Efficient Estimation for Longitudinal Network via Adaptive Merging

Haoran Zhang and Junhui Wang

Department of Statistics

The Chinese University of Hong Kong

Abstract

Longitudinal network consists of a sequence of temporal edges among multiple nodes, where the temporal edges are observed in real time. It has become ubiquitous with the rise of online social platform and e-commerce, but largely under-investigated in literature. In this paper, we propose an efficient estimation framework for longitudinal network, leveraging strengths of adaptive network merging, tensor decomposition and point process. It merges neighboring sparse networks so as to enlarge the number of observed edges and reduce estimation variance, whereas the estimation bias introduced by network merging is controlled by exploiting local temporal structures for adaptive network neighborhood. A projected gradient descent algorithm is proposed to facilitate estimation, where the upper bound of the estimation error in each iteration is established. A thorough analysis is conducted to quantify the asymptotic behavior of the proposed method, which shows that it can significantly reduce the estimation error and also provides guideline for network merging under various scenarios. We further demonstrate the advantage of the proposed method through extensive numerical experiments on synthetic datasets and a militarized interstate dispute dataset.

KEY WORDS: Dynamic network, embedding, multi-layer network, point process, tensor decomposition

1 Introduction

Longitudinal network, also known as temporal network or continuous-time dynamic network, consists of a sequence of temporal edges among multiple nodes, where the temporal edges may

be observed between each node pair in real time (Holme and Saramäki, 2012). It provides a flexible framework for modeling dynamic interactions between multiple objects and how network structure evolves over time (Aggarwal and Subbian, 2014). For instances, in online social platform such as Facebook, users send likes to the posts of their friends recurrently at different time (Perry-Smith and Shalley, 2003; Snijders et al., 2010); in international politics, countries may have conflict with others at one time but become allies at others (Cranmer and Desmarais, 2011; Kinne, 2013). Similar longitudinal networks have also been frequently encountered in biological science (Voytek and Knight, 2015; Avena-Koenigsberger et al., 2018) and ecological science (Ulanowicz, 2004; De Ruiter et al., 2005).

One of the key challenges in estimating longitudinal network resides in its scarce temporal edges, as the interactions between node pairs are instantaneous and come in a streaming fashion (Holme and Saramäki, 2012), and thus the observed network at each given time point can be extremely sparse. This makes longitudinal network substantially different from discrete-time dynamic network (Kim et al., 2018), where multiple snapshots of networks are collected each with much more observed edges. In literature, various methods have been proposed for discrete-time dynamic network, such as Markov chain based methods (Hanneke et al., 2010; Sewell and Chen, 2015, 2016; Matias and Miele, 2017), Markov process based methods (Snijders et al., 2010; Snijders, 2017) and tensor factorization methods (Lyu et al., 2021; Han et al., 2022). Whereas the former two assume that the discrete-time dynamic network is generated from some Markov chain or Markov process, tensor factorization methods treat the discrete-time dynamic network as an order-3 tensor and often require relatively dense network snapshots.

To circumvent the difficulty of severe under-sampling in longitudinal network, a common but rather ad-hoc approach is to merge longitudinal network into a multi-layer network based on equally spaced time intervals (Huang et al., ress). Such an overly simplified network merging scheme completely ignores the fact that network structure may change differently

2

during different time periods. Thus, it may introduce unnecessary estimation bias when network structure changes rapidly or incur large estimation variance when network structure stays unchanged for a long period. These negative impacts are yet neglected in literature, even though this ad-hoc network merging scheme has been widely employed to pre-process longitudinal networks in practice. Furthermore, some recent attempts were made from the perspective of survival and event history analysis (Vu et al., 2011a,b; Perry and Wolfe, 2013; Sit et al., 2021), with a keen focus on inference of the dependence of the temporal edge on some additional covariates. Some other recent works (Matias et al., 2018; Soliman et al., 2022) extend the stochastic block model to detect time-invariant communities in longitudinal network.

In this paper, we propose an efficient estimation method for longitudinal network, leveraging strengths of adaptive network merging, tensor decomposition and point process. Specifically, we introduce a two-step procedure for estimating the longitudinal network. The initial step merges the longitudinal network with some small intervals, leading to an initial estimate of the embeddings of the probability tensor. We then adaptively merge adjacent small intervals with similar estimated temporal embedding vectors, and re-estimate the probability tensor based on the adaptively merged intervals. We further give a regularized maximum likelihood estimator based on Poisson point process, whose intensity is determined by the out-node and in-node embedding vectors, as well as the temporal embedding vector at each specific time. A projected gradient descent algorithm is provided to facilitate estimation, as well as an information criteria for choosing the number of intervals.

A thorough theoretical analysis is conducted for the proposed estimation procedure. We first establish a general tensor estimation error bound based on a generic partition in each iteration of the projected gradient descent algorithm. The established error bound is tighter than most of the existing results in literature (Han et al., 2022), where the related empirical process is associated with a smaller parameter space with additional incoherence

conditions. This tighter bound enables us to derive the error bound for the tensor estimate based on equally spaced intervals, which consists of an interesting bias-variance tradeoff governed by the number of small intervals and leads to faster convergence rate than that in Han et al. (2022) and Cai et al. (2022). More importantly, the derived error bound does not require the strong intensity condition as required in Han et al. (2022) and Cai et al. (2022), which, to the best of our knowledge, is the first Poisson tensor estimation error bound in both medium and weak intensity regimes. Furthermore, it is shown that the tensor estimation error, including the estimation bias and variance, can be further reduced by adaptively merging intervals, which also provides guidelines for network merging under various scenarios. The advantage of the proposed method over other existing competitors is demonstrated in extensive numerical experiments on synthetic longitudinal networks. The proposed method is also applied to analyze a militarized interstate dispute dataset, where not only the prediction accuracy increases substantially, but the adaptively merged intervals also lead to clear and meaningful interpretation.

The rest of the paper is organized as follows. Section 2 first presents the two-step estimation procedure for longitudinal network, and then propose a regularized maximum likelihood estimator based on Poisson process. Section 3 provides the details of the computation algorithm. Section 4 establishes the error bound for the proposed method. Numerical experiments on synthetic and real-life networks are contained in Section 5. Section 6 concludes the paper with a brief discussion, and technical proofs and necessary lemmas are provided in the Appendix and a separate Supplementary File.

Notations. Before moving to Section 2, we introduce some notations and preliminaries for tensor decomposition. For any $n \geq r$, let $\mathbb{O}_{n,r} = \{\mathbf{U} \in \mathbb{R}^{n \times r} : \mathbf{U}^\top \mathbf{U} = \mathbf{I}_r\}$ and denote $\mathbb{O}_r = \mathbb{O}_{r,r}$. For a matrix $\mathbf{U}$, let $\mathbf{U}_{[i,]}, \mathbf{U}_{[,r]}$ and $(\mathbf{U})_{ir}$ denote the i-th row, r-th column and element (i,r) of $\mathbf{U}$, respectively. Let $\|\mathbf{U}\|_2, \|\mathbf{U}\|_F$ denote its spectral and Frobenius norm, and $\|\mathbf{U}\|_{2\to\infty} = \max_i \|\mathbf{U}_{[i,]}\|$. For any order-3 tensor $\mathcal{M} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, let $\mathcal{M}_{[i,,]}, \mathcal{M}_{[,j,]}, \mathcal{M}_{[,,k]}$

and $(\mathcal{M})_{ijk}$ denote the i-th horizontal slides, j-th lateral slides, k-th frontal slides and element (i,j,k) of $\mathcal{M}$, respectively. Let $\Psi_k(\mathcal{M}) \in \mathbb{R}^{n_k \times n_{-k}}$ be the mode-k unfolding of $\mathcal{M}$, where $n_{-k} = n_1 n_2 n_3 / n_k$ for $k = 1, 2, 3$. We denote $\mathrm{rank}(\mathcal{M}) \leq (r_1, r_2, r_3)$ if $\mathcal{M}$ admits the decomposition $\mathcal{M} = \mathcal{S} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W} =: [\mathcal{S}; \mathbf{U}, \mathbf{V}, \mathbf{W}]$ for some $\mathcal{S} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$, $\mathbf{U} \in \mathbb{R}^{n_1 \times r_1}$, $\mathbf{V} \in \mathbb{R}^{n_2 \times r_2}$ and $\mathbf{W} \in \mathbb{R}^{n_3 \times r_3}$. For any order-3 tensor $\mathcal{M}$ with $\mathrm{rank}(\mathcal{M}) \leq (r_1, r_2, r_3)$, define

$$\overline{\lambda}(\mathcal{M}) = \max \left\{ \|\Psi_1(\mathcal{M})\|_2, \|\Psi_2(\mathcal{M})\|_2, \|\Psi_3(\mathcal{M})\|_2 \right\},$$

$$\underline{\lambda}(\mathcal{M}) = \min \left\{ \sigma_{r_1}(\Psi_1(\mathcal{M})), \sigma_{r_2}(\Psi_2(\mathcal{M})), \sigma_{r_3}(\Psi_3(\mathcal{M})) \right\},$$

where $\sigma_r(\mathbf{M})$ denote the r-th largest singular value of matrix $\mathbf{M}$. Let $\|\mathcal{M}\|_F = \sqrt{\sum_{i,j,k} m_{ijk}^2}$ be the Frobenius norm of $\mathcal{M}$. Throughout the paper, we use c, C, ϵ and κ to denote positive constants whose values may vary according to context. For an integer m, let $[m]$ denote the set $\{1, ..., m\}$. For two number a and b, let $a \wedge b = \min(a, b)$. For two nonnegative sequences a_n and b_n, let $a_n \preceq b_n$ and $a_n \prec b_n$ denote $a_n = O(b_n)$ and $a_n = o(b_n)$, respectively. Denote $a_n \asymp b_n$ if $a_n \preceq b_n$ and $b_n \preceq a_n$. Further, $a_n \preceq_P b_n$ means that there exists a positive constant c such that $\Pr(a_n \geq c b_n) \to 0$ as n diverges.

