Journal of Machine Learning Research 23 (2022) 1-49 Submitted 10/21; Revised 5/22; Published 7/22

Efficient Inference for Dynamic Flexible Interactions of Neural Populations

Feng Zhou[1,2] ZHOUFENG6288@TSINGHUA.EDU.CN
Quyu Kong[3,4] QUYU.KONG@ANU.EDU.AU
Zhijie Deng[1] DZJ17@MAILS.TSINGHUA.EDU.CN
Jichao Kan[4] JICHAO.KAN@STUDENT.UTS.EDU.AU
Yixuan Zhang[4] YIXUAN.ZHANG@STUDENT.UTS.EDU.AU
Cheng Feng[2,5] CHENG.FENG@SIEMENS.COM
Jun Zhu[1,2∗] DCSZJ@TSINGHUA.EDU.CN

[1] *Dept. of Comp. Sci. & Tech., BNRist Center, THU-Bosch Joint ML Center, Tsinghua University*
[2] *THU-Siemens Joint Research Center for Industrial Intelligence and Internet of Things*
[3] *Research School of Computer Science, Australian National University*
[4] *Data Science Institute, University of Technology Sydney*
[5] *Siemens AG*

Editor: David Sontag

Abstract

Hawkes process provides an effective statistical framework for analyzing the interactions of neural spiking activities. Although utilized in many real applications, the classic Hawkes process is incapable of modeling inhibitory interactions among neural population. Instead, the nonlinear Hawkes process allows for modeling a more flexible influence pattern with excitatory or inhibitory interactions. This work proposes a flexible nonlinear Hawkes process variant based on sigmoid nonlinearity. To ease inference, three sets of auxiliary latent variables (Pólya-Gamma variables, latent marked Poisson processes and sparsity variables) are augmented to make functional connection weights appear in a Gaussian form, which enables simple iterative algorithms with analytical updates. As a result, the efficient Gibbs sampler, expectation-maximization (EM) algorithm and mean-field (MF) approximation are derived to estimate the interactions among neural populations. Furthermore, to reconcile with time-varying neural systems, the proposed time-invariant model is extended to a dynamic version by introducing a Markov state process. Similarly, three analytical iterative inference algorithms: Gibbs sampler, EM algorithm and mean-field approximation are derived. We compare the accuracy and efficiency of these inference algorithms on synthetic data, and further experiment on real neural recordings to demonstrate that the developed models achieve superior performance over the state-of-the-art competitors.

Keywords: nonlinear Hawkes process, Pólya-Gamma augmentation, conditional conjugate, time-varying interaction

1. Introduction

One of the most important tasks in neuroscience is to examine the neuronal activity in the cerebral cortex under varying experimental conditions. Recordings of neuronal activity are

∗. The corresponding author

represented through a series of action potentials or spike trains. The transmitted information and functional connection between neurons are considered to be primarily represented by spike trains (Brown et al., 2002, 2004; Kass and Ventura, 2001; Kass et al., 2014). A spike train is a sequence of recorded times at which a neuron fires an action potential and each spike specifies a timestamp. Spikes occur irregularly both within and across multiple trials, so it is reasonable to consider a spike train as a point process with the instantaneous firing rate being the intensity function of point processes (Perkel et al., 1967; Paninski, 2004; Eden et al., 2004). An example of spike trains for multiple neurons is presented in Figs. 8 and 11 in the real data experiment.

Despite many existing applications, the classic point process models, e.g., Poisson processes (Kingman, 1992), neglect the interactions within one neuron and between pairs of neurons, so they fail to capture the complex dependency within a neural population. In contrast, Hawkes process is one type of point processes which is able to express the *self-exciting* interaction between past and future events, finding applications in a wide range of domains including seismology (Ogata, 1998, 1999), criminology (Mohler et al., 2011; Lewis et al., 2012), financial engineering (Bacry et al., 2015; Filimonov and Sornette, 2015) and epidemics (Saichev and Sornette, 2011; Rizoiu et al., 2018). Unfortunately, due to the linearly additive intensity, the vanilla Hawkes process can only represent the purely excitatory interaction because a negative firing rate may coincide with inhibitory interaction. This makes the vanilla version inappropriate in the neuroscience domain where the interaction between neurons is a mixture of excitation and inhibition (Maffei et al., 2004; Mongillo et al., 2018).

In order to reconcile Hawkes process with inhibition, various nonlinear Hawkes process variants are proposed to allow for both excitatory and inhibitory interactions. The core point of the nonlinear Hawkes process is a nonlinearity which maps the convolution of the spike train with a causal influential kernel to a nonnegative conditional intensity, such as rectifier (Reynaud-Bouret et al., 2013), exponential (Gerhard et al., 2017), scaled softplus (Mei and Eisner, 2017) and sigmoid (Linderman, 2016; Apostolopoulou et al., 2019). The sigmoid mapping function is particularly appealing given that the Pólya-Gamma augmentation technique (Polson et al., 2013) can be utilized to convert the likelihood into a Gaussian form w.r.t. activation, namely, the non-conjugate model boils down to a conditional conjugate one. In this spirit, Apostolopoulou et al. (2019) augmented the nonlinear Hawkes process likelihood with thinned points and Pólya-Gamma random variables and derived a Gibbs sampler. However, the influence functions are confined to be purely exciting or inhibitive exponential decay. On the one hand, the parametric assumption of influence functions limits the model's expressiveness; on the other hand, due to the nonconjugacy of the excitation parameter of exponential decay influence function, a Metropolis-Hastings sampling step has to be embedded into the Gibbs sampler, which leads to inefficient Markov chain Monte Carlo (MCMC).

To address the issues on *effectiveness* and *efficiency* of the aforementioned work, we develop a flexible sigmoid nonlinear multivariate Hawkes processes (SNMHP) model in the continuous-time regime, which can represent *flexible excitation-inhibition-mixture* interactions among the neural population, as well as an *efficient conjugate* inference paradigm. Inspired by Donner and Opper (2017); Zhou et al. (2020), three sets of auxiliary latent variables: Pólya-Gamma variables, latent marked Poisson processes and sparsity variables, are augmented to convert the non-conjugate model to a conditional conjugate one making functional connection weights appear in a Gaussian form. Based on the augmented likelihood and prior, we propose three analytical iterative inference algorithms: a Gibbs sampler, an EM algorithm and a mean-field approximation, to fit neural spike trains. Each inference algorithm has its own pros and cons. The Gibbs sampler enables the direct characterization of the posterior over parameters without reliance on any approximation. Unfortunately, as revealed by our experiments in Sections 7.1 and 7.2, the Gibbs sampler suffers from an inefficiency issue. The EM algorithm is able to precisely find a maximum a posteriori probability (MAP) solution. Yet, as a point estimator, it precludes the modeling of the uncertainty over parameters. Mean-field approximation conjoins the merits of the Gibbs sampler and EM algorithm, capable of reasoning about parameter uncertainty in an efficient way, but it induces approximation error and lacks the guarantee of asymptotic consistency.

A typical assumption for spike train modeling is that the base spike rates of individual neurons and pairwise interactions among the neural population are *time-invariant*. Though hold for anesthetized animals, this assumption is routinely invalid when the brain state of the subject changes dynamically during behavior, stimulation and cognition (Vaadia et al., 1995; Donner et al., 2017). Ignoring such dynamics and abusing static models will lead to false model inference and misleading interpretation of interactions.

As a remedy, we extend the proposed SNMHP to a dynamic version. Specifically, we suggest coupling a Markov state process, which takes values in a discrete finite state space corresponding to the brain state, with Hawkes processes to construct a closed-loop dependency. Reciprocally, the base spike rates of individual neurons and interactions among the neural population depend on the current brain state; in the meantime, the brain state switches only when a spike occurs according to a state-transition matrix depending on the spiking neuron. The proposed dynamic-SNMHP empowers SNMHP to handle time-varying neural systems. Technically, the analytical Gibbs sampler, EM algorithm and mean-field approximation for SNMHP are extended to dynamic-SNMHP. As shown in our experiments, dynamic-SNMHP can recover flexible excitation-inhibition-mixture interactions in different brain states accurately and efficiently.

Conclusively, we make the following contributions in this work:

1. We propose a novel flexible nonlinear Hawkes process variant named SNMHP that has flexible influence patterns and is able to handle inhibitive interactions among neural populations.

2. To reconcile with time-varying neural systems, we extend SNMHP to dynamic-SNMHP by coupling a Markov state process with Hawkes processes, which can represent dynamic interactions in different brain states.

3. We develop three efficient Bayesian inference algorithms: a Gibbs sampler, an EM algorithm and a mean-field approximation, that leverage latent variable augmentation techniques to obtain closed-form iterative updates for both SNMHP and dynamic-SNMHP.

The paper is organized as follows: In Section 2, we introduce some background knowledge about temporal point processes and Pólya-Gamma distribution. In Section 3, we present how we build the SNMHP model starting from classic Hawkes processes step by step. In Section 4, we describe how the likelihood is augmented with Pólya-Gamma random variables and latent marked Poisson processes, and how the Laplace prior is augmented with sparsity variables. Based on the augmented likelihood and prior, we propose a Gibbs sampler, an EM algorithm and a mean-field approximation with analytical updates. In Section 5, we extend SNMHP to dynamic-SNMHP to reconcile with time-varying neural systems. In Section 6, similar to SNMHP, we derive the Gibbs sampler, EM algorithm and mean-field approximation for dynamic-SNMHP. In Section 7, we compare the accuracy and efficiency between these three inference algorithms on synthetic data, and use our proposed SNMHP and dynamic-SNMHP to analyze real neural recordings. We discuss the relationship between our approach and some related works in Section 8, the applicable scenarios for each inference algorithm in Section 9, then summarize the paper and provide some thoughts on possible future work in Section 10.

2. Preliminary

In this section, we introduce some background knowledge about temporal point processes and Pólya-Gamma augmentation technique.

2.1 Temporal Point Processes

The temporal point process is an essential stochastic process for modelling the mechanism of event occurrence in many real applications, e.g., neural spike train (Paninski, 2004) and high-frequency financial trade (Bacry and Muzy, 2014) where each event is represented as a point on the timeline. Although there are many methods of analyzing temporal point processes, in this work we focus on a convenient way of characterizing how the past history affects the present, which is called the conditional intensity function. Given a realization of a temporal point process $D = \{t_i\}_{i=1}^N \in [0, T]$ where t_i is the i-th event timestamp and T is the observation window of the process, the conditional intensity function specifies the mean number of events occurring in an infinitesimal interval $[t, t + dt)$ conditional on the history before t:

$$\lambda(t)dt = \mathbb{E}[N(t, t + dt) \mid \mathcal{H}_{t-}], \tag{1}$$

where $N(\cdot)$ defines the number of events in an interval and $\mathcal{H}_{t-}$ is the history right up to but not including t.

2.2 Pólya-Gamma Augmentation

The Bayesian inference for the probit regression is easy because of the simple latent variable method proposed in Albert and Chib (1993) for posterior sampling. However, the logistic regression model is more difficult due to the inconvenient form of the likelihood. Polson et al. (2013) proposed a novel data augmentation technique for Bayesian logistic regression, which is both exact and simple. The key idea is the binomial likelihood parametrized by log odds can be represented as a mixture of Gaussians w.r.t. a Pólya-Gamma distribution. The Pólya-Gamma augmentation technique is used in our model for inference in the following Section 4. In order to make the derivation of the subsequent inference algorithms more comprehensive, we provide the definition of the Pólya-Gamma distribution here, which is cited from Polson et al. (2013): A random variable ω has a Pólya-Gamma distribution with parameters $b > 0$ and $c \in \mathbb{R}$, denoted $\omega \sim p_{\mathrm{PG}}(\omega \mid b, c)$, if

$$\omega = \frac{1}{2\pi^2} \sum_{q=1}^{\infty} \frac{\gamma_q}{(q - 1/2)^2 + c^2/(4\pi^2)}, \tag{2}$$

where $\gamma_q \sim p_{\mathrm{Ga}}(b, 1)$ are independent Gamma random variables. Equation (2) provides a sampling definition of Pólya-Gamma distribution $p_{\mathrm{PG}}(\omega \mid b, c)$ whose probability density function is more complex (see Polson et al. (2013) for more details). Fortunately, in this work we do not need the density function but only the first order moment $\mathbb{E}[\omega] = \frac{b}{2c} \tanh(\frac{c}{2})$.

3. Our Model: SNMHP

Neurons communicate with each other by action potentials (spikes) and chemical neurotransmitters. A spike causes the pre-synaptic neuron to release a chemical neurotransmitter that induces impulse responses, either exciting or inhibiting the post-synaptic neuron from firing its own spikes. The addition of excitatory and inhibitory influence to a neuron determines whether a spike will occur. At the same time, the impulse response characterizes the exciting or inhibiting influence which can be complex and flexible (Purves et al., 2014; Squire et al., 2012; Bassett and Sporns, 2017). Arguably, the flexible nonlinear multivariate Hawkes processes are a suitable statistical model for representing mutually excitatory or inhibitory interactions and functional connectivity among neural populations.

3.1 Multivariate Hawkes Processes

The vanilla multivariate Hawkes processes (Hawkes, 1971) are sequences of timestamps $D = \{\{t_n^i\}_{n=1}^{N_i}\}_{i=1}^{M} \in [0, T]$ where t_n^i is the timestamp of n-th event on i-th dimension with N_i being the number of points on i-th dimension, M the number of dimensions, T the observation window. The i-th dimensional conditional intensity function, the mean number of events occurring on i-th dimension in $[t, t + dt)$ conditional on all dimensional history before t, has a particular functional form:

$$\lambda_i(t) = \mu_i + \sum_{j=1}^{M} \sum_{t_n^j < t} \phi_{ij}(t - t_n^j), \tag{3}$$

where $\mu_i > 0$ is the baseline rate of i-th dimension and $\phi_{ij}(\cdot) \geq 0$ is the causal influence function (impulse response) from j-th dimension to i-th dimension which is normally a parameterized function, e.g., exponential decay. The summation explains the self- and mutual-excitation phenomenon, i.e., the occurrence of previous events increases the intensity of events in the future. Unfortunately, one blemish is the vanilla multivariate Hawkes processes allow only nonnegative (excitatory) influence functions because negative (inhibitory) influence functions may yield a negative intensity which is meaningless. To reconcile the vanilla version with inhibitory effect and flexible influence function, we propose the SNMHP.

3.2 Sigmoid Nonlinear Multivariate Hawkes Processes

Similar to the classic nonlinear multivariate Hawkes processes (Brémaud and Massoulié, 1996), the i-th dimensional conditional intensity of SNMHP is defined as

$$\lambda_i(t) = \overline{\lambda}_i \sigma(h_i(t)), \quad h_i(t) = \mu_i + \sum_{j=1}^{M} \sum_{t_n^j < t} \phi_{ij}(t - t_n^j), \tag{4}$$

where μ_i is the base activation of neuron i, $h_i(t)$ is a real-valued activation and $\sigma(\cdot)$ is the logistic (sigmoid) function which maps the activation into a positive real value in $(0, 1)$ with $\overline{\lambda}_i$ being an upper bound to scale it to $(0, \overline{\lambda}_i)$. The sigmoid nonlinearity is chosen because as will be seen later, the Pólya-Gamma augmentation scheme (Polson et al., 2013) can be utilized to make inference easy and fast. After incorporating the nonlinearity, it is straightforward to see the influence functions, $\phi_{ij}(\cdot)$, can be positive or negative. If $\phi_{ij}(\cdot)$ is negative, the superposition of $\phi_{ij}(\cdot)$ will lead to a negative activation $h_i(t)$ that renders the intensity to 0; instead, the intensity tends to $\overline{\lambda}_i$ with a positive $\phi_{ij}(\cdot)$.

To achieve flexible modeling of interactions, the influence function is assumed to be a weighted sum of basis functions (Linderman, 2016)

$$\phi_{ij}(\cdot) = \sum_{b=1}^{B} w_{ijb} \widetilde{\phi}_b(\cdot), \tag{5}$$

where $\{\widetilde{\phi}_b\}_{b=1}^{B}$ are predefined basis functions and w_{ijb} is the weight capturing the influence from j-th dimension to i-th dimension by b-th basis function with positive indicating excitation and negative indicating inhibition. The basis functions are nonnegative functions capturing a wide spectrum of interactions. Although basis functions can be in any form, to make the weights indicative of functional connection strength, basis functions are chosen to be probability densities with compact support, which means that they have bounded support $[0, T_\phi]$ [1] and the integral is one. As a result, the i-th dimensional activation is

$$\begin{aligned} h_i(t) &= \mu_i + \sum_{j=1}^{M} \sum_{t_n^j < t} \sum_{b=1}^{B} w_{ijb} \widetilde{\phi}_b(t - t_n^j) = \mu_i + \sum_{j=1}^{M} \sum_{b=1}^{B} w_{ijb} \sum_{t_n^j < t} \widetilde{\phi}_b(t - t_n^j) \\ &= \mu_i + \sum_{j=1}^{M} \sum_{b=1}^{B} w_{ijb} \Phi_{jb}(t) = \mathbf{w}_i^\top \cdot \mathbf{\Phi}(t), \end{aligned} \tag{6}$$

where $\Phi_{jb}(t)$ is the convolution of j-th dimensional observation with b-th basis function and can be precomputed; $\mathbf{w}_i = [\mu_i, w_{i11}, \ldots, w_{iMB}]^\top$ and $\mathbf{\Phi}(t) = [1, \Phi_{11}(t), \ldots, \Phi_{MB}(t)]^\top$, both are $(MB + 1) \times 1$ vectors. A similar model is used in Linderman (2016) where a binary variable is included to characterize the sparsity of functional connection. As shown later, the sparsity in our model is guaranteed by employing a Laplace prior on weights instead.

In this paper, for the two conditions mentioned above (bounded support, probability density function), we choose the basis functions to be scaled (shifted) Beta densities, but

[1] The basis function can have unbounded support $[0, \infty]$. Here, we use a bounded support $[0, T_\phi]$ and assume the points before $t - T_\phi$ have negligible influence on t to accelerate the computation of $\mathbf{\Phi}(t)$.

alternatives also can be used. It is worth noting that if we assume the influence function has a bounded support, but try to approximate it using mixture of basis functions with unbounded support such as Gaussian, there will be edge effects when close to the endpoints of $[0, T_\phi]$. Please refer to Kottas (2006) for more details.

4. Inference for SNMHP

The likelihood of a point process model is provided by Daley and Vere-Jones (2003). Correspondingly, the probability density (likelihood) of SNMHP on the i-th dimension as a function of parameters in continuous time is

$$p(D \mid \mathbf{w}_i, \overline{\lambda}_i) = \prod_{n=1}^{N_i} \overline{\lambda}_i \sigma(h_i(t_n^i)) \exp\left(-\int_0^T \overline{\lambda}_i \sigma(h_i(t))dt\right). \tag{7}$$

It is worth noting that $h_i(t)$ depends on $\mathbf{w}_i$ and observations on all dimensions. Our goal is to infer the parameters, i.e., weights and intensity upper bounds, from observations, e.g., neural spike trains, over a time interval $[0, T]$. The functional connectivity in cortical circuits is demonstrated to be sparse in neuroscience (Thomson and Bannister, 2003; Sjöström et al., 2001). To include sparsity, a factorizing Laplace prior is applied on the weights which characterize the functional connection. With the likelihood Eq. (7), the Laplace prior $p(\mathbf{w}_i) = \prod_{j,b} \frac{1}{2\alpha} \exp(-\frac{|w_{ijb}|}{\alpha})$ and the improper prior $p(\overline{\lambda}_i) \propto \frac{1}{\overline{\lambda}_i}$ (Bishop, 2006), the i-th dimensional posterior of parameters can be expressed as

$$p(\mathbf{w}_i, \overline{\lambda}_i \mid D) \propto p(D \mid \mathbf{w}_i, \overline{\lambda}_i) p(\mathbf{w}_i) p(\overline{\lambda}_i). \tag{8}$$

It is straightforward to see the likelihood is non-conjugate to the priors because the sigmoid function exists in the likelihood term and the absolute value function exists in the prior term. As a result, we have no closed-form solution for the posterior. Many methods are proposed to circumvent the intractable problem, such as Laplace approximation (Tierney and Kadane, 1986), expectation propagation (Minka, 2001), directly applying MCMC (Gilks et al., 1995) or variational inference (Blei et al., 2017), but unfortunately their efficiency is poor due to the complex non-conjugate computation. Here we leverage three sets of auxiliary latent variables: Pólya-Gamma variables, latent marked Poisson processes and sparsity variables, to augment the likelihood and prior in such a way that the augmented likelihood becomes conditional conjugate to the augmented prior. Based on the augmented model, three efficient Bayesian inference algorithms: a Gibbs sampler, an EM algorithm and a mean-field approximation with analytical updates, are derived to perform inference for model parameters.

4.1 Augmentation of Pólya-Gamma Variables

Following Polson et al. (2013), the binomial likelihoods parametrized by log odds can be represented as mixtures of Gaussians w.r.t. a Pólya-Gamma distribution. Therefore, we can define a Gaussian representation of the sigmoid function

$$\sigma(z) = \int_0^\infty e^{f(\omega, z)} p_{\mathrm{PG}}(\omega \mid 1, 0) d\omega, \tag{9}$$

where $f(\omega,z)=z/2-z^2\omega/2-\log 2$ and $p_{\text{PG}}(\omega\mid 1,0)$ is the Pólya-Gamma distribution with $\omega\in\mathbb{R}^+$. Substituting Eq. (9) into the likelihood Eq. (7), the products of $\sigma(h_i(t_n^i))$ are transformed into a Gaussian form w.r.t. $\mathbf{w}_i$ because $h_i(t)$ is linear in $\mathbf{w}_i$.

