A Poisson Gamma Probabilistic Model for Latent Node-group Memberships in Dynamic Networks

Sikun Yang, Heinz Koeppl

Department of Electrical Engineering and Information Technology, Technische Universität Darmstadt
64283 Darmstadt, Germany
{sikun.yang, heinz.koeppl}@bcs.tu-darmstadt.de

Abstract

We present a probabilistic model for learning from dynamic relational data, wherein the observed interactions among networked nodes are modeled via the Bernoulli Poisson link function, and the underlying network structure are characterized by *nonnegative* latent node-group memberships, which are assumed to be gamma distributed. The latent memberships evolve according to Markov processes. The optimal number of latent groups can be determined by data itself. The computational complexity of our method scales with the number of non-zero links, which makes it scalable to large sparse dynamic relational data. We present batch and online Gibbs sampling algorithms to perform model inference. Finally, we demonstrate the model's performance on both synthetic and real-world datasets compared to state-of-the-art methods.

Introduction

Considerable work has been done on the analysis of static networks in terms of community detection or link prediction (Hoff et al. 2001; Kemp et al. 2006; Airoldi et al. 2008). However, due to the temporal evolution of nodes (e.g. individuals), their role within a network can change and hence observed links among nodes may appear or disappear over time (Mucha et al. 2010; Goldenberg et al. 2010). Given such dynamic network data, one may be interested in understanding the temporal evolution of groups in terms of their size and node-group memberships and in predicting missing or future unobserved links based on historical records.

A dynamic network of N nodes can be represented as a sequence of adjacency matrices $\mathbf{b}^{(t)} \in \{0,1\}^{N \times N}$, $t = 1, \ldots, T$, where $b_{mn}^{(t)} = 1$ indicates the presence of a link between node m and n at time point t and $b_{mn}^{(t)} = 0$ otherwise. For the sake of clarity we focus on undirected and unweighted networks but the presented method can be extended to weighted networks via the compound Poisson distribution (Basbug and Engelhardt 2016) and to multi-relational networks (Bordes et al. 2011; Nickel et al. 2016).

Many of the current probabilistic methods for dynamic networks map observed binary edge-variables (either links or non-links) to latent Gaussian random variables via the

logistic or probit function (Durante et al. 2014). The time-complexity of these approaches often scales quadratically with the number of nodes, i.e., $\mathcal{O}(N^2)$. This will become infeasible for large networks and will become especially inefficient for sparse networks. In this work we leverage the Bernoulli-Poisson link function (Dunson and Herring 2005; Zhou 2015; Caron and Fox 2015) to map an observed binary edge to a latent Poisson count random variable, which leads to a computational cost that only scales with the number of non-zero edges. As large, real-world networks are usually very sparse, the proposed method yields significant speed-up and enables the analysis of larger dynamic networks. We allow for a time-varying nonnegative degree of membership of a node to a group. We realize this by constructing a gamma Markov chain to capture the time evolution of those latent membership variables. Inspired by recent advances in data augmentation and marginalization techniques (Zhou and Carin 2015), we develop an easy-to-implement efficient Gibbs sampling algorithm for model inference. We also present an online Gibbs sampling algorithm that can process data in mini-batches and thus readily scales to massive sparse dynamic networks. The algorithms performs favorably on standard datasets when compared to state-of-the-art methods (Foulds et al. 2011; Acharya et al. 2015).

Dynamic Poisson Gamma Membership Model

In the proposed model, each node n is characterized by a time-dependent latent membership variable $\phi_{nk}^{(t)}$ that determines its interactions or involvement in group k at the t-th snapshot of the dynamic networks. This latent node-group membership is modeled by a gamma random variable and is, thus, naturally *nonnegative real-valued*. This is contrast to *multi-group memberships* models (or latent feature models) (Foulds et al. 2011; Heaukulani et al. 2013; Kim et al. 2013) where each node-group membership is represented by a *binary* latent feature vector. These models assume that each node either associates to one group or not – simply by a binary feature. The proposed model on the other hand can characterize how strongly each node associates with multiple groups.

Dynamics of latent node-group memberships. For dynamic networks, the latent node-group membership $\phi_{nk}^{(t)}$ can evolve over time to interpret the interaction dynamics among the nodes. For example, latent group k could mean "play soccer" and $\phi_{nk}^{(t)}$ could mean how frequently person n plays soccer or how strongly person n likes playing soccer. The person's degree of association to this group could be increasing over time due to, for instance, increased interaction with professional soccer players, or decreasing over time as a consequence of sickness. Hence, in order to model the temporal evolution of the latent node-group memberships, we assume the individual memberships to form a gamma Markov chain. More specifically, $\phi_{nk}^{(t)}$ is drawn from a gamma distribution, whose shape parameter is the latent membership at the previous time

$$\phi_{nk}^{(t)} \sim \mathrm{Gam}(\phi_{nk}^{(t-1)}/\tau, 1/\tau), \qquad \text{for } t = 1, \ldots, T$$
$$\phi_{nk}^{(0)} \sim \mathrm{Gam}(g_0, 1/h_0),$$

where the parameter τ controls the variance without affecting the mean, i.e., $\mathsf{E}[\phi_{nk}^{(t)} \mid \phi_{nk}^{(t-1)}, \tau] = \phi_{nk}^{(t-1)}$.

Model of latent groups. We characterize the interactions or correlations among latent groups by a matrix $\mathbf{\Lambda}$ of size $K \times K$, where $\lambda_{kk'}$ relates to the probability of there being a link between node n affiliated to group k and node m affiliated to group k'. Specifically, we assume the latent groups to be generated by the following hierarchical process: we first generate a separate weight for each group as

$$r_k \sim \mathrm{Gam}(\gamma_0/K, 1/c_0), \tag{1}$$

and then generate the inter-group interaction weight $\lambda_{kk'}$ and intra-group weight λ_{kk} as

$$\lambda_{kk'} \sim \begin{cases} \mathrm{Gam}(\xi r_k, 1/\beta), & \text{if } k = k' \\ \mathrm{Gam}(r_k r_{k'}, 1/\beta), & \text{otherwise} \end{cases} \tag{2}$$

