Investigating the impact of exposure in probabilistic models for dynamical networks

Johannes Schulz[1], **Marco Baity-Jesi**[2], **Caterina De Bacco**[1]

[1] Max Planck Institute for Intelligent Systems, Cyber Valley, Tuebingen, 72076, Germany

[2] Eawag, Überlandstrasse 133, CH-8600 Dübendorf, Switzerland

E-mail: `caterina.debacco@tuebingen.mpg.de`, `marco.baityjesi@eawag.ch`

Abstract. When modeling interactions between individuals in a network, probabilistic generative models often assume that any interaction could in principle exist. If an interaction is not observed, this is attributed to the fact that the chosen mechanism for tie formation is not occuring. For instance, two nodes may not have a high affinity. However, these models ignore the possibility that nodes may not have been exposed to each other. That prevents them from interacting in the first place, regardless the mechanism of tie formation. In particular, such approaches fail to capture situations where individuals may indeed be compatible and thus interact, if they were exposed to each other.

While this problem is known in recommender systems, it has only recently been investigated in network models. This study has been limited to static networks, where interactions do not evolve in time. As many real-world networks are constantly changing, interactions and exposure between nodes need to be modeled within the framework of temporal networks. Here, we introduce two approaches that tackle this problem by extending the concept of exposure to a dynamical setting. They both rely on a Bayesian probabilistic approach that considers community structure as the main underlying mechanism for tie formation.

By properly modeling how exposure between nodes changes in time, we obtain strong community reconstruction results for both synthetic and real data of recorded proximity interactions between individuals. On the latter, we additionally observe a significant improvement in terms of reconstructing unobserved interactions compared to a standard model that does not take exposure into account.

1. Introduction

Networks are a popular and effective representation of large-scale complex systems, encompassing socio-economical relations, the human brain, cell metabolism, ecosystems, informational infrastructure, and many more [?, ?]. A network or graph consists of a set of nodes representing distinct entities as well as a set of edges which represent interactions between these entities. This basic model is often extended to account for the specific properties of the system at hand. For cases in which not only interactions, but also their intensity is observed, weighted networks are used. Networks can be undirected, if interactions are symmetric, or directed if that is not the case. Additionally, many real-world systems are not static but evolve over time. This can be modeled through temporal networks [?, ?].

To understand the hidden patterns and processes in complex systems, it is helpful to further abstract a complex network structure composed of many microscopic variables into a lower-dimensional representation that coarsens the system. Probabilistic generative models [?] are mathematical approaches that aim at doing so by investigating the generative process, *i.e.* how exactly interactions between nodes are taking place, using a set of simplifying assumptions. Here, the challenge is to come up with a model that captures reality as accurately as possible while remaing relatively simple. One quite intuitive approach is to suppose that nodes can be grouped into equivalence classes and that interactions between them only depend on their class memberships. This assumption is called *Stochastic Equivalence* and lies at the heart of the *Stochastic Block Model* (SBM) [?], a popular and flexible probabilistic model for network data.

As the SBM assumes that nodes can be clustered into groups, it naturally lends itself to the task of community detection. Community detection is an inference task that aims at clustering similar nodes into communities [?, ?, ?]. Two nodes are considered similar if they interact in a similar way. In **??**, we show a synthetic network generated with a SBM as well as the corresponding adjacency matrix. An adjacency matrix represents a graph as a matrix A, such that the entry $A_{i,j}$ models the weight of an edge between nodes i and j. In the example in **??**, we can observe an assortative structure [?], *i.e.* nodes that belong to the same community have a higher likelihood of interacting than nodes that belong to different communities.

One main property observed in real networks is sparsity, *i.e.* the number of existing interactions is linear in the number of nodes. This means that one observes only a small fraction among the many possible interactions between pairs of nodes. For instance, online social networks can contain billions of users, but each user only interacts with a tiny fraction of that. The observed non-existing interactions are traditionally attributed to a low affinity between nodes (Alice not interacting with Jack implies that Alice does not like Jack). However, often the actual reason is not a low affinity, but rather the fact that the two users have never met, *i.e.* never been *exposed* to each other. **??** visualizes this exposure mechanism. The idea of exposure has been explored in the context of recommender systems [**?**, **?**, **?**, **?**]. However, adapting these techniques to social networks is non-trivial and the investigation of this problem is still missing. Nevertheless, it is crucial to do so, as non-existing links contain important information which can only partially be captured

by existing approaches [**?**, **?**, **?**, **?**]. In previous work [**?**], we explored the idea of including exposure as an additional binary latent variable to the SBM. This was done for static networks, *i.e.* settings where we only observe one network sample and assume that the network does not change over time. In practice though, many real networks are constantly evolving. This can have dramatic effects in how the exposure mechanism acts in these systems. If exposure models physical proximity and we consider social networks, we can for instance observe that exposure is not static at all. Also, once people are exposed to each other, they are more likely to stay exposed. Modelling exposure in a static way fails to take these two observations into account. Hence, it is paramount to investigate approaches that model exposure in dynamical settings and to understand its interplay with an underlying community structure. This is the main goal of this project.