2 Proposed method

Consider a bipartite longitudinal network with n_1 out-nodes and n_2 in-nodes on a given time interval $[0, T]$. Let $\mathcal{E} = \{(i_m, j_m, t_m) : m = 1, ..., M\}$ denote the set of all observed directed edges, where the triplet (i, j, t) denotes a temporal edge pointing from out-node i to in-node j at time t. Suppose the edges in $\mathcal{E}$ are generated from some distribution $f(\Theta(t))$ with $\Theta(t) = (\theta_{ij}(t))_{i,j=1}^n$, which admits a low rank structure so that

$$\theta_{ij}(t) = \mathcal{S} \times_1 \mathbf{u}_i^\top \times_2 \mathbf{v}_j^\top \times_3 \mathbf{w}(t)^\top, \tag{1}$$

where $\times_s$ denotes the mode-s product for $s \in [3]$, $\mathcal{S} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$ is an order-3 core tensor, and each out-node i, in-node j and time t are embedded as low-dimensional vectors $\mathbf{u}_i \in \mathbb{R}^{r_1}$, $\mathbf{v}_j \in \mathbb{R}^{r_2}$ and $\mathbf{w}(t) \in \mathbb{R}^{r_3}$, respectively. It is clear that the time-invariant network structure is captured by the network embedding vectors $\mathbf{u}$ and $\mathbf{v}$, while the temporal structure is captured by the temporal embedding vector $\mathbf{w}(t)$. Such a network embedding model has been widely employed for network data analysis (Hoff et al., 2002; Lyu et al., 2021; Zhang et al., 2022; Zhen and Wang, ress), which embeds the unstructured network in a low-dimensional Euclidean space to facilitate the subsequent analysis.

2.1 Adaptive merging

Let $\mathcal{G}_t = \{(i, j) : (i, j, t) \in \mathcal{E}\}$ as the observed network at time t, $\mathcal{T}_{ij} = \{t \in [0, T) : (i, j, t) \in \mathcal{E}\}$ as the time stamps for directed edges (i, j), and $\mathcal{T} = \bigcup_{i,j} \mathcal{T}_{ij}$. Since the directed edges in $\mathcal{E}$ are observed in real time, $\mathcal{G}_t$ can be extremely sparse and may consist of even only one observed edge, which casts great challenge for estimating the longitudinal network. To circumvent the difficulty of severe under-sampling, we propose to embed the longitudinal network by adaptively merging $\mathcal{G}_t$ into relatively dense networks based on their temporal structures, which leads to a substantially improved estimation of the longitudinal network.

We first split the time window $[0, T)$ into L equally spaced small intervals with endpoints $\{\delta_l\}_{l=1}^L$, where $\delta_l = l\Delta_{\boldsymbol{\delta}}$, $\delta_0 = 0$, and each interval $[\delta_{l-1}, \delta_l)$ is of width $\Delta_{\boldsymbol{\delta}} = T/L$. When $\Delta_{\boldsymbol{\delta}}$ is sufficiently small, it is expected that $\boldsymbol{\Theta}(t)$ shall be roughly constant within each time interval. As a direct consequence, $\boldsymbol{\Theta}(t)$ can be estimated by a low rank order-3 tensor $\mathcal{M} \in \mathbb{R}^{n_1 \times n_2 \times L}$, which admits a Tucker decomposition with rank (r_1, r_2, r_3),

$$\mathcal{M} = \mathcal{S} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W},$$

with $\mathbf{u}_i, \mathbf{v}_j$ and $\mathbf{w}_l$ being the corresponding rows of $\mathbf{U}, \mathbf{V}$ and $\mathbf{W}$, respectively. Let $\boldsymbol{\delta} =$

$(\delta_1, ..., \delta_L)^\top$ and $\mathcal{Y}_{\boldsymbol{\delta}} \in \mathbb{R}^{n_1 \times n_2 \times L}$ with $(\mathcal{Y}_{\boldsymbol{\delta}})_{ijl} = |\mathcal{T}_{ij} \cap [\delta_{l-1}, \delta_l]|$ representing the number of temporal edges in each small interval. An initial estimate $(\widehat{\mathcal{S}}_{\boldsymbol{\delta}}, \widehat{\mathbf{U}}_{\boldsymbol{\delta}}, \widehat{\mathbf{V}}_{\boldsymbol{\delta}}, \widehat{\mathbf{W}}_{\boldsymbol{\delta}})$ can be obtained by minimizing certain distance measure between $\mathcal{M}$ and $\mathcal{Y}_{\boldsymbol{\delta}}$, to be specified in Section 2.2.

Once we obtain the initial estimate $\widehat{\mathbf{W}}_{\boldsymbol{\delta}} = (\widehat{\mathbf{w}}_{1,\boldsymbol{\delta}}, ..., \widehat{\mathbf{w}}_{L,\boldsymbol{\delta}})^\top$, define

$$\widetilde{\mathbf{W}}_{\boldsymbol{\delta}} = (\widetilde{\mathbf{w}}_{1,\boldsymbol{\delta}}, ..., \widetilde{\mathbf{w}}_{L,\boldsymbol{\delta}})^\top = \sqrt{L} \widehat{\mathbf{W}}_{\boldsymbol{\delta}}((\widehat{\mathbf{W}}_{\boldsymbol{\delta}})^\top \widehat{\mathbf{W}}_{\boldsymbol{\delta}})^{-\frac{1}{2}},$$

where $(\widehat{\mathbf{W}}_{\boldsymbol{\delta}})^\top \widehat{\mathbf{W}}_{\boldsymbol{\delta}}$ is invertible with high probability as to be showed in the proof of Theorem 3. Though consistent, the estimation variance of $\widehat{\mathbf{W}}_{\boldsymbol{\delta}}$ can be exceedingly large when $\Delta_{\boldsymbol{\delta}}$ is too small. We then propose to merge adjacent small intervals with similar temporal embedding vectors $\widetilde{\mathbf{w}}_{l,\boldsymbol{\delta}}$, so as to shrink the estimation variance without compromising the estimation bias.

Let $\mathcal{P} = \{\mathcal{P}_1, ..., \mathcal{P}_K\}$ denote the adaptively merged intervals, where for any $l_1 \in \mathcal{P}_{k_1}$ and $l_2 \in \mathcal{P}_{k_2}$, it holds that $l_1 < l_2$ if $k_1 < k_2$. Then, it can be estimated as

$$\widehat{\mathcal{P}} = \arg\min_{\mathcal{P}} \sum_{k=1}^{K} \sum_{l \in \mathcal{P}_k} \|\widetilde{\mathbf{w}}_{l,\boldsymbol{\delta}} - \boldsymbol{\mu}_k\|^2, \tag{2}$$

where $\boldsymbol{\mu}_k = |\mathcal{P}_k|^{-1} \sum_{l \in \mathcal{P}_k} \widetilde{\mathbf{w}}_{l,\boldsymbol{\delta}}$. Note that (2) is equivalent to seeking change points in the sequence $(\widetilde{\mathbf{w}}_{1,\boldsymbol{\delta}}, ..., \widetilde{\mathbf{w}}_{L,\boldsymbol{\delta}})$, and thus can be efficiently solved by multiple change point detection algorithm (Hao et al., 2013; Niu et al., 2016). Further, define $\widehat{\eta}_k = \Delta_{\boldsymbol{\delta}} \max \widehat{\mathcal{P}}_k$, and thus $\widehat{\boldsymbol{\eta}} = (\widehat{\eta}_1, ..., \widehat{\eta}_K)^\top$ consists of the estimated endpoints of K adaptively merged intervals. Denote $\mathcal{Y}_{\widehat{\boldsymbol{\eta}}} \in \mathbb{R}^{n_1 \times n_2 \times K}$ with $(\mathcal{Y}_{\widehat{\boldsymbol{\eta}}})_{ijk} = |\mathcal{T}_{ij} \cap [\widehat{\eta}_{k-1}, \widehat{\eta}_k]|$ with $\widehat{\eta}_0 = 0$, the final estimate $(\widehat{\mathcal{S}}_{\widehat{\boldsymbol{\eta}}}, \widehat{\mathbf{U}}_{\widehat{\boldsymbol{\eta}}}, \widehat{\mathbf{V}}_{\widehat{\boldsymbol{\eta}}}, \widehat{\mathbf{W}}_{\widehat{\boldsymbol{\eta}}})$ is then obtained by minimizing the distance measure between $\mathcal{M}$ and $\mathcal{Y}_{\widehat{\boldsymbol{\eta}}}$.