4.2 Augmentation of Marked Poisson Processes

Inspired by Donner and Opper (2018); Zhou et al. (2020), a latent marked Poisson process can be augmented to render the exponential integral term in Eq. (7) appear in a Gaussian form w.r.t. $\mathbf{w}_i$. Utilizing Eq. (9) and the sigmoid symmetry property $\sigma(z)=1-\sigma(-z)$, the exponential integral term is transformed to

$$\exp\left(-\int_0^T\overline{\lambda}_i\sigma(h_i(t))dt\right)=\exp\left(-\int_0^T\int_0^\infty\left(1-e^{f(\omega,-h_i(t))}\right)\overline{\lambda}_i p_{\text{PG}}(\omega\mid 1,0)d\omega dt\right). \quad (10)$$

The right hand side is a characteristic functional of a marked Poisson process. According to Campbell's theorem (Kingman, 1992) in Appendix A, the exponential integral term can be rewritten as

$$\exp\left(-\int_0^T\overline{\lambda}_i\sigma(h_i(t))dt\right)=\mathbb{E}_{p_{\lambda_i}}\left[\prod_{(\omega,t)\in\Pi_i}e^{f(\omega,-h_i(t))}\right], \quad (11)$$

where $\Pi_i=\{(\omega_r^i,t_r^i)\}_{r=1}^{R_i}$ denotes a realization of a marked Poisson process and p_{λ_i} is the probability measure of the marked Poisson process Π_i with intensity $\lambda_i(t,\omega)=\overline{\lambda}_i p_{\text{PG}}(\omega\mid 1,0)$. The timestamps $\{t_r^i\}_{r=1}^{R_i}$ follow a homogeneous Poisson process with rate $\overline{\lambda}_i$ and the latent Pólya-Gamma variable $\omega_r^i\sim p_{\text{PG}}(\omega\mid 1,0)$ denotes the independent mark at each timestamp t_r^i. We can see that, after substituting Eq. (11) into the likelihood Eq. (7), the exponential integral term is also transformed into a Gaussian form w.r.t. $\mathbf{w}_i$.

4.3 Augmentation of Sparsity Variables

The augmentation of the two auxiliary latent variables above makes the augmented likelihood become a Gaussian form w.r.t. the weights. However, the absolute value in the exponent of the Laplace prior hampers the Gaussian form of weights in the posterior. To circumvent this issue, we augment the third set of auxiliary latent variables: sparsity variables. It has been proved that a Laplace distribution can be represented as an infinite mixture of Gaussians (Donner and Opper, 2017; Pontil et al., 2000)

$$p(w_{ijb})=\frac{1}{2\alpha}\exp\left(-\frac{|w_{ijb}|}{\alpha}\right)=\int_0^\infty\sqrt{\frac{\beta_{ijb}}{2\pi\alpha^2}}\exp\left(-\frac{\beta_{ijb}}{2\alpha^2}w_{ijb}^2\right)p(\beta_{ijb})d\beta_{ijb}, \quad (12)$$

where α is the hyperparameter of the Laplace prior, β is the latent sparsity variable and $p(\beta_{ijb})=(\frac{2}{\beta_{ijb}})^2\exp\left(-\frac{1}{2\beta_{ijb}}\right)$. It is straightforward to see the prior is transformed into a Gaussian form w.r.t. $\mathbf{w}_i$ after the augmentation of latent sparsity variables.

4.4 Augmented Likelihood and Prior

After the augmentation of three sets of latent variables, we obtain the augmented likelihood and prior (derivation in Appendix B)

$$p(D, \boldsymbol{\omega}_i, \Pi_i \mid \mathbf{w}_i, \overline{\lambda}_i) = \prod_{n=1}^{N_i} \left[\lambda_i(t_n^i, \omega_n^i) e^{f(\omega_n^i, h_i(t_n^i))}\right] \cdot p_{\lambda_i}(\Pi_i \mid \overline{\lambda}_i) \prod_{(\omega,t)\in\Pi_i} e^{f(\omega, -h_i(t))}, \tag{13a}$$

$$p(\mathbf{w}_i, \boldsymbol{\beta}_i) = \prod_{j,b}^{MB+1} \sqrt{\frac{\beta_{ijb}}{2\pi\alpha^2}} \exp\left(-\frac{\beta_{ijb}}{2\alpha^2} w_{ijb}^2\right) \left(\frac{2}{\beta_{ijb}}\right)^2 \exp\left(-\frac{1}{2\beta_{ijb}}\right), \tag{13b}$$

where $\boldsymbol{\omega}_i$ is the vector of ω_n^i on each t_n^i, $\boldsymbol{\beta}_i$ is a $(MB+1)\times 1$ vector of $[\beta_{i00}, \beta_{i11}, \ldots, \beta_{iMB}]^\top$, $\lambda_i(t_n^i, \omega_n^i) = \overline{\lambda}_i p_{\mathrm{PG}}(\omega_n^i \mid 1, 0)$. Combining the augmented likelihood in Eq. (13a), the augmented Laplace prior in Eq. (13b) and the improper prior $p(\overline{\lambda}_i) \propto 1/\overline{\lambda}_i$, we obtain the augmented joint distribution over all variables:

$$p(D, \boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i) = p(D, \boldsymbol{\omega}_i, \Pi_i \mid \mathbf{w}_i, \overline{\lambda}_i) p(\mathbf{w}_i, \boldsymbol{\beta}_i) p(\overline{\lambda}_i). \tag{14}$$

Notice that if we marginalize out the latent variables, the resulting marginal will be same as the original joint distribution. The motivation of augmenting auxiliary latent variables should now be clear: the augmented likelihood and prior contain the weights in a Gaussian form. As a result, the non-conjugate model is transformed to a conditional conjugate model to facilitate inference.

4.5 Gibbs Sampler

Based on the augmented joint distribution in Eq. (14), it is straightforward to derive the conditional densities of latent variables and parameters in closed forms. By sampling from these conditional densities iteratively, we construct an analytical Gibbs sampler. The i-th dimensional conditional densities are

$$p(\boldsymbol{\omega}_i \mid D, \mathbf{w}_i) = \prod_{n=1}^{N_i} p_{\mathrm{PG}}(\omega_n^i \mid 1, h_i(t_n^i)), \tag{15a}$$

$$\Lambda_i(t, \omega \mid D, \mathbf{w}_i, \overline{\lambda}_i) = \overline{\lambda}_i \sigma(-h_i(t)) p_{\mathrm{PG}}(\omega \mid 1, h_i(t)), \tag{15b}$$

$$p(\boldsymbol{\beta}_i \mid \mathbf{w}_i) = \prod_{j,b}^{MB+1} p_{\mathrm{IG}}(\beta_{ijb} \mid \frac{\alpha}{|w_{ijb}|}, 1), \tag{15c}$$

$$p(\overline{\lambda}_i \mid D, \Pi_i) = p_{\mathrm{Ga}}(\overline{\lambda}_i \mid N_i + R_i, T), \tag{15d}$$

$$p(\mathbf{w}_i \mid D, \boldsymbol{\omega}_i, \Pi_i) = \mathcal{N}(\mathbf{w}_i \mid \mathbf{m}_i, \boldsymbol{\Sigma}_i). \tag{15e}$$

Equation (15a) is the conditional posterior of Pólya-Gamma variables where we utilize the tilted Pólya-Gamma distribution $p_{\mathrm{PG}}(\omega \mid b, c) \propto e^{-c^2\omega/2} p_{\mathrm{PG}}(\omega \mid b, 0)$. An efficient sampling method (Polson et al., 2013) can be used to sample from the Pólya-Gamma density. Equation (15b) is the conditional posterior intensity of the marked Poisson process. For sampling, we first use the thinning algorithm (Ogata, 1998) to draw timestamps $\{t_r^i\}_{r=1}^{R_i}$ with the rate $\overline{\lambda}_i \sigma(-h_i(t))$ and then draw corresponding marks $\{\omega_r^i\}_{r=1}^{R_i}$ from $p_{\mathrm{PG}}(\omega \mid 1, h_i(t))$.

Equation (15c) is the conditional posterior of sparsity variables where p_{IG} is the inverse Gaussian distribution. Equation (15d) is the conditional posterior of the intensity upper bound where p_{Ga} is the Gamma distribution and $R_i = |\Pi_i|$ is the number of points on Π_i. Equation (15e) is the conditional posterior of activation weights. We define $\{t_n^i, \omega_n^i\}_{n=1}^{N_i}$ to be the observed timestamps and latent marks on the i-th dimension and $\{t_r^i, \omega_r^i\}_{r=1}^{R_i}$ to be the ones on Π_i. The covariance matrix $\mathbf{\Sigma}_i = [\mathbf{\Phi}_i \mathbf{D}_i {\mathbf{\Phi}_i}^\top + \mathrm{diag}(\alpha^{-2}\boldsymbol{\beta}_i)]^{-1}$ where $\mathbf{D}_i$ is a diagonal matrix with its first N_i entries being $\{\omega_n^i\}_{n=1}^{N_i}$ and the following R_i entries being $\{\omega_r^i\}_{r=1}^{R_i}$, $\mathbf{\Phi}_i = [\{\mathbf{\Phi}(t_n^i)\}_{n=1}^{N_i}, \{\mathbf{\Phi}(t_r^i)\}_{r=1}^{R_i}]$ is a $(MB+1) \times (N_i + R_i)$ matrix, $\mathrm{diag}(\cdot)$ constructs a diagonal matrix with the input vector. The mean is $\mathbf{m}_i = \mathbf{\Sigma}_i \mathbf{\Phi}_i \mathbf{v}_i$ where the first N_i entries of $\mathbf{v}_i$ are $1/2$ and the following R_i entries are $-1/2$. Sampling iteratively by Eq. (15), we obtain a sequence of samples to characterize the posterior of model parameters.

Complexity We define the number of observed points on all dimensions to be N, the number of latent points on $\{\Pi_i\}_{i=1}^M$ to be R and the average number of points on the support of T_ϕ on all dimensions to be N_{T_ϕ}. The computational complexity of the Gibbs sampler is $\mathcal{O}(NN_{T_\phi}B + L(RN_{T_\phi}B + C_{\mathrm{PG}}(N+R) + C_{\mathrm{TH}}M + (N+R)(MB+1)^2 + M(MB+1)^3))$ where L is the number of Gibbs loops, C_{PG} and C_{TH} are the complexities of Pólya-Gamma sampling and thinning algorithm. The sampling of other variables is ignored as it is fast. The first term corresponds to the precomputation of $\mathbf{\Phi}(t)$ on observed points, the second term to the computation of $\mathbf{\Phi}(t)$ on points of $\{\Pi_i\}_{i=1}^M$, the third and fourth terms to the sampling of Pólya-Gamma variables and marked Poisson processes, the fifth and sixth terms to the matrix multiplication and inversion, respectively. In the implementation, the efficiency bottleneck arises from the sampling of marked Poisson processes since the thinning algorithm is an inefficient acceptance-rejection sampling.

Hyperparameters The hyperparameters of the Gibbs sampler comprise α in the Laplace prior that encodes the sparsity of weights, the support of influence function T_ϕ, the number and parameters of basis functions, and the number of grids for sampling from $\{\Pi_i\}_{i=1}^M$. The hyperparameters α, T_ϕ and parameters of basis functions can be chosen by cross validation. The number of basis functions is essentially a trade-off between efficiency and flexibility, and we gradually increase it until a suitable value. Similarly, the number of grids is also gradually increased until no more significant improvement in accuracy. The pseudocode is provided in Algorithm 1.

4.6 EM Algorithm

The aforementioned Gibbs sampler is easy to implement but its efficiency is low due to the sampling of marked Poisson processes. To further improve efficiency, with the support of auxiliary latent variables, we propose an analytical EM algorithm to obtain the MAP estimate in this section. With the augmented likelihood in Eq. (13a) and the augmented Laplace prior in Eq. (13b), the log-posterior corresponds to a penalized log-likelihood. In the standard EM algorithm framework, the lower bound (surrogate function) of the log-posterior can be represented as

$$\mathcal{Q}(\mathbf{w}_i, \overline{\lambda}_i \mid \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1}) = \mathbb{E}_{\boldsymbol{\omega}_i, \Pi_i}\left[\log p(D, \boldsymbol{\omega}_i, \Pi_i \mid \mathbf{w}_i, \overline{\lambda}_i)\right] + \mathbb{E}_{\boldsymbol{\beta}_i}\left[\log p(\mathbf{w}_i, \boldsymbol{\beta}_i)\right], \tag{16}$$

with expectation w.r.t. posterior distributions $p(\boldsymbol{\omega}_i, \Pi_i \mid \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1})$ and $p(\boldsymbol{\beta}_i \mid \mathbf{w}_i^{s-1})$, $s-1$ indicating parameters from the last iteration.

Algorithm 1: Gibbs sampler for SNMHP

```
Result: $\{\lambda_i(t) = \overline{\lambda}_i \sigma(\mathbf{w}_i^\top \cdot \mathbf{\Phi}(t))\}_{i=1}^M$
Predefine basis functions $\{\widetilde{\phi}_b(\cdot)\}_{b=1}^B$;
Initialize the hyperparameter $\alpha$ and $\{\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i\}_{i=1}^M$;
for Iteration do
    for Dimension $i$ do
        Sample $\boldsymbol{\omega}_i$ by Eq. (15a);
        Sample $\Pi_i$ by Eq. (15b);
        Sample $\boldsymbol{\beta}_i$ by Eq. (15c);
        Sample $\overline{\lambda}_i$ by Eq. (15d);
        Sample $\mathbf{w}_i$ by Eq. (15e).
    end
    Update the hyperparameters.
end
```

E step The posterior of latent variables is already provided in Eq. (15). For the EM algorithm, the posterior distributions of Pólya-Gamma variables $\boldsymbol{\omega}_i$ and sparsity variables $\boldsymbol{\beta}_i$, and the posterior intensity of marked Poisson process Π_i are

$$p(\boldsymbol{\omega}_i \mid D, \mathbf{w}_i^{s-1}) = \prod_{n=1}^{N_i} p_{\mathrm{PG}}(\omega_n^i \mid 1, h_i^{s-1}(t_n^i)), \tag{17a}$$

$$\Lambda_i(t, \omega \mid D, \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1}) = \overline{\lambda}_i^{s-1} \sigma(-h_i^{s-1}(t)) p_{\mathrm{PG}}(\omega \mid 1, h_i^{s-1}(t)), \tag{17b}$$

$$p(\boldsymbol{\beta}_i \mid \mathbf{w}_i^{s-1}) = \prod_{j,b}^{MB+1} p_{\mathrm{IG}}(\beta_{ijb} \mid \frac{\alpha}{|w_{ijb}^{s-1}|}, 1). \tag{17c}$$

Comparing Eqs. (15) and (17), the difference is the weights and intensity upper bounds in Eq. (17) are from the last iteration. The first order moments, $\mathbb{E}[\omega_n^i] = 1/(2h_i^{s-1}(t_n^i)) \tanh(h_i^{s-1}(t_n^i)/2)$ and $\mathbb{E}[\beta_{ijb}] = \alpha/|w_{ijb}^{s-1}|$, are used in the M step.

M step Substituting Eq. (17) into Eq. (16), we obtain the lower bound $\mathcal{Q}(\mathbf{w}_i, \overline{\lambda}_i \mid \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1})$. The updated parameters can be obtained by maximizing the lower bound. The detailed derivation is provided in Appendix C. Due to the augmentation of auxiliary latent variables, the update of parameters has a closed-form solution

$$\overline{\lambda}_i^s = \frac{N_i + R_i}{T}, \tag{18a}$$

$$\mathbf{w}_i^s = \mathbf{\Sigma}_i \int_0^T B_i(t) \mathbf{\Phi}(t) dt, \tag{18b}$$

where $R_i = \int_0^T \int_0^\infty \Lambda_i(t, \omega \mid \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1}) d\omega dt$, $\mathbf{\Sigma}_i = \left[\int_0^T A_i(t) \mathbf{\Phi}(t) \mathbf{\Phi}^\top(t) dt + \mathrm{diag}\left(\alpha^{-2} \mathbb{E}[\boldsymbol{\beta}_i]\right)\right]^{-1}$, $A_i(t) = \sum_{n=1}^{N_i} \mathbb{E}[\omega_n^i] \delta(t - t_n^i) + \int_0^\infty \omega \Lambda_i(t, \omega) d\omega$, $B_i(t) = \frac{1}{2} \sum_{n=1}^{N_i} \delta(t - t_n^i) - \frac{1}{2} \int_0^\infty \Lambda_i(t, \omega) d\omega$ with $\delta(\cdot)$ being the Dirac delta function. It is worth noting that numerical quadrature methods, e.g., Gaussian quadrature (Golub and Welsch, 1969), need to be applied to the intractable integrals above.

Complexity We define the number of Gaussian quadrature nodes to be R_{gq}. The computational complexity of the EM algorithm is $\mathcal{O}((N+R_{gq})N_{T_\phi}B+L((N+MR_{gq})(MB+1)^2+M(MB+1)^3))$ where L is the number of EM iterations, the definition of other variables is same as that in the Gibbs sampler. The first term corresponds to the precomputation of $\mathbf{\Phi}(t)$ on observed points and Gaussian quadrature nodes, the second and third term to the matrix multiplication and inversion, respectively. We can see that the EM algorithm is faster than the Gibbs sampler because the computation of $\mathbf{\Phi}(t)$ on latent Poisson processes is replaced by that on Gaussian quadrature nodes, which is taken out of iterations, and the time-consuming sampling operations are avoided.

Hyperparameters The hyperparameters of the EM algorithm comprise α in the Laplace prior, the support of influence function T_ϕ, the number and parameters of basis functions, and the number of Gaussian quadrature nodes. The hyperparameters α, T_ϕ and parameters of basis functions can be chosen by cross validation or maximizing the lower bound $\mathcal{Q}$ using numerical methods. The number of basis functions is similarly determined as in the Gibbs sampler. Similarly, the number of quadrature nodes is also gradually increased until no more significant improvement in accuracy. The pseudocode is provided in Algorithm 2.

Algorithm 2: EM algorithm for SNMHP

```
Result: $\{\lambda_i(t)=\overline{\lambda}_i\sigma(\mathbf{w}_i^\top\cdot\mathbf{\Phi}(t))\}_{i=1}^M$
Predefine basis functions $\{\widetilde{\phi}_b(\cdot)\}_{b=1}^B$;
Initialize the hyperparameter $\alpha$, parameters $\{\mathbf{w}_i,\overline{\lambda}_i\}_{i=1}^M$ and the posterior of
  $\{\boldsymbol{\omega}_i,\Pi_i,\boldsymbol{\beta}_i\}_{i=1}^M$;
for Iteration do
    for Dimension $i$ do
        Update the posterior of $\boldsymbol{\omega}_i$ by Eq. (17a);
        Update the posterior intensity of $\Pi_i$ by Eq. (17b);
        Update the posterior of $\boldsymbol{\beta}_i$ by Eq. (17c);
        Update the intensity upper bound $\overline{\lambda}_i$ by Eq. (18a);
        Update the weights $\mathbf{w}_i$ by Eq. (18b).
    end
    Update the hyperparameters.
end
```

4.7 Mean-Field Approximation

The aforementioned EM algorithm is able to find a MAP estimate efficiently. Yet, as a point estimator, it precludes the modeling of the uncertainty over parameters. In order to conjoin the merits of efficiency and uncertainty quantification, we propose an analytical mean-field approximation to the true posterior of model parameters in this section. Variational inference (Blei et al., 2017) is an approximate inference method in which the exact posterior of latent variables is approximated by a variational distribution. The optimal variational distribution is obtained by minimising the Kullback-Leibler (KL) divergence between the variational distribution and the exact posterior, or equivalently maximizing the evidence lower bound (ELBO) (Bishop, 2006).

