Means of the summary statistics.

As the level of missing edges rises from 10% to 95%, the advantage of LPJMM becomes more pronounced. To compare the models' ability to recover group membership, we use MaxPEAR to estimate g for LPJMM and LPCM. (The other two methods, MinBinder and GreedyEPL, will give a similar result.) We also compare the summary statistics under different levels of missingness. (Note that SNSM is excluded from the comparison since it does not perform well even when data is complete.) The results are shown in Fig. 6.

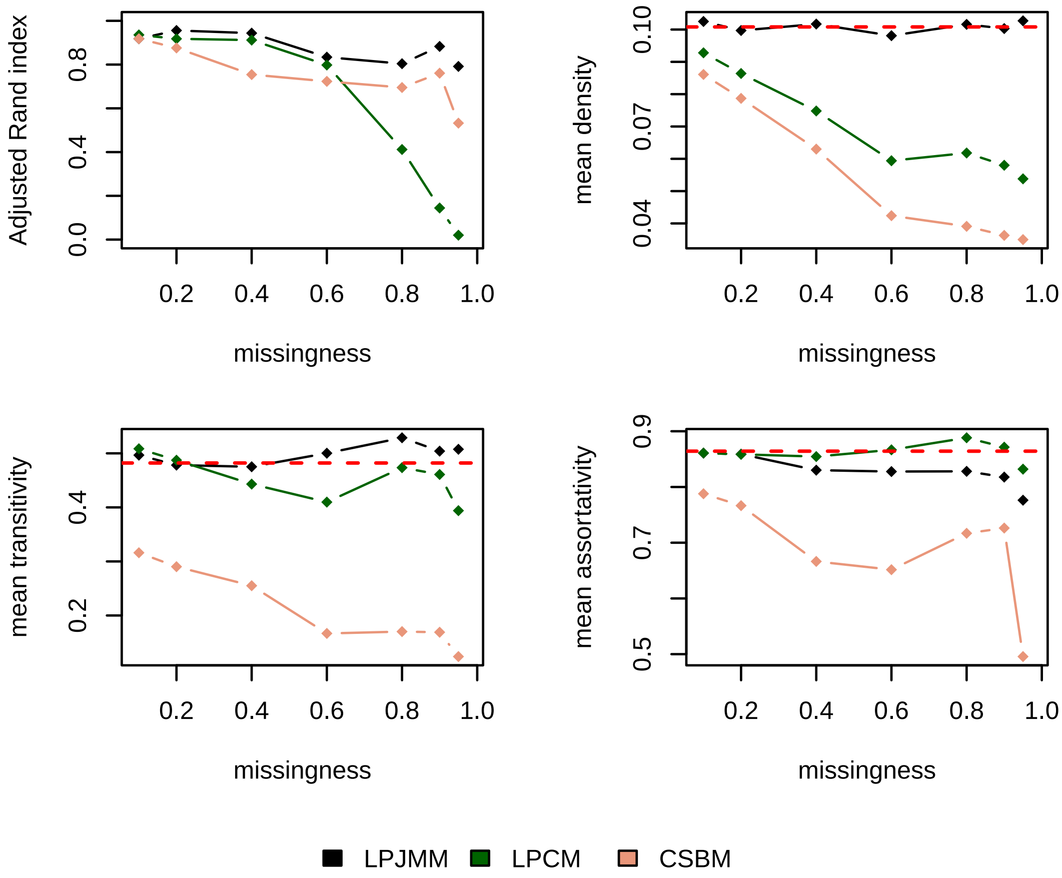


Figure 6: The upper left plot shows ARI as the percentage of the missing edges increases from 10% to 95%. In the rest of the three plots, the red horizontal dashed lines indicate the statistics summarized from the fully observed network.

The upper left plot in Fig. 6 reveals that LPCM closely matches LPJMM's performance when the missing data rate is below 60%. However, as the amount of missing data increases, LPCM's performance deteriorates rapidly. Additionally, the density estimation from LPCM deviates significantly from the true density as the missing data proportion rises. While CSBM's performance in terms of ARI appears less affected by the missing data rate, it struggles to recover all three summary statistics effectively. Consequently, when dealing with missing data, LPJMM emerges as a considerably more robust approach.

3.3 A network showing indiscernible clustering pattern

In this simulation scenario, we explore networks with well-blended group members. These networks do not show distinct clustering patterns in terms of the given group membership, which makes such membership more nominal than structural.

Consider a network shown in Fig. 7. This network does not display a distinct clustering pattern, as indicated in Fig. 7, along with a low assortativity coefficient, which is less than 0.1 (see Table 6). The lack of clear clustering makes it challenging to discern group memberships, which are minimally evident in the network data (see Table 5). However, the goodness-of-fit test given in Table 6 suggests that these models are still useful in recovering the network structure. Again, the results are consistent with the previous goodness-of-fit tests, with LPJMM slightly outperforming the other three models.