where $\xi \in \mathrm{R}_{>0}$ and $\beta \in \mathrm{R}_{>0}$. The reasonable number of latent groups can be inferred from dynamic relational data itself by the *shrinkage* mechanism of our model. More specifically, for fixed γ_0, the redundant groups will effectively be shrunk as many of the groups weights tend to be small for increasing K. Thus, the interaction weights $\lambda_{kk'}$ between the redundant group k and all the other groups k', and all the node-memberships to group k will be shrunk accordingly. In practice, the intra-group weight λ_{kk} would tend to almost zero if $\lambda_{kk} \sim \mathrm{Gam}(r_k^2, 1/\beta)$ for small r_k, and the corresponding groups will disappear inevitably. Hence, we use a separate variable ξ to avoid overly shrinking of small groups with less interactions with other groups. As γ_0 has a large effect on the number of the latent groups, we do not treat it as a fixed parameter but place a gamma prior over it, i.e., $\gamma_0 \sim \mathrm{Gam}(1, 1)$. Given the latent node-group membership $\phi_{nk}^{(t)}$ and the interaction weights $\lambda_{kk'}^{(t)}$ among groups, the probability of there being a link between node m and n is given by

$$b_{mn}^{(t)} \sim \mathrm{Bern}\left(1 - \exp\left\{-\sum_{k=1}^{K}\sum_{k'=1}^{K} \lambda_{kk'}\phi_{nk}^{(t)}\phi_{mk'}^{(t)}\right\}\right). \tag{3}$$
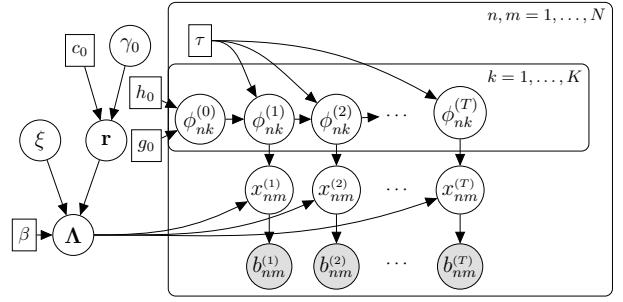

Figure 1: The graphical model of the Poisson gamma latent membership model; auxiliary variables introduced for inference are not shown.

Interestingly, we can also generate $b_{mn}^{(t)}$ by truncating a latent count random variable $x_{mn}^{(t)}$ at 1, where $x_{mn}^{(t)}$ can be seen as the integer-valued *weight* for node m and n and can be interpreted as the number of times the two nodes interacted. More specifically, $b_{mn}^{(t)}$ can be drawn as

$$b_{mn}^{(t)} = \mathbb{1}(x_{mn}^{(t)} \geq 1), \tag{4}$$

$$x_{mn}^{(t)} \sim \mathrm{Po}\left(\sum_{k=1}^{K}\sum_{k'=1}^{K} \lambda_{kk'}\phi_{nk}^{(t)}\phi_{mk'}^{(t)}\right), \tag{5}$$

where Po indicates the Poisson distribution. We can obtain Eq. (3) by marginalizing out the latent count $x_{mn}^{(t)}$ from the above expression. The conditional distribution of the latent count $x_{mn}^{(t)}$ can then be written as

$$(x_{mn}^{(t)} \mid b_{mn}^{(t)}, \mathbf{\Phi}, \mathbf{\Lambda}) \sim b_{mn}^{(t)}\mathrm{Po}_+\left(\sum_{k=1}^{K}\sum_{k'=1}^{K} \lambda_{kk'}\phi_{nk}^{(t)}\phi_{mk'}^{(t)}\right),$$

where $x \sim \mathrm{Po}_+(\sigma)$ is the zero-truncated Poisson distribution with probabilisty mass function (PMF) $f_X(x|\sigma) = (1 - e^{-\sigma})^{-1}\sigma^x e^{-\sigma}/x!$ and $x \in \mathbb{Z}_{>0}$ and $\mathbf{\Phi}$ denotes the set of all node-group membership variables. The usefulness of this construction for $b_{nm}^{(t)}$ will become clear in the inference section. We note that the latent count $x_{mn}^{(t)}$ only needs to be sampled for $b_{mn}^{(t)} = 1$, using rejection sampling detailed in (Zhou 2015). To this end the proposed hierarchical generative model is as follows.

$$\phi_{nk}^{(t)} \sim \mathrm{Gam}(\phi_{nk}^{(t-1)}/\tau, 1/\tau), \ \text{for } t = 1, \ldots, T$$
$$\phi_{nk}^{(0)} \sim \mathrm{Gam}(g_0, 1/h_0),$$
$$r_k \sim \mathrm{Gam}(\gamma_0/K, 1/c_0),$$
$$\lambda_{kk'} \sim \begin{cases} \mathrm{Gam}(\xi r_k, 1/\beta), & \text{if } k = k' \\ \mathrm{Gam}(r_k r_{k'}, 1/\beta), & \text{otherwise} \end{cases}$$
$$x_{mn}^{(t)} \sim \mathrm{Po}\left(\sum_{k=1}^{K}\sum_{k'=1}^{K} \lambda_{kk'}\phi_{nk}^{(t)}\phi_{mk'}^{(t)}\right),$$
$$b_{mn}^{(t)} = \mathbb{1}(x_{mn}^{(t)} \geq 1).$$

For our model's hyperparameters, we draw c_0, ξ and β from $\mathrm{Gam}(0.1, 1/0.1)$. The graphical model is shown in Fig. 1.

Related Work

Approaches to analyze dynamic networks range from non-Bayesian methods such as the exponential random graph models (Robins et al. 2007) or matrix and tensor factorization based methods (Dunlavy et al. 2011) to Bayesian latent variable models (Ishiguro et al. 2010; Foulds et al. 2011; Ho et al. 2011; Heaukulani et al. 2013; Kim et al. 2013). Our work falls into the latter class and we hence confine ourselves to discuss it's relation to this class.