To this end we develop two methods, which model exposure in a conceptually very different way. Through link prediction experiments, we obtain that one of them, ExpMarkov, is better able to capture the true generative process of social networks than the same model without exposure [**?**]. ExpMarkov models both exposure as well as affinity through distinct communities. We show that seperating these two concepts, which are typically implicitly mixed in traditional models, yields promising inference results.

This rest of this thesis is structured as follows: In chapter **??**, we provide the theoretical background regarding SBMs and the exposure mechanism. Then, we introduce two approaches, ExpMarkov and ExpHeaviside, that adapt the idea of exposure to a dynamical setting, as well as the inference algorithm used to infer the optimal latent variables. The respective derivations are shown in appendix **??**. In chapter **??**, the two methods are evaluated on synthetic and on real data. As real data, we use small temporal social networks collected in different environments [?, ?, ?, ?, ?] by the **SocioPatterns** collaboration (`http://www.sociopatterns.org`). We conclude by analysing the obtained results and by investigating potential avenues for future research.

2. Template Summary

- there is an incompatibility between amsmath.sty and iopart.cls which cannot be completely worked around. If your article relies on commands in amsmath.sty that are not available in iopart.cls, you may wish to consider using a different class file.
- The 'master LATEX file must read in all other LATEX and figure files from the current directory
- The words table and figure should be written in full and not abbreviaged to tab. and fig. Do not include 'eq.', 'equation' etc before an equation number or 'ref.' 'reference' etc before a reference number.
- All journals to which this document applies allow the use of either the Harvard or Vancouver system

3. Methods

For temporal networks, we model exposure dynamically. We choose a simple setting in which the probability of exposure is *i.i.d.* in time, but varies between node pairs. Additionally, in previous work the exposure prior was quite weak as the propensity to be exposed for node i is the same for each node j it could be exposed to. Here, we increase the expressiveness of μ by making it a $\tilde{K}$-dimensional vector. As is the case with u and v, μ can be interpreted as a mixed membership community vector, *i.e.* we are now able to model exposure communities as well as communities due to affinity.

Resulting model These two changes result in the following prior:

$$\begin{aligned} p(\mathbf{Z}) &= \prod_t \prod_{i<j} p(Z_{i,j}^{(t)}) \\ &= \prod_t \prod_{i<j} \mu_i^T \mu_j^{Z_{i,j}^{(t)}} (1 - \mu_i^T \mu_j)^{1-Z_{i,j}^{(t)}} \quad . \end{aligned} \tag{1}$$

We can see that the Bernulli prior on $\mathbf{Z}$ now depends on $\mu_i^T \mu_j$, that is the similarity between the exposure communities of i and j. Analogously to the static model, we choose the likelihood as:

$$p(\mathbf{A}\,|\,\theta,\mathbf{Z}) = \prod_t \prod_{i,j} Pois(A_{i,j}^{(t)};\lambda_{i,j})^{Z_{i,j}^{(t)}} \delta(A_{i,j}^{(t)})^{(1-Z_{i,j}^{(t)})} \quad . \tag{2}$$

4. Inference

We infer the optimal latent variables using *Maximum A Posteriori Estimation* (**MAP**), as we have prior information on $\mathbf{Z}$ that we can use.

$$\theta^* = \arg\max_\theta p(\theta\,|\,A) = \arg\max_\theta \log \sum_Z p(Z,\theta\,|\,A) \quad . \tag{3}$$

As the logarithm of a sum is hard to derive, we use Jensens inequality to approximate it:

$$\log \sum_Z p(Z,\theta\,|\,A) \geq \sum_Z q(Z) \log \frac{p(Z,\theta\,|\,A)}{q(Z)} := \mathcal{L}(q,\theta,\mu) \quad . \tag{4}$$

Here $q(\mathbf{Z})$ is any distribution satisfying $\sum_{\mathbf{Z}} q(\mathbf{Z}) = 1$.
However, exact equality is reached for

$$q(\mathbf{Z}) = \frac{p(\mathbf{Z},\theta\,|\,\mathbf{A})}{\sum_{\mathbf{Z}} p(\mathbf{Z},\theta\,|\,\mathbf{A})} \quad , \tag{5}$$

which is the posterior on $\mathbf{Z}$. Now we apply **EM** by iterating between updating $q(\mathbf{Z})$ according to equation **??** and updating θ and μ.