The mean-field approximation is a common type of variational inference where we assume the latent variables can be partitioned so that each partition is independent of the others. For the current problem, we need to approximate the i-th dimensional posterior $p(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i \mid D)$ by the variational distribution $q(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i)$. We assume the i-th dimensional variational distribution factorizes as

$$q(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i) = q_1(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i) q_2(\mathbf{w}_i, \overline{\lambda}_i).$$

By using the calculus of variations, it can be shown that the optimal distribution for each factor (Bishop, 2006), in terms of minimizing the KL divergence, can be expressed as

$$\begin{aligned} q_1(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i) &\propto \exp\left(\mathbb{E}_{q_2}\left[\log p(D, \boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i)\right]\right), \\ q_2(\mathbf{w}_i, \overline{\lambda}_i) &\propto \exp\left(\mathbb{E}_{q_1}\left[\log p(D, \boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i)\right]\right). \end{aligned} \tag{19}$$

Substituting the augmented joint distribution $p(D, \boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i)$ in Eq. (14) into Eq. (19), we obtain the optimal distribution for each variable:

$$q_1(\boldsymbol{\omega}_i) = \prod_{n=1}^{N_i} p_{\mathrm{PG}}(\omega_n^i \mid 1, \widetilde{h}_i(t_n^i)), \tag{20a}$$

$$\Lambda_i^1(t, \omega) = \overline{\lambda}_i^1 \sigma(-\widetilde{h}_i(t)) p_{\mathrm{PG}}(\omega \mid 1, \widetilde{h}_i(t)) \exp\left(\frac{1}{2}(\widetilde{h}_i(t) - \overline{h}_i(t))\right), \tag{20b}$$

$$q_1(\boldsymbol{\beta}_i) = \prod_{j,b}^{MB+1} p_{\mathrm{IG}}(\beta_{ijb} \mid \frac{\alpha}{\widetilde{w}_{ijb}}, 1), \tag{20c}$$

$$q_2(\overline{\lambda}_i) = p_{\mathrm{Ga}}(\overline{\lambda}_i \mid N_i + \widetilde{R}_i, T), \tag{20d}$$

$$q_2(\mathbf{w}_i) = \mathcal{N}(\mathbf{w}_i \mid \widetilde{\mathbf{m}}_i, \widetilde{\boldsymbol{\Sigma}}_i). \tag{20e}$$

Equation (20a) is the optimal density of Pólya-Gamma variables where $\widetilde{h}_i(t) = \sqrt{\mathbb{E}[h_i^2(t)]}$. The subsequent required expectation in Eq. (20e) is $\mathbb{E}[\omega_n^i] = 1/(2\widetilde{h}_i(t_n^i)) \tanh(\widetilde{h}_i(t_n^i)/2)$. Equation (20b) is the intensity of the optimal marked Poisson processes where $\overline{\lambda}_i^1 = e^{\mathbb{E}[\log \overline{\lambda}_i]}$ and $\overline{h}_i(t) = \mathbb{E}[h_i(t)]$. Equation (20c) is the optimal density of sparsity variables where $\widetilde{w}_{ijb} = \sqrt{\mathbb{E}[w_{ijb}^2]}$. The subsequent required expectation in Eq. (20e) is $\mathbb{E}[\beta_{ijb}] = \alpha/\widetilde{w}_{ijb}$. Equation (20d) is the optimal density of intensity upper bounds where $\widetilde{R}_i = \int_0^T \int_0^\infty \Lambda_i^1(t, \omega) d\omega dt$ that can be solved by Gaussian quadrature. The required expectation in Eq. (20b) is $\mathbb{E}[\log \overline{\lambda}_i] = \psi(N_i + \widetilde{R}_i) - \log(T)$ where $\psi(\cdot)$ is the digamma function. Equation (20e) is the optimal density of activation weights where $\widetilde{\boldsymbol{\Sigma}}_i = [\int_t A_i(t)\boldsymbol{\Phi}(t)\boldsymbol{\Phi}^\top(t)dt + \mathrm{diag}(\alpha^{-2}\mathbb{E}[\boldsymbol{\beta}_i])]^{-1}$, $\widetilde{\mathbf{m}}_i = \widetilde{\boldsymbol{\Sigma}}_i \int_t B_i(t)\boldsymbol{\Phi}(t)dt$ with $A_i(t) = \sum_{n=1}^{N_i} \mathbb{E}[\omega_n^i]\delta(t - t_n^i) + \int_0^\infty \omega \Lambda_i^1(t, \omega) d\omega$ and $B_i(t) = \frac{1}{2}\sum_{n=1}^{N_i} \delta(t - t_n^i) - \frac{1}{2}\int_0^\infty \Lambda_i^1(t, \omega) d\omega$. All intractable integrals can be solved by Gaussian quadrature. The required expectations in Eqs. (20a) and (20b) are $\mathbb{E}[h_i(t)] = \boldsymbol{\Phi}^\top(t)\widetilde{\mathbf{m}}_i$ and $\mathbb{E}[h_i^2(t)] = (\boldsymbol{\Phi}^\top(t)\widetilde{\mathbf{m}}_i)^2 + \boldsymbol{\Phi}^\top(t)\widetilde{\boldsymbol{\Sigma}}_i\boldsymbol{\Phi}(t)$. Update the posterior of $\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{w}_i, \overline{\lambda}_i$ iteratively by Eq. (20) and we obtain a mean-field approximation.

Complexity The computational complexity of the mean-field approximation is $\mathcal{O}((N+R_{gq})N_{T_\phi}B+L((N+MR_{gq})(MB+1)^2+M(MB+1)^3))$ where L is the number of mean-field iterations, the definition of other variables is same as that in the EM algorithm. The first term corresponds to the precomputation of $\mathbf{\Phi}(t)$ on observed points and Gaussian quadrature nodes, the second and third term to the matrix multiplication and inversion, respectively. We can see the complexity of the mean-field approximation is same as that of the EM algorithm. In practice, the mean-field approximation is slightly slower than the EM algorithm because the computation in Eq. (20) is more complex than Eqs. (17) and (18), but much faster than the Gibbs sampler as we compute the expectation rather than sampling.

Hyperparameters Similar to the EM algorithm, the hyperparameters of the mean-field approximation comprise α in the Laplace prior, the support of influence function T_ϕ, the number and parameters of basis functions, and the number of Gaussian quadrature nodes. The hyperparameters α, T_ϕ and parameters of basis functions can be chosen by cross validation or maximizing the ELBO using numerical methods. The number of basis functions is a trade-off between efficiency and flexibility, and we gradually increase it until a suitable value. Similarly, the number of quadrature nodes is also gradually increased until no more significant improvement in accuracy. The pseudocode is provided in Algorithm 3.

Algorithm 3: Mean-field approximation for SNMHP

```
Result: $\{\lambda_i(t)=\overline{\lambda}_i\sigma(\mathbf{w}_i^\top\cdot\mathbf{\Phi}(t))\}_{i=1}^M$
Predefine basis functions $\{\widetilde{\phi}_b(\cdot)\}_{b=1}^B$;
Initialize the hyperparameter $\alpha$ and variational distributions of
  $\{\boldsymbol{\omega}_i,\Pi_i,\boldsymbol{\beta}_i,\mathbf{w}_i,\overline{\lambda}_i\}_{i=1}^M$;
for Iteration do
    for Dimension $i$ do
        Update $q_1$ of $\boldsymbol{\omega}_i$ by Eq. (20a);
        Update $\Lambda^1$ of $\Pi_i$ by Eq. (20b);
        Update $q_1$ of $\boldsymbol{\beta}_i$ by Eq. (20c);
        Update $q_2$ of $\overline{\lambda}_i$ by Eq. (20d);
        Update $q_2$ of $\mathbf{w}_i$ by Eq. (20e).
    end
    Update the hyperparameters.
end
```

5. Our Model: Dynamic-SNMHP

Our proposed SNMHP model can represent flexible excitation-inhibition-mixture interactions among the neural population and has three efficient analytical inference algorithms with the support of auxiliary latent variables. However, a fundamental problem of SNMHP is that it assumes *time-invariant* base spike rates for individual neurons and interactions among the neural population. This assumption does not hold when the brain of the subject is active, e.g., the internal state of the brain changes dynamically during behavior, stimulation and cognition (Vaadia et al., 1995; Donner et al., 2017). Ignoring such dynamics leads to incorrect model inference and misleading interpretation of interactions among the neural population. To further address the *time-invariant* problem in SNMHP, we extend SNMHP to the dynamic-SNMHP by incorporating a Markov state process representing the brain state, to couple with Hawkes processes. The proposed dynamic-SNMHP empowers SNMHP to handle time-varying neural systems.

In the following, we elaborate on how to make SNMHP embrace with the state process to form the dynamic-SNMHP. An M-dimensional dynamic-SNMHP consists of M sequences of random timestamps and their corresponding states $D = \{\{\{t_n^i, z(t_n^i)\}_{n=1}^{N_i}\}_{i=1}^M, z(T)\}$ in the observation window $[0, T]$.

The State Process In the dynamic-SNMHP, we introduce a state process $z(t)$ that takes values in a discrete finite state space $\mathcal{Z} = \{1, \dots, K\}$ to represent the brain state. Inspired by Morariu-Patrichi and Pakkanen (2018), we establish a Markov state process which is coupled with point processes to form a closed-loop dependency. Reciprocally, the underlying parameters of point processes depend on the current state; at the meantime, the state process switches only when a spike occurs on point processes by a state-transition matrix depending on the spiking neuron. Given a set of state-transition matrices $\mathbf{G} = \{\mathbf{G}_1, \dots, \mathbf{G}_M\}$ with $\mathbf{G}_i$ being a $K \times K$ transition probability matrix for the i-th neuron, the transition probability of $z(t)$ at spike timestamp t_n^i is

$$p(z({t_n^i}^+) = k' \mid z(t_n^i) = k) = g_i(k, k'), \tag{21}$$

where we assume the state process $z(t)$ is left continuous, i.e., $\lim_{t \to c^-} z(t) = z(c)$ and $z({t_n^i}^+)$ is the right limit of $z(t_n^i)$. $g_i(k, k')$ with $k, k' \in \{1, \dots, K\}$ is the entry of $\mathbf{G}_i$.

The Point Processes Equation (21) describes how the state process evolves with point processes. In turn, we define how point processes depend on the state process. We extend SNMHP to dynamic-SNMHP that base activations and influence functions depend on the system state. The i-th dimensional conditional intensity of dynamic-SNMHP is

$$\lambda_i(t, z(t)) = \overline{\lambda}_i \sigma(h_i(t, z(t))), \quad h_i(t, z(t)) = \mu_i^{z(t)} + \sum_{j=1}^{M} \sum_{t_n^j < t} \phi_{ij}^{z(t)}(t - t_n^j), \tag{22}$$

where $h_i(t, z(t))$ is a real-valued state-dependent activation, which is passed through a sigmoid function $\sigma(\cdot)$ to guarantee the non-negativity of intensity and then scaled by an upper bound $\overline{\lambda}_i$. $\mu_i^{z(t)}$ and $\phi_{ij}^{z(t)}$ are the $z(t)$-state base activation and influence function.

Similar to SNMHP, $\phi_{ij}^{z(t)}$ is assumed to be a mixture function $\phi_{ij}^{z(t)}(\cdot) = \sum_{b=1}^{B} w_{ijb}^{z(t)} \widetilde{\phi}_b(\cdot)$ where $\{\widetilde{\phi}_b\}_{b=1}^B$ are predefined basis functions and $w_{ijb}^{z(t)}$ is the state-dependent activation weight characterizing the influence from j-th dimension to i-th dimension by b-th basis function in $z(t)$ state. The i-th dimensional activation can be rewritten in a vector form

$$h_i(t, z(t)) = \mu_i^{z(t)} + \sum_{j=1}^{M} \sum_{t_n^j < t} \sum_{b=1}^{B} w_{ijb}^{z(t)} \widetilde{\phi}_b(t - t_n^j) = {\mathbf{w}_i^{z(t)}}^\top \cdot \mathbf{\Phi}(t), \tag{23}$$

where $\mathbf{w}_i^{z(t)} = [\mu_i^{z(t)}, w_{i11}^{z(t)}, \dots, w_{iMB}^{z(t)}]^\top$ and $\mathbf{\Phi}(t) = [1, \Phi_{11}(t), \dots, \Phi_{MB}(t)]^\top$. Similar to SNMHP, the basis functions are assumed to be the scaled (shifted) Beta densities on the support $[0, T_\phi]$. Combining Eqs. (21) to (23), we obtain the dynamic-SNMHP. The dynamic-SNMHP successfully addresses the time-invariant limitation, as its base activations and influence functions are dependent on the system state. The dynamic-SNMHP can be considered as a closed-loop system in which the parameters $\boldsymbol{\theta} = \{\{\mathbf{G}_i\}_{i=1}^M, \{\overline{\lambda}_i\}_{i=1}^M, \{\mathbf{w}_i^k\}_{i=1,k=1}^{M,K}\}$.

6. Inference for Dynamic-SNMHP

The Gibbs sampler, EM algorithm and mean-field approximation for SNMHP can be extended to dynamic-SNMHP. The likelihood of dynamic-SNMHP on the i-th dimension is

$$p(D \mid \mathbf{G}_i, \overline{\lambda}_i, \{\mathbf{w}_i^k\}_{k=1}^K) = \prod_{n=1}^{N_i} g_i(z(t_n^i), z(t_n^{i+}))\overline{\lambda}_i \sigma(h_i(t_n^i, z(t_n^i))) \exp\left(-\int_0^T \overline{\lambda}_i \sigma(h_i(t, z(t)))dt\right). \tag{24}$$

We place a conjugate Dirichlet prior on each row of the state-transition matrix $\mathbf{G}_i$, an improper prior on $\overline{\lambda}_i$ and a factorizing Laplace prior on $\mathbf{w}_i^k$ to induce sparsity, writing

$$p(\mathbf{g}_k^i) = p_{\mathrm{Dir}}(\mathbf{g}_k^i \mid \boldsymbol{\eta}), \quad p(\overline{\lambda}_i) \propto \frac{1}{\overline{\lambda}_i}, \quad p(\mathbf{w}_i^k) = \prod_{j,b} \frac{1}{2\alpha} \exp(-\frac{|w_{ijb}^k|}{\alpha}), \tag{25}$$

where p_{Dir} is the Dirichlet distribution, $\mathbf{g}_k^i$ is the k-th row of $\mathbf{G}_i$.

Combining Eqs. (24) and (25), we obtain the joint distribution over all variables. The posterior of the state-transition matrix is tractable because the Dirichlet prior is conjugate to the state process likelihood (categorical distribution). However, the non-conjugacy between the point process likelihood and the Laplace prior renders the inference intractable. Similar to SNMHP, we here leverage auxiliary latent variable augmentation to facilitate the inference. Based on the augmented model, we extend the Gibbs sampler, EM algorithm and mean-field approximation for SNMHP to dynamic-SNMHP. The derivation of all algorithms for dynamic-SNMHP is similar to that for SNMHP, so we omit the derivation and only provide the final results.

6.1 Gibbs Sampler

Similar to Section 4.5, the i-th dimensional conditional densities are

$$p(\boldsymbol{\omega}_i \mid D, \mathbf{w}_i) = \prod_{n=1}^{N_i} p_{\mathrm{PG}}(\omega_n^i \mid 1, h_i(t_n^i, z(t_n^i))), \tag{26a}$$

$$\Lambda_i(t, \omega \mid D, \mathbf{w}_i, \overline{\lambda}_i) = \overline{\lambda}_i \sigma(-h_i(t, z(t))) p_{\mathrm{PG}}(\omega \mid 1, h_i(t, z(t))), \tag{26b}$$

$$p(\boldsymbol{\beta}_i \mid \mathbf{w}_i) = \prod_{k=1}^K \prod_{j,b}^{MB+1} p_{\mathrm{IG}}(\beta_{ijb}^k \mid \frac{\alpha}{|w_{ijb}^k|}, 1), \tag{26c}$$

$$p(\mathbf{G}_i \mid D) = \prod_{k=1}^K p_{\mathrm{Dir}}(\mathbf{g}_k^i \mid \mathbf{u}_k^i + \boldsymbol{\eta}), \tag{26d}$$

$$p(\overline{\lambda}_i \mid D, \Pi_i) = p_{\mathrm{Ga}}(\overline{\lambda}_i \mid N_i + R_i, T), \tag{26e}$$

$$p(\mathbf{w}_i \mid D, \boldsymbol{\omega}_i, \Pi_i) = \prod_{k=1}^K \mathcal{N}(\mathbf{w}_i^k \mid \mathbf{m}_i^k, \boldsymbol{\Sigma}_i^k). \tag{26f}$$

Equation (26a) is the conditional posterior of Pólya-Gamma variables. Equation (26b) is the conditional posterior intensity of the marked Poisson process. Equation (26c) is the conditional posterior of sparsity variables. Equation (26d) is the posterior of the state-transition matrix where $\mathbf{u}_k^i = [u_{k,1}^i, \ldots, u_{k,K}^i]$ is the count of state transition from k to $k' \in \{1, \ldots, K\}$ on the i-th dimension. Equation (26e) is the conditional posterior of the intensity upper bound where $R_i = |\Pi_i|$ is the number of points on Π_i. Equation (26f) is the conditional posterior of activation weights. We define $\{t_n^{i,k}, \omega_n^{i,k}\}_{n=1}^{N_{i,k}}$ to be the observed timestamps and latent marks on the i-th dimension with state k and $\{t_r^{i,k}, \omega_r^{i,k}\}_{r=1}^{R_{i,k}}$ to be the ones on Π_i with state k. The covariance matrix $\mathbf{\Sigma}_i^k = [\mathbf{\Phi}_i^k \mathbf{D}_i^k {\mathbf{\Phi}_i^k}^\top + \text{diag}(\alpha^{-2}\boldsymbol{\beta}_i^k)]^{-1}$ where $\mathbf{D}_i^k$ is a diagonal matrix with its first $N_{i,k}$ entries being $\{\omega_n^{i,k}\}_{n=1}^{N_{i,k}}$ and the following $R_{i,k}$ entries being $\{\omega_r^{i,k}\}_{r=1}^{R_{i,k}}$, $\mathbf{\Phi}_i^k = [\{\mathbf{\Phi}(t_n^{i,k})\}_{n=1}^{N_{i,k}}, \{\mathbf{\Phi}(t_r^{i,k})\}_{r=1}^{R_{i,k}}]$ is a $(MB+1) \times (N_{i,k} + R_{i,k})$ matrix, diag$(\cdot)$ indicates the diagonal matrix of a vector. The mean $\mathbf{m}_i^k = \mathbf{\Sigma}_i^k \mathbf{\Phi}_i^k \mathbf{v}_i^k$ where the first $N_{i,k}$ entries of $\mathbf{v}_i^k$ are $1/2$ and the following $R_{i,k}$ entries are $-1/2$.

Complexity The computational complexity of the Gibbs sampler is $\mathcal{O}(NN_{T_\phi}B + L(RN_{T_\phi}B + C_{\text{PG}}(N+R) + C_{\text{TH}}M + (N+R)(MB+1)^2 + KM(MB+1)^3))$. We can see that the incorporation of state process only increases the complexity in the last term.

Hyperparameters Compared with the Gibbs sampler for SNMHP, here we only have one extra hyperparameter $\boldsymbol{\eta}$. In experiments, $\boldsymbol{\eta}$ is set to $\mathbf{1}$ to represent a uniform Dirichlet prior; the choice of other hyperparameters is similar to that for SNMHP.

6.2 EM Algorithm

Similar to Section 4.6, we propose an analytical EM algorithm to obtain the MAP estimate for dynamic-SNMHP.

E step For the EM algorithm, the posterior distributions of Pólya-Gamma variables $\boldsymbol{\omega}_i$ and sparsity variables $\boldsymbol{\beta}_i$, and the posterior intensity of marked Poisson process Π_i are

$$p(\boldsymbol{\omega}_i \mid D, \mathbf{w}_i^{s-1}) = \prod_{n=1}^{N_i} p_{\text{PG}}(\omega_n^i \mid 1, h_i^{s-1}(t_n^i, z(t_n^i))), \tag{27a}$$

$$\Lambda_i(t, \omega \mid D, \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1}) = \overline{\lambda}_i^{s-1} \sigma(-h_i^{s-1}(t, z(t))) p_{\text{PG}}(\omega \mid 1, h_i^{s-1}(t, z(t))), \tag{27b}$$

$$p(\boldsymbol{\beta}_i \mid \mathbf{w}_i^{s-1}) = \prod_{k=1}^{K} \prod_{j,b}^{MB+1} p_{\text{IG}}(\beta_{ijb}^k \mid \frac{\alpha}{|{w_{ijb}^{k}}^{s-1}|}, 1). \tag{27c}$$

The first order moments, $\mathbb{E}[\omega_n^i] = 1/(2h_i^{s-1}(t_n^i, z(t_n^i))) \tanh(h_i^{s-1}(t_n^i, z(t_n^i))/2)$ and $\mathbb{E}[\beta_{ijb}^k] = \alpha/|{w_{ijb}^{k}}^{s-1}|$, are used in the M step.

M step The update of parameters has closed-form solutions:

$$\mathbf{g}_k^i = \frac{\mathbf{u}_k^i}{\sum_{k'=1}^K u_{k,k'}^i}, \tag{28a}$$

$$\overline{\lambda}_i^s = \frac{N_i + R_i}{T}, \tag{28b}$$

$$\mathbf{w}_i^{k^s} = \mathbf{\Sigma}_i^k \int_{t \in k} B_i(t) \mathbf{\Phi}(t) dt, \tag{28c}$$

where $\mathbf{u}_k^i = [u_{k,1}^i, \dots, u_{k,K}^i]$ is the count of state transition from k to k' on the i-th dimension, $R_i = \int_0^T \int_0^\infty \Lambda_i(t, \omega \mid \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1}) d\omega dt$, $\mathbf{\Sigma}_i^k = \left[\int_{t \in k} A_i(t) \mathbf{\Phi}(t) \mathbf{\Phi}^\top(t) dt + \operatorname{diag}\left(\alpha^{-2} \mathbb{E}[\boldsymbol{\beta}_i^k]\right)\right]^{-1}$, $A_i(t) = \sum_{n=1}^{N_i} \mathbb{E}[\omega_n^i] \delta(t - t_n^i) + \int_0^\infty \omega \Lambda_i(t, \omega) d\omega$, $B_i(t) = \frac{1}{2} \sum_{n=1}^{N_i} \delta(t - t_n^i) - \frac{1}{2} \int_0^\infty \Lambda_i(t, \omega) d\omega$ with $\delta(\cdot)$ being the Dirac delta function. The $\int_{t \in k}$ means the integral over time intervals with state k. All intractable integrals can be solved by Gaussian quadrature. It is worth noting that the computation of $\mathbf{g}_k^i$ is not iterative as it does not depend on other variables.

Complexity The computational complexity of the EM algorithm is $\mathcal{O}((N + R_{gq}) N_{T_\phi} B + L((N + M R_{gq})(MB + 1)^2 + KM(MB + 1)^3))$. Similarly, the incorporation of state process only increases the complexity in the last term.

Hyperparameters The choice of hyperparameters is similar to that for SNMHP.