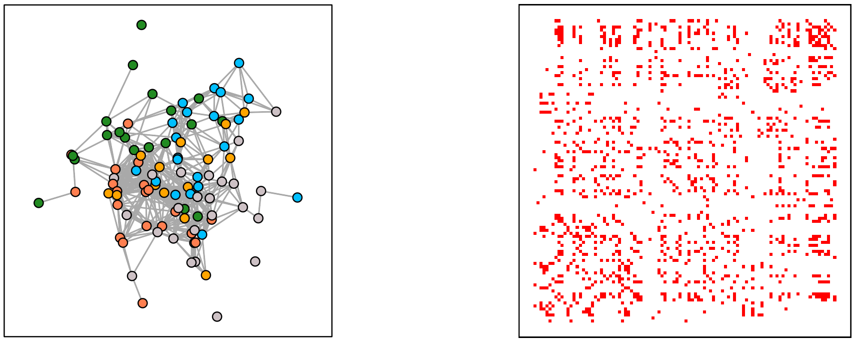


Figure 7: Network visualization and heatmap of the adjacency matrix.

	LPJMM	LPCM	CSBM
MaxPEAR	0.072 (3)	0.083 (8)	–
MinBinder	0.096 (25)	0.058 (25)	–
GreedyEPL	0.000 (1)	0.067 (3)	–
Variational-EM	–	–	0.075 (6)

Table 5: ARI and numbers of estimated groups (in parentheses).

	true value	LPJMM	LPCM	SNSM	CSBM
density	0.1242	0.1246	0.1241	0.1213	0.1226
transitivity	0.5181	0.5286	0.5476	0.3875	0.3931
assortativity	0.0840	0.0787	0.0742	0.0715	0.0481

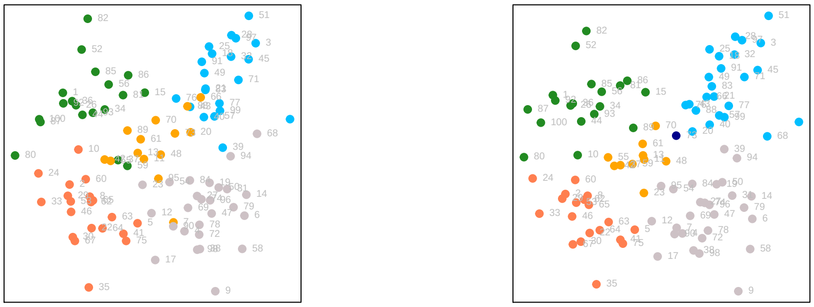
Table 6: Means of the summary statistics.

3.4 A two-layer network

Continue using the parameter setup in Section 3.1 and its generated network as the first layer, we generate a second layer of the network with $a_2 = 3$, $b_2 = 1$, $\theta_2 = 4$. As in Section 3.1, we fit LPJMM with $K = 2$ and $H = 5$ and evaluate the model's ability to recover the group membership using ARI. The results are given in Table 7, which shows similar clustering estimates as in Section 3.1 where only one layer is considered. The plot of the estimated latent position configurations is given in Fig. 8 (B), which visualizes the model's recovery of latent positions and group membership.

	MaxPEAR	MinBinder	GreedyEPL
ARI	0.748 (6)	0.753 (12)	0.662 (4)

Table 7: ARI that corresponds to different estimation methods for group membership. Numbers in the parentheses represent numbers of estimated groups.



(a) Truth

(b) Estimated **z** and **g**

Figure 8: (A): Points are plotted based on true latent position **z** and true group membership **g**. (B): Points are plotted using the estimated latent positions, and color represents the estimated group membership using the MaxPEAR method.

We also carry out the goodness-of-fit test and the result is given in Table 8, which shows that LPJMM captures these structural features accurately, and the result for layer 1 is similar to the result in Section 3.1.

		true value	mean
density	layer 1	0.1531	0.1535
	layer 2	0.1024	0.1023
transitivity	layer 1	0.5049	0.5088
	layer 2	0.5477	0.5546
assortativity	layer 1	0.5512	0.5466
	layer 2	0.6923	0.6890

Table 8: Means of the summary statistics.

Recall that θ_1 and θ_2 are of no direct interest since they are not identifiable. However, we are still interested in the ratio θ_1/θ_2 since it reflects the relative strength of borrowing information from the latent space of each

	true value	posterior mean	95% credible interval
θ_1/θ_2	0.680	0.653	(0.579, 0.721)
a_1	5	5.01	(4.719, 5.262)
a_2	3	3.25	(2.976, 3.572)
b_1	-2	-1.919	(-2.053, -1.766)
b_2	1	1.058	(0.901, 1.252)
β	0	-0.047	(-1.027, 1.01)
τ^2	0.3	0.409	(0.261, 0.592)
σ^2	1	0.642	(0.230, 1.684)

Table 9: Posterior means and 95% credible intervals.

layer. Although a_ℓ and b_ℓ are of no direct interest, we pay attention to their signs, especially that of b_ℓ because different signs of b_ℓ have different interpretations of the effect of attributes as discussed in Section 2. We also assess the model's ability to estimate parameters β, τ^2, and σ^2 using posterior means and 95% credible intervals. The results are given in Table 9. Overall, the performance of LPJMM in recovering the true values of these model parameters is pretty well, except for τ^2 and σ^2. Both LPJMM and SNSM tend to underestimate σ^2 and overestimate τ^2. That is, the covariance of the attributes tends to be underestimated, and although τ^2 is slightly overestimated, the variance of the attributes ($\tau^2 + \sigma^2$) still tends to be underestimated.

