entropy

MDPI

Article

Variational Bayesian Inference for Nonlinear Hawkes Process with Gaussian Process Self-Effects

Noa Malem-Shinitski [1,*], César Ojeda [2] and Manfred Opper [2,3]

[1] Institute of Mathematics, University of Potsdam, 14476 Potsdam, Germany
[2] Artificial Intelligence Group, Technische Universität Berlin, 10623 Berlin, Germany; ojedamarin@tu-berlin.de (C.O.); manfred.opper@tu-berlin.de (M.O.)
[3] Centre for Systems Modelling and Quantitative Biomedicine, University of Birmingham, Birmingham B15 2TT, UK
* Correspondence: malem@uni-potsdam.de

Abstract: Traditionally, Hawkes processes are used to model time-continuous point processes with history dependence. Here, we propose an extended model where the self-effects are of both excitatory and inhibitory types and follow a Gaussian Process. Whereas previous work either relies on a less flexible parameterization of the model, or requires a large amount of data, our formulation allows for both a flexible model and learning when data are scarce. We continue the line of work of Bayesian inference for Hawkes processes, and derive an inference algorithm by performing inference on an aggregated sum of Gaussian Processes. Approximate Bayesian inference is achieved via data augmentation, and we describe a mean-field variational inference approach to learn the model parameters. To demonstrate the flexibility of the model we apply our methodology on data from different domains and compare it to previously reported results.

Keywords: Bayesian inference; point process; Gaussian process

Citation: Malem-Shinitski, N.; Ojeda, C.; Opper, M. Variational Bayesian Inference for Nonlinear Hawkes Process with Gaussian Process Self-Effects. *Entropy* **2022**, *24*, 356. https://doi.org/10.3390/e24030356

Academic Editors: Udo Von Toussaint and Philip Broadbridge

Received: 27 December 2021
Accepted: 22 February 2022
Published: 28 February 2022

1. Introduction

Sequences of self-exciting, or inhibiting, temporal events are frequent footmarks of natural phenomena: earthquakes are known to be temporally clustered as aftershocks are commonly triggered following the occurrence of a main event [1]; in social networks, the propagation of news can be modeled in terms of information cascades over the edges of a graph [2]; and in neuronal activity, the occurrence of one spike may increase or decrease the probability of the occurrence of the next spike over some time period [3].

Traditionally, sequences of events in continuous time are modeled by point processes, of which Cox processes [4], or doubly stochastic processes, use a stochastic process for the intensity function, which depends only on time and is not effected by the occurrences of the events. The Hawkes process [5] extends the Cox process to capture phenomena in which the past events affect future arrivals, by introducing a memory dependence via a memory kernel, which is also referred to as the causal influence function. When incorporating the dependence of the process on its own history, due to the superposition theorem of the point process, new events will depend on either an exogenous rate, which is independent of the history, or an endogenous rate from past arrivals.

Originally, the dependence on history in the Hawkes process is assumed to be self-excitatory, and the memory kernel is parameterized by an exponential or power law decay, which results in a model with low flexibility. Furthermore, assuming only an excitatory relation between the events does not hold for other phenomena we wish to model. For example, inhibitory effects between neurons [6], and even self-inhibition [7], are crucial for regulating the neuronal activity. Thus, the memory kernel should also include inhibitory relations between the events and by doing so the intensity may become negative. To ensure that the intensity function is non-negative, a nonlinear link function is applied on

the memory kernel, and the resulting model is often referred to as a nonlinear Hawkes process [8–10]. Theoretical results for nonlinear Hawkes processes have been developed for many years and they include stability estimates [8] as well as convergence rates for Bayesian estimators [11].

In this work, we present a Nonlinear Hawkes Process with Gaussian Process Self-Effects (NH–GPS), which extends the class of nonlinear Hawkes processes. As the causal influence between events can be either excitatory or inhibitory, we use the term *self-effects* to refer to the influence of past events on future events.