7

2.2 Poisson point process

To devise a proper distance measure, let $y_{ij}(\cdot)$ be the point process that counts the number of directed edges out-node i sends to in-node j during $[0, T)$. Particularly, out-node i sends a directed edge to in-node j at time t if and only if $dy_{ij}(t) = 1$. Given $\mathbf{\Theta}(t) = (\theta_{ij}(t))_{n_1 \times n_2}$, we assume that $y_{ij}(\cdot)$'s are mutually independent Poisson processes such that

$$\mathbb{E}(dy_{ij}(t) \mid \theta_{ij}(t)) = \lambda_0 e^{\theta_{ij}(t)} dt \tag{3}$$

where $\theta_{ij}(t)$ is the underlying propensity for node pair (i, j) at time t, and $\lambda_0 > 0$ is the baseline intensity. The larger $\theta_{ij}(t)$ is, the more likely out-node i will send a directed edge to in-node j during $[t, t + dt)$. The log-likelihood function of $\{y_{ij}(t)\}_{1 \le i \le n_1, 1 \le j \le n_2}$ can become

$$l(\mathbf{\Theta}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \left\{ \sum_{t \in \mathcal{T}_{ij}} \log \lambda_{ij}(t) - \int_0^T \lambda_{ij}(s) ds \right\}, \tag{4}$$

where $\lambda_{ij}(t) = \lambda_0 \exp(\theta_{ij}(t))$. Note that λ_0 is fixed throughout the paper, but it could also be varying with t, which may require more involved treatment.

Let $\boldsymbol{\tau} = (\tau_0, \ldots, \tau_{n_3})^T$ denote a generic partition of $[0, T)$ with $0 = \tau_0 < \tau_1 < \ldots < \tau_{n_3} = T$. Particularly, $\boldsymbol{\tau}$ could be the equally spaced intervals $\boldsymbol{\delta}$ for the initial estimate or the adaptively merged intervals $\widehat{\boldsymbol{\eta}}$ for the final estimate in Section 2.1, and n_3 could be L or K, correspondingly. For any $\mathcal{M} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we define

$$l(\mathcal{M}; \boldsymbol{\tau}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{l=1}^{n_3} \left\{ m_{ijl} |\mathcal{T}_{ij} \cap [\tau_{l-1}, \tau_l)| - e^{m_{ijl}} \lambda_0 (\tau_l - \tau_{l-1}) \right\}. \tag{5}$$

Note that if $\mathbf{\Theta}(t)$ is roughly constant in each interval, we consider the regularized formulation,

$$(\widehat{\mathcal{S}}_{\boldsymbol{\tau}}, \widehat{\mathbf{U}}_{\boldsymbol{\tau}}, \widehat{\mathbf{V}}_{\boldsymbol{\tau}}, \widehat{\mathbf{W}}_{\boldsymbol{\tau}}) = \underset{\mathcal{S}, \mathbf{U}, \mathbf{V}, \mathbf{W}}{\arg \min} \left\{ -l(\mathcal{M}; \boldsymbol{\tau}) + \gamma_{\boldsymbol{\tau}} \mathcal{J}_{\boldsymbol{\tau}}(\mathbf{U}, \mathbf{V}, \mathbf{W}) \right\}, \tag{6}$$

where $\gamma_{\boldsymbol{\tau}}$ is the tuning parameter, $\mathcal{J}_{\boldsymbol{\tau}}(\mathbf{U}, \mathbf{V}, \mathbf{W})$ is the regularization term which takes the form

$$\mathcal{J}_{\boldsymbol{\tau}}(\mathbf{U}, \mathbf{V}, \mathbf{W}) = \frac{1}{4} \left\{ \left\| \frac{1}{n_1} \mathbf{U}^{\top} \mathbf{U} - \mathbf{I}_{r_1} \right\|_F^2 + \left\| \frac{1}{n_2} \mathbf{V}^{\top} \mathbf{V} - \mathbf{I}_{r_2} \right\|_F^2 + \left\| \frac{1}{n_3} \mathbf{W}^{\top} \mathbf{W} - \mathbf{I}_{r_3} \right\|_F^2 \right\},$$

encouraging the orthogonality among columns in $\mathbf{U}, \mathbf{V}$ and $\mathbf{W}$. A similar regularization term has also been employed in Han et al. (2022), which involves some additional tuning parameter and thus requires more computational efforts.

3 Computation

Define $\mathcal{C}_{\mathcal{S}} = \{\mathcal{S} \in \mathbb{R}^{r_1 \times r_2 \times r_3} : \|\mathcal{S}\|_F \leq c_{\mathcal{S}}\}$, $\mathcal{C}_{\mathbf{U}} = \{\mathbf{U} \in \mathbb{R}^{n_1 \times r_1} : \|\mathbf{U}\|_{2\to\infty} \leq c_1\}$, $\mathcal{C}_{\mathbf{V}} = \{\mathbf{V} \in \mathbb{R}^{n_2 \times r_2} : \|\mathbf{V}\|_{2\to\infty} \leq c_2\}$, and $\mathcal{C}_{\mathbf{W}} = \{\mathbf{W} \in \mathbb{R}^{n_3 \times r_3} : \|\mathbf{W}\|_{2\to\infty} \leq c_3\}$, where $c_{\mathcal{S}}, c_1, c_2$ and c_3 are constants. Here n_3 could be L and K and with a little abuse of notation, we use a generic $\mathcal{C}_{\mathbf{W}}$. For any convex set $\mathcal{C}$, denote $\mathcal{P}_{\mathcal{C}}$ to be the projection operator onto $\mathcal{C}$.

We develop an efficient projected gradient descent (PGD) updating algorithm to solve the optimization task in (6). Choose an initializer $(\mathcal{S}_{\boldsymbol{\tau}}^{(0)}, \mathbf{U}_{\boldsymbol{\tau}}^{(0)}, \mathbf{V}_{\boldsymbol{\tau}}^{(0)}, \mathbf{W}_{\boldsymbol{\tau}}^{(0)})$ such that $\mathcal{S}_{\boldsymbol{\tau}}^{(0)} \in \mathcal{C}_{\mathcal{S}}$, $\mathbf{U}_{\boldsymbol{\tau}}^{(0)} \in \mathcal{C}_{\mathbf{U}}$, $\mathbf{V}_{\boldsymbol{\tau}}^{(0)} \in \mathcal{C}_{\mathbf{V}}$ and $\mathbf{W}_{\boldsymbol{\tau}}^{(0)} \in \mathcal{C}_{\mathbf{W}}$, with $\mathbf{U}_{\boldsymbol{\tau}}^{(0)\top} \mathbf{U}_{\boldsymbol{\tau}}^{(0)} = n_1 \mathbf{I}_{r_1}$, $\mathbf{V}_{\boldsymbol{\tau}}^{(0)\top} \mathbf{V}_{\boldsymbol{\tau}}^{(0)} = n_2 \mathbf{I}_{r_2}$ and $\mathbf{W}_{\boldsymbol{\tau}}^{(0)\top} \mathbf{W}_{\boldsymbol{\tau}}^{(0)} = n_3 \mathbf{I}_{r_3}$. Given $(\mathcal{S}_{\boldsymbol{\tau}}^{(r)}, \mathbf{U}_{\boldsymbol{\tau}}^{(r)}, \mathbf{V}_{\boldsymbol{\tau}}^{(r)}, \mathbf{W}_{\boldsymbol{\tau}}^{(r)})$ and $\mathcal{M}_{\boldsymbol{\tau}}^{(r)} = [\mathcal{S}_{\boldsymbol{\tau}}^{(r)}; \mathbf{U}_{\boldsymbol{\tau}}^{(r)}, \mathbf{V}_{\boldsymbol{\tau}}^{(r)}, \mathbf{W}_{\boldsymbol{\tau}}^{(r)}]$, we implement the following updating scheme with step size $\zeta_{\boldsymbol{\tau}}$:

$$\begin{aligned}
\mathbf{U}_{\boldsymbol{\tau}}^{(r+1)} &= \mathcal{P}_{\mathcal{C}_{\mathbf{U}}} \left\{ \mathbf{U}_{\boldsymbol{\tau}}^{(r)} + \zeta_{\boldsymbol{\tau}} \left[n_1 \frac{\partial l(\mathcal{M}_{\boldsymbol{\tau}}^{(r)}; \boldsymbol{\tau})}{\partial \mathbf{U}} - \gamma_{\boldsymbol{\tau}} \mathbf{U}_{\boldsymbol{\tau}}^{(r)} \left(\frac{1}{n_1} \mathbf{U}_{\boldsymbol{\tau}}^{(r)\top} \mathbf{U}_{\boldsymbol{\tau}}^{(r)} - \mathbf{I}_{r_1} \right) \right] \right\}; \\
\mathbf{V}_{\boldsymbol{\tau}}^{(r+1)} &= \mathcal{P}_{\mathcal{C}_{\mathbf{V}}} \left\{ \mathbf{V}_{\boldsymbol{\tau}}^{(r)} + \zeta_{\boldsymbol{\tau}} \left[n_2 \frac{\partial l(\mathcal{M}_{\boldsymbol{\tau}}^{(r)}; \boldsymbol{\tau})}{\partial \mathbf{V}} - \gamma_{\boldsymbol{\tau}} \mathbf{V}_{\boldsymbol{\tau}}^{(r)} \left(\frac{1}{n_2} \mathbf{V}_{\boldsymbol{\tau}}^{(r)\top} \mathbf{V}_{\boldsymbol{\tau}}^{(r)} - \mathbf{I}_{r_2} \right) \right] \right\}; \\
\mathbf{W}_{\boldsymbol{\tau}}^{(r+1)} &= \mathcal{P}_{\mathcal{C}_{\mathbf{W}}} \left\{ \mathbf{W}_{\boldsymbol{\tau}}^{(r)} + \zeta_{\boldsymbol{\tau}} \left[n_3 \frac{\partial l(\mathcal{M}_{\boldsymbol{\tau}}^{(r)}; \boldsymbol{\tau})}{\partial \mathbf{W}} - \gamma_{\boldsymbol{\tau}} \mathbf{W}_{\boldsymbol{\tau}}^{(r)} \left(\frac{1}{n_3} \mathbf{W}_{\boldsymbol{\tau}}^{(r)\top} \mathbf{W}_{\boldsymbol{\tau}}^{(r)} - \mathbf{I}_{r_3} \right) \right] \right\}; \\
\mathcal{S}_{\boldsymbol{\tau}}^{(r+1)} &= \mathcal{P}_{\mathcal{C}_{\mathcal{S}}} \left\{ \mathcal{S}_{\boldsymbol{\tau}}^{(r)} + \zeta_{\boldsymbol{\tau}} \frac{\partial l(\mathcal{M}_{\boldsymbol{\tau}}^{(r)}; \boldsymbol{\tau})}{\partial \mathcal{S}} \right\},
\end{aligned} \tag{7}$$