4 Real data analysis

In this section, we consider a three-layer network data set collected by [Lazega, 2001] from a corporate law firm from 1988-1991 in New England. This network describes three types of relationships (namely, networks of advice, friendship, and coworker contacts) between 71 lawyers in the law firm. Several actor attributes are also collected: age, gender, seniority (years with the firm), office (located in Boston, Hartford, or Providence), practice (litigation or corporate law), law school the lawyers attended (Harvard or Yale, University of Connecticut, or other universities) and status (partner or associate). A principal component analysis (PCA) is performed on age and seniority attributes, and the first principal component explains 89% of the variance which is of no surprise since age and seniority are highly correlated with a correlation coefficient being 0.78. We chose the first principal component to be the attribute $\mathbf{x}$ and let $H = 8$ since it is the largest number of clusters we expect in the network. Then the model is fitted to the network using the same prior and Markov chain setup as in Section 3.

The study of the Lazega network in this paper is meant to find out how the three types of relations can be explained by the findings deduced from the model fitting. We first visualize the estimated latent positions $\mathbf{z}$ colored by different categorical attributes (gender, office, practice, law school, and status) in Fig. 9. As we can see from these plots, the estimated positions $\mathbf{z}$ are well separated by the office (especially offices

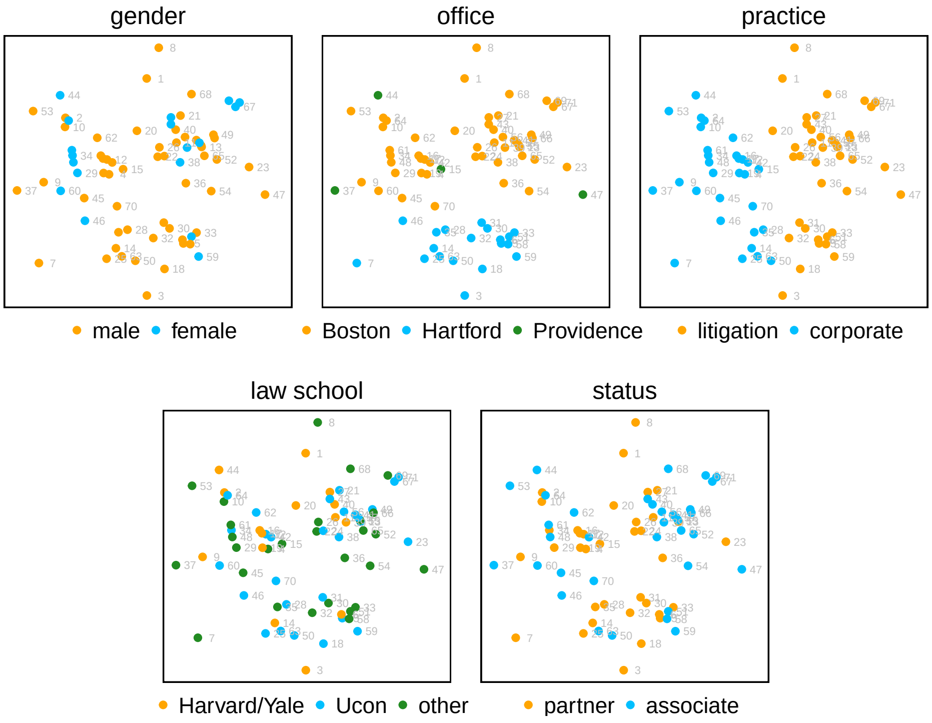
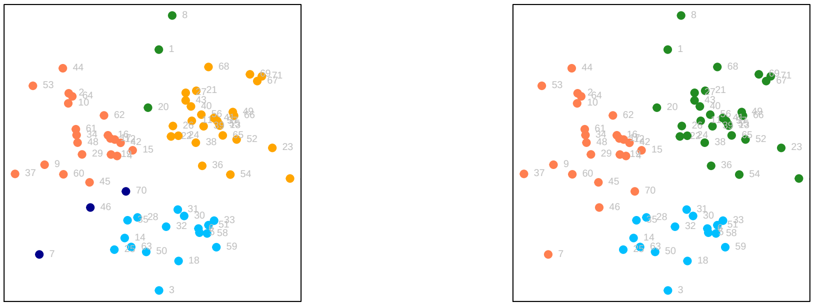


Figure 9: Points in all plots are drawn based on the estimated latent positions $\mathbf{z}$ and are colored based on their categories in gender, office, practice, law school, and status.