To obtain the parameter updates, we compute the derivative of $\mathcal{L}(q,\theta,\mu)$ *w.r.t.* θ and μ and set it to zero (**??**), resulting in the following update equations:

4.1. Updating θ

$\mathcal{L}(q,\theta,\mu)$ simplifies, as we only care about the parts which depend on θ:

$$\begin{aligned}
\mathcal{L}(q,\theta,\mu) &\propto \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(A\,|\,Z,\theta) \\
&= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left(\prod_t \prod_{i,j} \delta(A_{i,j}^{(t)})^{1-Z_{i,j}^{(t)}} Pois(A_{i,j}^{(t)};\lambda_{i,j})^{Z_{i,j}^{(t)}} \right) \\
&\propto \sum_{i,j} \sum_t \sum_{\mathbf{Z}} q(\mathbf{Z}) Z_{i,j}^{(t)} \log(Pois(A_{i,j}^{(t)};\lambda_{i,j})) \\
&\overset{(\mathbf{??})}{=} \sum_{i,j} \sum_t Q_{i,j}^{(t)} \left(A_{i,j}^{(t)} \log(\lambda_{i,j}) - \lambda_{i,j} \right) \quad .
\end{aligned} \tag{6}$$

Now, we can take the derivative of the simplified $\mathcal{L}(q,\theta,\mu)$ and set it to zero to obtain the updates on θ. As $\log \lambda_{i,j}$ is difficult to derive, we use Jensens inequality [?] to estimate it.

$$\log(\lambda_{i,j}) = \log\left(\sum_{k,q} u_{i,k}v_{j,q}w_{k,q}\right) \geq \sum_{k,q} \rho_{ijkq} \log\left(\frac{u_{i,k}v_{j,q}w_{k,q}}{\rho_{ijkq}}\right) \quad . \tag{7}$$

For

$$\rho_{ijkq} = \frac{u_{i,k}v_{j,q}w_{k,q}}{\sum_{k,q} u_{i,k}v_{j,q}w_{k,q}} \quad , \tag{8}$$

exact equality is reached.

Now, we apply **EM** a second time by iterating between updating ρ_{ijkq} according to equation **??** and updating θ.

We show the derivations for $u_{m,n}$. However, $v_{m,n}$ and $w_{m,n}$ can be obtained in a similar way:

$$\begin{aligned} \frac{\partial}{\partial u_{m,n}}\mathcal{L}(q,\theta,\mu) &= \sum_{i,j}\sum_{t} Q_{i,j}^{(t)}\left(A_{i,j}^{(t)}\frac{\partial}{\partial u_{m,n}}\log(\lambda_{i,j}) - \frac{\partial}{\partial u_{m,n}}\lambda_{i,j}\right) \\ &= \sum_{j,t} Q_{m,j}^{(t)}\left(A_{m,j}^{(t)}\frac{\sum_q \rho_{mjnq}}{u_{mn}} - \sum_q v_{jq}w_{nq}\right) \stackrel{!}{=} 0 \\ \Leftrightarrow u_{m,n} &= \frac{\sum_j\sum_q \rho_{mjnq}\sum_t Q_{m,j}^{(t)}A_{mj}^{(t)}}{\sum_j\sum_q v_{jq}w_{nq}\sum_t Q_{m,j}^{(t)}} \quad , \end{aligned} \tag{9}$$

$$v_{m,n} = \frac{\sum_i\sum_k \rho_{imkn}\sum_t Q_{i,m}^{(t)}A_{im}^{(t)}}{\sum_i\sum_k u_{ik}w_{kn}\sum_t Q_{i,m}^{(t)}} \quad , \tag{10}$$

$$w_{m,n} = \frac{\sum_{ij} \rho_{ijmn}\sum_t Q_{i,j}^{(t)}tA_{ij}^{(t)}}{\sum_{ij} u_{im}v_{jn}\sum_t Q_{i,j}^{(t)}} \quad . \tag{11}$$