Dynamic extensions of *mixed membership models*, where each node is assigned to a set of latent groups represented by multinomial distribution, have been developed (Fu et al. 2009; Ho et al. 2011). One limitation of mixed membership models is that if the probability that node i associates to group k is increased, the probability that node i associates to group k' has to be decreased. The multi-group memberships models use a binary latent feature vector to characterize each node's multi-group memberships. In multi-group memberships models, a node's membership to one group does not limit its memberships to other groups. However, differences in the degree associations of a node to different groups cannot be captured by such models (Foulds et al. 2011; Heaukulani et al. 2013; Kim et al. 2013). One possible extension is to introduce a Gaussian distributed random variables to characterize how strongly each node is associated to different groups as previously done for latent factor models (Rai and Daume III 2008). Such approaches where membership variables evolve according to linear dynamical systems (Xing et al. 2010) can exploit the rich and efficient toolset for inference, such as Kalman filtering. However, the resulting signed-valued latent features lack an intuitive interpretation, e.g., in terms of degree of membership to a group.

In contrast to these approaches, our model is based on a bilinear Poisson factor model (Zhou 2015), where each node's memberships to groups are represented by a nonnegative real-valued memberships variable. The model does not only allow each node to be associated with multiple groups but also captures the degree at which each node is associated to a group. It means that our model combines the advantages of both mixed membership and multi-group memberships models. We exploit recent data augmentation technique (Zhou and Carin 2015), to construct a sampling scheme for the time evolution of the nonnegative latent features. Related to our work is the dynamic gamma process Poisson factorization model (D-GPPF) (Acharya et al. 2015) where the underlying groups' structure can evolve over time but each node-group membership is static. This is in constrast to our approach where the node's memberships evolve over time. We note that the gamma Markov chain used by our method and by D-GPPF is motivated by the augmentation techniques in (Zhou and Carin 2015). In the experiment section we compare our model to (1) the hierarchical gamma process edge partition model (HGP-EPM) (Zhou 2015), which is the static counterpart of our model, (2) the dynamic relational infinite feature model (DRIFT) (Foulds et al. 2011) which uses binary latent features to represent the node-group memberships, and characterizes the temporal dependences of latent features via a hidden Markov process and (3) the D-GPPF model.

Inference

We present a Gibbs sampling procedure to draw samples of $\{\phi_{nk}^{(t)}, \lambda_{kk'}, r_k, \xi, \gamma_0, \beta, c_0\}$ from their posterior distribution given the observed dynamic relational data and the hyperparameters $\{\tau, g_0, h_0\}$. In order to circumvent the technical challenges of drawing samples from the gamma Markov chain which does not yield closed-form posterior, we make use of the idea of data augmentation and marginalization technique and of the gamma-Poisson conjugacy to derive a closed-form update.

Notation. When expressing the full conditionals for Gibbs sampling we will use the shorthand "−" to denote all other variables or equivalently those in the Markov blanket for the respective variable according to Fig. 1. We use "·" as a index summation shorthand, e.g., $x_{\cdot j} = \sum_i x_{ij}$.

We repeatedly exploit the following three results (Quenouille 1949; Dunson and Herring 2005; Zhou and Carin 2015) to derive the conditional distributions used in our sampling algorithm.

Result 1. A negative binomially (NB) distributed random variable $y \sim \text{NB}(r, p)$ can be generated from a gamma mixed Poisson distribution as, i.e., $y \sim \text{Po}(\lambda)$ and $\lambda \sim \text{Gam}(r, \frac{p}{1-p})$, as seen by marginalizing over λ.

Result 2. The Poisson-logarithmic bivariate distributed variable (y, l) (Zhou and Carin 2015) with $y \sim \text{NB}(y; r, p)$ and a Chinese restaurant table (CRT) (Pitman 2006) distributed variables $l \sim \text{CRT}(l; y, r)$, can equivalently be expressed as a sum-logarithmic (SumLog) and Poisson variable, i.e., $y \sim \sum_{s=1}^{l} u_s$ with $u_s \sim \text{Logarithmic}(p)$ and $l \sim \text{Po}(-r \log(1 - p))$.

Result 3. Let $y_{\cdot} = \sum_{n=1}^{N} y_n$, where $y_n \sim \text{Po}(\lambda_n)$ are independently drawn from a Poisson distribution with rate λ_n, then according to the Poisson-multinomial equivalence (Dunson and Herring 2005), we have $(y_1, \dots, y_N) \sim \text{Mult}(y_{\cdot}; \frac{\lambda_1}{\sum_n \lambda_n}, \dots, \frac{\lambda_N}{\sum_n \lambda_n})$ and $y_{\cdot} \sim \text{Po}(\sum_n \lambda_n)$.

Gibbs sampling

Sampling latent counts $x_{mn}^{(t)}$. We sample a latent count for each time dependent observed edge $b_{mn}^{(t)}$ as

$$(x_{mn}^{(t)} \mid -) \sim b_{mn}^{(t)} \text{Po}_+ \left(\sum_{k=1}^{K} \sum_{k'=1}^{K} \lambda_{kk'} \phi_{nk}^{(t)} \phi_{mk'}^{(t)} \right). \quad (6)$$

Sampling individual counts $x_{mkk'n}^{(t)}$. We can partition the latent count $x_{mn}^{(t)} \sim \text{Po}(\sum_{k,k'=1}^{K} \lambda_{kk'} \phi_{nk}^{(t)} \phi_{mk'}^{(t)})$ using the Poisson additive property as $x_{mn}^{(t)} = \sum_{k,k'}^{K} x_{mkk'n}^{(t)}$, where $x_{mkk'n}^{(t)} \sim \text{Po}(\lambda_{kk'} \phi_{nk}^{(t)} \phi_{mk'}^{(t)})$. Then, via the Poisson-multinomial equivalence, we sample the latent count $x_{mkk'n}^{(t)}$ as

$$(x_{mkk'n}^{(t)} \mid -) \sim \text{Mult} \left(x_{mn}^{(t)}; \frac{\lambda_{kk'} \phi_{nk}^{(t)} \phi_{mk'}^{(t)}}{\sum_{k=1}^{K} \sum_{k'=1}^{K} \lambda_{kk'} \phi_{nk}^{(t)} \phi_{mk'}^{(t)}} \right). \quad (7)$$