6.3 Mean-Field Approximation

We assume the variational distribution of i-th dimensional point process factorizes as $q(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i, \mathbf{G}_i, \overline{\lambda}_i, \mathbf{w}_i) = q_1(\boldsymbol{\omega}_i, \Pi_i, \boldsymbol{\beta}_i) q_2(\mathbf{G}_i, \overline{\lambda}_i, \mathbf{w}_i)$ where $q_2(\mathbf{G}_i, \overline{\lambda}_i, \mathbf{w}_i) = p(\mathbf{G}_i \mid D) q_2(\overline{\lambda}_i, \mathbf{w}_i)$. Similarly, the optimal distribution for each factor is expressed as

$$q_1(\boldsymbol{\omega}_i) = \prod_{n=1}^{N_i} p_{\mathrm{PG}}(\omega_n^i \mid 1, \widetilde{h}_i(t_n^i, z(t_n^i))), \tag{29a}$$

$$\Lambda_i^1(t, \omega) = \overline{\lambda}_i^1 \sigma(-\widetilde{h}_i(t, z(t))) p_{\mathrm{PG}}(\omega \mid 1, \widetilde{h}_i(t, z(t))) \exp\left(\frac{1}{2}(\widetilde{h}_i(t, z(t)) - \overline{h}_i(t, z(t)))\right), \tag{29b}$$

$$q_1(\boldsymbol{\beta}_i) = \prod_{k=1}^{K} \prod_{j,b}^{MB+1} p_{\mathrm{IG}}(\beta_{ijb}^k \mid \frac{\alpha}{\widetilde{w}_{ijb}^k}, 1), \tag{29c}$$

$$p(\mathbf{G}_i \mid D) = \prod_{k=1}^{K} p_{\mathrm{Dir}}(\mathbf{g}_k^i \mid \mathbf{u}_k^i + \boldsymbol{\eta}), \tag{29d}$$

$$q_2(\overline{\lambda}_i) = p_{\mathrm{Ga}}(\overline{\lambda}_i \mid N_i + \widetilde{R}_i, T), \tag{29e}$$

$$q_2(\mathbf{w}_i) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{w}_i^k \mid \widetilde{\mathbf{m}}_i^k, \widetilde{\mathbf{\Sigma}}_i^k). \tag{29f}$$

Equation (29a) is the optimal distribution of Pólya-Gamma variables where $\widetilde{h}_i(t, z(t)) = \sqrt{\mathbb{E}[h_i^2(t, z(t))]}$. The required expectation is $\mathbb{E}[\omega_n^i] = 1/(2\widetilde{h}_i(t_n^i, z(t_n^i))) \tanh(\widetilde{h}_i(t_n^i, z(t_n^i))/2)$. Equation (29b) is the intensity of the optimal marked Poisson processes where $\overline{\lambda}_i^1 = e^{\mathbb{E}[\log \overline{\lambda}_i]}$ and $\overline{h}_i(t, z(t)) = \mathbb{E}[h_i(t, z(t))]$. Equation (29c) is the optimal density of sparsity variables where $\widetilde{w}_{ijb}^k = \sqrt{\mathbb{E}[{w_{ijb}^k}^2]}$. The required expectation is $\mathbb{E}[\beta_{ijb}^k] = \alpha / \widetilde{w}_{ijb}^k$. Equation (29d) is the posterior of state-transition matrix where $\mathbf{u}_k^i = [u_{k,1}^i, \dots, u_{k,K}^i]$ is the count of state transition from k to $k' \in \{1, \dots, K\}$ on the i-th dimension. Equation (29e) is the optimal density of intensity upper bounds where $\widetilde{R}_i = \int_0^T \int_0^\infty \Lambda_i^1(t, \omega) d\omega dt$ that can be solved by Gaussian quadrature. The required expectation in Eq. (29b) is $\mathbb{E}[\log \overline{\lambda}_i] = \psi(N_i + \widetilde{R}_i) - \log(T)$ where $\psi(\cdot)$ is the digamma function. Equation (29f) is the optimal density of activation weights where $\widetilde{\mathbf{\Sigma}}_i^k = [\int_{t \in k} A_i(t) \mathbf{\Phi}(t) \mathbf{\Phi}^\top(t) dt + \operatorname{diag}(\alpha^{-2} \mathbb{E}[\boldsymbol{\beta}_i^k])]^{-1}$, $\widetilde{\mathbf{m}}_i^k = \widetilde{\mathbf{\Sigma}}_i^k \int_{t \in k} B_i(t) \mathbf{\Phi}(t) dt$ with $A_i(t) = \sum_{n=1}^{N_i} \mathbb{E}[\omega_n^i] \delta(t - t_n^i) + \int_0^\infty \omega \Lambda_i^1(t, \omega) d\omega$ and $B_i(t) =$

$\frac{1}{2}\sum_{n=1}^{N_i}\delta(t-t_n^i)-\frac{1}{2}\int_0^\infty\Lambda_i^1(t,\omega)d\omega$. The $\int_{t\in k}$ means the integral over time intervals with state k. All intractable integrals can be solved by Gaussian quadrature. The required expectations in Eqs. (29a) and (29b) are $\mathbb{E}[h_i(t,z(t)=k)]=\mathbf{\Phi}^\top(t)\widetilde{\mathbf{m}}_i^k$ and $\mathbb{E}[h_i^2(t,z(t)=k)]=(\mathbf{\Phi}^\top(t)\widetilde{\mathbf{m}}_i^k)^2+\mathbf{\Phi}^\top(t)\widetilde{\mathbf{\Sigma}}_i^k\mathbf{\Phi}(t)$. It is worth noting that Eq. (29d) is not iterative as it does not depend on other variables. Computing the posterior of $\mathbf{G}_i$ by Eq. (29d) directly and updating the posterior of $\boldsymbol{\omega}_i,\Pi_i,\boldsymbol{\beta}_i,\overline{\lambda}_i,\mathbf{w}_i$ iteratively by Eq. (29), we obtain a mean-field approximation.

Complexity The computational complexity of the mean-field approximation is $\mathcal{O}((N+R_{gq})N_{T_\phi}B+L((N+MR_{gq})(MB+1)^2+KM(MB+1)^3))$. Similarly, the incorporation of state process only increases the complexity in the last term.

Hyperparameters Compared with the mean-field approximation for SNMHP, here we only have one extra hyperparameter $\boldsymbol{\eta}$. In experiments, $\boldsymbol{\eta}$ is set to $\mathbf{1}$ to represent a uniform Dirichlet prior; the choice of other hyperparameters is similar to that for SNMHP.

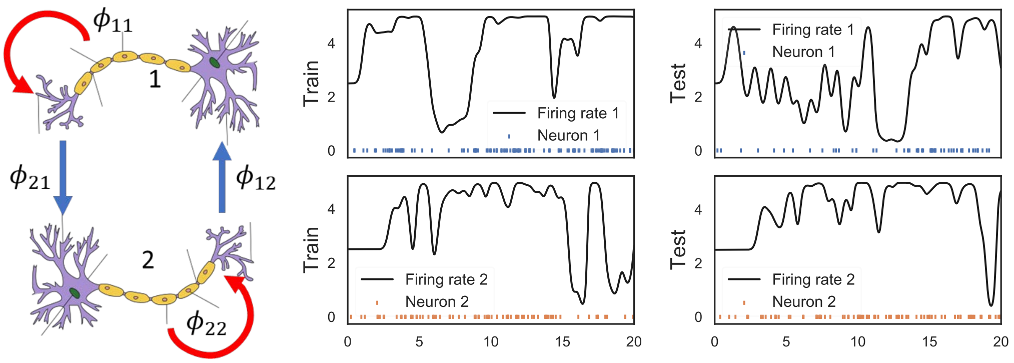
7. Experiments

In this section, we first conduct experiments to compare Gibbs sampler, EM algorithm and mean-field approximation for SNMHP and dynamic-SNMHP on synthetic spike data; and then we consider a more complicated problem: check if our proposed models can recover some predefined influence functions; finally we use our proposed SNMHP and dynamic-SNMHP to analyze two real-world spike datasets. The implementation code is publicly available at `https://github.com/zhoufeng6288/DFN-Hawkes`.

7.1 Comparison of Gibbs, EM and Mean-Field for SNMHP

In this section, we compare the *accuracy* and *efficiency* of Gibbs sampler, EM algorithm and mean-field approximation for SNMHP. We analyze spike trains obtained from the synthetic neural population model shown in Fig. 1a. The synthetic neural population contains two neurons which are self-exciting and mutual-inhibitive. We assume 4 scaled (shifted) Beta distributions: $\tilde{\phi}_{\{1,2,3,4\}}=\text{Beta}(\tilde{\alpha}=50,\tilde{\beta}=50,\text{scale}=6,\text{shift}=\{-2,-1,0,1\})$ as basis functions with support $[0,T_\phi=6]$. For the influence functions, it is assumed that $\phi_{11}=\tilde{\phi}_1$, $\phi_{22}=\tilde{\phi}_3$, $\phi_{12}=-\frac{1}{2}\tilde{\phi}_2$, $\phi_{21}=-\frac{1}{2}\tilde{\phi}_4$ with positive indicating excitation and negative indicating inhibition. With base activations $\mu_1=\mu_2=0$ and upper bounds $\overline{\lambda}_1=\overline{\lambda}_2=5$, we use the thinning algorithm (Ogata, 1998) to generate two sets of synthetic spike data on the time window $[0,T=400]$ with one being the training dataset and the other one test dataset, which contain 2700 and 2602 spikes respectively. The spike times of our simulated training and test data are shown in Figs. 1b and 1c where we zoom in $[0,20]$. We aim to identify the interactions between two neurons from statistically dependent spike trains.

We use the proposed Gibbs sampler, EM algorithm and mean-field approximation to perform inference on the training data. For hyperparameters, because the ground-truth basis functions are known, the number, support and parameters of basis functions are chosen as the ground truth. By cross validation, the hyperparameter α is chosen to be 0.2 for the three algorithms. The number of grids in the Gibbs sampler, quadrature nodes in the EM algorithm and mean-field approximation is set to 2000, and the number of iterations for three inference algorithms is set to 200, which is large enough for convergence.

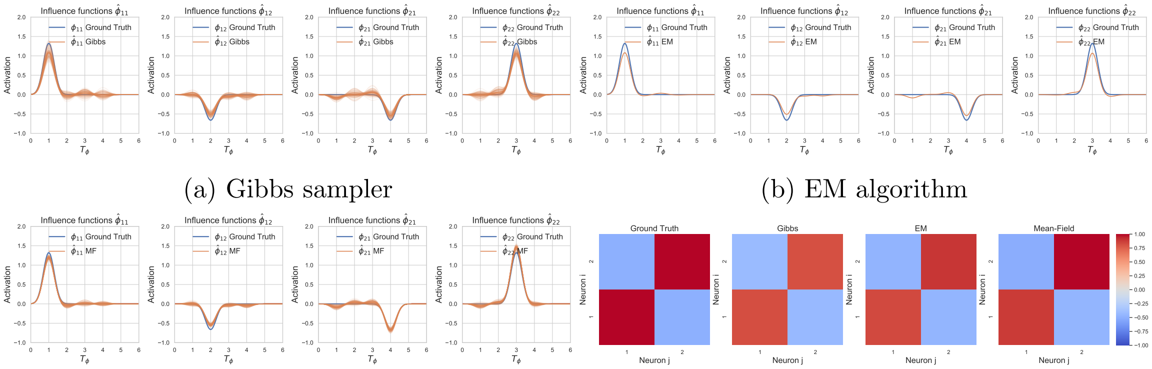


(a) (b) (c)

Figure 1: The synthetic model and data for SNMHP. (a): The synthetic neural population contains 2 neurons; the interactions between 2 neurons are self-exciting and mutual-inhibitive with red arrows indicating excitation and blue arrows indicating inhibition. (b), (c): The spike times (colorful dots) and firing rates (black lines) of 2 neurons in the synthetic training and test data (zoomed in $[0, 20]$).

For accuracy, the estimated interactions among the neural population are shown in Figs. 2a to 2c. It is easy to see that the interactions between the two neurons estimated by three inference algorithms are flexible and closely aligned with the ground truth. The posterior variance of the mean-field algorithm is lower than that of the Gibbs sampler, which is a well-known result (Blei et al., 2017). Besides, as shown in Fig. 2d, the functional connectivity estimated by three methods is close to the ground-truth structure. The functional connectivity is defined as $\int \phi_{ij}(t)dt$ with positive indicating excitation and negative indicating inhibition. The MAP estimate, posterior samples of intensity upper bounds and base activations of three methods are shown in Figs. 3a and 3b with their estimation statistics shown in Fig. 3d. Again, the posterior means of Gibbs and mean-field are close to the MAP estimate of EM, and the posterior variance of Gibbs is larger than that of mean-field. We also compare the training/test log-likelihood of the three methods by depicting their curves in Fig. 3c. We notice that all inference algorithms converge to a near plateau.

For efficiency, we compare the running time of three inference methods w.r.t. the number of observations in Fig. 3e where the number of dimensions is fixed to 2, basis functions to 4, the grid for Gibbs and quadrature nodes for EM and mean-field to 200, iterations of all methods to 200. As a result, the Gibbs sampler is the least efficient; the EM algorithm is slightly faster than the mean-field approximation since the computation in EM is simpler than that in mean-field. The efficiency bottleneck of the Gibbs sampler is the sampling of the marked Poisson process. On the one hand, we use the thinning algorithm to draw timestamps which produces a large number of events to reject; on the other hand, for each timestamp, we need to draw the corresponding mark from a Pólya-Gamma distribution. Both the sampling of timestamps and marks are slower than the computation of expectation in EM and MF.



(a) Gibbs sampler

(b) EM algorithm

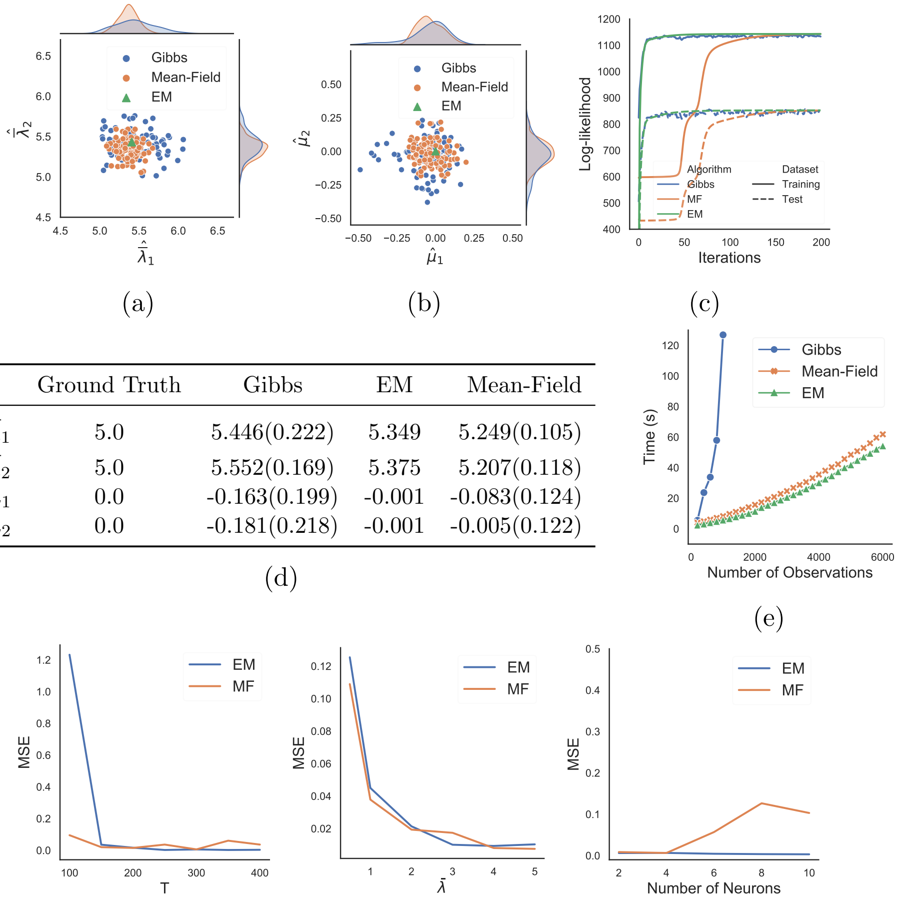
(c) Mean-field approximation

(d) Functional connectivity

Figure 2: For SNMHP: (a): The 100 posterior trajectories of interactions between two neurons by Gibbs sampler. (b): The MAP estimate of interactions between two neurons by EM algorithm. (c): The 100 posterior trajectories of interactions between two neurons by mean-field approximation. The interactions estimated by three inference algorithms are close to the ground truth. (d): The heat map of functional connectivity between two neurons with red indicating excitation and blue indicating inhibition. From left to right, we present the ground truth and functional connectivity estimated by three inference algorithms.

We also increase the observation window, the firing rate and the number of neurons to demonstrate how the estimation accuracy scales with them. To evaluate the accuracy, we use the mean squared error (MSE) to measure the distance between the estimated parameters and the ground truth (intensity upper bounds and activation weights). Taking efficiency into account, we only conduct experiments with EM algorithm and mean-field approximation because the Gibbs sampler is time-consuming. The result is shown in Figs. 3f to 3h. For the observation window, we still use the two-neurons setting with the same model parameters and increase the observation window T from 100 to 400. We can observe that the MSE becomes smaller when we increase the observation window. MF performs better than EM when given few observation data, but they finally converge to a similar MSE. For the firing rate, we fix the observation window to 400, use the same model parameters as before and increase the intensity upper bound $\overline{\lambda}$ from 0.5 to 5. We can observe that the MSE becomes smaller when we increase the intensity upper bound. Both cases above are easy to understand: we have more observation data when we increase T or $\overline{\lambda}$, which contributes to better accuracy. For the number of neurons, we generate 5 groups of 2 neurons as described above with $T = 400$, $\overline{\lambda} = 5$, and then gradually construct a larger population by concatenating the independent groups. We perform inference on the first group, then two groups until five groups (2 neurons to 10 neurons). We can observe that the MSE does not change significantly when we have more neurons.

Conclusively, three inference algorithms for SNMHP provide similar estimations, which are close to the ground truth. The Gibbs sampler can accurately characterize the posterior but is the least efficient; the EM algorithm can provide an accurate MAP estimate efficiently



(a)

(b)

(c)

	Ground Truth	Gibbs	EM	Mean-Field
$\hat{\bar{\lambda}}_1$	5.0	5.446(0.222)	5.349	5.249(0.105)
$\hat{\bar{\lambda}}_2$	5.0	5.552(0.169)	5.375	5.207(0.118)
$\hat{\mu}_1$	0.0	-0.163(0.199)	-0.001	-0.083(0.124)
$\hat{\mu}_2$	0.0	-0.181(0.218)	-0.001	-0.005(0.122)

(d)

(e)

(f)

(g)

(h)

Figure 3: For SNMHP: The 100 posterior samples and MAP estimate of (a): intensity upper bounds $\bar{\boldsymbol{\lambda}}$ and (b): base activations $\boldsymbol{\mu}$ from Gibbs sampler, mean-field approximation and EM algorithm. (c): The training/test log-likelihood curves of three methods w.r.t. iterations (for mean-field, it is evaluated by the mean). (d): The estimation statistics of intensity upper bounds and base activations by three algorithms based on 100 posterior samples and MAP estimate. The mean and standard deviation (in brackets) are provided. (e): The running time of three methods w.r.t. the number of observations (the precomputation of $\boldsymbol{\Phi}(t)$ is included). (f), (g), (h): The MSE between estimated parameters and ground truth w.r.t. the observation window, the intensity upper bound and the number of neurons.

but can not characterize the uncertainty as a point estimator; the mean-field approximation has merits in uncertainty quantification and inference efficiency but can only provide an approximated posterior. In practice, which inference algorithm to use depends on what we desire in specific applications.

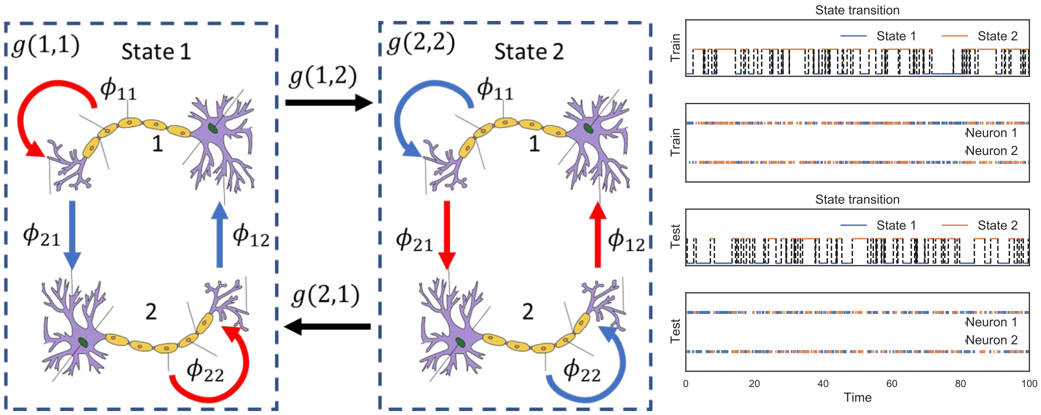
7.2 Comparison of Gibbs, EM and Mean-Field for Dynamic-SNMHP

In this section, we compare the *accuracy* and *efficiency* of Gibbs sampler, EM algorithm and mean-field approximation for dynamic-SNMHP. We still analyze spike trains obtained from the synthetic neural population model in Section 7.1, but extend it to the time-varying neural system shown in Fig. 4a, which is a 2-neuron 2-state dynamic-SNMHP with *self-exciting* and *mutual-inhibitive* interactions in the first state and *self-inhibitive* and *mutual-exciting* interactions in the second state.