(a) MaxPEAR

(b) GreedyEPL

Figure 10: Points are plotted using the estimated latent positions and color indicates the estimated group membership using MaxPEAR and GreedyEPL methods respectively.

in Boston and Hartford) and practice. Compare these plots with $\mathbf{z}$ colored by MaxPEAR and GreedyEPL estimated clustering $\mathbf{g}$ in Fig. 10, we can see that both estimated $\mathbf{g}$ roughly clusters lawyers into three groups: lawyers in Hartford office, litigation lawyers in Boston or Providence offices, and corporate lawyers in Boston or Providence offices.

Plots of adjacency matrices of the three layers (where lawyers are grouped by the MaxPEAR estimate of $\mathbf{g}$) and their corresponding networks are given in Fig. 11, where we could see that the coworker network shows the most estimated clustering pattern, while the advice network presents the least of such pattern, which could also be seen from the relative ratios of θ_ℓs in Table 10. This means that lawyers from the same office and doing the same practice are more likely to become coworkers and friends, but who they seek advice from does not depend much on office locations and practices. Furthermore, we can deduce from the posteriors of b_ℓ in Table 10 that these lawyers tend to seek advice from people of similar age (or seniority) since the posterior estimate of b_1 is negative. In contrast, lawyers of different ages (or seniority) are more likely to become friends and coworkers. This conclusion is in line with the assortativity coefficients found based on the nodal attributes (lawyer's age) given in Table 11.

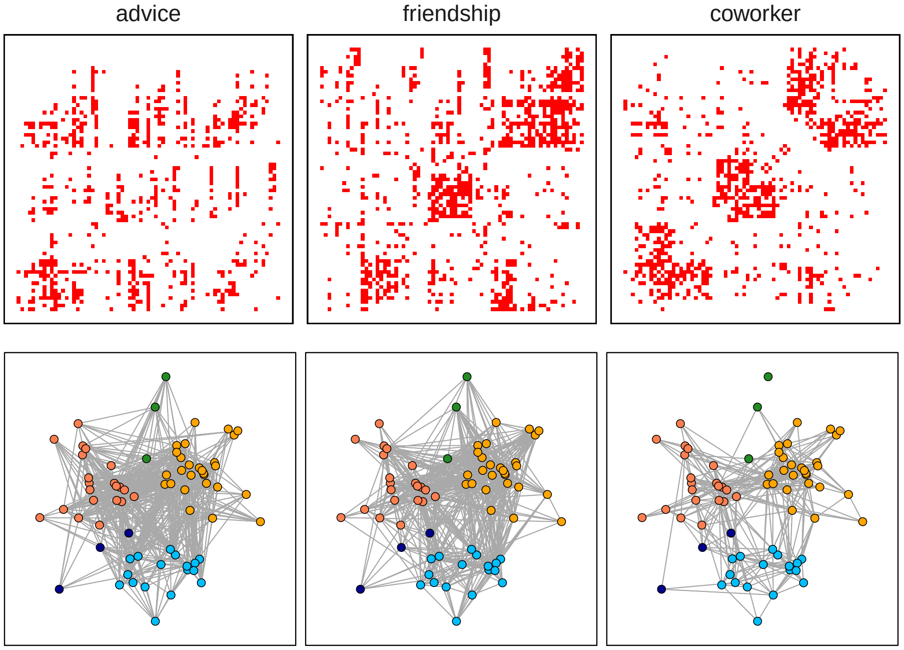


Figure 11: Upper: Heatmaps of the adjacency matrices (where lawyers are reordered according to the MaxPEAR estimate of $\mathbf{g}$). Lower: A visualization of the three layers based on the estimated $\mathbf{z}$ and color indicates the MaxPEAR estimate of $\mathbf{g}$.

	posterior mean	95% credible interval
θ_1/θ_2	0.3229	(0.2352, 0.4152)
θ_1/θ_3	0.2035	(0.1479, 0.2606)
θ_2/θ_3	0.6319	(0.5536, 0.7198)
b_1	-0.0986	(-0.1401, -0.0579)
b_2	0.0708	(0.0263, 0.1137)
b_3	0.133	(0.0854, 0.186)

Table 10: Posterior means and 95% credible intervals.

	advice	friendship	coworker
assortativity	0.2536	-0.1107	-0.1224

Table 11: Assortativity coefficients based on lawyer's age.