Sampling group weights r_k**.** Via the Poisson additive property, we have

$$x_{\cdot kk'\cdot}^{(\cdot)} \sim \text{Po}(\lambda_{kk'}\theta_{kk'}), \tag{8}$$

where we defined $x_{\cdot kk'\cdot}^{(\cdot)} \equiv \sum_t \sum_{m,n\neq m} x_{mkk'n}^{(t)}$ and $\theta_{kk'} \equiv \sum_t \sum_n \sum_{m\neq n} \phi_{nk}^{(t)}\phi_{mk'}^{(t)}$. We can marginalize out $\lambda_{kk'}$ from Eq. (8) and (2) using the gamma-Poisson conjugacy, which gives

$$x_{\cdot kk'\cdot}^{(\cdot)} \sim \text{NB}(r_k \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}}, \tilde{p}_{kk'}),$$

where $\tilde{p}_{kk'} = \frac{\theta_{kk'}}{\theta_{kk'}+\beta}$ and $\delta_{kk'}$ denotes the Kronecker delta. According to Result 2, we introduce the auxiliary variables as

$$l_{kk'} \sim \text{CRT}(x_{\cdot kk'\cdot}^{(\cdot)}, r_k \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}}). \tag{9}$$

We then re-express the bivariate distribution over $x_{\cdot kk'\cdot}^{(\cdot)}$ and $l_{kk'}$ as

$$x_{\cdot kk'\cdot}^{(\cdot)} \sim \text{SumLog}(l_{kk'}, r_k \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}}),$$
$$l_{kk'} \sim \text{Po}[-r_k \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}} \log(1-\tilde{p}_{kk'})]. \tag{10}$$

Using Eq. (1) and (10), via the gamma-Poisson conjugacy, we obtain the conditional distribution of r_k as

$$(r_k \mid -) \sim \text{Gam}\Big[\frac{\gamma_0}{K} + \sum_{k'} l_{kk'},$$
$$\frac{1}{c_0 - \sum_{k'} \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}} \log(1-\tilde{p}_{kk'})}\Big]. \tag{11}$$

Sampling intra-group weight ξ**.** We resample the auxiliary variables l_{kk} using Eq. (9), and then exploit the gamma-Poisson conjugacy to sample ξ as

$$(\xi \mid -) \sim \text{Gam}\Big[0.1 + \sum_k l_{kk}, \frac{1}{0.1 - \sum_k r_k \log(1-\tilde{p}_{kk})}\Big]. \tag{12}$$

Sampling inter-group weights $\lambda_{kk'}$**.** We sample $\lambda_{kk'}$ from its conditional obtained from Eq. (2) and (8) via the gamma-Poisson conjugacy as

$$(\lambda_{kk'} \mid -) \sim \text{Gam}\Big[x_{\cdot kk'\cdot}^{(\cdot)} + r_k \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}}, \frac{1}{(\beta+\theta_{kk'})})\Big]. \tag{13}$$

Sampling hyperparameter γ_0**.** Using Eq. (10) and the Poisson additive property, we have $l_{k\cdot} \equiv \sum_{k'} l_{kk'}$ as

$$l_{k\cdot} \sim \text{Po}[-r_k \sum_{k'} \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}} \log(1-\tilde{p}_{kk'})].$$

Marginalizing out r_k using the gamma-Poisson conjugacy, we have

$$l_{k\cdot} \sim \text{NB}(\gamma_0/K, \hat{p}_k),$$

where $\hat{p}_k = \frac{\sum_{k'} \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}} \log(1-\tilde{p}_{kk'})}{c_0 - \sum_{k'} \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}} \log(1-\tilde{p}_{kk'})}$. We introduce the auxiliary variables $\tilde{l}_k \sim \text{CRT}(l_{k\cdot}, \gamma_0/K)$ and re-express the bivariate distribution over $l_{k\cdot}$ and $\tilde{l}_k$ as

$$l_{k\cdot} \sim \text{SumLog}(\tilde{l}_k, \hat{p}_k),$$
$$\tilde{l}_k \sim \text{Po}\Big[-\frac{\gamma_0}{K}\log(1-\hat{p}_k)\Big]. \tag{14}$$

Using Eq. (14), we can then sample γ_0 via the gamma-Poisson conjugacy as

$$(\gamma_0 \mid -) \sim \text{Gam}\Big[1 + \sum_k \tilde{l}_k, \frac{1}{1 - \frac{1}{K}\sum_k \log(1-\hat{p}_k)}\Big]. \tag{15}$$

Sampling latent memberships $\phi_{nk}^{(t)}$**.** Since the latent memberships $\phi_{nk}^{(t)}$ evolve over time according to our Markovian construction, the backward and forward information need to be incorporated into the updates of $\phi_{nk}^{(t)}$. We start from time slice $t = T$,

$$x_{nk\cdot\cdot}^{(T)} \sim \text{Po}(\phi_{nk}^{(T)}\omega_{nk}^{(T)}), \qquad \phi_{nk}^{(T)} \sim \text{Gam}(\phi_{nk}^{(T-1)}/\tau, 1/\tau),$$

where

$$x_{nk\cdot\cdot}^{(t)} \equiv \sum_{m\neq n, k'} x_{mkk'n}^{(t)}, \qquad \omega_{nk}^{(t)} \equiv \sum_{m\neq n, k'} \phi_{mk'}^{(t)}\lambda_{kk'}.$$

Via the gamma-Poisson conjugacy, we have

$$(\phi_{nk}^{(T)} \mid -) \sim \text{Gam}\Big[\phi_{nk}^{(T-1)}/\tau + x_{nk\cdot\cdot}^{(T)}, 1/(\tau + \omega_{nk}^{(T)})\Big]. \tag{16}$$

Marginalizing out $\phi_{nk}^{(T)}$ yields

$$x_{nk\cdot\cdot}^{(T)} \sim \text{NB}(\phi_{nk}^{(T-1)}/\tau, \varrho_{nk}^{(T)}), \tag{17}$$

where $\varrho_{nk}^{(T)} = \frac{\omega_{nk}^{(T)}}{\tau+\omega_{nk}^{(T)}}$. According to Result 2, the NB distribution can be augmented with an auxiliary variable as

$$y_{nk}^{(T)} \sim \text{CRT}(x_{nk\cdot\cdot}^{(T)}, \phi_{nk}^{(T-1)}/\tau). \tag{18}$$