We choose a semi-parametric approach, which avoids the limiting parameterization of the memory kernel and the background rate. We assume a Gaussian process (GP) prior to the exogenous events' intensity and on the memory kernel, which allows also for an inhibitory effect between the events. To ensure that the intensity function is non-negative we use the sigmoid link function. This modeling approach is not only descriptive but also allows us to obtain a fast inference procedure. The history of self-effects defines an aggregated Gaussian process, and we perform the inference directly on this aggregation rather than obtaining a posterior over each self-effect.

While highly flexible approaches to modeling the intensity function of nonlinear Hawkes processes have been presented before, they mainly rely on deep neural network solutions [12,13]. These approaches are hindered by the necessity of large datasets. In contrast, our methodology retains modeling flexibility due to the nonparametric nature of Gaussian processes, while being able to perform well when data are scarce.

Outline

In Section 2, we describe the NH–GPS model and emphasize how its structure allows for efficient Bayesian inference. In Section 3, we describe the augmentation scheme and derive the mean-field variational inference algorithm. In Section 4, we discuss related work and describe how it relates to ours. In Section 5, we present the results of our experiments both on synthetic data and different real-world examples, and compare it to existing work when possible. In Section 6, we conclude by discussing our work and future research directions.

2. Proposed Model

2.1. Classical Hawkes Process

Let $\mathcal{T}_t = [0, t] \in \mathbb{R}$. We define the counting measure $N(\mathcal{T}_t)$ as the number of arrivals in the interval $\mathcal{T}_t$. Furthermore, we define the history $\mathcal{H}_t$ or the realizations of a given process, as the set of arrivals in the interval $\mathcal{T}_t$, namely $\mathcal{H}_t = \{T_1, \ldots, T_{N(\mathcal{T}_t)} : T_i \in \mathcal{T}_t \wedge T_{i-1} < T_i\}$, and T_i corresponds to the time of arrival i. For a temporal point process, the counting measure $N(\cdot)$ has an associated intensity defined as

$$\Lambda(t) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[N(\mathcal{T}_{t+\Delta t}) - N(\mathcal{T}_t)|\mathcal{H}_t]}{\Delta t}.$$

The intensity function may depend on the history of the process. An example of such a process is the Hawkes process, or self-exciting point process [14], which defines self-excitations [15] around *exogenous events*. Following Hawkes and Oakes [5], the intensity of the Hawkes process is defined by

$$\Lambda(t|\mathcal{H}_t) = s(t) + \sum_{t_n < t} g(t - t_n), \tag{1}$$

where $s(t)$ is the base intensity of exogenous arrivals and $g(t - t_n)$ is the memory kernel, or causal influence function, defining the change in the excitation or inhibition value following each arrival. In the classical Hawkes process, causal influence of only excitatory nature is allowed and the memory kernel is usually of the form $g(t - t_n) = \beta e^{-\alpha(t-t_n)}$ for an exponentially decaying memory.

2.2. Nonlinear Hawkes Process with Gaussian Process Self-Effects

In the classical Hawkes process, the memory kernel g in Equation (1) must be non-negative, to prevent the intensity function from being negative. As a result the history of the model has only excitatory effect on future events. We are interested in a model that includes inhibition between events, and we release the constraint over g so it can be negative, and define the following nonlinear intensity function

$$\Lambda(t) = \lambda\sigma(\phi(t)) \tag{2}$$

$$\sigma(\phi(t)) = \frac{1}{1+\exp\left(-\phi(t)\right)} \tag{3}$$

$$\phi(t) = s(t) + \sum_{t_n<t} g(t-t_n)\exp(-\alpha(t-t_n)). \tag{4}$$

Here, we choose the sigmoid function to ensure that the intensity function $\Lambda(\cdot)$ is non-negative. λ is the intensity bound and we refer to $\phi(\cdot)$ as the linear intensity function.