and let $\mathcal{M}_{\boldsymbol{\tau}}^{(r+1)} = [\mathcal{S}_{\boldsymbol{\tau}}^{(r+1)}; \mathbf{U}_{\boldsymbol{\tau}}^{(r+1)}, \mathbf{V}_{\boldsymbol{\tau}}^{(r+1)}, \mathbf{W}_{\boldsymbol{\tau}}^{(r+1)}]$. We repeat the above updating scheme for a relative large number of iterations, say R, and let $(\widehat{\mathcal{S}}_{\boldsymbol{\tau}}, \widehat{\mathbf{U}}_{\boldsymbol{\tau}}, \widehat{\mathbf{V}}_{\boldsymbol{\tau}}, \widehat{\mathbf{W}}_{\boldsymbol{\tau}}) = (\mathcal{S}_{\boldsymbol{\tau}}^{(R)}, \mathbf{U}_{\boldsymbol{\tau}}^{(R)}, \mathbf{V}_{\boldsymbol{\tau}}^{(R)}, \mathbf{W}_{\boldsymbol{\tau}}^{(R)})$ be the initial estimation.

Remark 1. *We point out that the updating scheme in* (7) *differs from the standard projected gradient descent update, as different step sizes are used for updating different variables. Specifically, the step sizes for updating* $\mathbf{U}_{\boldsymbol{\tau}}^{(r)}, \mathbf{V}_{\boldsymbol{\tau}}^{(r)}, \mathbf{W}_{\boldsymbol{\tau}}^{(r)}, \mathcal{S}_{\boldsymbol{\tau}}^{(r)}$ *are* $n_1\zeta_{\boldsymbol{\tau}}, n_2\zeta_{\boldsymbol{\tau}}, n_3\zeta_{\boldsymbol{\tau}}$ *and* $\zeta_{\boldsymbol{\tau}}$, *respectively. This is the key difference from the algorithm in Han et al. (2022), which is also the reason that we do not require additional tuning parameter in* $\mathcal{J}_{\boldsymbol{\tau}}(\mathbf{U}, \mathbf{V}, \mathbf{W})$ *and* $\mathcal{J}_{\boldsymbol{\eta}}(\mathbf{U}, \mathbf{V}, \mathbf{W})$.

It remains to determine the number of merged interval K in (2), which differs from estimating the number of change points in $(\widetilde{\mathbf{w}}_{1,\boldsymbol{\delta}}, ..., \widetilde{\mathbf{w}}_{L,\boldsymbol{\delta}})$, since $\widetilde{\mathbf{w}}_{1,\boldsymbol{\delta}}, ..., \widetilde{\mathbf{w}}_{L,\boldsymbol{\delta}}$ are dependent with each other. In particular, we set

$$\widehat{K} = \arg\min_{S} \big\{ \min_{\mathcal{P}} \mathcal{L}(\mathcal{P}; S) + \nu_{nT}S \big\}, \tag{8}$$

where $\mathcal{P} = \{\mathcal{P}_1, ..., \mathcal{P}_S\}$ is an ordered partition of $[L]$ with S subsets, ν_{nT} is a quantity to be specified in Theorem 3, and

$$\mathcal{L}(\mathcal{P}; S) = \frac{1}{L} \sum_{s=1}^{S} \sum_{l \in \mathcal{P}_s} \| \widetilde{\mathbf{w}}_{l,\boldsymbol{\delta}} - \boldsymbol{\mu}_s \|^2, \tag{9}$$

with $\boldsymbol{\mu}_s = |\mathcal{P}_s|^{-1} \sum_{l \in \mathcal{P}_s} \widetilde{\mathbf{w}}_{l,\boldsymbol{\delta}}$. More importantly, $\widehat{K}$ can be showed to be a consistent estimator of K under mild conditions as in Theorem 3.

4 Theory

4.1 A new error bound for the PGD algorithm

We first derive the upper bound for the tensor estimation error in each iteration for the PGD algorithm (7). Let $\boldsymbol{\tau} = (\tau_1, ..., \tau_{n_3})^\top \in \mathbb{R}^{n_3}$ with $0 = \tau_0 < \tau_1 < ... < \tau_{n_3} = T$ be a generic partition of $[0, T)$. In particular, $\boldsymbol{\tau}$ could be $\boldsymbol{\delta}$ and $\widehat{\boldsymbol{\eta}}$. Further suppose there exists a quantity $H \in (0, T]$ such that $\min_{1 \leq l \leq n_3}(\tau_l - \tau_{l-1}) \asymp \max_{1 \leq l \leq n_3}(\tau_l - \tau_{l-1}) \asymp H$. Let

$$l(\mathcal{M}; \boldsymbol{\tau}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} \left\{ m_{ijk} \big| \mathcal{T}_{ij} \cap [\tau_{k-1}, \tau_k) \big| - e^{m_{ijk}} \lambda_0(\tau_k - \tau_{k-1}) \right\}, \quad \text{and define}$$

$$\widetilde{\mathcal{C}}_{\mathcal{M}, \boldsymbol{\tau}} = \Big\{ \mathcal{M} = [\mathcal{S}; \mathbf{U}, \mathbf{V}, \mathbf{W}] : \ \mathcal{S} \in \mathbb{R}^{r_1 \times r_2 \times r_3}, \mathbf{U} \in \mathbb{R}^{n_1 \times r_1}, \mathbf{V} \in \mathbb{R}^{n_2 \times r_2}, \mathbf{W} \in \mathbb{R}^{n_3 \times r_3},$$

$$\|\mathcal{S}\|_F = \|\mathbf{U}\|_F = \|\mathbf{V}\|_F = \|\mathbf{W}\|_F = 1, \ \text{and at least two of the followings hold:}$$

$$\|\mathbf{U}\|_{2 \to \infty} \leq \sqrt{n_1^{-1} \log n_1}, \ \ \|\mathbf{V}\|_{2 \to \infty} \leq \sqrt{n_2^{-1} \log n_2}, \ \ \|\mathbf{W}\|_{2 \to \infty} \leq \sqrt{n_3^{-1} \log n_3} \Big\}.$$

Theorem 1 establishes an upper bound for $\|\mathcal{M}_{\boldsymbol{\tau}}^{(R)} - \overline{\mathcal{M}}\|_F^2$ for any given value R.

Theorem 1. *Let* $\overline{\mathcal{M}} = [\overline{\mathcal{S}}; \overline{\mathbf{U}}, \overline{\mathbf{V}}, \overline{\mathbf{W}}] \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ *be an order-3 tensor with* $\overline{\mathcal{S}} \in \mathcal{C}_{\mathcal{S}}$, $\overline{\mathbf{U}} \in \mathcal{C}_{\mathbf{U}}$, $\overline{\mathbf{V}} \in \mathcal{C}_{\mathbf{V}}$, $\overline{\mathbf{W}} \in \mathcal{C}_{\mathbf{W}}$, $\overline{\mathbf{U}}^\top \overline{\mathbf{U}} = n_1 \mathbf{I}_{r_1}$, $\overline{\mathbf{V}}^\top \overline{\mathbf{V}} = n_2 \mathbf{I}_{r_2}$ *and* $\overline{\mathbf{W}}^\top \overline{\mathbf{W}} = n_3 \mathbf{I}_{r_3}$. *Suppose that* $\underline{\lambda}(\overline{\mathcal{M}}) \asymp \overline{\lambda}(\overline{\mathcal{M}}) \asymp \sqrt{n_1 n_2 n_3}$, $\gamma_{\boldsymbol{\tau}} \asymp n_1 n_2 n_3 H$ *and* $\xi_{\boldsymbol{\tau}} = \sup_{\mathcal{M} \in \widetilde{\mathcal{C}}_{\mathcal{M}, \boldsymbol{\tau}}} \big| \langle \nabla l(\overline{\mathcal{M}}; \boldsymbol{\tau}), \mathcal{M} \rangle \big| \preceq \sqrt{n_1 n_2 n_3} H$. *Then, there exists* $c_0 > 0$ *such that for any step size* $\zeta_{\boldsymbol{\tau}} = \frac{c}{n_1 n_2 n_3 H}$ *with* $0 < c < c_0$, *we have*

$$\frac{1}{n_1 n_2 n_3} \|\mathcal{M}_{\boldsymbol{\tau}}^{(R)} - \overline{\mathcal{M}}\|_F^2 \preceq \frac{\xi_{\boldsymbol{\tau}}^2}{n_1 n_2 n_3 H^2} + \frac{(1 - \kappa)^R}{n_1 n_2 n_3} \|\mathcal{M}_{\boldsymbol{\tau}}^{(0)} - \overline{\mathcal{M}}\|_F^2, \qquad (10)$$

for some constant $0 < \kappa < 1$.