We re-express the bivariate distribution over $x_{nk\cdot\cdot}^{(T)}$ and $y_{nk}^{(T)}$ as

$$x_{nk\cdot\cdot}^{(T)} \sim \text{SumLog}(y_{nk}^{(T)}, \varrho_{nk}^{(T)}),$$
$$y_{nk}^{(T)} \sim \text{Po}\Big[-\frac{\phi_{nk}^{(T-1)}}{\tau}\log(1-\varrho_{nk}^{(T)})\Big]. \tag{19}$$

where

$$\varrho_{nk}^{(t)} = \frac{\omega_{nk}^{(t)} - \frac{1}{\tau}\log(1-\varrho_{nk}^{(t+1)})}{\tau + \omega_{nk}^{(t)} - \frac{1}{\tau}\log(1-\varrho_{nk}^{(t+1)})}. \tag{20}$$

Given $x_{nk\cdot\cdot}^{(T-1)} \sim \text{Po}(\phi_{nk}^{(T-1)}\omega_{nk}^{(T-1)})$, via the Poisson additive property, we have

$$y_{nk}^{(T)} + x_{nk\cdot\cdot}^{(T-1)} \sim \text{Po}\Big(\phi_{nk}^{(T-1)}\Big[\omega_{nk}^{(T-1)} - \frac{1}{\tau}\log(1-\varrho_{nk}^{(T)})\Big]\Big). \tag{21}$$

Combing the likelihood in Eq. (21) with the gamma prior placed on $\phi_{nk}^{(T-1)}$, we immediately have the conditional distribution of $\phi_{nk}^{(T-1)}$ via the gamma-Poisson conjugacy as

$$(\phi_{nk}^{(T-1)} \mid -) \sim \text{Gam}\Big[\phi_{nk}^{(T-2)}/\tau + y_{nk}^{(T)} + x_{nk\cdot\cdot}^{(T-1)}, \tag{22}$$
$$\frac{1}{\tau + \omega_{nk}^{(T-1)} - \frac{1}{\tau}\log(1-\varrho_{nk}^{(T)})}\Big].$$

Here, $y_{nk}^{(T)}$ can be considered as the *backward* information passed from $t = T$ to $T - 1$. Recursively, we augment $\phi_{nk}^{(t)}$ at each time slice with an auxiliary variable $y_{nk}^{(t)}$ as

$$y_{nk}^{(t+1)} + x_{nk\cdot\cdot}^{(t)} \sim \text{NB}(\phi_{nk}^{(t-1)}/\tau, \varrho_{nk}^{(t)}),$$
$$y_{nk}^{(t)} \sim \text{CRT}(x_{nk\cdot\cdot}^{(t)} + y_{nk}^{(t+1)}, \phi_{nk}^{(t-1)}/\tau), \quad (23)$$

where the NB distribution over $y_{nk}^{(t+1)} + x_{nk\cdot\cdot}^{(t)}$ is obtained via the Poisson additive property and gamma-Poisson conjugacy with $x_{nk\cdot\cdot}^{(t)} \sim \text{Po}(\phi_{nk}^{(t)}\omega_{nk}^{(t)})$. Repeatedly using Result 2, we have

$$y_{nk}^{(t+1)} + x_{nk\cdot\cdot}^{(t)} \sim \text{SumLog}(y_{nk}^{(t)}, \varrho_{nk}^{(t)}),$$
$$y_{nk}^{(t)} \sim \text{Po}\Big[- \frac{\phi_{nk}^{(t-1)}}{\tau} \log(1 - \varrho_{nk}^{(t)}) \Big].$$

By repeatedly exploiting the Poisson additive property and gamma-Poisson conjugacy, we obtain

$$(\phi_{nk}^{(t-1)} \mid -) \sim \text{Gam}\Big[y_{nk}^{(t)} + \phi_{nk}^{(t-2)}/\tau + x_{nk\cdot\cdot}^{(t-1)}, \quad (24)$$
$$\frac{1}{\tau + \omega_{nk}^{(t-1)} - \frac{1}{\tau}\log(1 - \varrho_{nk}^{(t)})} \Big].$$

We sample the auxiliary variables $y_{nk}^{(t)}$ and update $\varrho_{nk}^{(t)}$ recursively from $t = T$ to $t = 1$, which can be considered as the *backward filtering* step. Then, in the *forward* pass we sample $\phi_{nk}^{(t)}$ from $t = 1$ to $t = T$.

Sampling hyperparameters. Via the gamma-gamma conjugacy, we sample c_0 and β as

$$(c_0 \mid -) \sim \text{Gam}\Big[0.1 + \gamma_0, 1/(0.1 + \sum_k r_k) \Big], \quad (25)$$

$$(\beta \mid -) \sim \text{Gam}\Big[0.1 + \sum_{k,k'} r_k \xi^{\delta_{kk'}} r_{k'}^{1-\delta_{kk'}}, \frac{1}{0.1 + \sum_{k,k'} \lambda_{kk'}} \Big]$$

Algorithm 1 summarizes the full procedure.

Online Gibbs sampling

To make our model applicable to large-scale dynamic networks, we propose an online Gibbs sampling algorithm based on the recent developed Bayesian conditional density filtering (BCDF) (Guhaniyogi et al. 2014), which has been adapted for Poisson tensor factorization (Hu et al. 2015) recently. The main idea of BCDF is to partition the data into small mini-batches, and then to perform inference by updating the sufficient statistics using each mini-batch in each iteration. Specifically, the sufficient statistics used in our model are the latent count numbers. We use J and J^i to denote the indices of the entire data and the mini-batch in i-th iteration respectively. We define the quantities updated with the mini-batch in i-th iteration as:

$$x_{mk\cdot\cdot}^{(t)i} = \frac{|J|}{|J^i|} \sum_{\substack{k', n \neq m, \\ m,n \in J^i}} x_{mkk'n}^{(t)}, \quad x_{\cdot kk'}^{i} = \frac{|J|}{|J^i|} \sum_{\substack{t, m, n \neq m, \\ m,n \in J^i}} x_{mkk'n}^{(t)}.$$