We explicitly add the exponential decay to enforce the forgetting constraint, which is essential for most realistic processes. Although we choose here a specific parameterization of the memory decay, one can choose other forms of memory decay with minimal adaptation to the learning procedure of the model parameters.

Besides α and λ, the functions $s(\cdot)$ and $g(\cdot)$ are the unknown parameters of the model to be inferred from the data. In this paper, we will use a nonparametric Bayesian inference approach based on the definition of a prior probability measure over these functions. In contrast to alternative Bayesian models for Hawkes processes where the positive rate function is directly modeled by a Dirichlet process, in our case we have to deal with random functions which are not constrained to be positive. This suggests that we should use a simple, but still highly flexible approach by modeling the two functions independently as realizations of Gaussian random processes. We write symbolically:

$$s \sim GP(0, K^s) \tag{5}$$

$$g \sim GP(0, K^g) \tag{6}$$

$$K^{s/g}(t_1, t_2) = a_{s/g} \cdot \exp\left(-\frac{\|t_1 - t_2\|^2}{\sigma_{s/g}^2}\right). \tag{7}$$

The corresponding Gaussian prior measures are uniquely defined by the mean functions (which we set to be equal to zero) and the second moments given by the covariance kernel functions $K^{g/s}$. The latter are defined by the prior expectations

$$K^s(t_1, t_2) \doteq E[s(t_1)s(t_2)] \tag{8}$$

$$K^g(t_1, t_2) \doteq E[g(t_1)g(t_2)] \tag{9}$$

By a proper choice of kernels, we can encode further prior beliefs on typical realizations of $s(\cdot)$ and $g(\cdot)$. Throughout the paper, we will work with the so-called RBF kernels from Equation (7). This kernel corresponds to the prior assumptions that the Gaussian processes are stationary (the kernels depend on time differences only) and that the functions $s(\cdot)$ and $g(\cdot)$ are infinitely often differentiable. The kernels depend on two hyperparameters a and σ, which reflect the typical amplitude and length scale of the functions. The reasonable values of these hyperparameters will also be inferred from the data.

Finally, we assume a prior distribution also on the upper intensity bound

$$\lambda \sim Gamma(\alpha_0, \beta_0).$$

and we identify the hyperparameters of the model as $\{\sigma_g, a_g, \alpha, \sigma_s, a_s\}$.

In this work, we propose Bayesian inference for fitting the model to data. Due to the nonlinearity over $\phi(\cdot)$, we are no longer able to easily utilize the branching structure of the

Hawkes process, which allowed for the estimation of $s(\cdot)$ and $g(\cdot)$ [16,17]. Thus, a natural solution is to perform the inference directly on $\phi(\cdot)$.

Next, we identify the prior over the entire linear intensity $p(\phi)$. From Equation (4) we see that the linear intensity function ϕ is nothing but the sum of GPs, and as such it is also a GP:

$$\phi \sim GP(0, \tilde{K}) \tag{10}$$

$$\tilde{K}_{lk} = K^s_{lk} + \sum_{t_i<t_l} \sum_{t_j<t_k} K^g_{t_l-t_i,t_k-t_j} \exp\big(-\alpha\big(t_l - t_i + t_k - t_j\big)\big). \tag{11}$$

Multivariate Model

We propose an extension of the model to multiple dimensions. This is useful in applications where different types of events are observed, or the events originate from different processes that affect each other. We define an R-dimensional point process with intensity in dimension r

$$\Lambda^r(t) = \lambda^r \sigma(\phi^r(t))$$

$$\phi^r(t) = s^r(t) + \sum_{m=1}^{R} \sum_{t_n^m<t} g_{r,m}(t - t_n^m) \exp(-\alpha_{r,m}(t - t_n^m))$$

where t_n^m is the time of event number n of type m. We assume that every dimension has its own intensity bound λ^k and background rate $s^r(\cdot)$. The different dimensions interact with each other via the self-effects term. $g_{r,m}(\cdot)$ defined the effects of the events of type m on the events of type r. As in the univariate case, this effect may change over time.

