Investigating the impact of exposure in probabilistic models for dynamical networks

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

[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. 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:

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

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:

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

2. 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 **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:

2.1. Updating θ

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

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

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:

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

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

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

2.2. Updating μ

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

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

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

2.3. Updating $Q^{(t)}_{i,j}$

$$\begin{aligned}
Q^{(t)}_{i,j} &= p(Z^{(t)}_{i,j} = 1 \,|\, \theta, \mathbf{A}) = p(Z^{(t)}_{i,j} = 1 \,|\, \theta, A^{(t)}_{i,j}, A^{(t)}_{j,i}) \\
&= \frac{p(Z^{(t)}_{i,j} = 1) p(A^{(t)}_{i,j} \,|\, Z^{(t)}_{i,j} = 1, \theta) p(A^{(t)}_{j,i} \,|\, Z^{(t)}_{i,j} = 1, \theta)}{\sum_{\mathbf{Z}} p(Z^{(t)}_{i,j} = 1) p(A^{(t)}_{i,j} \,|\, Z^{(t)}_{i,j} = 1, \theta) p(A^{(t)}_{j,i} \,|\, Z^{(t)}_{i,j} = 1, \theta)} \\
&= \frac{\mu_{ij} \mathrm{Pois}(A^{(t)}_{i,j}; \lambda_{i,j}) \mathrm{Pois}(A^{(t)}_{j,i}; \lambda_{j,i})}{\mu_{ij} \mathrm{Pois}(A^{(t)}_{i,j}; \lambda_{i,j}) \mathrm{Pois}(A^{(t)}_{j,i}; \lambda_{j,i}) + (1-\mu_{ij}) \delta(A^{(t)}_{i,j}) \delta(A^{(t)}_{j,i})}
\end{aligned} \tag{14}$$

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 $\rho$ 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 $\theta$ (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 $\mu_{ik}$(acc. to ??)
```