Algorithm 1: Batch Gibbs Sampling

input : dynamic relational data $\mathbf{b}^{(1)}, \ldots, \mathbf{b}^{(T)}$
Initialize the maximum number of groups K,
hyperparameters $\beta, \tau, c_0, g_0, h_0$, and parameters
$\gamma_0, r_k, \xi, \lambda_{kk'}, \phi_{nk}^{(t)}$
repeat
  Sample $x_{mn}^{(t)}$ for non-zero links (Eq. 6)
  Sample $x_{mkk'n}^{(t)}$ (Eq. 7) and update
    $x_{\cdot kk'}^{(\cdot)} = \sum_t \sum_{m,n \neq m} x_{mkk'n}^{(t)}$
    $x_{mk\cdot}^{(t)} = \sum_{n \neq m,k'} x_{mkk'n}^{(t)}$
  Sample $l_{kk'}$ (Eq. 9) and calculate the quantities:
    $\theta_{kk'} = \sum_{t,n,m \neq n} \phi_{nk}^{(t)} \phi_{mk'}^{(t)}, \quad \tilde{p}_{kk'} = \frac{\theta_{kk'}}{\theta_{kk'} + \beta}$
  Sample r_k (Eq. 11), ξ (Eq. 12), and $\lambda_{kk'}$ (Eq. 13)
  for $t = T$ to 1 **do**
    Sample $y_{nk}^{(t)}$ (Eq. 23) and update $\varrho_{nk}^{(t)}$ (Eq. 20)
  end
  for $t = 0$ to T **do**
    Sample $\phi_{nk}^{(t)}$ (Eq. 24)
  end
  Sample c_0, β (Eq. 25) and γ_0 (Eq. 15)
until *convergence*
output: posterior mean $\phi_{nk}^{(t)}, r_k, \xi, \lambda_{kk'}, \beta, \gamma_0, c_0$

The main procedure of our online Gibbs sampler is then as follows. We first update the sufficient statistics used to sample model parameters as

$$x_{mk\cdot\cdot}^{(t)i} = (1 - \rho^i)x_{mk\cdot\cdot}^{i-1} + \rho^i \frac{|J|}{|J^i|} \sum_{\substack{n \neq m, \\ m,n \in J^i}} \sum_{k'} x_{mkk'n}^{(t)},$$

$$x_{\cdot kk'}^{i} = (1 - \rho^i)x_{\cdot kk'}^{i} + \rho^i \frac{|J|}{|J^i|} \sum_t \sum_{\substack{m,n \neq m, \\ m,n \in J^i}} x_{mkk'n}^{(t)},$$

where $\rho^i = (i + i_0)^{-\kappa}$, where $i_0 > 0$ and $\kappa \in (1/2, 1]$ is the decay factor commonly used for online methods (Guhaniyogi et al. 2014). We calculate the sufficient statistics for each mini-batch and then resample the model parameters using the procedure in batch Gibbs sampling algorithm outlined in Algorithm 1.

Experiments

We evaluate our model and its inference algorithm by performing experiments on both synthetic and real-world datasets. First, we generate a synthetic data with the true underlying network structure evolving over time to test our model on dynamic community detection. For quantitive evaluation, we determine the model's ability to predict held-out missing links. Our baseline methods include DRIFT, D-GPPF and HGP-EPM as we discussed before. For DRIFT, we use default settings as the code released online. [1] We im-

---

plemented D-GPPF by ourselves and set the hyperparameters and initialize the model parameters with the values provided in (Acharya et al. 2015). For HGP-EPM, we used the code released for (Zhou 2015). [2] In the following, we refer to our model as DPGM (**D**ynamic **P**oisson **G**amma **M**embership model). For DPGM, we set $\tau = 1, g_0 = 0.1, h_0 = 0.1$ and use $K = N/2$, where N is the number of nodes, for initilization. We obtain similar results when instead setting $\tau = 0.1, \tau = 10$ in a sensitivity analysis. For online Gibbs sampling, we set $\kappa = 0.5, i_0 = 100$, and mini-batch size $|J_i| = N/4$. All our experiments were performed on a standard desktop with 2.7 GHz CPU and 24 GB RAM. Following (Foulds et al. 2011; Heaukulani et al. 2013; Kim et al. 2013), we generate a set of small-scale dynamic networks from real-world relational data while we use held-out relational data to evaluate our model. The following three real-world datasets are used in our experiments, the detail of which are summarized in Table 1.

NIPS. The dataset records the co-authorship information among 5722 authors on publications in NIPS conferences over the past ten years (Rosenfeld et al. 2014). We first take the 70 authors who are most connected across all years to evaluate all methods (NIPS 70). We also use the whole dataset (NIPS 5K) for evaluation.

Enron. The dataset contains 136776 emails among 2359 persons over 28 months ($T = 28$) (Tang et al. 2008). We generate a binary symmetric matrix for each monthly snapshot. The presence or absence of an email between each pair of persons during one month is described by the binary link at that time. We first select 61 persons by taking a 7-core of the aggregated network for the entire time and filter out the authors with email records less than 5 snapshots (Enron 61). We also use the whole dataset for evaluation (Enron 2K).

DBLP. The DBLP dynamic networks dataset (Tang et al. 2008) are generated from the co-authorship recordings among 347013 authors over 25 years, which is a subset of data contained in the DBLP database. We first choose 7750 authors by taking a 7-core of the aggregated network for the entire time (DBLP 7K) and subsequently filter out authors with less than 10 years of consecutive publication activity to generate a small dataset (DBLP 96).

The proposed method and the two baselines are applied to all six datasets, except for DRIFT. DRIFT could not be applied to NIPS 5K, Enron 2K and DBLP 7K due to its unfavorable computational complexity. Most of these datasets exhibit strong sparsity that the proposed algorithm can exploit through its Bernoulli-Poisson link function.

Table 1: Details of the dataset used in our experiments.

Datasets	NIPS 70	DBLP 96	Enron 61
Nodes #	70	96	61
Time slices #	10	25	28
Non-zero links #	528	1392	1386
	NIPS 5K	DBLP 7K	Enron 2K
Nodes #	5722	7750	2359
Time slices #	10	10	28
Non-zero links #	5514	108980	76828

---

[2]https://github.com/mingyuanzhou/EPM.

Dynamic community detection

We adapt the synthetic example used in (Acharya et al. 2015) to generate a dynamic network of size 65 that evolve over five time slices as shown in Fig. 2. More specifically, we generate three groups at $t = 1$, and split the second group at $t = 2$. From $t = 3$ to 4, the second and third group merge into one group. In Fig. 2, column (b) and (d) show the discovered latent groups over time by D-GPPF and DPGM, respectively. D-GPPF captures the evolution of the discovered groups but has difficulties to characterize the changes of node-group memberships over time. Our model (DPGM) can detect the dynamic groups quite distinctively. We also show the associations of the nodes to the inferred latent groups by D-GPPF and DPGM in column (c) and (e), respectively. In particular, we calculate the association weights of each node to the latent groups as $r_{tk}\phi_{nk}$ for D-GPPF and $\phi_{nk}^t \lambda_{kk}$ for DPGM. For both models, most of the redundant groups can effectively be shrunk even though we initialize both algorithms with $K = 50$. The node-group association weights estimated by DPGM vary smoothly over time and capture the evolution of the node-group memberships, which is consistent to the ground truth shown in column (a).