Given the observations, the different dimensions are independent of each other and we can learn their parameters separately. Thus, in the following section we present the inference for the univariate model, and the extension to the multivariate case is straight forward.

3. Inference

Conditioned on the intensity function $\Lambda(\cdot)$, the likelihood of observations $\{t_1, \dots t_N\}$ from a Hawkes process is [18]

$$\ell(\{t_1, \dots t_n\}|\Lambda(\cdot)) = \exp\Big\{-\int_0^T \Lambda(t')dt'\Big\} \prod_{n=1}^{N} \Lambda(t_n).$$

In order to obtain posterior distributions for the latent variables either in the form of a Gibbs sampler or through approximate posteriors in variational inference, we require certain computations that are not tractable or efficient under the current form of the likelihood. Similarly to previous work on Cox and Hawkes processes [17,19,20], we follow an augmentation procedure. We do so via the introduction of auxiliary variables, which expand the model to a different likelihood form. Under the marginalization of the aforementioned auxiliary variables, the new likelihood will conserve the form of the original model likelihood. The new form of the likelihood is constructed such that the computations required for the inference procedure are either tractable, computationally fast, or both.

3.1. Model Augmentation

The first step we take in treating the likelihood function is using the Pólya–Gamma (PG) augmentation scheme. Following Theorem 1 in Polson et al. [21], we can rewrite the nonlinear intensity function as

$$\sigma(\phi(t)) = \int_0^\infty e^{f(w,\phi(t))} PG(w; 1, 0)dw \tag{12}$$

$$f(w, \phi(t)) = -\frac{\phi(t)^2 w}{2} + \frac{\phi(t)}{2} - \ln 2. \tag{13}$$

As we augment each observation with a variable w_n from a PG distribution, the joint likelihood of the observed events $\{t_n\}$ and PG variables $\{w_n\}$ is

$$p\left(\{t_n\}_{n=1}^N, \{w_n\}_{n=1}^N | \phi, \lambda\right) = \exp\left(-\int_0^T \lambda\sigma(\phi(t))dt\right) \cdot \prod_{n=1}^N \lambda e^{f(w_n,t_n)} PG(w_n; 1, 0) \tag{14}$$

with

$$\exp\left\{-\int_0^T \lambda\sigma(\phi(t))dt\right\} = \exp\left(-\int_0^T \int_0^\infty \lambda PG(w; 1, 0)\left(1 - e^{f(w,-\phi(t))}\right)dwdt\right). \tag{15}$$

where we used $\sigma(t) = 1 - \sigma(-t)$.

Next, we utilize the Campbell's theorem [14], which states that for a Poisson process Π with intensity φ

$$\mathbb{E}_\varphi\left(\prod_{x\in\Pi} \exp(h(x))\right) = \exp\left(-\int (1 - \exp(h(x)))\varphi(x)dx\right).$$

Looking at Equation (15), we identify $x = (t, w)$ and $\varphi(t, w) = \lambda PG(w|1, 0)$ is the intensity of a marked Poisson process in $\mathcal{T}$ with marks $w \sim PG(0, 1)$. Furthermore, we determine $h(x) = f(w, -\phi(t))$. We can now rewrite the exponential in Equation (14) as

$$\exp\left\{-\int_0^T \lambda\sigma(\phi(t))dt\right\} = \mathbb{E}_\varphi\left(\prod_{m=1}^M e^{f\left(\hat{w}_m, \hat{t}_m\right)}\right) \tag{16}$$

for realizations $\{\hat{t}_m, \hat{w}_m\}_{m=1}^M$.