The first term in the right-hand side of (10) quantifies the statistical error while the second term is the computational error in each iteration of (7). Note that the bias between $\overline{\mathcal{M}}$ and the "true" tensor that generates the process data through (3) is implicitly quantified by $\xi_{\boldsymbol{\tau}}$.

Particularly, if $\overline{\mathcal{M}}$ is chosen to be close to the "true" underlying tensor, then $\mathbb{E}\nabla l(\overline{\mathcal{M}}; \boldsymbol{\tau}) \approx \mathbf{0}$, which leads to a small $\xi_{\boldsymbol{\tau}}$ and an accurate estimation. On the other hand, choosing an $\overline{\mathcal{M}}$ that is far away from the "true" underlying tensor will lead to a large $\xi_{\boldsymbol{\tau}}$ and a relatively worse estimation.

It is interesting to remark that a similar upper bound for the tensor estimation error is established in Han et al. (2022). Yet, Theorem 1 differs from Han et al. (2022) in that the space $\widetilde{\mathcal{C}}_{\mathcal{M},\boldsymbol{\tau}}$ associated with the empirical process $\xi_{\boldsymbol{\tau}}$ is reduced by requiring additional incoherence conditions that $\|\mathbf{U}\|_{2\to\infty} \le \mu_1$, $\|\mathbf{V}\|_{2\to\infty} \le \mu_2$ and $\|\mathbf{W}\|_{2\to\infty} \le \mu_3$ with $\mu_k = \sqrt{\log(n_k)/n_k}$. The incoherence conditions for $\mathbf{U}, \mathbf{V}, \mathbf{W}$ in $\widetilde{\mathcal{C}}_{\mathcal{M},\boldsymbol{\tau}}$ are the key ingredient to derive the convergence rate for the tensor estimation error in a complete regime, which is in sharp contrast to the results in Han et al. (2022) and Cai et al. (2022) focusing only on the strong intensity regime.

4.2   Error analysis based on equally spaced intervals

Suppose the longitudinal network $\mathcal{G}_t$ is generated with $\boldsymbol{\Theta}^*(t) = \mathcal{S}^* \times_1 \mathbf{U}^* \times_2 \mathbf{V}^* \times_3 \mathbf{w}^*(t)$, where $\text{rank}(\Psi_s(\mathcal{S}^*)) = r_s$ for $s = 1, 2, 3$, $\mathbf{U}^{*\top}\mathbf{U}^* = n_1\mathbf{I}_{r_1}$, $\mathbf{V}^{*\top}\mathbf{V}^* = n_2\mathbf{I}_{r_2}$ and $\int_0^T \mathbf{w}^*(t)\mathbf{w}^*(t)^\top dt = T\mathbf{I}_{r_3}$. Further, suppose $\mathbf{w}^*(t)$ is a piecewise constant function of t in that $\mathbf{w}^*(t) = \mathbf{w}^*_{k,\boldsymbol{\eta}}$ for $t \in [\eta_{k-1}, \eta_k)$, where $\boldsymbol{\eta} = (\eta_1, ..., \eta_{K_0})^\top$ with $0 = \eta_0 < \eta_1 < ... < \eta_{K_0} = T$. Let $\mathbf{W}^*_{\boldsymbol{\eta}} \in \mathbb{R}^{K_0 \times r_3}$ with $(\mathbf{W}^*_{\boldsymbol{\eta}})_{[k,]} = \mathbf{w}^*_{k,\boldsymbol{\eta}}$, and $\mathcal{M}^*_{\boldsymbol{\eta}} = [\mathcal{S}^*; \mathbf{U}^*, \mathbf{V}^*, \mathbf{W}^*_{\boldsymbol{\eta}}]$. Define $d_{\min} = \min_{1 \le k \le K_0}(\eta_k - \eta_{k-1})/T$, $d_{\max} = \max_{1 \le k \le K_0}(\eta_k - \eta_{k-1})/T$ and $\Delta_{\boldsymbol{\eta}} = d_{\min}T$. Suppose $d_{\min} \asymp d_{\max} \asymp 1/K_0$, $\|\mathcal{S}^*\|_F \le c_{\mathcal{S}}/\max\{2, (K_0 d_{\min})^{-1/2}\}$, $\|\mathbf{U}^*\|_{2\to\infty} \le c_1$, $\|\mathbf{V}^*\|_{2\to\infty} \le c_2$ and $\sup_{t \in [0,T)} \|\mathbf{w}^*(t)\| \le c_3/\max\{2, \sqrt{K_0 d_{\max}}\}$. Let $n = \max\{n_1, n_2\}$ for simplicity and suppose $n_1 \asymp n_2 \asymp n$.

Recall that $\Delta_{\boldsymbol{\delta}} = T/L$. Theorem 2 establishes the tensor error bound for the initial estimate based on the equally spaced interval $\boldsymbol{\delta}$.

Theorem 2. (Initial estimate) *Suppose $L \succ K_0$ and $\gamma_{\boldsymbol{\delta}} \asymp n^2 L \Delta_{\boldsymbol{\delta}}$. Then, there exists*

a constant $c_0 > 0$ such that for a step size $\zeta_\delta = \frac{c}{n^2 L \Delta_\delta}$ with $0 < c < c_0$, with probability approaching 1, it holds true that

$$\frac{1}{n_1 n_2 L} \|\mathcal{M}_\delta^{(R)} - \mathcal{M}_\delta^*\|_F^2 \preceq I_{1,\delta} + I_{2,\delta} + I_{3,\delta},$$

where $\mathcal{M}_\delta^* = [\mathcal{S}^*; \mathbf{U}^*, \mathbf{V}^*, \mathbf{W}_\delta^*]$ with $\mathbf{W}_\delta^* \in \mathbb{R}^{L \times r_3}$ such that $(\mathbf{W}_\delta^*)_{[l,]} = \mathbf{w}^*(\delta_{l-1})$. Here

$$I_{1,\delta} = \begin{cases} \dfrac{1}{nT} + \dfrac{L}{n^2 T}, & \text{if } \log(nL) \prec \Delta_\delta \prec \dfrac{T}{K_0}, \\[2mm] \dfrac{\log(nL)}{nL} + \dfrac{\log(nL)}{n^2}, & \text{if } 1 \preceq \Delta_\delta \preceq \log(nL), \\[2mm] \dfrac{\log(nL)}{nT} + \dfrac{L\log(nL)}{n^2 T}, & \text{if } \dfrac{(n+L)^2 \log(nL)^3}{n^2 L} \prec \Delta_\delta \prec 1, \end{cases}$$

$I_{2,\delta} = K_0/L$ and $I_{3,\delta} = C(1-\kappa)^R$ for some constants C and $0 < \kappa < 1$.

Respectively, $I_{1,\delta}$, $I_{2,\delta}$ and $I_{3,\delta}$ correspond to the estimation variance, the bias induced by network merging, and the computational error of (7) after R iterations. If we suppress the logarithmic factor, the estimation variance $I_{1,\delta} \asymp \frac{1}{nT} + \frac{L}{n^2 T}$, which matches up with the minimax lower bound for the tensor estimation error in Poisson PCA (Han et al., 2022). More importantly, thanks to the new error bound for the PGD algorithm in Theorem 1, the strong intensity condition $\Delta_\delta \succ \log(nL)$ as required in Han et al. (2022) and Cai et al. (2022) can be relaxed, and similar upper bound can be obtained even when Δ_δ decays to 0. This is due to the fact that the shrunk $\widetilde{\mathcal{C}}_{\mathbf{M},\delta}$ with additional incoherence conditions enables us to take advantage of a tighter concentration inequalities to bound the empirical process ξ_δ. To the best of our knowledge, Theorem 2 gives the first Poisson tensor estimation error bound in both weak intensity regime with $\Delta_\delta \prec 1$ and medium intensity regime with $1 \preceq \Delta_\delta \preceq \log(nL)$.

Furthermore, with a relatively large value of R, the computational error $I_{3,\delta}$ is dominated by $I_{1,\delta} + I_{2,\delta}$. Then, the convergence rate of $\|\mathcal{M}_\delta^{(R)} - \mathcal{M}_\delta^*\|_F^2$ is largely determined by the

trade-off between $I_{1,\boldsymbol{\delta}}$ and $I_{2,\boldsymbol{\delta}}$. Corollary 1 specifies the convergence rate for the estimation error in the weak intensity regime.