5 Discussion

This paper presents a latent position model that extends LPCM of Handcock et al. [2007] and SNSM of Ciminelli et al. [2019] to jointly model network data and the nodal attributes and perform model-based clustering. By jointly modeling the network and the attributes, we can describe how the attributes change over the network and explain how relations could be influenced by attributes. LPJMM also provides an extension to multi-layer network settings on the assumption that all layers share the same latent position structure but with different strengths of borrowing such latent structure information. We applied our method to four simulated networks and found LPJMM to give more satisfactory fits and is competitive in terms of goodness-of-fit and group detection compared with SNSM, LPCM, and CSBM. In addition, the advantage of LPJMM is more pronounced when there is missing data in the network and LPJMM is shown to be much more robust than the other models. LPJMM is also applied to a three-layer real network data set and we are able to draw reasonable conclusions from the modeling results.

We have suggested choosing the number of groups H to be the largest number of groups that one is willing to accept in the network because we have found that varying the number of groups has almost no impact on the model fit and prediction outcome as long as it is in a reasonable range. One could also fit the CSBM to the network first, and choose H based on its estimated number of groups. One problem we have not addressed in the paper is choosing the dimension of the latent space. This can be done by using Bayesian model selection like WAIC as in Sosa and Betancourt [2022].

Our model could be extended in several ways. Firstly, other extensions of our model to multi-layer settings could be considered. For example, Sosa and Betancourt [2022] assumed conditionally independent layer-specific latent positions, whereas MacDonald et al. [2022] assumed that the latent position of an actor in all layers is $(d_0 + d_1)$-dimensional, where the first d_0 components of the latent position are the same across all layers, and only the last d_1 components are layer-specific. Secondly, instead of assigning a user-specified number of groups H to the model, we could learn the number of groups by using a Bayesian nonparametric approach with a Dirichlet Process prior to model community memberships (see, e.g., Amini et al., 2019).

LPJMM could also be extended to leverage multivariate covariates. So far, we have limited ourselves to modeling univariate nodal attributes that are approximately Gaussian. For continuous nodal attributes with more than one dimension, we have used the first principal component from the principal component analysis. To take full advantage of high-dimensional nodal attributes, one could use multivariate spatial process modeling to replace Eq. (2). Other extensions of more sophisticated spatial modeling include spatiotemporal modeling of attributes for time-varying networks, which would help to describe changes in actors over time.

Data Availability Statement

The code and data used in this paper can be found online at `https://github.com/zjin3/LPJMM`.

Declaration of interest

This work was supported by the National Science Foundation [grant number 2051911].

Appendix

A Model Specifications for SNSM, LPCM and CSBM

Note that the original SNSM in Ciminelli et al. [2019] uses the logit link. In order to make a fair comparison, we also use the probit link in SNSM as in LPJMM. The model specification for SNSM used in this paper is given as follows:

$$
\begin{gathered}
y_{i,j} \mid \mathbf{z}, \mathbf{x}, a_\ell, b_\ell, \theta_\ell \overset{\text{ind}}{\sim} \mathrm{Ber}\big(\Phi(a + b|x_i - x_j| - \|z_i - z_j\|)\big)\,, \\
\mathbf{x} \mid \mathbf{z}, \beta, \sigma, \tau, \phi \sim \mathrm{N}_N(\beta \mathbf{1}_N,\, \sigma^2 M(\mathbf{z}, \phi) + \tau^2 I_N)\,,
\end{gathered}
$$

and the priors are set to be the same as the priors in LPJMM (if possible). To be specific,

$$
z_i \overset{\text{i.i.d.}}{\sim} \mathrm{N}_2(\mathbf{0},\, I_2)\,, \quad \beta \sim \mathrm{N}(0, 10^4)\,, \quad \sigma^2 \sim \mathrm{InvG}(2,1)\,, \quad \tau^2 \sim \mathrm{InvG}(2,1)\,, \quad \phi \sim \mathrm{U}(0,1)\,,
$$

and the priors on the parameters in the probit regression tier are given by:

$$
a \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,9)\,, \quad b \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,9)\,.
$$

SNSM in this paper is implemented using JAGS.

The model specification for LPCM (see Handcock et al., 2007) is given as the follows,

$$y_{i,j} \mid \mathbf{z}, \mathbf{x}, \beta_0, \beta_1 \overset{\text{ind}}{\sim} \text{Ber}\big(\text{logit}(\beta_0^\intercal x_{i,j} - \beta_1 \|z_i - z_j\|)\big)\,,$$
$$z_i \mid \boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\kappa} \overset{\text{i.i.d.}}{\sim} \sum_{h=1}^{5} \omega_h \text{N}_5(\mu_h,\, \kappa_h^2 I_K)\,,$$

and we use the default priors given in the `latentnet` package for prior specifications.