Missing link prediction

For the task of missing link prediction, we randomly hold out 20% of the observed interactions (either links or nonlinks) at each time as test data. The remaining data is used for training. HGP-EPM, DRIFT and D-GPPF are considered as the baseline methods. We train a HGP-EPM model on the training entries for each time slice separately. For each method, we use 2000 burn-in iterations, and collect 1000 samples of the model posterior. We estimate the posterior mean of the link probability for each held-out edge in the test data by averaging over the collected Gibbs samples. We then use these link probabilities to evaluate the predictive performance of each model by calculating the area under the curve of the receiver operating characteristic (AUC-ROC) and of the precision-recall (AUC-PR). Table 2 shows the average evaluation metrics for each model over 10 runs. Overall, our model (DPGM) shows the best performance. We observe that both DRIFT and DPGM outperform D-GPPF because the evolution of individual node-group memberships are explicitly captured in these two models. D-GPPF essentially assumes that the nodes' memberships are static over time and thus has difficulties to fully capture the dynamics of each node's interactions caused by the same node's memberships' evolution. We see that DPGM outperforms its static counterpart, HGP-EPM, via capturing the evolution of nodes' memberships over time. We also compare per-iteration computation time of each model (all models are implemented in Matlab), as shown in Table 3. The computational cost of DRIFT scales quadratically with the number of nodes. Both D-GPPF and DPGM are much faster than DRIFT because the former two models scale only with the number of nonzero edges. We also report per-iteration computation time of GPPF and DPGM with Matlab/MEX/C implementation on medium-scale data in Table 3.
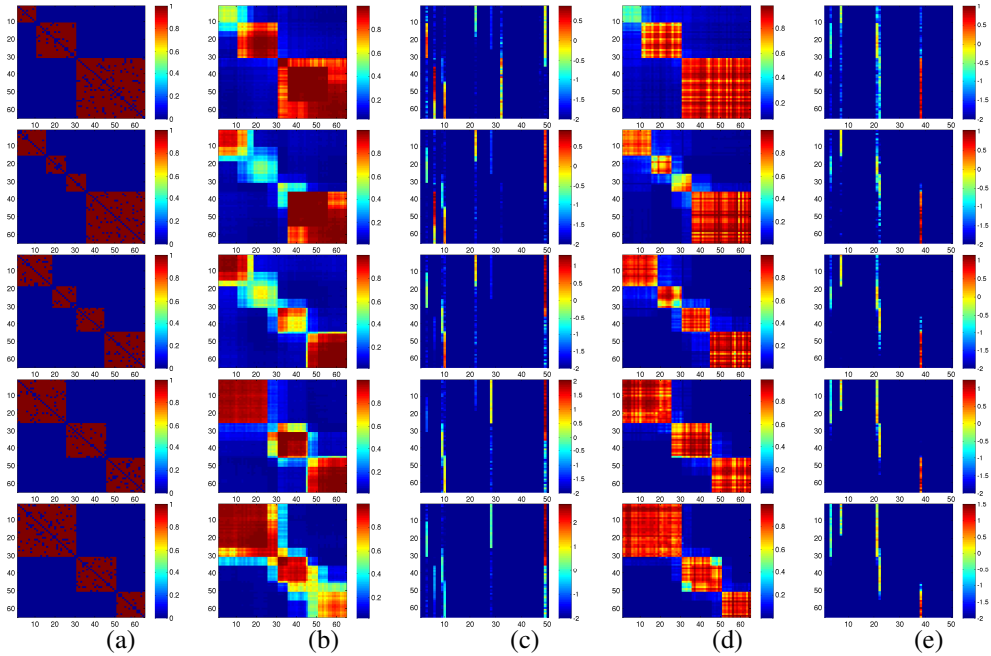
Figure 2: Dynamic community detection on synthetic data. We generate a dynamic network with five time snapshots as shown in column (a) ordered from top to bottom. The link probabilities estimated by D-GPPF and DPGM are shown in column (b) and (d). The association weights of each node to the latent groups can be calculated by $r_{tk}\phi_{nk}$ for D-GPPF and $\phi_{nk}^t \lambda_{kk}$ for DPGM as shown in column (c) and (e), respectively. The pixel values are displayed on $\log_{10}$ scale.

Table 2: Missing links prediction. We highlight the performance of the best scoring model in bold.

| Model | NIPS 70 | | DBLP 96 | | Enron 61 | |
	AUC-ROC	AUC-PR	AUC-ROC	AUC-PR	AUC-ROC	AUC-PR
HGP-EPM	0.947 ± 0.003	0.146 ± 0.030	0.936 ± 0.004	0.268 ± 0.025	0.931 ± 0.003	0.432 ± 0.016
D-GPPF	0.943 ± 0.009	0.072 ± 0.023	0.924 ± 0.004	0.249 ± 0.010	0.963 ± 0.002	0.649 ± 0.022
DRIFT	0.963 ± 0.002	0.148 ± 0.015	0.965 ± 0.001	0.335 ± 0.001	$\mathbf{0.975 \pm 0.006}$	$\mathbf{0.736 \pm 0.030}$
DPGM (batch)	$\mathbf{0.970 \pm 0.003}$	$\mathbf{0.169 \pm 0.028}$	$\mathbf{0.967 \pm 0.002}$	$\mathbf{0.339 \pm 0.006}$	0.971 ± 0.002	0.712 ± 0.004

| Model | NIPS 5K | | DBLP 7K | | Enron 2K | |
	AUC-ROC	AUC-PR	AUC-ROC	AUC-PR	AUC-ROC	AUC-PR
HGP-EPM	0.928 ± 0.010	0.121 ± 0.008	0.897 ± 0.016	0.055 ± 0.008	0.975 ± 0.003	0.324 ± 0.006
D-GPPF	0.928 ± 0.006	0.119 ± 0.006	0.858 ± 0.002	0.054 ± 0.003	0.976 ± 0.001	0.305 ± 0.003
DPGM (batch)	$\mathbf{0.930 \pm 0.002}$	$\mathbf{0.129 \pm 0.004}$	$\mathbf{0.923 \pm 0.011}$	$\mathbf{0.057 \pm 0.001}$	$\mathbf{0.983 \pm 0.001}$	$\mathbf{0.398 \pm 0.006}$
DPGM (online)	0.929 ± 0.002	0.104 ± 0.005	0.911 ± 0.001	0.051 ± 0.003	0.976 ± 0.001	0.346 ± 0.005

Table 3: Comparison of per-iteration computation time (seconds).