Corollary 1. *Suppose all the conditions in Theorem 2 are satisfied, and $\log(n)^3 \prec T \prec \frac{n^2}{\log(n)^3}$. Then, choosing $\frac{\sqrt{T}\log(n)^{3/2}}{n} \prec \Delta_{\boldsymbol{\delta}} \prec 1$, we have*

$$\frac{1}{n_1 n_2 L}\|\mathcal{M}_{\boldsymbol{\delta}}^{(R)} - \mathcal{M}_{\boldsymbol{\delta}}^*\|_F^2 \preceq_P \frac{K_0}{L}.$$

Remark 2. *Corollary 1 assures the validity of employing small intervals with $\Delta_{\boldsymbol{\delta}} \prec 1$ in estimating the underlying tensor, to which the existing results (Han et al., 2022; Cai et al., 2022) requiring the strong intensity assumption may not apply. It is also interesting to point out that the derived error bound in the weak intensity regime also provides practical guideline for network merging. By Corollary 1, we will get a faster convergence rate as $\frac{K_0 \log(n)^2}{n\sqrt{T}}$ with $\Delta_{\boldsymbol{\delta}} \asymp \frac{\sqrt{T}\log(n)^2}{n}$ or $L \asymp \frac{n\sqrt{T}}{\log(n)^2}$, in contrast to the rate $\frac{K_0 \log(n)^2}{T}$ obtained in the strong intensity regime with $\Delta_{\boldsymbol{\delta}} \asymp \log(n)^2$ or $L \asymp \frac{T}{\log(n)^2}$ (Han et al., 2022; Cai et al., 2022). The intuition is that if T diverges very slowly, then one prefers to choose a relatively small $\Delta_{\boldsymbol{\delta}}$ or large L to reduce the bias $I_{2,\boldsymbol{\delta}} = K_0\Delta_{\boldsymbol{\delta}}/T$.*

4.3 Error analysis based on adaptively merged intervals

Define $\rho = \min_{k\in[K_0]} \|\mathbf{w}_{k,\boldsymbol{\eta}}^* - \mathbf{w}_{k-1,\boldsymbol{\eta}}^*\|$ and suppose $\rho \succeq 1$. Denote $r_{nT} = I_{1,\boldsymbol{\delta}} + I_{2,\boldsymbol{\delta}}$ as the upper bound in Theorem 2. Theorem 3 shows that (8) gives a consistent estimate of K_0, and (2) further results in a precise recovery of the true partition $\boldsymbol{\eta}$ with overwhelming probability.

Theorem 3. (Consistency of partition) *Suppose all the conditions of Theorem 2 are satisfied, and $r_{nT} \prec \nu_{nT} \prec 1/K_0$. Then as n and T grow to infinity, we have $\Pr(\widehat{K} = K_0) \to 1$ and $\|\widehat{\boldsymbol{\eta}} - \boldsymbol{\eta}\|_\infty \preceq_P Tr_{nT}$.*

Remark 3. *It is clear that the consistency of $\widehat{K}$ is guaranteed with a wide range of ν_{nT}. Specifically, the condition $\nu_{nT} \prec 1/K_0$ implies that $\widehat{K} \geq K_0$, whereas $\nu_{nT} \succ r_{nT}$ guarantees $\widehat{K} \leq K_0$. More importantly, Theorem 3 provides valuable guidelines for choosing L and ν_{nT}. For fixed K_0, if $T \prec \frac{n^2}{\log(n)^3}$, we can choose $L = \frac{n\sqrt{T}}{\log(n)^2}$ and $\nu_{nT} = \frac{\log(n)}{n^{1/2}T^{1/4}}$; if $T \succeq \frac{n^2}{\log(n)^3}$, we can choose $L = \frac{n\sqrt{T}}{\log(T)^3}$ and $\nu_{nT} = \frac{\log(T)^{3/2}}{n^{1/2}T^{1/4}}$.*

Given that the true partition $\boldsymbol{\eta}$ is accurately estimated by $\widehat{\boldsymbol{\eta}}$, Theorem 4 further shows that the estimate $\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)}$ based on the adaptively merged intervals $\widehat{\boldsymbol{\eta}}$ can attain a faster rate of convergence than that in Theorem 2.

Theorem 4. (Improved estimate via adaptive merging) *Suppose all the conditions of Theorem 3 are satisfied, $T \succeq K_0 \log(nK_0)^3$ and $\gamma_{\boldsymbol{\eta}} \asymp n^2 K_0 \Delta_{\boldsymbol{\eta}}$. Then, there exists $c_0 > 0$ such that for a step size $\zeta_{\boldsymbol{\eta}} = \frac{c}{n^2 K_0 \Delta_{\boldsymbol{\eta}}}$ with $0 < c < c_0$, with probability approaching 1, we have*

$$\frac{1}{n_1 n_2 K_0} \|\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)} - \mathcal{M}_{\boldsymbol{\eta}}^*\|_F^2 \preceq I_{1,\boldsymbol{\eta}} + I_{2,\boldsymbol{\eta}} + I_{3,\boldsymbol{\eta}},$$

where $I_{1,\boldsymbol{\eta}} = \frac{1}{nT} + \frac{K_0}{n^2 T}$,

$$I_{2,\boldsymbol{\eta}} = \begin{cases} K_0^2 r_{nT}^2, & \text{if } Tr_{nT} \succ \log(nK_0), \\ K_0^2 r_{nT}^2 \log(nK_0)^4, & \text{if } 1 \preceq Tr_{nT} \preceq \log(nK_0), \\ \dfrac{K_0^2 \log(nK_0)^4}{T^2}, & \text{if } Tr_{nT} \prec 1, \end{cases}$$

and $I_{3,\boldsymbol{\eta}} = C(1-\kappa)^R$ for some constants C and $0 < \kappa < 1$.

Similarly, $I_{1,\boldsymbol{\eta}}$, $I_{2,\boldsymbol{\eta}}$ and $I_{3,\boldsymbol{\eta}}$ correspond to the estimation variance, the bias induced by adaptively merging, and the computational error of (7) after R iterations, respectively. It is clear that $I_{1,\boldsymbol{\eta}}$ is much smaller than $I_{1,\boldsymbol{\delta}}$ in Theorem 2 where the term $\frac{L}{n^2 T}$ is reduced to $\frac{K_0}{n^2 T}$. The convergence rate for the bias term, $I_{2,\boldsymbol{\eta}}$, takes different forms depending on the term Tr_{nT}. Specifically, Corollary 2 gives the convergence rate for the estimation error of $\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)}$.

15

Corollary 2. *Suppose all the conditions in Theorem 4 are satisfied. If $T \succeq \frac{n^2}{\log(n)^3}$, then choosing $\Delta_{\boldsymbol{\delta}} \succ \log(nL)$ leads to*

$$\frac{1}{n_1 n_2 K_0}\|\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)} - \mathcal{M}_{\boldsymbol{\eta}}^{*}\|_F^2 \preceq_P \frac{1}{nT} + \frac{K_0}{n^2 T} + \frac{K_0^2 L^2}{n^4 T^2} + \frac{K_0^4}{L^2};$$

and if $T \prec \frac{n^2}{\log(n)^3}$, choosing $\frac{\sqrt{T}\log(n)^{3/2}}{n} \prec \Delta_{\boldsymbol{\delta}} \prec \frac{1}{K_0}$ leads to

$$\frac{1}{n_1 n_2 K_0}\|\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)} - \mathcal{M}_{\boldsymbol{\eta}}^{*}\|_F^2 \preceq_P \frac{1}{nT} + \frac{K_0}{n^2 T} + \frac{K_0^2 \log(nK_0)^4}{T^2}.$$

Remark 4. *Let K_0 be a fixed constant, and we compare the estimates $\mathcal{M}_{\widehat{\boldsymbol{\delta}}}^{(R)}$ based on the equally spaced intervals and $\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)}$ based on the adaptively merged intervals. If $T \succeq \frac{n^2}{\log(n)^3}$, then $\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)}$ converges to 0 at a faster rate of $\frac{1}{nT} + \frac{\log(T)^6}{n^2 T}$, whereas the convergence rate of $\mathcal{M}_{\widehat{\boldsymbol{\delta}}}^{(R)}$ with $L = \frac{n\sqrt{T}}{\log(T)^3}$ is of order $\frac{\log(T)^3}{n\sqrt{T}}$. If $T \prec \frac{n^2}{\log(n)^3}$, then the convergence rates of $\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)}$ and $\mathcal{M}_{\widehat{\boldsymbol{\delta}}}^{(R)}$ are of order $\frac{1}{nT} + \frac{\log(n)^4}{T^2}$ and $\frac{\log(n)^2}{n\sqrt{T}}$ with $L = \frac{n\sqrt{T}}{\log(n)^2}$, where $\mathcal{M}_{\widehat{\boldsymbol{\eta}}}^{(R)}$ is still advantageous as long as $T \succ n^{2/3}\log(n)^{4/3}$.*

By Corollaries 1 and 2, Table 1 summarizes the convergence rates of the proposed method. Comparing with the error rate $\frac{\log(n)^2}{T}$ obtained in the strong intensity regime (Han et al., 2022; Cai et al., 2022), it is clear that the proposed method achieves much faster convergence rates in all regimes.

Regime	Choice of L	Adaptive Merging	Error Rate
$T \succeq \frac{n^2}{\log(n)^3}$	$\frac{n\sqrt{T}}{\log(T)^3}$	Yes	$\frac{1}{nT} + \frac{\log(T)^6}{n^2 T}$
$n^{2/3}\log(n)^{4/3} \preceq T \prec \frac{n^2}{\log(n)^3}$	$\frac{n\sqrt{T}}{\log(n)^2}$	Yes	$\frac{1}{nT} + \frac{\log(n)^4}{T^2}$
$T \preceq n^{2/3}\log(n)^{4/3}$	$\frac{n\sqrt{T}}{\log(n)^2}$	No	$\frac{\log(n)^2}{n\sqrt{T}}$

Table 1: Convergence rates for the proposed method in different regimes.

5 Numerical experiments

5.1 Simulation examples

We investigate the finite-sample performance of the proposed method, and compare it with a number of existing tensor decomposition methods, including a modified Poisson tensor PCA (Han et al., 2022), higher-order SVD (De Lathauwer et al., 2000a) and higher-order orthogonal iteration (De Lathauwer et al., 2000b). We denote them as $\mathrm{AM}(\widehat{K})$, $\mathrm{ES}(L)$, HOSVD and HOOI for short, where $\mathrm{AM}(\widehat{K})$ is based on an estimated number of adaptively merged intervals and $\mathrm{ES}(L)$ is a slightly modified version of Poisson tensor PCA built on L equally spaced intervals. Their numeric performance is assessed by the average tensor estimation error based on the corresponding intervals.