We first introduce several notations before presenting CSBM in Leger [2016]. Suppose there are Q groups in the network. Denote the $N \times Q$ group membership matrix as $\boldsymbol{Z} = \{Z_{iq}\}$, and $Z_{iq} = 1$ if actor i belongs to group q, $Z_{iq} = 0$ if otherwise. It is assumed that an actor can only belong to one group. The model specification for CSBM is given as follows,

$$y_{i,j} \mid Z_i, Z_j, \mathbf{x}, \beta \overset{\text{ind}}{\sim} \text{Ber}\big(\text{logit}(m_{q_i,q_j} + \beta^\intercal x_{i,j})\big)\,,$$

where Z_i is the i-th row of $\boldsymbol{Z}$, q_i is the group membership for actor i and the group effect $m_{q_i,q_j} \in \mathbb{R}$.

B Comparing Model Performances for Different Number of Groups

We conduct a comparison of LPJMM with different $H \in \{3, 4, \ldots, 9\}$ using the data set in Section 3.1. Table 12 presents ARI, and the results are similar for models that assume H to be equal to or larger than the true number of groups (which is 5 in this example). However, ARI for all three estimates is significantly smaller when the model assumes H to be smaller than 5. Also, notice that the estimated number of groups increases with H. Visualizations of how ARI and the estimated number of groups change over H are given in Fig. 12.

H	MaxPEAR	MinBinder	GreedyEPL
3	0.4067 (3)	0.4008 (5)	0.4321 (3)
4	0.4882 (3)	0.4977 (6)	0.6521 (4)
5	0.7374 (5)	0.7115 (11)	0.6635 (4)
6	0.7237 (6)	0.7442 (20)	0.7134 (4)
7	0.7449 (7)	0.6624 (25)	0.7313 (4)
8	0.7422 (8)	0.6674 (25)	0.7293 (8)
9	0.7056 (12)	0.7041 (25)	0.7043 (11)

Table 12: ARI of different estimates under LPJMM with different H. Numbers in the parentheses denote the numbers of estimated groups.

The goodness-of-fit test outlined in Section 3 is also carried out here to compare the means of several summary statistics, which are plotted in Fig. 13. As we can see from the plots, the model's fit is not affected by

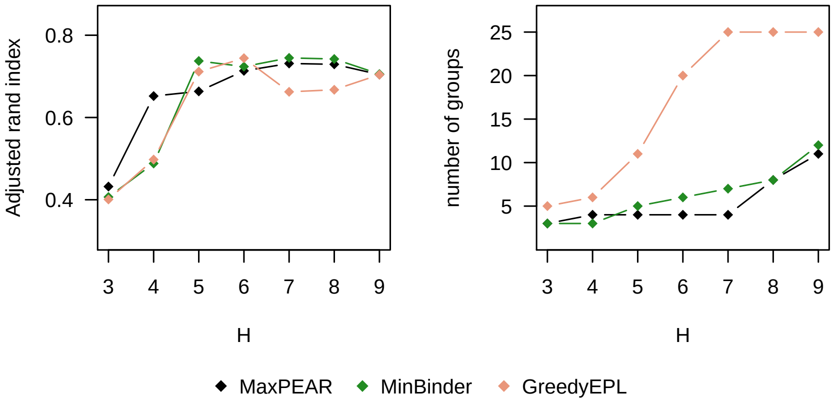
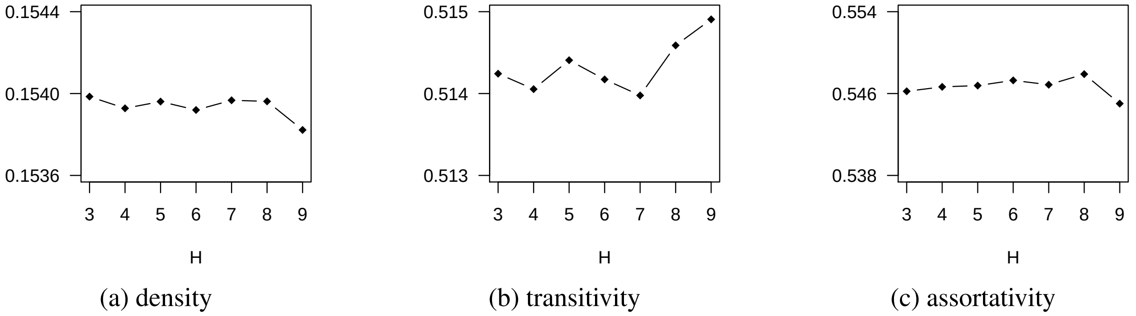


Figure 12: Left: ARI of the clustering estimates found by using the MaxPear, MinBinder, and GreedyEPL methods. Right: Estimated number of groups using the three methods.

the choice of H even for H smaller than the actual number of clusters in the network.