In the following, we use the superscript in brackets to indicate states and the subscript to indicate neurons or basis functions. We use the same 4 scaled (shifted) Beta distributions: $\tilde{\phi}_{\{1,2,3,4\}} = \text{Beta}(\tilde{\alpha} = 50, \tilde{\beta} = 50, \text{scale} = 6, \text{shift} = \{-2, -1, 0, 1\})$ as basis functions with support $[0, T_\phi = 6]$. The state-dependent interactions between two neurons are designed as $\phi_{11}^{(1)} = \widetilde{\phi}_1$, $\phi_{12}^{(1)} = -\frac{1}{2}\widetilde{\phi}_2$, $\phi_{21}^{(1)} = -\frac{1}{2}\widetilde{\phi}_4$, $\phi_{22}^{(1)} = \widetilde{\phi}_3$ in the first state and $\phi_{11}^{(2)} = -\frac{1}{2}\widetilde{\phi}_1$, $\phi_{12}^{(2)} = \widetilde{\phi}_2$, $\phi_{21}^{(2)} = \widetilde{\phi}_4$, $\phi_{22}^{(2)} = -\frac{1}{2}\widetilde{\phi}_3$ in the second state with positive indicating excitation and negative indicating inhibition. The state-dependent base activations are $\mu_1^{(1)} = \mu_2^{(1)} = \mu_1^{(2)} = \mu_2^{(2)} = 0$ in two states. The intensity upper bounds are $\overline{\lambda}_1 = \overline{\lambda}_2 = 5$. The dimension-dependent state-transition matrices are $\mathbf{G}_1 = [[g_1(1,1) = 0.99, g_1(1,2) = 0.01], [g_1(2,1) = 0.01, g_1(2,2) = 0.99]]$, $\mathbf{G}_2 = [[g_2(1,1) = 0.80, g_2(1,2) = 0.20], [g_2(2,1) = 0.20, g_2(2,2) = 0.80]]$, which means it has a high probability to keep the original state. We use the thinning algorithm to generate two sets of synthetic spike data on $[0, T = 500]$ as the training and test datasets, which contain 4025 and 3760 spikes respectively. The state process and spike times of our simulated training and test data on $[0, 100]$ are shown in Fig. 4b where the state process switches between two states and the spike dynamics are temporally heterogeneous according to the state. We aim to identify the state-dependent interactions between two neurons and dimension-dependent state-transition matrices from statistically dependent spike trains.

We use the proposed Gibbs sampler, EM algorithm and mean-field approximation for dynamic-SNMHP to perform inference on the training data. For hyperparameters, the number, support and parameters of basis functions are chosen as the ground truth. The hyperparameter $\boldsymbol{\eta}$ is set to $\mathbf{1}$ to represent a uniform Dirichlet prior. By cross validation, the hyperparameter α is chosen to be 0.2 for Gibbs, EM and mean-field. The number of grids in the Gibbs sampler, quadrature nodes in the EM algorithm and mean-field approximation is set to 5000, and the number of iterations for three inference algorithms is set to 600, which is large enough for convergence.

For accuracy, the estimated interactions between two neurons in two states by three inference algorithms are shown in Figs. 5a to 5c, which are all close to the ground truth. The estimated functional connectivity in two states by three methods is shown in Fig. 5d, which successfully recovers the dynamic ground-truth structure. The posterior samples and MAP estimate of intensity upper bounds and base activations in two states by three methods are shown in Figs. 6a to 6c with their estimation statistics shown in Fig. 6e where the posterior means of Gibbs and mean-field are close to the MAP estimate of EM and the posterior variance of Gibbs is larger than that of mean-field. The estimated dimension-dependent state-transition matrices are shown in Fig. 6e in which the estimations of three methods are close to the ground truth. We also compare the training/test log-likelihood of three inference algorithms in Fig. 6d where we obtain a similar result as SNMHP: the training/test log-likelihood curves of three algorithms converge to a similar maximum.



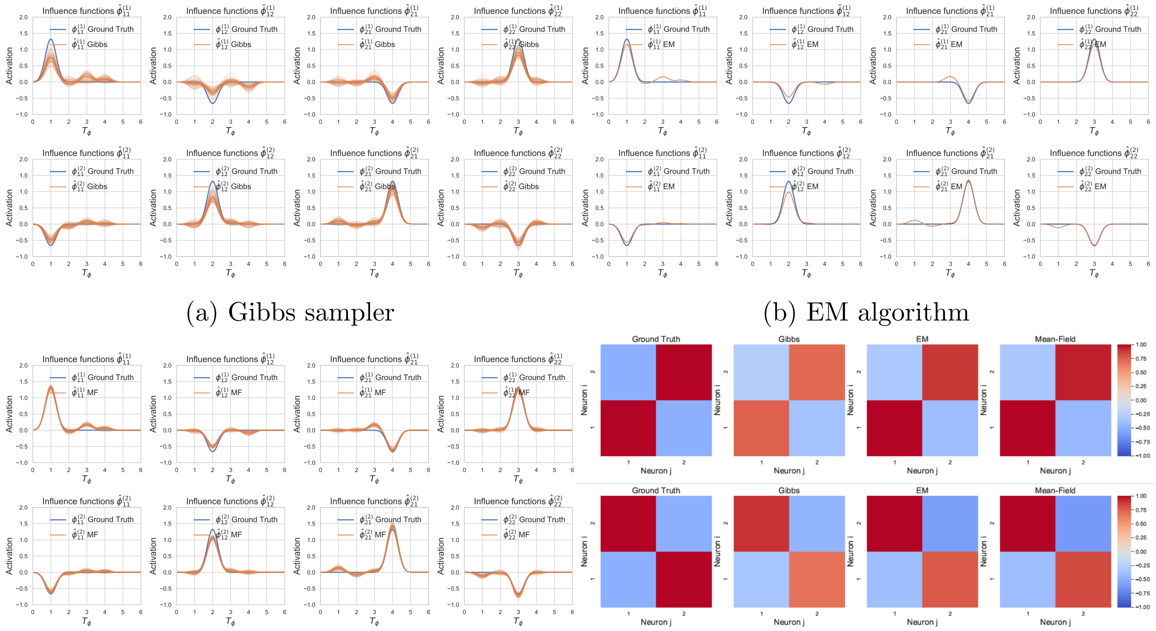
(a) (b)

Figure 4: The synthetic model and data for dynamic-SNMHP. (a): The synthetic neural population contains 2 neurons; the interactions are self-exciting and mutual-inhibitive in the 1-st state and self-inhibitive and mutual-exciting in the 2-nd state. The switching between two states follows the dimension-dependent state-transition matrices **G**. (b): The state process and spike times of 2 neurons in the synthetic training and test data.

For efficiency, we compare the running time of three inference methods w.r.t. the number of observations in Fig. 6f where the number of states is fixed to 2, dimensions to 2, basis functions to 4, grid for Gibbs and quadrature nodes for EM and mean-field to 200, iterations of all methods to 200. As expected, a similar conclusion as SNMHP is obtained: the Gibbs sampler is the least efficient due to the time-consuming sampling operation; the EM algorithm is slightly faster than the mean-field approximation because the computation in EM is simpler than that in mean-field. The same inference algorithm for dynamic-SNMHP is slower than that for SNMHP because of more parameters for estimation.

As for SNMHP, we also increase the observation window, the firing rate and the number of neurons to demonstrate how the estimation accuracy scales with them for dynamic-SNMHP. Taking efficiency into account, we only conduct experiments with EM algorithm and mean-field approximation. All experimental settings remain same as Section 7.1. The result is shown in Figs. 6g to 6i. We obtain the same conclusion as that for SNMHP: the MSE becomes smaller when we increase the observation window and intensity upper bound, but does not change significantly when we have more neurons.

Conclusively, three inference algorithms for dynamic-SNMHP provide similar estimations close to the ground truth. Similar to SNMHP, each method has its own pros and cons in terms of accuracy, efficiency and uncertainty quantification. The choice of inference algorithms depends on specific requirements of the application.



(a) Gibbs sampler

(b) EM algorithm

(c) Mean-field approximation

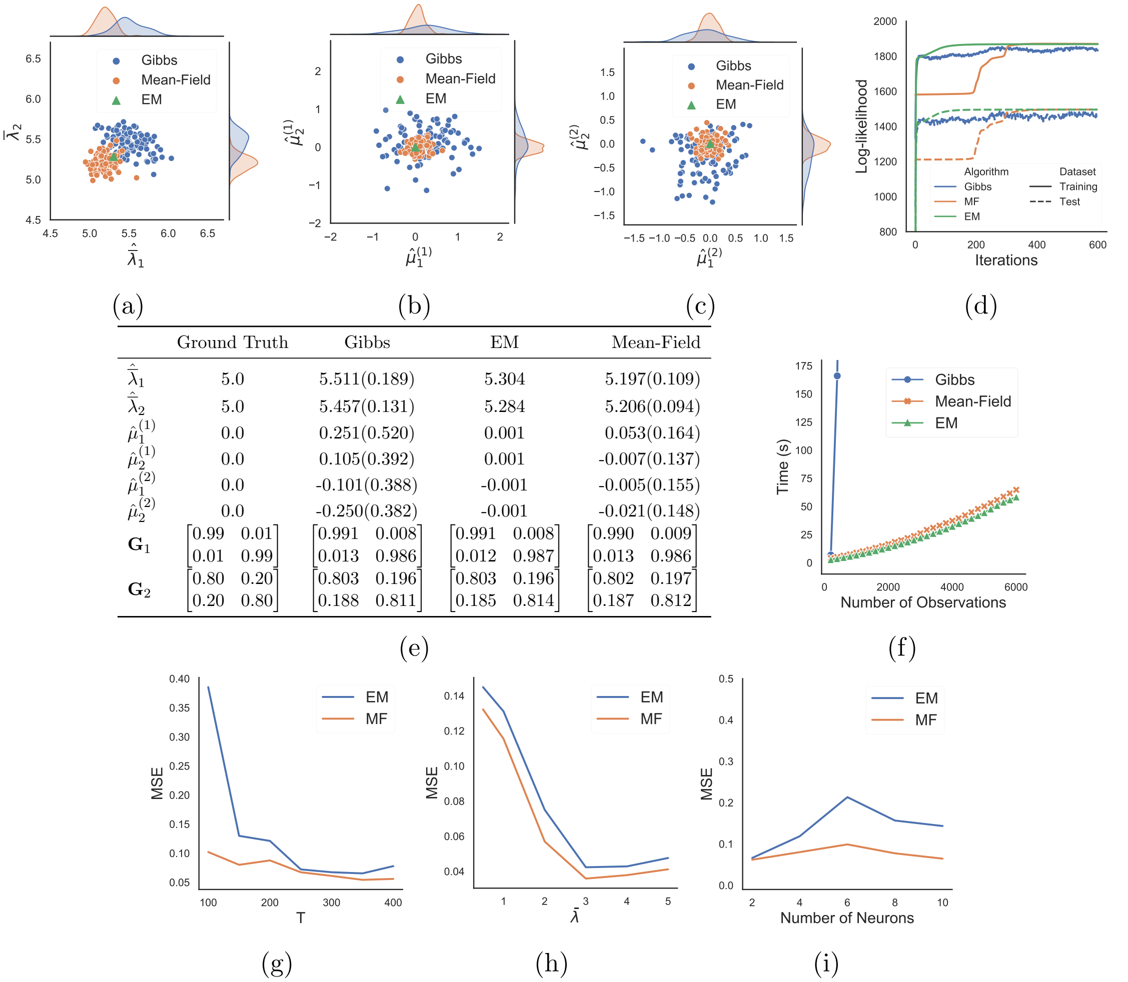
(d) Functional connectivity

Figure 5: For dynamic-SNMHP: (a): The 100 posterior trajectories of interactions between two neurons in two states by Gibbs sampler (up: 1-st state, down: 2-nd state). (b): The MAP estimate of interactions between two neurons in two states by EM algorithm. (c): The 100 posterior trajectories of interactions between two neurons in two states by mean-field approximation. The interactions estimated by three methods are close to the ground truth. (d): The heat map of functional connectivity between two neurons in two states with red indicating excitation and blue indicating inhibition. From left to right, we present the ground truth and functional connectivity estimated by three inference algorithms.

7.3 Influence Function Recovery

In the above two sections, we use the presumed model setting (predefined basis functions to formulate influence functions) to generate the data and analyze the difference among three inference methods. In this section, we consider a more complicated setting: we use some predefined influence functions to generate the data directly and check if our model (mixture of Beta densities) can recover them.

Because SNMHP is a special case of dynamic-SNMHP, we only analyze dynamic-SNMHP in this section; and due to the inefficiency of Gibbs, the inference is performed by only EM algorithm and mean-field approximation. We still analyze the simulated data from the 2-neuron 2-state neural population model in Section 7.2. We define two kinds of parameterized influence functions with support $T_\phi = 2\pi$, (1) sine function: $\phi_{11}^{(1)}(\cdot) = \sin(\cdot)$, $\phi_{12}^{(1)}(\cdot) = -\frac{1}{2}\sin(\cdot)$, $\phi_{21}^{(1)}(\cdot) = -\frac{1}{2}\sin(\cdot)$, $\phi_{22}^{(1)}(\cdot) = \sin(\cdot)$ in the first state and $\phi_{11}^{(2)}(\cdot) = -\frac{1}{2}\sin(\cdot)$, $\phi_{12}^{(2)}(\cdot) = \sin(\cdot)$, $\phi_{21}^{(2)}(\cdot) = \sin(\cdot)$, $\phi_{22}^{(2)}(\cdot) = -\frac{1}{2}\sin(\cdot)$ in the second state; (2) ex-



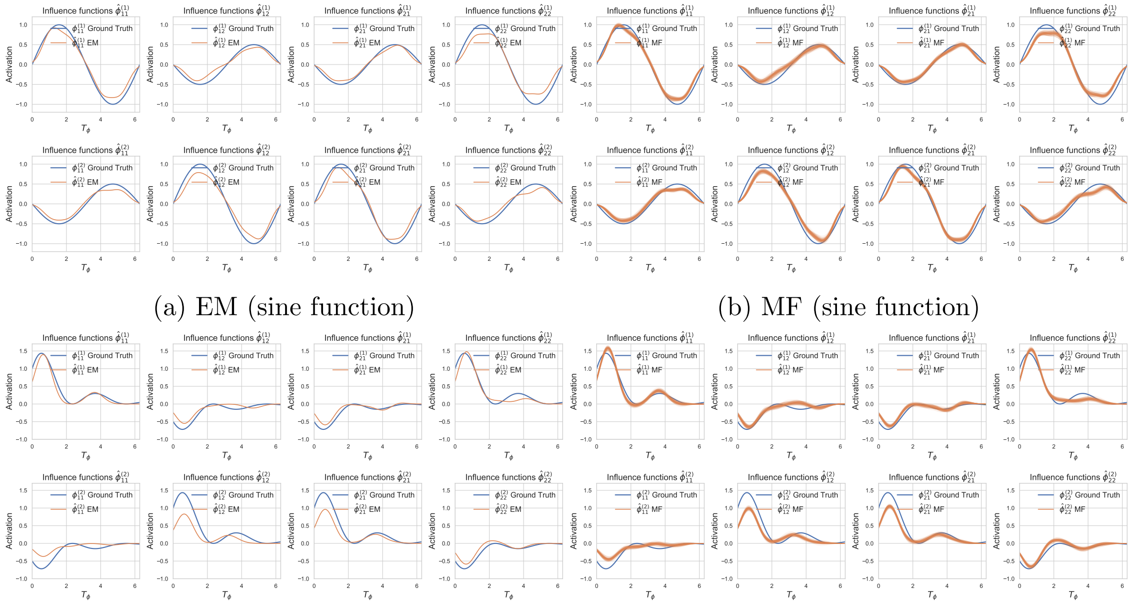
(a) (b) (c) (d)

	Ground Truth	Gibbs	EM	Mean-Field
$\hat{\bar{\lambda}}_1$	5.0	5.511(0.189)	5.304	5.197(0.109)
$\hat{\bar{\lambda}}_2$	5.0	5.457(0.131)	5.284	5.206(0.094)
$\hat{\mu}_1^{(1)}$	0.0	0.251(0.520)	0.001	0.053(0.164)
$\hat{\mu}_2^{(1)}$	0.0	0.105(0.392)	0.001	-0.007(0.137)
$\hat{\mu}_1^{(2)}$	0.0	-0.101(0.388)	-0.001	-0.005(0.155)
$\hat{\mu}_2^{(2)}$	0.0	-0.250(0.382)	-0.001	-0.021(0.148)
$\mathbf{G}_1$	$\begin{bmatrix}0.99 & 0.01\\0.01 & 0.99\end{bmatrix}$	$\begin{bmatrix}0.991 & 0.008\\0.013 & 0.986\end{bmatrix}$	$\begin{bmatrix}0.991 & 0.008\\0.012 & 0.987\end{bmatrix}$	$\begin{bmatrix}0.990 & 0.009\\0.013 & 0.986\end{bmatrix}$
$\mathbf{G}_2$	$\begin{bmatrix}0.80 & 0.20\\0.20 & 0.80\end{bmatrix}$	$\begin{bmatrix}0.803 & 0.196\\0.188 & 0.811\end{bmatrix}$	$\begin{bmatrix}0.803 & 0.196\\0.185 & 0.814\end{bmatrix}$	$\begin{bmatrix}0.802 & 0.197\\0.187 & 0.812\end{bmatrix}$

(e) (f)

(g) (h) (i)

Figure 6: For dynamic-SNMHP: The 100 posterior samples and MAP estimate of (a): intensity upper bounds $\bar{\boldsymbol{\lambda}}$, (b): base activations $\boldsymbol{\mu}^{(1)}$ in the 1-st state and (c): $\boldsymbol{\mu}^{(2)}$ in the 2-nd state from Gibbs sampler, mean-field approximation and EM algorithm. (d): The training/test log-likelihood curves of three inference algorithms w.r.t. iterations (for mean-field, it is evaluated by the mean). (e): The estimation statistics of intensity upper bounds and base activations in two states by three algorithms based on 100 posterior samples and MAP estimate. The mean and standard deviation (in brackets) are provided (for state-transition matrices, we only show the mean). (f): The running time of three inference algorithms w.r.t. the number of observations (the precomputation of $\boldsymbol{\Phi}(t)$ is included). (g), (h), (i): The MSE between estimated parameters and ground truth w.r.t. the observation window, the intensity upper bound and the number of neurons.



(a) EM (sine function)

(b) MF (sine function)

(c) EM (exponential decay sine function)

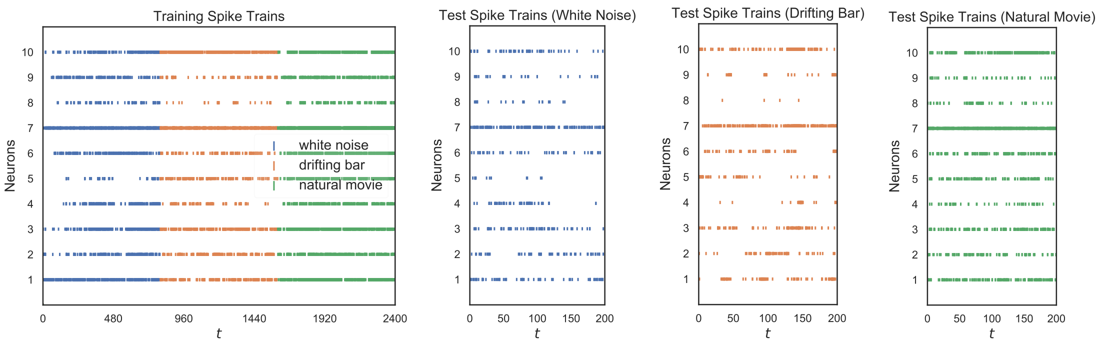
(d) MF (exponential decay sine function)

Figure 7: For the sine influence function: (a): the MAP estimate of interactions between two neurons in two states by EM algorithm (up: 1-st state, down: 2-nd state), (b): the 100 posterior trajectories of interactions between two neurons in two states by mean-field approximation. For the exponential decay sine influence function: (c): that by EM algorithm, (d): that by mean-field approximation.

ponential decay sine function: $\phi_{11}^{(1)}(\cdot) = e^{-\frac{1}{2}t}(\sin(2t)+1)$, $\phi_{12}^{(1)}(\cdot) = -\frac{1}{2}e^{-\frac{1}{2}t}(\sin(2t)+1)$, $\phi_{21}^{(1)}(\cdot) = -\frac{1}{2}e^{-\frac{1}{2}t}(\sin(2t)+1)$, $\phi_{22}^{(1)}(\cdot) = e^{-\frac{1}{2}t}(\sin(2t)+1)$ in the first state and $\phi_{11}^{(2)}(\cdot) = -\frac{1}{2}e^{-\frac{1}{2}t}(\sin(2t)+1)$, $\phi_{12}^{(2)}(\cdot) = e^{-\frac{1}{2}t}(\sin(2t)+1)$, $\phi_{21}^{(2)}(\cdot) = e^{-\frac{1}{2}t}(\sin(2t)+1)$, $\phi_{22}^{(2)}(\cdot) = -\frac{1}{2}e^{-\frac{1}{2}t}(\sin(2t)+1)$ in the second state. The state-dependent base activations, the intensity upper bounds and the dimension-dependent state-transition matrices all follow Section 7.2. We use the thinning algorithm to generate the synthetic spike data on $[0, T = 1000]$.