Example 1. The dataset is generated based on the longitudinal network embedding model, where the ranks are set as $r_1 = r_2 = r_3 = 2$. The columns of $\mathbf{U}^*/\sqrt{n_1}$ and $\mathbf{V}^*/\sqrt{n_2}$ are generated uniformly from $\mathbb{O}_{n,2}$, while the columns of $\mathbf{W}_{\boldsymbol{\eta}}^*$ are randomly generated such that $\int_0^T \mathbf{w}^*(t)\mathbf{w}^*(t)^\top dt = T\mathbf{I}_2$. For $\mathcal{S}^*$, the diagonal entries are set to be 0.5 and the rest entries 0.

Example 2. The dataset is generated based on the dynamic stochastic co-block model, where the block membership remains the same within each time interval but changes from one interval to the next. Specifically, let $(\mathcal{M}_{\boldsymbol{\eta}}^*)_{[,,k]} = \mathbf{Z}_k \mathbf{B}(\mathbf{X}_k)^\top$, where $\mathbf{Z}_k \in \{0,1\}^{n_1 \times d}, \mathbf{X}_k \in \{0,1\}^{n_2 \times d}$ and $\mathbf{B} \in (0,1)^{d \times d}$. Here, each row of $\mathbf{Z}_k$ and $\mathbf{X}_k$ has exact one 1, representing the block membership for corresponding out-node and in-node, respectively. Let $\psi_{ik} \in [d]$ and $\phi_{jk} \in [d]$ denote the membership for the i-th out-node and the j-th in-node in within the time interval $[\eta_{k-1}, \eta_k)$. We generate ψ_{ik} and ϕ_{jk} uniformly from $[d]$, and $(\mathbf{B})_{ij}$ uniformly from $[-0.5, 0.5]$, where d is set to be 3.

In both examples, we let T take value in $\{300, 400, 500\}$, and $n_1 = n_2 = n \in \{50, 100\}$. We set $\lambda_0 = 0.05$, $K = T/100$, and the partition $\boldsymbol{\eta} \in \mathbb{R}^K$ is constructed in a way such that each η_k

17

is randomly generated from $[0, T)$, where the length ratio for the largest and smallest intervals is no larger than 3. For the proposed method, we initialize it with $ES(L)$ with $L = 100K$ on equally spaced intervals, and obtain the finial estimate $AM(\widehat{K})$ based on the adaptively merged intervals. In addition, we also apply HOSVD and HOOI on $\log((\mathcal{Y}_{\boldsymbol{\delta}} + 1/2)/(\lambda_0 \Delta_{\boldsymbol{\delta}}))$ based on the equally spaced intervals (Han et al., 2022), where $(\mathcal{Y}_{\boldsymbol{\delta}})_{ijl} = |\mathcal{T}_{ij} \cap [\delta_{l-1}, \delta_l)|$. The averaged tensor estimation errors over 50 independent replications and their standard errors for each method are summarized in Tables 2 and 3.

Table 2: The averaged tensor estimation errors and their standard errors (in parentheses) for various methods over 50 independent replications in Example 1. All numbers are multiplied by 100.

	Method	$T = 300$	$T = 400$	$T = 500$
	$AM(\widehat{K})$	**0.376**(0.039)	**0.272**(0.032)	**0.252**(0.035)
$n = 50$	$ES(L)$	1.064(0.065)	1.124(0.055)	1.002(0.052)
	HOSVD	610.2(0.139)	609.2(0.111)	610.1(0.119)
	HOOI	610.0(0.127)	608.8(0.105)	608.9(0.100)
	$AM(\widehat{K})$	**0.204**(0.017)	**0.152**(0.013)	**0.115**(0.008)
$n = 100$	$ES(L)$	0.350(0.024)	0.311(0.015)	0.560(0.020)
	HOSVD	606.0(0.060)	608.5(0.062)	610.3(0.052)
	HOOI	606.1(0.065)	608.2(0.059)	609.6(0.050)

It is evident that $AM(\widehat{K})$ has delivered superior numerical performance and outperforms the other three competitors in all scenarios of both examples, showing that adaptively merged intervals indeed lead to substantial advantage in terms of estimation. It is also interesting to remark that $ES(L)$ shows great advantage over HOSVD and HOOI, suggesting superiority of likelihood-based methods in longitudinal network estimation.

We now scrutinize how the tensor estimation error, as well as the embedding error, is affected by different choices of L in Example 1 with $n = 50$, $T = 400$ and $K = 4$. The left panel of Figure 1 shows the average tensor estimation errors of $ES(L)$ over 50 independent replications with different L. Clearly, as L increases, the error dramatically decreases at

Table 3: The averaged tensor estimation errors and their standard errors (in parentheses) for various methods over 50 independent replications in Example 2. All numbers are multiplied by 100.

	Method	$T = 300$	$T = 400$	$T = 500$
	$\text{AM}(\widehat{K})$	**3.708**(0.289)	**4.357**(0.253)	**4.909**(0.211)
$n = 50$	$\text{ES}(L)$	5.938(0.836)	7.521(1.179)	10.15(1.133)
	HOSVD	574.5(0.131)	571.4(0.092)	573.7(0.096)
	HOOI	574.2(0.130)	571.1(0.104)	573.4(0.093)
	$\text{AM}(\widehat{K})$	**1.500**(0.080)	**1.624**(0.067)	**1.816**(0.077)
$n = 100$	$\text{ES}(L)$	1.528(0.165)	1.787(0.259)	2.117(0.269)
	HOSVD	571.1(0.065)	562.8(0.059)	568.7(0.052)
	HOOI	570.6(0.064)	562.4(0.069)	568.3(0.056)

first, and then slowly increases. This is because the bias induced by the partition with a small number of intervals dominates the tensor estimation error in each interval, which will be reduced dramatically as L increases. Yet, as L becomes larger, the estimation variance begins to dominate the tensor estimation error, and it increases along with L. This phenomenon validates the asymptotic upper bound in Theorem 2. The averaged tensor estimation error of $\text{AM}(\widehat{K})$ with $\widehat{K} = 4$ adaptively merged intervals is represented by the red dotted line, which is substantially smaller than that of all the methods based on equally spaced intervals, demonstrating the advantage of the proposed methods in Theorem 4.

5.2 Real example

We apply the proposed method to analyze a longitudinal network based on the militarized interstate dispute dataset (Palmer et al., 2022). The dataset consists of all the major interstate disputes and involved countries during 1895-2014. It can be converted into a longitudinal network with nodes representing all countries ever involved in any dispute over the years. Particularly, we set $dy_{ij}(t) = 1$ if country i cooperated with country j in a militarized in-
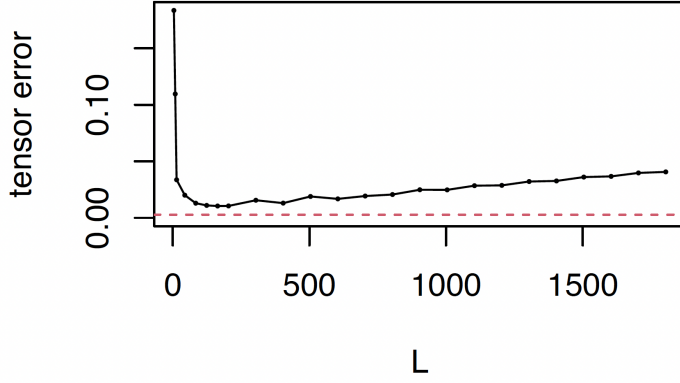
Figure 1: The average tensor estimation errors based on equal spaced intervals with different values of L over 50 independent replications. The red dotted lines are the average estimation errors of the estimate based on adaptively merged intervals.

terstate dispute occurred at time t. We keep it as 1 for the following years until a dispute occurred between themselves, and then $dy_{ij}(t)$ changes to 0 and remains until the next cooperation. This pre-processing step leads to a longitudinal network with $n_1 = n_2 = 195$ nodes and 110066 temporal edges, and the time stamps range from 0 to $T = 120$ years. We apply the proposed method with $\Delta_{\boldsymbol{\delta}} = 5$ years and thus $L = 24$, where the ranks are set to be $(r_1, r_2, r_3) = (2, 2, 2)$ following a similar rank selection procedure in Han et al. (2022).