(a) density

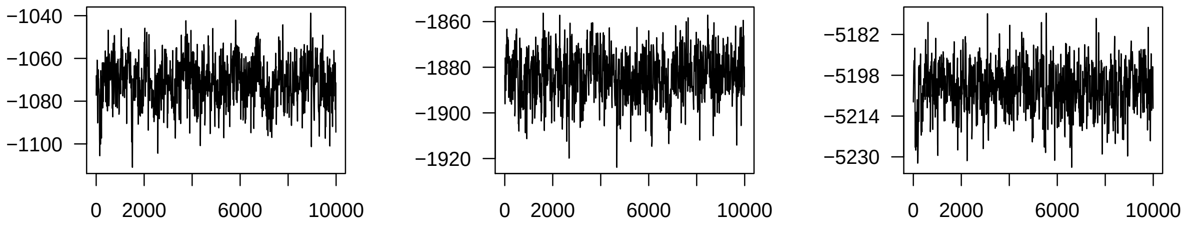
(b) transitivity

(c) assortativity

Figure 13: The means of summary statistics for different H.

C Traceplots of Log-likelihood

The traceplots of the log-likelihood (after thinning the Markov chain every 10 iterations) in simulation studies and real applications in Sections 3 and 4 are given in Fig. 14.



(a) network in Section 3.1

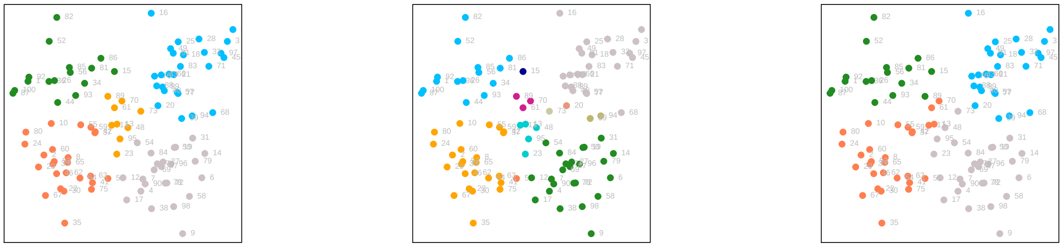
(b) network in Section 3.4

(c) Lazega network

Figure 14: Traceplots of the log-likelihood.

D Visualizations of Results from LPJMM and LPCM

Visualizations of the estimated latent positions and estimated group membership in Section 3.1 using the MaxPEAR, MinBinder, and GreedyEPL methods under LPJMM and LPCM are shown in Figs. 15 and 16 respectively.

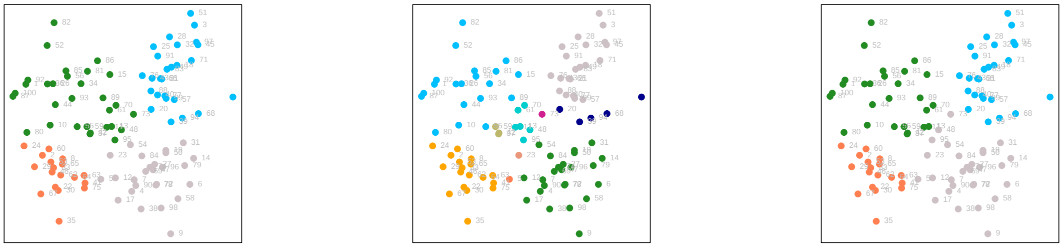


(a) MaxPEAR

(b) MinBinder

(c) GreedyEPL

Figure 15: Points are plotted based on the estimated latent position $\mathbf{z}$ and three estimated group memberships $\hat{\mathbf{g}}$ of LPJMM.



(a) MaxPEAR (LPCM)

(b) MinBinder (LPCM)

(c) GreedyEPL (LPCM)

Figure 16: Points are plotted based on estimated $\mathbf{z}$ and three estimated $\hat{\mathbf{g}}$ of LPCM.

References

J. H. Albert and S. Chib. Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association*, 88:669–679, 1993.

D. J. Aldous. *Exchangeability and related topics*. Springer, Berlin, Heidelberg, 1985.

A. A. Amini, M. S. Paez, and L. Lin. Hierarchical stochastic block model for community detection in multiplex networks. *arXiv:1904.05330*, 2019.

A. Athreya, D. E. Fishkind, M. Tang, C. E. Priebe, Y. Park, J. T. Vogelstein, K. Levin, V. Lyzinski, Y. Qin, and D. L. Sussman. Statistical inference on random dot product graphs: a survey. *Journal of Machine Learning Research*, 18:1–92, 2017.

S. Banerjee, B. Carlin, and A. Gelf. *Hierarchical Modeling and Analysis for Spatial Data*. CRC Press, 2nd edition, 2015.

J. Chiquet, S. Donnet, and P. Barbillon. `sbm`*: Stochastic Blockmodels*, 2022. URL `https://CRAN.R-project.org/package=sbm`. R package version 0.4.4.

J. T. Ciminelli, T. Love, and T. T. Wu. Social network spatial model. *Spatial Statistics*, 29:129–144, 2019.

J. Diebolt and C. P. Robert. Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society. Series B*, 56:363–375, 1994.

D. Durante and D. Dunson. Bayesian inference and testing of group differences in brain networks. *Bayesian Analysis*, 13:29–58, 2018.