4.2. Updating μ

$$\begin{aligned} \mathcal{L}(q,\theta,\mu) &\propto \sum_{\mathbf{Z}} q(\mathbf{Z})\log p(Z) \\ &= \sum_{\mathbf{Z}} q(\mathbf{Z})\sum_t\sum_{i<j} Z_{i,j}^{(t)}\log(\mu_i^T\mu_j) + (1-Z_{i,j}^{(t)})\log(1-\mu_i^T\mu_j) \\ &= \sum_{i<j}\log(\mu_i^T\mu_j)\left(\sum_t Q_{i,j}^{(t)}\right) + \log(1-\mu_i^T\mu_j)\left(T - \sum_t Q_{i,j}^{(t)}\right) \quad . \end{aligned} \tag{12}$$

Again, the parameter updates on μ can be obtained by computing the derivative with respect to μ and setting it to zero:

$$
\begin{aligned}
\frac{\partial}{\partial \mu_{k,l}} \mathcal{L}(q, \theta, \mu) &= \sum_{i<j} \left(\sum_t Q_{i,j}^{(t)} \right) \frac{\delta_{k,i}\mu_{j,l} + \delta_{k,j}\mu_{i,l}}{\mu_i^T \mu_j} \\
&\qquad - \left(T - \sum_t Q_{i,j}^{(t)} \right) \frac{\delta_{k,i}\mu_{j,l} + \delta_{k,j}\mu_{i,l}}{1 - \mu_i^T \mu_j} \\
&= \sum_{i \neq k} \mu_{il} \left(\frac{\sum_t Q_{i,k}^{(t)}}{\mu_{ik}} - \frac{T - \sum_t Q_{i,k}^{(t)}}{1 - \mu_{ik}} \right) \stackrel{!}{=} 0 \quad .
\end{aligned}
\tag{13}
$$

4.3. Updating $Q_{i,j}^{(t)}$

$$
\begin{aligned}
Q_{i,j}^{(t)} &= p(Z_{i,j}^{(t)} = 1 \,|\, \theta, \mathbf{A}) = p(Z_{i,j}^{(t)} = 1 \,|\, \theta, A_{i,j}^{(t)}, A_{j,i}^{(t)}) \\
&= \frac{p(Z_{i,j}^{(t)} = 1) p(A_{i,j}^{(t)} \,|\, Z_{i,j}^{(t)} = 1, \theta) p(A_{j,i}^{(t)} \,|\, Z_{i,j}^{(t)} = 1, \theta)}{\sum_{\mathbf{Z}} p(Z_{i,j}^{(t)} = 1) p(A_{i,j}^{(t)} \,|\, Z_{i,j}^{(t)} = 1, \theta) p(A_{j,i}^{(t)} \,|\, Z_{i,j}^{(t)} = 1, \theta)} \\
&= \frac{\mu_i^T \mu_j \mathrm{Pois}(A_{i,j}^{(t)}; \lambda_{i,j}) \mathrm{Pois}(A_{j,i}^{(t)}; \lambda_{j,i})}{\mu_i^T \mu_j \mathrm{Pois}(A_{i,j}^{(t)}; \lambda_{i,j}) \mathrm{Pois}(A_{j,i}^{(t)}; \lambda_{j,i}) + (1 - \mu_i^T \mu_j) \delta(A_{i,j}^{(t)}) \delta(A_{j,i}^{(t)})}
\end{aligned}
\tag{14}
$$

The complete inference algorithm is summarized below (**??**)

Algorithm 1 EM

Input: network $A \in \mathbb{N}^{N\times N\times T}$
number of affinity communities K
number of exposure communities $\tilde{K}$

Output: memberships $u, v \in \mathbb{R}^{N\times K}$
network affinity matrix $w \in \mathbb{R}^{K\times K}$
exposure memberships $\mu \in \mathbb{R}^{N\times\tilde{K}}$
posterior estimate $Q \in \mathbb{R}^{N\times N}$ for Z

Initialize $\theta : (u, v, w), \mu$ at random.

Repeat until $\mathcal{L}(q, \theta, \mu)$ converges:

1. Calculate ρ and Q (E-step):

$$\rho_{ijkq} = \frac{u_{i,k}v_{j,q}w_{k,q}}{\sum_{k,q} u_{i,k}v_{j,q}w_{k,q}} \, ,$$

update $Q_{i,j}^{(t)}$ *acc.* to **??**,

2. Update parameters θ (M-step):

i) for each node i and community k update memberships:

update u_{ik}(*acc.* to **??**)
update v_{ik}(*acc.* to **??**)

ii) for each pair of communities k, q update affinity matrix:

update w_{kq}(*acc.* to **??**)

iii) update prior on exposure indicator for each node i and exposure community k:

update μ_{ik}(*acc.* to **??**)