For hyperparameters, we choose 9 scaled shifted Beta distributions $\tilde{\phi}_{\{1,\ldots,9\}} = \text{Beta}(\tilde{\alpha} = 20, \tilde{\beta} = 20, \text{scale} = 2\pi, \text{shift} = \{-2, -1.5, -1, -0.5, 0, 0.5, 1, 1.5, 2\})$ with support $[0, T_\phi = 2\pi]$ as basis functions for the first case, and $\tilde{\phi}_{\{1,\ldots,9\}} = \text{Beta}(\tilde{\alpha} = 20, \tilde{\beta} = 20, \text{scale} = 2\pi, \text{shift} = \{-2.5, -2, -1.5, -1, -0.5, 0, 0.5, 1, 1.5\})$ as basis functions for the second case; the hyperparameter $\boldsymbol{\eta}$ is set to $\mathbf{1}$ to represent a uniform Dirichlet prior; α is chosen to be 0.2 by cross validation; the number of quadrature nodes is set to 5000.

The estimated influence functions for two cases in two states by EM algorithm and mean-field approximation are shown in Fig. 7. It is straightforward to see our model successfully recovers the predefined influence functions.



(a) (b) (c) (d)

Figure 8: The training and test spike trains in the dataset of cat primary visual cortex.

7.4 Neural Spike Data from Cat Primary Visual Cortex

In this section, we analyze the performance of SNMHP (single-state dynamic-SNMHP) and dynamic-SNMHP on a real multi-neuron evoked spike train data in cat primary visual cortex by visual stimulation. The neural data (Blanche, 2009) was recorded by Tim Blanche in the laboratory of Nicholas Swindale, University of British Columbia, and downloaded from the Collaborative Research in Computational Neuroscience data sharing website. Several multi-channel silicon electrode arrays are designed to record simultaneously spike times of 10 cells from a single penetration in cat primary visual cortex areas 17. Neural spikes are evoked by three visual stimuli on a monitor: spatiotemporal white noise, drifting bars and a natural movie that simulate retinal stimulation under natural views.

We extract the spikes evoked by three stimuli in the time window $[0, 800]$ (time unit: 100ms) as the training data and $[800, 1000]$ as the test data. To evaluate the performance of SNMHP and dynamic-SNMHP, we concatenate 3 sets of training spike trains sequentially to constitute the state-switching training data on $[0, 2400]$. The training dataset contains 15050 spikes and the test dataset contains 637, 1194 and 2089 spikes by stimuli of white noise, drifting bar and natural movie, respectively. The training and test datasets are shown in Fig. 8.

To compare the performance of single- and multi-state models, we pre-process the data in two different ways: in the first case, we assume base spike rates and interactions among the neural population are time-invariant and ignore the dynamics caused by three different stimuli to make the data eligible for single-state models, e.g., SNMHP and some baselines below; in the second case, we regard spike times evoked by three different stimuli as neural responses in three different brain states to make the data well-suited for multi-state models, e.g., dynamic-SNMHP and some baselines below.

We compare our proposed SNMHP and dynamic-SNMHP with cutting-edge multivariate Hawkes process models in recent years, including single- and multi-state statistical Hawkes process models and deep Hawkes process models. Specifically, the following most relevant baselines are considered:

- The *Granger causality Hawkes processes (GC-Hawkes)* (Xu et al., 2016) which are single-state nonparametric linear multivariate Hawkes processes with influence functions formulated by mixture of basis functions. The inference is performed by maximum likelihood estimation (MLE) with regularization terms.

- The *inhibitive Hawkes processes (IN-Hawkes)* (Mei and Eisner, 2017) which are parametric nonlinear multivariate Hawkes processes with the scaled softplus link function and exponential decay influence functions. The inference is performed by MLE.
- The *neural Hawkes processes (NE-Hawkes)* (Mei and Eisner, 2017) which are deep nonlinear multivariate Hawkes processes with the scaled softplus link function and the activation is modeled by an LSTM. The inference is performed by MLE.
- The *state-dependent Hawkes processes (SD-Hawkes)* (Morariu-Patrichi and Pakkanen, 2018) which are multi-state parametric linear multivariate Hawkes processes with exponential decay influence functions. It is a multi-state statistical model and the inference is by MLE.
- The *mutually regressive point processes (MR-PP)* (Apostolopoulou et al., 2019) which are parametric nonlinear multivariate Hawkes processes with the sigmoid link function and exponential decay influence functions. It is a single-state statistical model whose inference is by MCMC based on Pólya-Gamma augmentation and Poisson thinning.
- The *self-attentive Hawkes processes (SA-Hawkes)* (Zhang et al., 2020) whose framework is similar to NE-Hawkes except that the activation is modeled by the self-attention mechanism. The inference is performed by MLE.
- The *Transformer Hawkes processes (TR-Hawkes)* (Zuo et al., 2020) whose framework is similar to NE-Hawkes except that the activation is modeled by a Transformer architecture. The inference is performed by MLE.

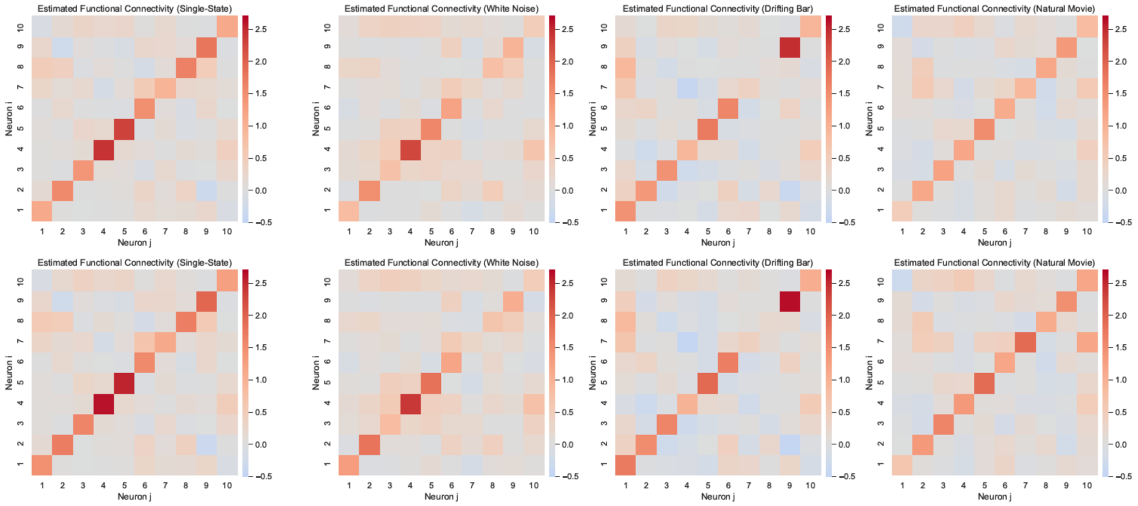
Taking efficiency into account, the inference is performed by EM algorithm and mean-field approximation. All hyperparameters are carefully tuned to obtain the optimal test log-likelihood. Specifically, the basis functions are chosen as: $\tilde{\phi}_{\{1,2,3\}} = \text{Beta}(\tilde{\alpha} = 500, , \tilde{\beta} = 500, \text{scale} = 10, \text{shift} = \{-5, -4, -3\})$ with support $[0, T_\phi = 10]$; the hyperparameter α of Laplace prior is optimised to be 0.1; the hyperparameter $\boldsymbol{\eta}$ is set to $\mathbf{1}$ to represent a uniform Dirichlet prior; the number of quadrature nodes is set to 2000 and the number of iterations to 100, which is large enough for convergence.

For SD-Hawkes, there are no additional hyperparameters needed to be tuned. We employ the same sets of hyperparameters provided in the code repositories from Apostolopoulou et al. (2019); Xu et al. (2016); Mei and Eisner (2017); Zhang et al. (2020) and Zuo et al. (2020) for tuning MR-PP, GC-Hawkes, NE-Hawkes, SA-Hawkes and TR-Hawkes, respectively. For IN-Hawkes, we find the original implementation in Mei and Eisner (2017) is hard to converge on our data, so we implement it by ourselves. We use a workstation with Intel Xeon Gold 6240R CPU and Nvidia Quadro RTX 6000 GPU for training these models.

Because we do not know the ground-truth functional connectivity and model parameters for the real data, we evaluate the models by the log-likelihood on test data. Our goal is to compare the test log-likelihood between deep models (NE-Hawkes, SA-Hawkes, TR-Hawkes), single-state models (GC-Hawkes, IN-Hawkes, MR-PP, SNMHP) and multi-state models (SD-Hawkes, dynamic-SNMHP), infer the interactions among the neural population, and provide some macroscopic property analysis about the functional connectivity of cortical circuits.

Results One important advantage of our proposed model comes from the interpretability, which is in stark contrast to the 'black-box' deep Hawkes process models. As we stated in the previous sections, the influence functions in our model represent the interactions between neurons. By estimating the activation weights, our model can characterize the functional connectivity among the neural population, which most deep models cannot provide. For example, the functional connectivity estimated by SNMHP and dynamic-SNMHP is shown in Fig. 9. SNMHP can characterize the static functional connectivity among the neuron population while dynamic-SNMHP can represent the time-varying functional connectivity under different stimuli. Each neuron in area 17 has its own receptive field which contains regions that exert an exciting influence on the neuron response and regions that exert an inhibitive influence. If we put a lot of light on the exciting region and only a little on the inhibitive region of the receptive field, the corresponding neuron will exhibit a strong self-exciting response; but if the light shines on both exciting and inhibitive regions, the corresponding neuron cannot display strong response. We speculate the white noise stimuli coincide with the exciting region of #2, #4 and #5 neurons, which leads to their strong self-excitation. In the work of Hubel and Wiesel (1962), they found the neuron in area 17 is more likely to respond to the bar stimuli. This explains why the functional connectivity strength is larger in face of drifting bar stimuli. For the natural movie, the stimuli are much more complex and changeable w.r.t. position and orientation, and this leads to the generally moderate response for all neurons. Moreover, the estimated dynamic functional connectivity under different stimuli from dynamic-SNMHP (Fig. 9b) is quite different from the static one from SNMHP (Fig. 9a). This validates our speculation that ignoring dynamics in neural spike trains leads to incorrect model inference and misleading interpretation of interactions, because spikes in different states may interfere with the inference of each state if they are incorrectly assumed to be static.

The main motivation for modelling the time-varying interactions of the neural population is to understand the macroscopic properties of the time-varying network (Donner et al., 2017). In the following, we mainly focus on two macroscopic metrics. The first is the log-likelihood on test data which characterizes the fitting performance of the estimated time-varying interaction network. For deep models and single-state statistical models, the training/test data are considered as single-state spike trains. The test log-likelihood of EM algorithm and mean-field approximation for SNMHP and dynamic-SNMHP are compared with baselines in Table 1. As expected, the two algorithms for each model provide similar results. The multi-state statistical models generally surpass the single-state statistical models with dynamic-SNMHP being the champion in most cases. This is because dynamic-SNMHP inherits the flexibility from SNMHP to represent the flexible interactions; in the meantime, dynamic-SNMHP can characterize the time-varying interactions that vary in different states leading to better goodness-of-fit. Deep Hawkes process models are strong baselines w.r.t. test log-likelihood because deep models have better expressiveness due to a large amount of parameters, but, as we stated above, they lack interpretability which is a crucial requirement in the neuroscience domain. An example of estimated interactions among 10 neurons in cat primary visual cortex is provided in Appendix D.



(a) SNMHP

(b) Dynamic-SNMHP (white noise, drifting bar, natural movie)

Figure 9: The heat map of functional connectivity among 10 neurons estimated by EM algorithm (top) and mean-field approximation (bottom). (a): For SNMHP. (b): For dynamic-SNMHP, left: white noise, middle: drifting bar, right: natural movie. For mean-field, it is evaluated by the mean. The estimations by two algorithms are similar to each other.

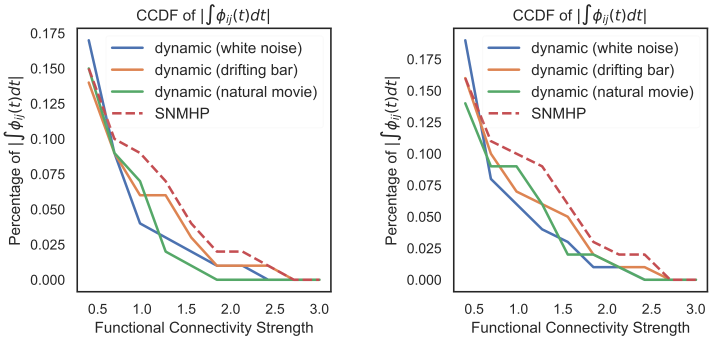
The second is the complementary cumulative distribution function (CCDF) of functional connectivity which is defined as the percentage of $|\int \phi_{ij}(t)dt|$ greater than a given strength. This metric characterizes the functional connectivity strength distribution of the interaction network. The CCDF of SNMHP and dynamic-SNMHP in three states from EM algorithm and mean-field approximation are demonstrated in Fig. 10. The results of two algorithms are similar to each other. For the natural movie stimuli, the strength of all interactions is below 2.0 (EM) and 2.5 (MF); this is consistent with our observation in Fig. 9 that most neurons generally have moderate responses in face of natural movie stimuli. More importantly, for the drifting bar stimuli, more interactions are concentrated in the domain of high strength (> 1.2) than white noise and natural movie; this is also consistent with the existing finding that cells in area 17 have elongated receptive fields and consequently respond best to elongated stimuli such as bars. The CCDF of dynamic-SNMHP under three stimuli are different from that of SNMHP, which means the functional connectivity strength distributions of static and dynamic models are different. This demonstrates the necessity of using a dynamic model to characterize the time-varying interactions in different states.

An extra advantage of our proposed inference algorithms is the efficiency due to closed-form expressions. We compare the running time of SNMHP and dynamic-SNMHP with statistical baselines[2]: IN-Hawkes, MR-PP and vanilla multivariate Hawkes processes with exponential decay influence functions. For vanilla Hawkes processes, we use two methods,

[2] SD-Hawkes is excluded because the optimization provided in the code repository from Morariu-Patrichi and Pakkanen (2018) is implemented in C but our methods are implemented in Python.

Models		White Noise	Drifting Bar	Natural Movie
Deep	NE-Hawkes	-1109.18	-615.84	-1361.29
	SA-Hawkes	-1697.73	-1558.88	-1898.13
	TR-Hawkes	**-1038.24**	-791.99	-1186.12
Single-State	GC-Hawkes	-1804.97	-2650.66	-3246.12
	IN-Hawkes	-1111.19	-588.78	-1046.27
	MR-PP	-1377.26	-1475.23	-1720.89
	SNMHP	-1068.09(EM) -1066.67(MF)	-609.47(EM) -632.87(MF)	-1048.48(EM) -1047.28(MF)
Multi-State	SD-Hawkes	-1085.06	-590.12	-975.33
	Dynamic-SNMHP	-1048.62(EM) -1040.98(MF)	**-579.97**(EM) -602.08(MF)	**-974.71**(EM) -975.28(MF)

Table 1: The comparison of log-likelihood on test real data stimulated by white noise, drifting bar and natural movie between our proposed models (SNMHP, dynamic-SNMHP) and alternatives.



(a) EM (b) MF

Figure 10: The CCDF of SNMHP and dynamic-SNMHP in three states from EM algorithm and mean-field approximation.

numerical differentiation and analytical expressions (Ozaki, 1979), to compute the gradient of log-likelihood; and use the 'SLSQP' method in 'scipy.optimize.minimize' for optimization. For IN-Hawkes, we implement the log-likelihood by ourselves, use 'autograd' to compute the gradient and use 'SLSQP' for optimization. All aforementioned models use CPU and a full batch of data for computation to achieve a fair comparison. As shown in Table 2, for SNMHP, our EM algorithm and mean-field approximation cost 6 minutes and 6 minutes 30 seconds respectively; for dynamic-SNMHP, our EM algorithm and mean-field approximation cost 6 minutes 10 seconds and 6 minutes 41 seconds respectively; MR-PP costs 2 hours 15 minutes, IN-Hawkes costs 8 hours 34 minutes and the analytical gradient implementation

Models	vanilla-Hawkes (numerical grad)	vanilla-Hawkes (analytical grad)	IN-Hawkes	MR-PP	SNMHP	Dynamic-SNMHP
Running Time	$>$ 2 days	24mins 40secs	8hrs 34mins	2hrs 15mins	6mins (EM) 6mins 30secs (MF)	6mins 10secs (EM) 6mins 41secs (MF)

Table 2: The running time of different models on the dataset of cat primary visual cortex.

of MLE for vanilla Hawkes processes costs 24 minutes 40 seconds with the same number of iterations, while the numerical differentiation implementation takes more than 2 days.

Conclusively, our proposed SNMHP can achieve competitive test log-likelihood because SNMHP can represent the flexible excitation-inhibition-mixture interactions among the neural population. More importantly, our proposed dynamic-SNMHP demonstrates prominent advantages over other single- and multi-state baselines, because on the one hand dynamic-SNMHP inherits the flexibility from SNMHP, on the other hand it can describe the dynamics in neural spike trains driven by brain states, which serves as a source of advantages for dynamic-SNMHP over single-state models that are unable to represent a time-varying neural system. In the meantime, our proposed inference algorithms achieve a competitive performance on efficiency due to closed-form expressions.

7.5 Neural Spike Data from Rat Frontal Cortex

In this section, we use the proposed SNMHP and dynamic-SNMHP to analyze a more challenging real multi-neuron spike train dataset which contains 50 neurons. In the frontal cortex of male Long-Evans rats, the spike train data (Watson et al., 2016) was recorded by silicon probe electrodes. There are no stimuli rather the rats are left alone in the cage with a ‘wake-sleep’ episode where the wake state is at least 7 minutes and followed by at least 20 minutes sleep state. Due to no stimuli, the spike train data is mainly composed of spontaneous activities and the macroscopic properties of the interaction network should remain similar in different states. In Section 7.4, we have shown the dynamic-SNMHP can find the time-varying interactions when given spike train data whose macroscopic properties change with brain state. On the contrary, in this section, we check if dynamic-SNMHP can find the consistent interactions when the spike train data whose macroscopic properties do not change significantly is forced into several states. More importantly, we can verify whether our proposed methods are applicable to high-dimensional spike train data.

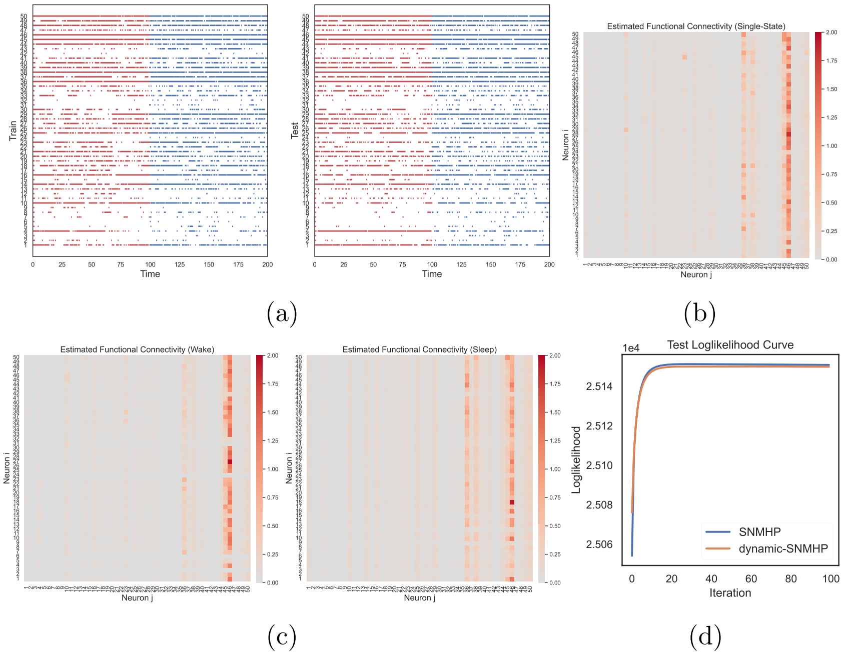
The dataset includes simultaneous records of 50 neurons and indicates a threshold time τ separating the wake and sleep episodes. We extract the spikes in the time window $[\tau-100,\tau]$ (time unit: 1s) as the wake-state training data, $[\tau-200,\tau-100]$ as the wake-state test data, $[\tau,\tau+100]$ as the sleep-state training data, and $[\tau+100,\tau+200]$ as the sleep-state test data. We concatenate the training (test) sequences in two states in chronological order to constitute a two-state training (test) data on $[0,200]$. The training dataset contains 30510 spikes and test dataset contains 31872 spikes, respectively. The training and test datasets are shown in Fig. 11a.

Similar to Section 7.4, we pre-process the data in two different ways to make it eligible for single-state or multi-state models. Due to the similarity of results from EM algorithm and mean-field approximation shown in above sections and the efficiency issue of Gibbs, we only use mean-field approximation for inference in this section. All hyperparameters are carefully tuned to obtain the optimal test log-likelihood. Specifically, we choose the basis function as $\tilde{\phi} = \text{Beta}(\tilde{\alpha} = 10, , \tilde{\beta} = 10, \text{scale} = 10, \text{shift} = 0)$ with support $[0, T_\phi = 10]$; the hyperparameter α of Laplace prior is optimised to be 0.5; the hyperparameter $\boldsymbol{\eta}$ is set to $\mathbf{1}$ to represent a uniform Dirichlet prior; the number of quadrature nodes is set to 4000 and the number of iterations to 100, which is large enough for convergence. We employ the same sets of hyperparameters provided in the code repositories from the corresponding baseline models. Our goal is to compare the test log-likelihood with baseline models, infer the interactions among the neural population in different states, and check if dynamic-SNMHP can find the similarity of interaction networks when the spike train data of consistent macroscopic properties is forced into two states.