	NIPS 70	DBLP 96	Enron 61		NIPS 5K	DBLP 7K	Enron 2K
D-GPPF	0.0388	0.1350	0.2161	D-GPPF	7.6440	7.9160	7.8227
DRIFT	11.7047	42.1853	24.7505	DPGM (batch)	10.6240	15.6576	15.8584
DPGM (batch)	0.1283	0.6302	0.7364	DPGM (online)	8.9501	10.8152	10.4521

Conclusion

We have presented a probabilistic model for learning from dynamic relational data. The evolution of the underlying structure is characterized by the Markovian construction of latent memberships. We also proposed efficient batch and online Gibbs algorithms that make use of the data augmentation technique. Experimental results on synthetic and real datasets illustrate our model's interpretable latent representations and competitive performance. Our model is dedicated to dynamic networks modeling but can be considered for other related problems such as dynamic multi-relational graph model (Tay et al. 2017; Trivedi et al. 2017). Another interesting direction is to scale up the model inference algorithm via stochastic gradient variational Bayes (Kingma and Welling 2014; Knowles 2015).

Acknowledgments

References

Acharya, A., et al. 2015. Nonparametric dynamic network modeling. In *KDD Workshop on MLT*, 104–113.

Airoldi, E. M., et al. 2008. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.* 9:1981–2014.

Basbug, M. E., and Engelhardt, B. E. 2016. Hierarchical compound Poisson factorization. In *ICML*, 1795–1803.

Bordes, A., et al. 2011. Learning structured embeddings of knowledge bases. In *AAAI*, 301–306.

Caron, F., and Fox, E. B. 2015. Sparse graphs using exchangeable random measures. *CoRR* arXiv:1401.1137.

Dunlavy, D. M., et al. 2011. Temporal link prediction using matrix and tensor factorizations. *ACM Trans. Knowl. Discov. Data* 5(2):10:1–10:27.

Dunson, D. B., and Herring, A. H. 2005. Bayesian latent variable models for mixed discrete outcomes. *Biostatistics* 6(1):11–25.

Durante, D., et al. 2014. Bayesian logistic Gaussian process models for dynamic networks. In *AISTATS*, 194–201.

Foulds, J. R., et al. 2011. A dynamic relational infinite feature model for longitudinal social networks. In *AISTATS*, 287–295.

Fu, W., et al. 2009. Dynamic mixed membership blockmodel for evolving networks. In *ICML*, 329–336.

Goldenberg, A., et al. 2010. A survey of statistical network models. *Found. Trends Mach. Learn.* 2(2):129–233.

Guhaniyogi, R., et al. 2014. Bayesian conditional density filtering for big data. *CoRR* abs/1401.3632.

Heaukulani, C., et al. 2013. Dynamic probabilistic models for latent feature propagation in social networks. In *ICML*, 275–283.

Ho, Q., et al. 2011. Evolving cluster mixed-membership blockmodel for time-evolving networks. In *AISTATS*, 342–350.

Hoff, P. D., et al. 2001. Latent space approaches to social network analysis. *JASA* 97:1090–1098.

Hu, C., et al. 2015. Zero-truncated Poisson tensor factorization for massive binary tensors. In *UAI*, 375–384.

Ishiguro, K., et al. 2010. Dynamic infinite relational model for time-varying relational data analysis. In *NIPS*, 919–927.

Kemp, C., et al. 2006. Learning systems of concepts with an infinite relational model. In *AAAI*, 381–388.

Kim, M., et al. 2013. Nonparametric multi-group membership model for dynamic networks. In *NIPS*, 1385–1393.

Kingma, D. P., and Welling, M. 2014. Auto-encoding variational Bayes. In *ICLR*, 1–14.

Knowles, D. A. 2015. Stochastic gradient variational Bayes for gamma approximating distributions. *arXiv preprint arXiv:1509.01631*.

Mucha, P. J., et al. 2010. Community structure in time-dependent, multiscale, and multiplex networks. *Science* 328(5980):876–878.

Nickel, M., et al. 2016. A review of relational machine learning for knowledge graphs. *Proc. of the IEEE* 104(1):11–33.

Pitman, J. 2006. *Combinatorial stochastic processes*. Berlin: Springer-Verlag. Lectures on Probability Theory.

Quenouille, M. H. 1949. A relation between the logarithmic, Poisson, and negative binomial series. *Biometrics* 5(2):162–164.

Rai, P., and Daume III, H. 2008. The infinite hierarchical factor regression model. In *NIPS*.

Robins, G.; Pattison, P.; Kalish, Y.; and Lusher, D. 2007. An introduction to exponential random graph (p*) models for social networks. *Social Networks* 29(2):173–191.

Rosenfeld, N., et al. 2014. Learning structured models with the auc loss and its generalizations. In *AISTATS*, 841–849.

Tang, L., et al. 2008. Community evolution in dynamic multi-mode networks. In *KDD*, 677–685.

Tay, Y., et al. 2017. Non-parametric estimation of multiple embeddings for link prediction on dynamic knowledge graphs. In *AAAI*, 1243–1249.

Trivedi, R., et al. 2017. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *ICML*, 3462–3471.

Xing, E. P., et al. 2010. A state-space mixed membership blockmodel for dynamic network tomography. *Ann. Appl. Stat.* 4(2):535–566.

Zhou, M., and Carin, L. 2015. Negative binomial process count and mixture modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* 37(2):307–320.

Zhou, M. 2015. Infinite edge partition models for overlapping community detection and link prediction. In *AISTATS*, 1135–1143.