S. D'Angelo, T. B. Murphy, and M. Alfò. Latent space modelling of multidimensional networks with application to the exchange of votes in Eurovision song contest. *The Annals of Applied Statistics*, 13:900–930, 2019.

S. D'Angelo, M. Alfò, and M. Fop. Model-based clustering for multidimensional social networks. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 00:1–27, 2023.

P. Erdös and A. Rényi. On random graphs. I. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.

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

O. Frank and D. Strauss. Markov graphs. *Journal of the American Statistical Association*, 81:832–842, 1986.

A. Fritsch. `mcclust`*: Process an MCMC Sample of Clusterings*, 2022. URL `https://CRAN.R-project.org/package=mcclust`. R package version 1.0.1.

A. Fritsch and K. Ickstadt. Improved criteria for clustering based on the posterior similarity matrix. *Bayesian Analysis*, 4:367–391, 2009.

I. Gollini and T. B. Murphy. Joint modeling of multiple network views. *Journal of Computational and Graphical Statistics*, 25:246–265, 2016.

S. Guha and A. Rodriguez. Bayesian regression with undirected network predictors with an application to brain connectome data. *Journal of the American Statistical Association*, 116(534):581–593, 2021.

M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A*, 170:301–354, 2007.

P. Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in Neural Information Processing Systems*, pages 657–664, 2007.

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

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

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

D. N. Hoover. Row-column exchangeability and a generalized model for probability. *Exchangeability in probability and statistics (Rome, 1981)*, pages 281–291, 1982.

M. Kim and J. Leskovec. Multiplicative attribute graph model of real-world networks. *Internet Mathematics*, 8:113–160, 2012.

E. D. Kolaczyk and G. Csárdi. *Statistical Analysis of Network Data with R*. Springer, 2nd edition, 2020.

P. N. Krivitsky and M. S. Handcock. `latentnet`*: Latent Position and Cluster Models for Statistical Networks*. The Statnet Project (`https://statnet.org`), 2022. URL `https://CRAN.R-project.org/package=latentnet`. R package version 2.10.6.

P. N. Krivitsky, M. S. Handcock, A. E. Raftery, and P. D. Hoff. Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social Networks*, 31:204–213, 2009.

J. W. Lau and P. J. Green. Bayesian model based clustering procedures. *Journal of Computational and Graphical Statistics*, 16:526–558, 2007.

E. Lazega. *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership*. Oxford University Press, 2001.

C. Lee and D. J. Wilkinson. A review of stochastic block models and extensions for graph clustering. *Applied Network Science*, 4:1–50, 2019.

J.-B. Leger. Blockmodels: A R-package for estimating in Latent Block Model and Stochastic Block Model, with various probability functions, with or without covariates. *arXiv:1602.07587*, 2016.

Crystal D Linkletter. *Spatial process models for social network analysis*. PhD thesis, Citeseer, 2007.

P. W. MacDonald, E. Levina, and J. Zhu. Latent space models for multiplex networks with shared structure. *arXiv:2012.14409*, 2020.

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

C. Matias and S. Robin. Modeling heterogeneity in random graphs through latent space models: a selective review. *ESAIM: Proceedings and Surveys*, 47:55–74, 2014.

S. Minhas, P. D. Hoff, and M. D. Ward. Inferential approaches for network analysis: AMEN for latent factor models. *Political Analysis*, 27:208–222, 2019.

K. Nowicki and T. A. B. Snijders. Estimation and prediction for stochastic block structures. *Journal of the American Statistical Association*, 96:1077–1087, 2001.

M. Plummer. `rjags`*: Bayesian Graphical Models using MCMC*, 2022. URL `https://CRAN.R-project.org/package=rjags`. R package version 4-13.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2021. URL `https://www.R-project.org/`.

R. Rastelli. `GreedyEPL`*: Greedy Expected Posterior Loss*, 2021. URL `https://CRAN.R-project.org/package=GreedyEPL`. R package version 1.2.

R. Rastelli and N. Friel. Optimal Bayesian estimators for latent variable cluster models. *Statistics and Computing*, 28:1169–1186, 2018.

M. Salter-Townshend and T. H. McCormick. Latent space models for multiview network data. *The Annals of Applied Statistics*, 11:1217–1244, 2017.

M. Schweinberger and A. B. Snijders. Settings in social networks: a measurement model. *Sociological Methodology*, 33:307–341, 2003.

J. Sosa. A review on latent space models for social networks. *Revista Colombiana de Estadistica*, 44: 171–200, 2021.

J. Sosa and B. Betancourt. A latent space model for multilayer network data. *Computational Statistics & Data Analysis*, 169:107432, 2022.

L. Wang, Z. Zhang, and D. Dunson. Common and individual structure of brain networks. *Annals of Applied Statistics*, 13:85–112, 2019.

Y. J. Wang and G. Y. Wong. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82:8–19, 1987.