Results The functional connectivity estimated by SNMHP and dynamic-SNMHP is shown in Figs. 11b and 11c respectively. Most of the interactions are excitatory. Obviously, the interaction networks estimated in wake and sleep states from dynamic-SNMHP are similar to that from SNMHP. Some neurons, e.g., #36, #38, #46, have a strong impact on all the other neurons. This observation is consistent with the common sense that the neurons of ‘output’ type always correspond to the ones with a high firing rate (Fig. 11a). Similar to Section 7.4, we analyze the macroscopic properties of the network quantificationally w.r.t. test log-likelihood and CCDF. The test log-likelihood curves of SNMHP and dynamic-SNMHP are shown in Fig. 11d where the converged test log-likelihood of SNMHP and dynamic-SNMHP are close to each other. This demonstrates the similarity of macroscopic properties in two states. Moreover, as shown in Fig. 11e, the test log-likelihood of our models is competitive with TR-Hawkes being the champion, SNMHP and dynamic-SNMHP being the runners-up. This again demonstrates the excellent fitting performance of SNMHP and dynamic-SNMHP. Besides, the CCDF of dynamic-SNMHP in two states is approximately consistent with that of SNMHP (shown in Fig. 11f), which also demonstrates that the macroscopic properties of networks are similar in wake and sleep states.

The 50-dimensional neural spike train data is a challenge for the inference. For example, both SA-Hawkes and TR-Hawkes produce an out-of-memory error with the original model size on this dataset due to the high dimensionality. To address this problem, we reduce the size of both models to 1 head and 1 layer. For efficiency, the mean-field approximation of SNMHP costs 40 minutes 20 seconds, that of dynamic-SNMHP costs 40 minutes 34 seconds, while IN-Hawkes and MR-PP are slow and cannot finish in 24 hours. This demonstrates that SNMHP and dynamic-SNMHP can perform inference for high-dimensional spike train data with a reasonable running time.

Conclusively, all the experimental results above demonstrate that for spike train data with consistent macroscopic properties, dynamic-SNMHP can find a similar pattern even though the spike train is forced into several states. This validates that dynamic-SNMHP will automatically degrade to SNMHP when the patterns of spike train data in different states are highly consistent. More importantly, our proposed SNMHP and dynamic-SNMHP can be applied to high-dimensional neural spike train data.

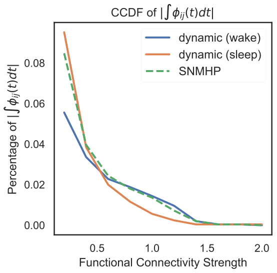


(a) (b)

(c) (d)

Models		Test Log-likelihood
Deep	NE-Hawkes	18801.93
	SA-Hawkes	24024.99
	TR-Hawkes	**25390.56**
Single-State	GC-Hawkes	23542.86
	IN-Hawkes	—[3]
	MR-PP	—[3]
	SNMHP	25150.28
Multi-State	SD-Hawkes	22269.46
	Dynamic-SNMHP	25149.47

(e)



(f)

Figure 11: The dataset of rat frontal cortex. (a): The training (left) and test (right) spike trains with red indicating wake state and blue indicating sleep state. (b): The heat map of functional connectivity among 50 neurons from SNMHP. (c): That from dynamic-SNMHP, left: wake state, right: sleep state. (d): The test log-likelihood curves w.r.t. iterations from SNMHP and dynamic-SNMHP. (e): The test log-likelihood results of baseline models. (f): The CCDF of functional connectivity of SNMHP and dynamic-SNMHP in two states.

8. Related Work

In this section, we introduce some related works about linear/nonlinear Hawkes processes, deep point processes and latent variable augmentation.

Linear Hawkes Processes The traditional linear Hawkes processes model is assumed to be in a parametric form, as we introduced in Eq. (3), which limits its expressiveness

[3] The baseline models cannot finish in 24 hours.

severely because the actual influence function or base rate can be flexible in different applications. Extending the expressiveness of linear Hawkes processes has been a long-standing research topic and many papers have been published in this area. Lewis and Mohler (2011) proposed to extend both the base rate μ and influence function $\phi(\cdot)$ to flexible functions and perform estimation by solving Euler-Lagrange equations, and Zhou et al. (2013) extended this method to multivariate Hawkes processes. Similarly, Zhou et al. (2020, 2021a) employed the Gaussian processes to model the flexible base rate and influence function, and performed Bayesian inference with different methods. To model stochastic influence functions, Dassios et al. (2013) assumed the exponential decay influence functions have i.i.d. random excitations, and Lee et al. (2016) further assumed all excitations are subject to a stochastic process and modeled by stochastic differential equations. Alaa et al. (2017); Xu et al. (2017); Morariu-Patrichi and Pakkanen (2018); Wu et al. (2019) further extended the traditional linear Hawkes processes to the time-varying versions.

Nonlinear Hawkes Processes The nonlinear Hawkes processes are more general than the classic linear Hawkes processes because nonlinear variant can characterize the inhibitive influence from past events to future ones. This makes the nonlinear Hawkes processes well suited in neuroscience where both excitatory and inhibitory interactions exist among neuron populations. Various nonlinear functions are used to map the real-valued activation to a nonnegative conditional intensity, such as rectifier (Reynaud-Bouret et al., 2013), exponential (Gerhard et al., 2017), sigmoid (Linderman, 2016; Apostolopoulou et al., 2019) or directly learned from data (Wang et al., 2016). Escola et al. (2011) also proposed to use Markov-modulated nonlinear Hawkes processes to study switching neural responses. For most nonlinear Hawkes processes, the nonlinear maps complicate the likelihood function to constitute a non-conjugate problem, but the sigmoid mapping function has the advantage that the Pólya-Gamma augmentation technique can be utilized to transform the non-conjugate problem into a conditional conjugate one. For this reason, we also utilize sigmoid nonlinearity in this work. Besides, most of these works employ parametric influence functions, e.g., exponential decay, to characterize interactions between neurons, but this is inconsistent with complex interactions in real neural recordings. In contrast with these works, the influence function is assumed to be a weighted sum of basis functions in our proposed models, which is more flexible in practice.

Deep Point Processes Another line of point process models is the deep point processes which are a class of point process models based on cutting-edge deep neural networks. Du et al. (2016) proposed a recurrent neural network (RNN) based temporal point process whose conditional intensity is formulated as a function of the hidden state of the RNN. Boyd et al. (2020) further extended the RNN-based temporal point process to the multi-source scenarios, where event sequences are assumed to be from different distributions. Some other variant models were also developed to use long short-term memory (LSTM) (Mei and Eisner, 2017; Mei et al., 2020) and attention-based transformer architecture (Zhang et al., 2020; Zuo et al., 2020). To circumvent the intensity integral issue, some recent works proposed to model the cumulative intensity function rather than the intensity function itself with deep neural networks (Omi et al., 2019; Shchur et al., 2020). The deep point processes have superior expressiveness due to the powerful fitting ability of neural networks. However, they are prone to be overfitting and lack the interpretability in neuroscience domain because they always directly model the intensity function and ignore the influence function.

Latent Variable Augmentation The auxiliary latent variable augmentation technique has been introduced into the Bayesian nonparametric inference of Poisson process and linear Hawkes process in many existing works. Adams et al. (2009) proposed a nonparametric Poisson process whose intensity is modeled as a sigmoid transformed Gaussian process and the Poisson process likelihood is augmented with thinned points to construct a tractable but inefficient MCMC inference algorithm. To improve the inference efficiency, Donner and Opper (2018) proposed to augment the Poisson process likelihood with Pólya-Gamma random variables and latent marked Poisson processes. As a result, the augmented likelihood is conditional conjugate to the Gaussian process prior and efficient Bayesian inference algorithms can be derived. Zhou et al. (2020) further extended the approach in Donner and Opper (2018) to linear single-variate Hawkes process by introducing an additional branching latent variable. For nonlinear Hawkes process, Linderman (2016) proposed a discrete-time model to convert the likelihood from a Poisson process to a Poisson distribution; then Pólya-Gamma random variables are augmented on discrete observations to propose a Gibbs sampler. This method is further extended to a continuous-time regime in Apostolopoulou et al. (2019) by augmenting thinned points and Pólya-Gamma random variables to propose a Gibbs sampler. However, the influence function is limited to be purely exciting or inhibitive exponential decay. Besides, due to the nonconjugacy of the excitation parameter of exponential decay influence function, a Metropolis-Hastings sampling step has to be embedded into the Gibbs sampler making the inference inefficient. Unlike aforementioned works, our models utilize multiple basis functions to characterize influence functions to guarantee flexibility; for inference, except Pólya-Gamma random variables, we also augment the nonlinear Hawkes process likelihood with latent marked Poisson processes and the Laplace prior with sparsity variables to construct a conditional conjugate model. As a result, three efficient Bayesian inference algorithms can be derived. It is worth noting that although our proposed Gibbs sampler is less efficient than the EM algorithm and mean-field approximation, but it has better efficiency than Apostolopoulou et al. (2019) since the time-consuming Metropolis-Hasting sampling is not needed. In Apostolopoulou et al. (2019), a tighter intensity upper bound is used to reduce the number of thinned points to accelerate the sampler. Instead, our EM algorithm and mean-field approximation do not encounter this problem as we compute the expectation rather than sampling.

9. Discussion

It is worth noting that the SNMHP model was originally proposed in Zhou et al. (2021b) and the corresponding EM algorithm is derived there for the inference. In this work, we further derive the additional Gibbs sampler and mean-field approximation for it. Moreover, we extend SNMHP to dynamic-SNMHP in this work to handle a time-varying neural system, for which three efficient inference algorithms: Gibbs sampler, EM algorithm and mean-field approximation, are derived.

We have proposed three inference algorithms for both the static and dynamic models in this work. Ones may wonder if there is a dominant method surpassing all the others and particularly recommended in practice. We remark that each inference algorithm has its own pros and cons. Theoretically, the Gibbs sampler enables the direct characterization of the posterior over parameters without reliance on any approximation. Unfortunately, as revealed by our experiments, the Gibbs sampler suffers from an inefficiency issue. The EM algorithm is able to precisely find the MAP solution. Yet, as a point estimator, it precludes the modeling of the uncertainty over parameters. Mean-field approximation conjoins the merits of Gibbs sampler and EM algorithm, capable of reasoning about parameter uncertainty in an efficient way, but it induces approximation error and lacks the guarantee of asymptotic consistency. In real-world applications, if the accuracy is the primary demand regardless of how much the cost is, the Gibbs sampler is recommended. If the efficiency is of key concern and the uncertainty over parameters is dispensable, the EM algorithm is the most appropriate. If we want to keep efficiency without compromising uncertainty quantification, it is better to use mean-field approximation.

10. Conclusion

In this paper, we develop an SNMHP model in the continuous-time regime which can characterize excitation-inhibition-mixture interactions among the neural population. To address the non-conjugate problem, three auxiliary latent variables are augmented into the likelihood and prior to convert the non-conjugate model to a conditional conjugate model. As a result, three efficient Bayesian inference algorithms: Gibbs sampler, EM algorithm and mean-field approximation are derived in closed form with superior efficiency. To empower SNMHP to reconcile with time-varying neural systems, we extend SNMHP to dynamic-SNMHP by incorporating a Markov state process to interact with point processes constituting a closed-loop framework. For inference, three efficient Bayesian inference algorithms for SNMHP are extended to dynamic-SNMHP.

The synthetic data experimental results confirm that three inference algorithms have similar accuracy; the EM algorithm and mean-field approximation have better efficiency than the Gibbs sampler. In practice, which inference algorithm to use depends on desired requirements of the application. The experimental comparison with state-of-the-art competitors on real neural recordings demonstrates that: the fitting performance of SNMHP is superior to single-state baselines and that of dynamic-SNMHP surpasses other single- and multi-state models; SNMHP and dynamic-SNMHP are applicable to high-dimensional neural spike train data; dynamic-SNMHP degrades to SNMHP automatically when the patterns of spike train data in different states are highly consistent.

From the application perspective, although our models are proposed in the neuroscience domain, they can be applied to other applications where the inhibition is a vital factor or the event dynamics are changing with system state, e.g., in the coronavirus (COVID-19) spread, the inhibitive effect may represent the medical treatment or cure, or the forced quarantine by the government; in the high-frequency trading markets, the state of limit order book has a vital impact on the arrival rate of orders because it implies the trend of price change.

In this work, we utilize the mixture of basis functions to characterize flexible influence functions. Future work can be done to represent influence functions in a nonparametric way, e.g., Gaussian process, which raises a greater challenge for inference. For dynamic-SNMHP, the system state takes value in a discrete finite state space and it can be extended to be a continuously varying quantity in the future, where the activation weights will also change continuously over time. For efficiency, the stochastic EM and mean-field using mini-batch can be developed in the future to further accelerate the inference.

Acknowledgments

The authors would like to thank the anonymous reviewers for insightful comments which greatly improved the paper. This work was supported by the National Key Research and Development Program of China (2021ZD0110502, 2017YFA0700904), NSFC Projects (Nos. 62061136001, 61621136008, 62106121, U19B2034, U19A2081, U1811461), the major key project of PCL (No. PCL2021A12), and Tsinghua Guo Qiang Institute, and the High Performance Computing Center, Tsinghua University. F. Zhou was partially funded by China Postdoctoral Science Foundation.

Appendix A. Campbell's Theorem

Let $\Pi_{\hat{\mathcal{Z}}} = \{(\mathbf{z}_n, \boldsymbol{\omega}_n)\}_{n=1}^N$ be a marked Poisson process on the product space $\hat{\mathcal{Z}} = \mathcal{Z} \times \Omega$ with intensity $\Lambda(\mathbf{z}, \boldsymbol{\omega}) = \Lambda(\mathbf{z})p(\boldsymbol{\omega} \mid \mathbf{z})$. $\Lambda(\mathbf{z})$ is the intensity for the unmarked Poisson process $\{\mathbf{z}_n\}_{n=1}^N$ with $\boldsymbol{\omega}_n \sim p(\boldsymbol{\omega}_n \mid \mathbf{z}_n)$ being an independent mark drawn at each $\mathbf{z}_n$. Furthermore, we define a function $h(\mathbf{z}, \boldsymbol{\omega}) : \mathcal{Z} \times \Omega \to \mathbb{R}$ and the sum $H(\Pi_{\hat{\mathcal{Z}}}) = \sum_{(\mathbf{z},\boldsymbol{\omega})\in\Pi_{\hat{\mathcal{Z}}}} h(\mathbf{z}, \boldsymbol{\omega})$. If $\Lambda(\mathbf{z}, \boldsymbol{\omega}) < \infty$, then

$$\mathbb{E}_{\Pi_{\hat{\mathcal{Z}}}}\left[\exp\left(\xi H(\Pi_{\hat{\mathcal{Z}}})\right)\right] = \exp\left[\int_{\hat{\mathcal{Z}}}\left(e^{\xi h(\mathbf{z},\boldsymbol{\omega})} - 1\right)\Lambda(\mathbf{z}, \boldsymbol{\omega})d\boldsymbol{\omega}d\mathbf{z}\right],$$

for any $\xi \in \mathbb{C}$. The above equation defines the characteristic functional of a marked Poisson process. This proves Eq. (11) in the paper. The mean is

$$\mathbb{E}_{\Pi_{\hat{\mathcal{Z}}}}\left[H(\Pi_{\hat{\mathcal{Z}}})\right] = \int_{\hat{\mathcal{Z}}} h(\mathbf{z}, \boldsymbol{\omega})\Lambda(\mathbf{z}, \boldsymbol{\omega})d\boldsymbol{\omega}d\mathbf{z},$$

which is used when substituting Eq. (17) into Eq. (16) in the EM algorithm, and Eq. (20) into Eq. (19) in the mean-field approximation for SNMHP. The same applies to the dynamic-SNMHP.

Appendix B. Derivation of Augmented Likelihood and Prior

Substituting Eqs. (9) and (11) into Eq. (7), we obtain

$$\begin{aligned}
p(D \mid \mathbf{w}_i, \overline{\lambda}_i) &= \prod_{n=1}^{N_i} \overline{\lambda}_i \sigma(h_i(t_n^i)) \exp\left(-\int_0^T \overline{\lambda}_i \sigma(h_i(t))dt\right) \\
&= \prod_{n=1}^{N_i}\left(\int_0^\infty \overline{\lambda}_i e^{f(\omega_n^i, h_i(t_n^i))} p_{\mathrm{PG}}(\omega_n^i \mid 1, 0)d\omega_n^i\right) \cdot \mathbb{E}_{p_{\lambda_i}}\left[\prod_{(\omega,t)\in\Pi_i} e^{f(\omega,-h_i(t))}\right] \\
&= \iint \prod_{n=1}^{N_i}\left[\lambda_i(t_n^i, \omega_n^i)e^{f(\omega_n^i, h_i(t_n^i))}\right] \cdot p_{\lambda_i}(\Pi_i \mid \overline{\lambda}_i) \prod_{(\omega,t)\in\Pi_i} e^{f(\omega,-h_i(t))} d\boldsymbol{\omega}_i d\Pi_i.
\end{aligned}$$

where $\boldsymbol{\omega}_i$ is the vector of ω_n^i and $\lambda_i(t_n^i, \omega_n^i) = \overline{\lambda}_i p_{\mathrm{PG}}(\omega_n^i \mid 1, 0)$. It is straightforward to see the integrand is the augmented likelihood

$$p(D, \Pi_i, \boldsymbol{\omega}_i \mid \mathbf{w}_i, \overline{\lambda}_i) = \prod_{n=1}^{N_i}\left[\lambda_i(t_n^i, \omega_n^i)e^{f(\omega_n^i, h_i(t_n^i))}\right] \cdot p_{\lambda_i}(\Pi_i \mid \overline{\lambda}_i) \prod_{(\omega,t)\in\Pi_i} e^{f(\omega,-h_i(t))},$$

which is Eq. (13a). Similarly, the integrand in Eq. (12) is the augmented prior in Eq. (13b). The same applies to the dynamic-SNMHP where the difference is the incorporation of the state-transition matrix.

Appendix C. Derivation of Gibbs, EM and Mean-Field

C.1 Gibbs Sampler

Based on the augmented joint distribution in Eq. (14), we can derive the conditional densities of latent variables and parameters in closed form. By sampling from these conditional densities iteratively, we construct an analytical Gibbs sampler. Because the derivation is relatively easy for $\boldsymbol{\omega}_i$, $\boldsymbol{\beta}_i$, $\overline{\lambda}_i$ and $\mathbf{w}_i$ but difficult for Π_i, we here elaborate the derivation for Π_i and omit that for other variables. The posterior of Π_i is dependent on $\mathbf{w}_i$ and $\overline{\lambda}_i$

$$p(\Pi_i \mid \mathbf{w}_i, \overline{\lambda}_i) = \frac{p_{\lambda_i}(\Pi_i \mid \overline{\lambda}_i)\prod_{(\omega,t)\in\Pi_i} e^{f(\omega,-h_i(t))}}{\int p_{\lambda_i}(\Pi_i \mid \overline{\lambda}_i)\prod_{(\omega,t)\in\Pi_i} e^{f(\omega,-h_i(t))} d\Pi_i},$$

where Campbell's theorem can be applied to convert the denominator, the equation above can be transformed as

$$\begin{aligned} p(\Pi_i \mid \mathbf{w}_i, \overline{\lambda}_i) &= \frac{p_{\lambda_i}(\Pi_i \mid \overline{\lambda}_i)\prod_{(\omega,t)\in\Pi_i} e^{f(\omega,-h_i(t))}}{\exp\left(-\iint (1-e^{f(\omega,-h_i(t))})\overline{\lambda}_i p_{\mathrm{PG}}(\omega \mid 1,0) d\omega dt\right)} \\ &= \prod_{(\omega,t)\in\Pi_i}\left(e^{f(\omega,-h_i(t))}\overline{\lambda}_i p_{\mathrm{PG}}(\omega \mid 1,0)\right)\cdot \exp\left(-\iint e^{f(\omega,-h_i(t))}\overline{\lambda}_i p_{\mathrm{PG}}(\omega \mid 1,0) d\omega dt\right). \end{aligned}$$

The above posterior is in the likelihood form of a marked Poisson process with intensity

$$\Lambda_i(t,\omega \mid \mathbf{w}_i, \overline{\lambda}_i) = e^{f(\omega,-h_i(t))}\overline{\lambda}_i p_{\mathrm{PG}}(\omega \mid 1,0) = \overline{\lambda}_i \sigma(-h_i(t)) p_{\mathrm{PG}}(\omega \mid 1, h_i(t)).$$

C.2 EM Algorithm

In the standard EM algorithm framework, the lower bound of log-posterior has been provided in Eq. (16). For the E step, the posterior of latent variables is already provided in Eq. (15); the only difference is the activation weights and intensity upper bounds are from the last iteration. For the M step, we elaborate the derivation below.