To assess the numeric performance, we randomly split the node pairs into 5 disjoint subsets $\{\mathcal{P}_p\}_{p=1}^5$. For each p, we obtain the estimated tensor $\widehat{\mathcal{M}}^{(p)}$ on $\mathcal{P}_{-p} = [n_1] \times [n_2] \backslash \mathcal{P}_p$, and validate the estimation accuracy on $\mathcal{P}_p$,

$$\text{err}^{(p)} = \frac{\|(\mathcal{T} - \widehat{\mathcal{Y}}^{(p)}) \circ \mathbf{1}_{\mathcal{P}_p}\|_F}{\|\mathcal{T} \circ \mathbf{1}_{\mathcal{P}_p}\|_F},$$

where $\mathcal{T} = (\mathcal{T}_{ij})_{n_1 \times n_2}$ and $\widehat{\mathcal{Y}}^{(p)} \in \mathbb{R}^{n_1 \times n_2}$ contain the true and estimated numbers of temporal edges for each node pair (i, j), $\mathbf{1}_{\mathcal{P}_p} \in \mathbb{R}^{n_1 \times n_2}$ is the indicator matrix for $\mathcal{P}_p$, and $\circ$ denotes the matrix Hadamard product. Then, the testing error is calculated as $\text{err} = \sum_{p=1}^5 \text{err}^{(p)}/5$. For AM($\widehat{K}$), $\widehat{\mathcal{Y}}_{\widehat{\boldsymbol{\eta}}}^{(p)}$ is obtained by $(\widehat{\mathcal{Y}}_{\widehat{\boldsymbol{\eta}}}^{(p)})_{ij} = \sum_{k=1}^{\widehat{K}} \lambda_0 \exp((\widehat{\mathcal{M}}_{\widehat{\boldsymbol{\eta}}}^{(p)})_{ijk})(\widehat{\eta}_k - \widehat{\eta}_{k-1})$, whereas $(\widehat{\mathcal{Y}}_{\boldsymbol{\delta}}^{(p)})_{ij} = \sum_{l=1}^{\widehat{L}} \lambda_0 \exp((\widehat{\mathcal{M}}_{\boldsymbol{\delta}}^{(p)})_{ijk})(\delta_l - \delta_{l-1})$ for ES(L). The estimates by HOSVD and HOOI

are obtained in the same way as in Section 5.1. The averaged testing errors and their standard errors for the competing methods over 50 times replications are provided in Table 4, supporting the advantage of adaptively merged intervals in $\text{AM}(\widehat{K})$ over the other three competitors.

Table 4: The average testing errors and standard errors (in parentheses) for various methods over 50 replications.

$\text{AM}(\widehat{K})$	ES(L)	HOSVD	HOOI
0.739(0.037)	0.752(0.087)	1.160(0.002)	1.163(0.002)

Furthermore, the output of $\text{AM}(\widehat{K})$ yields that $\widehat{K} = 6$ and $\widehat{\boldsymbol{\eta}} = (20, 45, 50, 95, 105, 120)$, and thus the adaptively merged time intervals are 1895-1914, 1915-1939, 1940-1944, 1945-1989, 1990-1999 and 2000-2014. These intervals appear to be closely related with a number of major world-wide events: before WWI, recess between WWI and WWII, WWII, Cold War, the 90s, and the 21st century. The estimated temporal embedding vectors $\{\widehat{\mathbf{w}}_{l,\boldsymbol{\delta}}\}_{l=1}^{L}$ are shown in Figure 2, where $\widehat{\mathbf{w}}_{l,\boldsymbol{\delta}}$ in different merged time intervals, represented by different colors, are well separated.
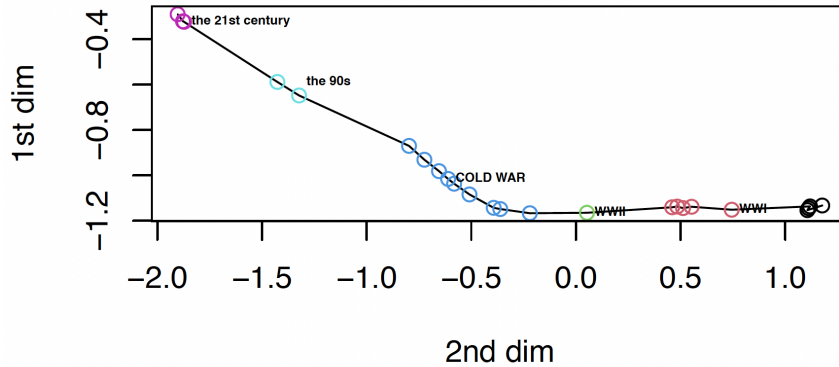

Figure 2: The estimated temporal embedding vectors $\{\widehat{\mathbf{w}}_{l,\boldsymbol{\delta}}\}_{l=1}^{9}$, where colors represent different merged time intervals.

6 Discussion

In this paper, we propose an efficient estimation framework for longitudinal network, leveraging strengths of adaptive network merging, tensor decomposition and point process. A thorough analysis is conducted to quantify the asymptotic behavior of the proposed method, which shows that adaptively network merging leads to substantially improved estimation accuracy compared with existing competitors in literature. The theoretical analysis also provides a guideline for network merging under various scenarios. The advantage of the proposed method is supported in the numerical experiments on both synthetic and real longitudinal networks. The proposed estimation framework can be further extended to incorporate edge-wise or node-wise covariates or employ some more general counting processes, which will be left for future investigation.

Acknowledgment

References

Aggarwal, C. and Subbian, K. (2014). Evolutionary network analysis: A survey. *ACM Computing Surveys (CSUR)*, 47:1–36.

Avena-Koenigsberger, A., Misic, B., and Sporns, O. (2018). Communication dynamics in complex brain networks. *Nature Reviews Neuroscience*, 19:17–33.

Cai, J.-F., Li, J., and Xia, D. (2022). Generalized low-rank plus sparse tensor estimation

by fast riemannian optimization. *Journal of the American Statistical Association*, pages 1–17.

Cranmer, S. J. and Desmarais, B. A. (2011). Inferential network analysis with exponential random graph models. *Political Analysis*, 19:66–86.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000a). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21:1253–1278.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000b). On the best rank-1 and rank-$(r_1, r_2, ..., r_n)$ approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21:1324–1342.

De Ruiter, P. C., Wolters, V., and Moore, J. C. (2005). *Dynamic food webs: multispecies assemblages, ecosystem development and environmental change.* Elsevier.

Han, R., Willett, R., and Zhang, A. R. (2022). An optimal statistical and computational framework for generalized tensor estimation. *The Annals of Statistics*, 50:1–29.

Hanneke, S., Fu, W., and Xing, E. P. (2010). Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605.

Hao, N., Niu, Y. S., and Zhang, H. (2013). Multiple change-point detection via a screening and ranking algorithm. *Statistica Sinica*, 23:1553.

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

Holme, P. and Saramäki, J. (2012). Temporal networks. *Physics Reports*, 519:97–125.

Huang, S., Weng, H., and Feng, Y. (2023, in press). Spectral clustering via adaptive layer aggregation for multi-layer networks. *Journal of Computational and Graphical Statistics*.

Kim, B., Lee, K. H., Xue, L., and Niu, X. (2018). A review of dynamic network models with latent variables. *Statistics Surveys*, 12:105.

Kinne, B. J. (2013). Network dynamics and the evolution of international cooperation. *American Political Science Review*, 107:766–785.

Lyu, Z., Xia, D., and Zhang, Y. (2021). Latent space model for higher-order networks and generalized tensor decomposition. *arXiv preprint arXiv:2106.16042*.

Matias, C. and Miele, V. (2017). Statistical clustering of temporal networks through a dynamic stochastic block model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79:1119–1141.

Matias, C., Rebafka, T., and Villers, F. (2018). A semiparametric extension of the stochastic block model for longitudinal networks. *Biometrika*, 105:665–680.

Niu, Y. S., Hao, N., and Zhang, H. (2016). Multiple change-point detection: A selective overview. *Statistical Science*, 31:611–623.

Palmer, G., McManus, R. W., D'Orazio, V., Kenwick, M. R., Karstens, M., Bloch, C., Dietrich, N., Kahn, K., Ritter, K., and Soules, M. J. (2022). The mid5 dataset, 2011–2014: Procedures, coding rules, and description. *Conflict Management and Peace Science*, 39:470–482.

Perry, P. O. and Wolfe, P. J. (2013). Point process modelling for directed interaction networks. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75:821–849.

Perry-Smith, J. E. and Shalley, C. E. (2003). The social side of creativity: A static and dynamic social network perspective. *Academy of Management Review*, 28:89–106.

Sewell, D. K. and Chen, Y. (2015). Latent space models for dynamic networks. *Journal of the American Statistical Association*, 110:1646–1657.

Sewell, D. K. and Chen, Y. (2016). Latent space models for dynamic networks with weighted edges. *Social Networks*, 44:105–116.

Sit, T., Ying, Z., and Yu, Y. (2021). Event history analysis of dynamic networks. *Biometrika*, 108:223–230.

Snijders, T. A. (2017). Stochastic actor-oriented models for network dynamics. *Annual Review of Statistics and Its Application*, 4:343–363.

Snijders, T. A., Koskinen, J., and Schweinberger, M. (2010). Maximum likelihood estimation for social network dynamics. *The Annals of Applied Statistics*, 4:567.

Soliman, H., Zhao, L., Huang, Z., Paul, S., and Xu, K. S. (2022). The multivariate community hawkes model for dependent relational events in continuous-time networks. In *International Conference on Machine Learning*, pages 20329–20346. PMLR.

Ulanowicz, R. E. (2004). Quantitative methods for ecological network analysis. *Computational Biology and Chemistry*, 28:321–339.

Voytek, B. and Knight, R. T. (2015). Dynamic network communication as a unifying neural basis for cognition, development, aging, and disease. *Biological Psychiatry*, 77:1089–1097.

Vu, D., Hunter, D., Smyth, P., and Asuncion, A. (2011a). Continuous-time regression models for longitudinal networks. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc.

Vu, D. Q., Asuncion, A. U., Hunter, D. R., and Smyth, P. (2011b). Dynamic egocentric models for citation networks. In *International Conference on Machine Learning*, page 857–864.

Zhang, J., He, X., and Wang, J. (2022). Directed community detection with network embedding. *Journal of the American Statistical Association*, 117:1809–1819.

Zhen, Y. and Wang, J. (2023, in press). Community detection in general hypergraph via graph embedding. *Journal of the American Statistical Association*.