Substituting the posterior distributions of latent variables into Eq. (16), we obtain the lower bound $\mathcal{Q}$. The first term of Eq. (16) is

$$\begin{aligned} \mathbb{E}_{\boldsymbol{\omega}_i,\Pi_i}\left[\log p(D, \boldsymbol{\omega}_i, \Pi_i \mid \mathbf{w}_i, \overline{\lambda}_i)\right] = &-\frac{1}{2}\mathbf{w}_i^\top \cdot \int_0^T A_i(t)\boldsymbol{\Phi}(t)\boldsymbol{\Phi}^\top(t) dt \cdot \mathbf{w}_i + \mathbf{w}_i^\top \cdot \int_0^T B_i(t)\boldsymbol{\Phi}(t) dt \\ &- \overline{\lambda}_i T + \left(N_i + \iint \Lambda_i(t,\omega) d\omega dt\right)\log \overline{\lambda}_i + C \end{aligned}$$

where we utilize the mean rule in Campbell's theorem, C is a constant and

$$\begin{aligned} A_i(t) &= \sum_{n=1}^{N_i}\mathbb{E}[\omega_n^i]\delta(t-t_n^i) + \int_0^\infty \omega\Lambda_i(t,\omega) d\omega, \\ B_i(t) &= \frac{1}{2}\sum_{n=1}^{N_i}\delta(t-t_n^i) - \frac{1}{2}\int_0^\infty \Lambda_i(t,\omega) d\omega, \end{aligned}$$

with $\delta(\cdot)$ being the Dirac delta function and $\mathbb{E}[\omega_n^i] = 1/(2h_i^{s-1}(t_n^i))\tanh(h_i^{s-1}(t_n^i)/2)$ (Polson et al., 2013). The integral of intensity function has no closed-form solution but can be solved by numerical quadrature methods. The second term of Eq. (16) is

$$\mathbb{E}_{\boldsymbol{\beta}_i}\left[\log p(\mathbf{w}_i, \boldsymbol{\beta}_i)\right] = -\frac{1}{2}\mathbf{w}_i^\top \cdot \mathrm{diag}\left(\frac{\mathbb{E}[\boldsymbol{\beta}_i]}{\alpha^2}\right)\cdot \mathbf{w}_i + C,$$

where C is a constant, $\mathbb{E}[\boldsymbol{\beta}_i] = \{\mathbb{E}[\beta_{ijb}]\}_{jb}^{MB+1} = \{\alpha / w_{ijb}^{s-1}\}_{jb}^{MB+1}$ and $\text{diag}(\cdot)$ indicates the diagonal matrix of a vector.

The updated parameters $\overline{\lambda}_i^s$ and $\mathbf{w}_i^s$ can be obtained by setting the gradient of $\mathcal{Q}$ to zero. Due to auxiliary variables augmentation, we obtain an analytical expression

$$\overline{\lambda}_i^s = \left(N_i + R_i\right)/T,$$
$$\mathbf{w}_i^s = \boldsymbol{\Sigma}_i \int_0^T B_i(t)\boldsymbol{\Phi}(t)dt,$$

where $R_i = \int_0^T \int_0^\infty \Lambda_i(t, \omega \mid \mathbf{w}_i^{s-1}, \overline{\lambda}_i^{s-1}) d\omega dt$, $\boldsymbol{\Sigma}_i = \left[\int_0^T A_i(t)\boldsymbol{\Phi}(t)\boldsymbol{\Phi}^\top(t)dt + \text{diag}\left(\alpha^{-2}\mathbb{E}[\boldsymbol{\beta}_i]\right)\right]^{-1}$. Numerical quadrature methods need to be applied to intractable integrals above.

C.3 Mean-Field Approximation

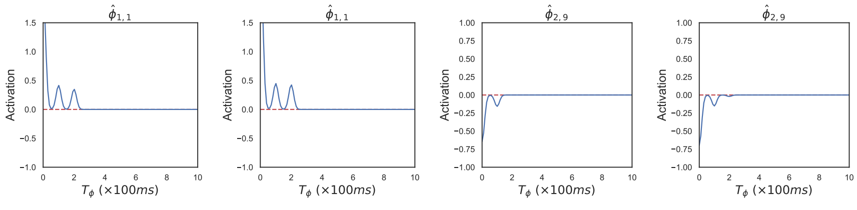
In the mean-field approximation framework, substituting the augmented joint distribution Eq. (14) into Eq. (19), we can obtain the optimal distribution for each factor. For $\boldsymbol{\omega}_i$ and $\boldsymbol{\beta}_i$, the derivation is relatively easy; for Π_i, the derivation is similar to that in Appendix C.1; for $\overline{\lambda}_i$ and $\mathbf{w}_i$, the derivation is similar to that in Appendix C.2.

C.4 Extension to Dynamic-SNMHP

The above derivation of Gibbs sampler, EM algorithm and mean-field approximation for SNMHP can be easily extended to dynamic-SNMHP. The only difference is the incorporation of state-transition matrix. Because the likelihood of dynamic-SNMHP factorizes as state process likelihood and point process likelihood, and the Dirichlet prior is conjugate to the state process likelihood, the incorporation of state process does not complicate the inference severely.

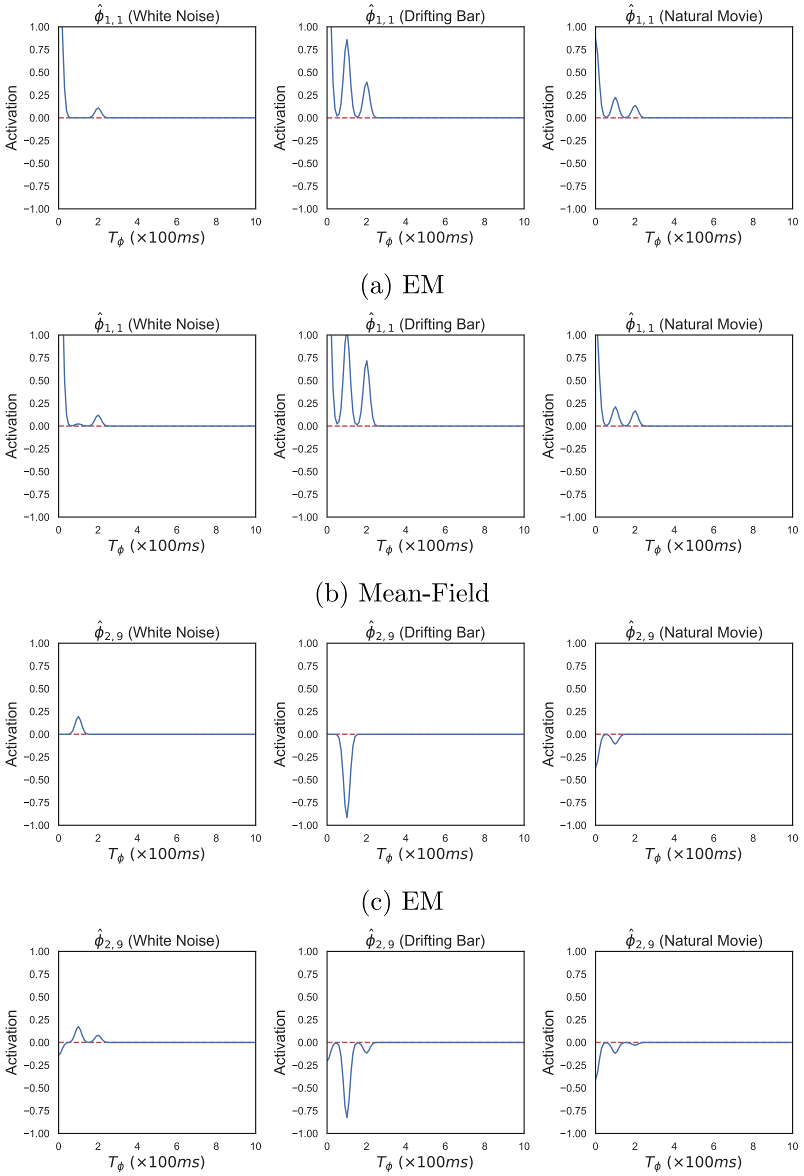
Appendix D. Interactions among Neural Populations

In this section, we visualize some estimated interactions among the neurons in cat primary visual cortex. Due to sparsity, most of the interactions are almost 0, here we show the influence functions $\hat{\phi}_{11}$ and $\hat{\phi}_{29}$ estimated by SNMHP and dynamic-SNMHP as an example.



(a) EM (b) Mean-Field (c) EM (d) Mean-Field

Figure 12: For SNMHP: The influence functions $\hat{\phi}_{11}$ and $\hat{\phi}_{29}$ estimated by EM algorithm and mean-field approximation. For mean-field, it is evaluated by the mean. We can see the estimations by two algorithms are similar to each other.



(a) EM

(b) Mean-Field

(c) EM

(d) Mean-Field

Figure 13: For dynamic-SNMHP: The influence functions $\hat{\phi}_{11}$ and $\hat{\phi}_{29}$ under three different stimuli estimated by EM algorithm and mean-field approximation. For mean-field, it is evaluated by the mean. We can see the estimations by two algorithms are similar to each other and the influence functions under different stimuli are quite different from the one estimated by SNMHP.

Appendix E. Maximization of Log-posterior with Numerical Optimization

One anonymous reviewer pointed out that an important baseline to compare against is to maximize the log-posterior (Eq. (8)) of the nonlinear Hawkes process model proposed in our work directly, using Gaussian quadrature to approximate the integral of intensity without any augmentation, since this can demonstrate the advantage of the proposed EM algorithm on the extended space. We compare the convergence of our proposed EM for SNMHP with the numerical optimization implemented by the 'SLSQP' method in 'scipy.optimize.minimize' on the real data in Fig. 14. We can see both methods finally converge to a similar training log-likelihood, which proves the efficacy of our EM algorithm. More importantly, our EM algorithms converges faster than the SLSQP method: the former converges in 40 steps while the latter in more than 300 steps.

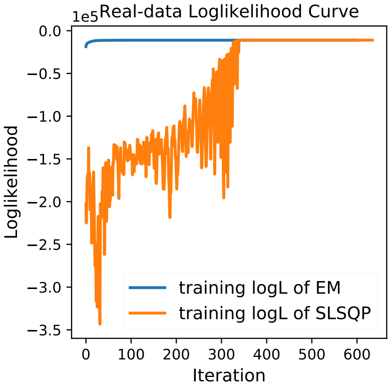


Figure 14: The training log-likelihood curves of our proposed EM algorithm for SNMHP and that of the numerical optimization implemented by 'SLSQP' method in 'scipy.optimize.minimize' on the real data.

References

Ryan Prescott Adams, Iain Murray, and David JC MacKay. Tractable nonparametric Bayesian inference in Poisson processes with Gaussian process intensities. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 9–16. ACM, 2009.

Ahmed M Alaa, Scott Hu, and Mihaela Schaar. Learning from clinical judgments: semi-Markov-modulated marked Hawkes processes for risk prognosis. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2017.

James H Albert and Siddhartha Chib. Bayesian analysis of binary and polychotomous response data. *Journal of the American statistical Association*, 88(422):669–679, 1993.

Ifigeneia Apostolopoulou, Scott Linderman, Kyle Miller, and Artur Dubrawski. Mutually regressive point processes. In *Advances in Neural Information Processing Systems*, pages 5116–5127, 2019.

Emmanuel Bacry and Jean-François Muzy. Hawkes model for price and trades high-frequency dynamics. *Quantitative Finance*, 14(7):1147–1166, 2014.

Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005, 2015.

Danielle S Bassett and Olaf Sporns. Network neuroscience. *Nature neuroscience*, 20(3):353, 2017.

Christopher M Bishop. *Pattern Recognition and Machine Learning.* springer, 2006.

Tim Blanche. Multi-neuron recordings in primary visual cortex. CRCNS.org. http://dx.doi.org/10.6080/K0MW2F2J, 2009.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

Alex Boyd, Robert Bamler, Stephan Mandt, and Padhraic Smyth. User-dependent neural sequence models for continuous-time event data. *Advances in Neural Information Processing Systems*, 33:21488–21499, 2020.

Pierre Brémaud and Laurent Massoulié. Stability of nonlinear Hawkes processes. *The Annals of Probability*, pages 1563–1588, 1996.

Emery N Brown, Riccardo Barbieri, Valérie Ventura, Robert E Kass, and Loren M Frank. The time-rescaling theorem and its application to neural spike train data analysis. *Neural computation*, 14(2):325–346, 2002.

Emery N Brown, Robert E Kass, and Partha P Mitra. Multiple neural spike train data analysis: state-of-the-art and future challenges. *Nature neuroscience*, 7(5):456–461, 2004.

Daryl J Daley and David Vere-Jones. An introduction to the theory of point processes. vol. i. probability and its applications, 2003.

Angelos Dassios, Hongbiao Zhao, et al. Exact simulation of Hawkes process with exponentially decaying intensity. *Electronic Communications in Probability*, 18, 2013.

Christian Donner and Manfred Opper. Inverse Ising problem in continuous time: A latent variable approach. *Physical Review E*, 96(6):062104, 2017.

Christian Donner and Manfred Opper. Efficient Bayesian inference of sigmoidal Gaussian Cox processes. *Journal of Machine Learning Research*, 19(1):2710–2743, 2018.

Christian Donner, Klaus Obermayer, and Hideaki Shimazaki. Approximate inference for time-varying interactions and macroscopic dynamics of neural populations. *PLoS computational biology*, 13(1):e1005309, 2017.

Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. Recurrent marked temporal point processes: embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1555–1564. ACM, 2016.

Uri T Eden, Loren M Frank, Riccardo Barbieri, Victor Solo, and Emery N Brown. Dynamic analysis of neural encoding by point process adaptive filtering. *Neural computation*, 16 (5):971–998, 2004.

Sean Escola, Alfredo Fontanini, Don Katz, and Liam Paninski. Hidden markov models for the stimulus-response relationships of multistate neural systems. *Neural computation*, 23 (5):1071–1132, 2011.

Vladimir Filimonov and Didier Sornette. Apparent criticality and calibration issues in the Hawkes self-excited point process model: application to high-frequency financial data. *Quantitative Finance*, 15(8):1293–1314, 2015.

Felipe Gerhard, Moritz Deger, and Wilson Truccolo. On the stability and dynamics of stochastic spiking neuron models: Nonlinear Hawkes process and point process GLMs. *PLoS computational biology*, 13(2), 2017.

Walter R Gilks, Sylvia Richardson, and David Spiegelhalter. *Markov chain Monte Carlo in practice*. CRC press, 1995.

Gene H Golub and John H Welsch. Calculation of Gauss quadrature rules. *Mathematics of Computation*, 23(106):221–230, 1969.

Alan G Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106, 1962.

Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*, 13(8):1713–1720, 2001.

Robert E Kass, Uri T Eden, and Emery N Brown. *Analysis of neural data*, volume 491. Springer, 2014.

John Frank Charles Kingman. *Poisson processes*, volume 3. Clarendon Press, 1992.

Athanasios Kottas. Dirichlet process mixtures of Beta distributions, with applications to density and intensity estimation. In *Workshop on Learning with Nonparametric Bayesian Methods, 23rd International Conference on Machine Learning (ICML)*, volume 47, 2006.

Young Lee, Kar Wai Lim, and Cheng Soon Ong. Hawkes processes with stochastic excitations. In *International Conference on Machine Learning*, pages 79–88, 2016.

Erik Lewis and George Mohler. A nonparametric EM algorithm for multiscale Hawkes processes. *Journal of Nonparametric Statistics*, 1(1):1–20, 2011.

Erik Lewis, George Mohler, P Jeffrey Brantingham, and Andrea L Bertozzi. Self-exciting point process models of civilian deaths in Iraq. *Security Journal*, 25(3):244–264, 2012.

Scott Warren Linderman. *Bayesian Methods for Discovering Structure in Neural Spike Trains*. PhD thesis, Harvard University, 2016.

Arianna Maffei, Sacha B Nelson, and Gina G Turrigiano. Selective reconfiguration of layer 4 visual cortical circuitry by visual deprivation. *Nature neuroscience*, 7(12):1353–1359, 2004.

Hongyuan Mei and Jason M Eisner. The neural Hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, pages 6754–6764, 2017.

Hongyuan Mei, Guanghui Qin, Minjie Xu, and Jason Eisner. Neural datalog through time: Informed temporal modeling via logical specification. In *International Conference on Machine Learning*, pages 6808–6819. PMLR, 2020.

Thomas P. Minka. Expectation propagation for approximate Bayesian inference. In *Conference on Uncertainty in Artificial Intelligence*, pages 362–369, 2001.

George O Mohler, Martin B Short, P Jeffrey Brantingham, Frederic Paik Schoenberg, and George E Tita. Self-exciting point process modeling of crime. *Journal of the American Statistical Association*, 106(493):100–108, 2011.

Gianluigi Mongillo, Simon Rumpel, and Yonatan Loewenstein. Inhibitory connectivity defines the realm of excitatory plasticity. *Nature neuroscience*, 21(10):1463–1470, 2018.

Maxime Morariu-Patrichi and Mikko S Pakkanen. State-dependent Hawkes processes and their application to limit order book modelling. *arXiv preprint arXiv:1809.08060*, 2018.

Yosihiko Ogata. Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402, 1998.

Yosihiko Ogata. Seismicity analysis through point-process modeling: A review. In *Seismicity patterns, their statistical significance and physical meaning*, pages 471–507. Springer, 1999.

Takahiro Omi, Kazuyuki Aihara, et al. Fully neural network based model for general temporal point processes. *Advances in neural information processing systems*, 32, 2019.

Tohru Ozaki. Maximum likelihood estimation of Hawkes' self-exciting point processes. *Annals of the Institute of Statistical Mathematics*, 31(1):145–155, 1979.

Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.

Donald H Perkel, George L Gerstein, and George P Moore. Neuronal spike trains and stochastic point processes: II. Simultaneous spike trains. *Biophysical journal*, 7(4):419–440, 1967.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya-Gamma latent variables. *Journal of the American statistical Association*, 108 (504):1339–1349, 2013.

Massimiliano Pontil, Sayan Mukherjee, and Federico Girosi. On the noise model of support vector machines regression. In *International Conference on Algorithmic Learning Theory*, pages 316–324. Springer, 2000.

Dale Purves, George J Augustine, David Fitzpatrick, WC Hall, AS LaMantia, JO McNamara, and L White. Neuroscience, 2008. *De Boeck, Sinauer, Sunderland, Mass*, pages 15–16, 2014.

Patricia Reynaud-Bouret, Vincent Rivoirard, and Christine Tuleau-Malot. Inference of functional connectivity in neurosciences via Hawkes processes. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 317–320. IEEE, 2013.

Marian-Andrei Rizoiu, Swapnil Mishra, Quyu Kong, Mark Carman, and Lexing Xie. SIR-Hawkes: linking epidemic models and Hawkes processes to model diffusions in finite populations. In *Proceedings of the 2018 World Wide Web Conference*, pages 419–428, 2018.

AI Saichev and Didier Sornette. Generating functions and stability study of multivariate self-excited epidemic processes. *The European Physical Journal B*, 83(2):271, 2011.

Oleksandr Shchur, Nicholas Gao, Marin Bilos, and Stephan Günnemann. Fast and flexible temporal point processes with triangular maps. In *Advances in neural information processing systems*, 2020.

Per Jesper Sjöström, Gina G Turrigiano, and Sacha B Nelson. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, 32(6):1149–1164, 2001.

Larry Squire, Darwin Berg, Floyd E Bloom, Sascha Du Lac, Anirvan Ghosh, and Nicholas C Spitzer. *Fundamental neuroscience*. Academic Press, 2012.

Alex M Thomson and A Peter Bannister. Interlaminar connections in the neocortex. *Cerebral cortex*, 13(1):5–14, 2003.

Luke Tierney and Joseph B Kadane. Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86, 1986.

E Vaadia, I Haalman, M Abeles, Hagit Bergman, Y Prut, Hi Slovin, and AMHJ Aertsen. Dynamics of neuronal interactions in monkey cortex in relation to behavioural events. *Nature*, 373(6514):515–518, 1995.

Yichen Wang, Bo Xie, Nan Du, and Le Song. Isotonic Hawkes processes. In *International conference on machine learning*, pages 2226–2234. PMLR, 2016.

Brendon O Watson, Daniel Levenstein, J Palmer Greene, Jennifer N Gelinas, and György Buzsáki. Network homeostasis and state dynamics of neocortical sleep. *Neuron*, 90(4): 839–852, 2016.

Jing Wu, Owen Ward, James Curley, and Tian Zheng. Markov-modulated Hawkes processes for sporadic and bursty event occurrences. *arXiv preprint arXiv:1903.03223*, 2019.

Hongteng Xu, Mehrdad Farajtabar, and Hongyuan Zha. Learning Granger causality for Hawkes processes. In *International conference on machine learning*, pages 1717–1726. PMLR, 2016.

Hongteng Xu, Dixin Luo, and Hongyuan Zha. Learning Hawkes processes from short doubly-censored event sequences. In *International Conference on Machine Learning*, pages 3831–3840. PMLR, 2017.

Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. Self-attentive Hawkes process. In *International Conference on Machine Learning*, pages 11183–11193. PMLR, 2020.

Feng Zhou, Zhidong Li, Xuhui Fan, Yang Wang, Arcot Sowmya, and Fang Chen. Efficient inference for nonparametric Hawkes processes using auxiliary latent variables. *Journal of Machine Learning Research*, 21(241):1–31, 2020.

Feng Zhou, Simon Luo, Zhidong Li, Xuhui Fan, Yang Wang, Arcot Sowmya, and Fang Chen. Efficient em-variational inference for nonparametric hawkes process. *Stat. Comput.*, 31 (4):46, 2021a.

Feng Zhou, Yixuan Zhang, and Jun Zhu. Efficient inference of flexible interaction in spiking-neuron networks. In *International Conference on Learning Representations*, 2021b.

Ke Zhou, Hongyuan Zha, and Le Song. Learning triggering kernels for multi-dimensional Hawkes processes. In *International Conference on Machine Learning*, pages 1301–1309, 2013.

Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer Hawkes process. In *International conference on machine learning*, pages 11692–11702. PMLR, 2020.