We substitute Equation (16) into Equation (14), which results in the full augmented likelihood. Given the prior distributions over ϕ and λ, we can now write the model's posterior distribution as

$$p\left(\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \lambda | \{t_n\}\right) \propto \exp(-\lambda T) \times \prod_{m=1}^M \lambda e^{f\left(\hat{w}_m, -\phi\left(\hat{t}_m\right)\right)} PG(w_m; 1, 0) \times \prod_{n=1}^N \lambda e^{f(w_n, \phi(t_n))} PG(w_n; 1, 0) \times p(\phi)p(\lambda). \tag{17}$$

To summarize, we augment the model with two sets of variables—the PG variables $\{w_n\}$, which augment the actual realizations, and the tuples $\{\hat{t}_m, \hat{w}_m\}$, which are the realizations and marks of the auxiliary marked Poisson process.

As mentioned above, we intend to learn directly the linear intensity function $\phi(\cdot)$. This allows us to utilize the mean-field variational inference previously introduced in Donner and Opper [19] and Donner and Opper [22]. Next, we go through the steps of the algorithm, and we refer the reader to the two papers mentioned above for further details. As a baseline we compare the performance of the variational inference algorithm to a Gibbs sampler. The details of the Gibbs sampler can be found in Appendix A and Algorithm 2.

3.2. Variational Inference

In the variational inference [23,24] we define a tractable distribution family and adapt it to approximate the posterior by maximizing the lower bound $\mathcal{L}(Q)$ defined below. This procedure minimizes the Kullback–Leibler divergence between the unknown posterior and the proposed approximating distribution. The posterior density is approximated by

$$\begin{aligned} &p(\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \lambda | \{t_n\}) \\ &\approx q_1(\phi, \lambda) q_2\Big(\{w_n\}_{n=1}^N, \{\hat{t}_m, \hat{w}_m\}_{m=1}^M\Big). \end{aligned}$$

This leads to the following lower bound on the evidence

$$\mathcal{L}(Q) = \mathbb{E}_Q\left[\log\left\{\frac{p(\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \lambda | \{t_n\})}{q_1(\phi, \lambda) q_2\big(\{w_n\}_{n=1}^N, \{\hat{t}_m, \hat{w}_m\}_{m=1}^M\big)}\right\}\right].$$

Here, Q refers to the probability measure of the variational posterior. We can maximize the bound by alternating the maximization over each of the factors [24]. The optimal solution for each factor is

$$\begin{aligned} &\log q_1^*(\phi, \lambda) = \qquad (18) \\ &\mathbb{E}_{q_2\left(\{w_n\}_{n=1}^N, \{\hat{t}_m, \hat{w}_m\}_{m=1}^M\right)}[\log P(\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \lambda, \{t_n\})] \\ &\log q_2^*\Big(\{w_n\}_{n=1}^N, \{\hat{t}_m, \hat{w}_m\}_{m=1}^M\Big) = \qquad (19) \\ &\mathbb{E}_{q_1(\phi,\lambda)}[\log P(\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \lambda, \{t_n\})]. \end{aligned}$$

Thus, to obtain the optimal distribution of one of the factors, one must calculate the expectations of the logarithm of the joint distribution over the remaining factors in the approximation, resulting in an iterative algorithm.

In the following subsections, we explicitly express the functional form of the optimal distributions, and obtain the corresponding expectations required for updating the factors.

3.3. Optimal Q_1

We find that the optimal q_1 is factorized as

$$q_1(\phi, \lambda) = q_1(\lambda) q_1(\phi)$$

The first factor is identified as a Gamma distribution

$$\begin{aligned} &q_1(\lambda) = Gamma(\alpha, \beta) \qquad (20) \\ &\alpha = \alpha_0 + N + \int_{\mathcal{T}x\mathcal{W}} \Lambda_{q_2}(t, w) dt dw \\ &\beta = \beta_0 + T \end{aligned}$$

with known expectations.

The optimal distribution for the second factor is of the form

$$\begin{aligned} q_1^\star &\propto e^{-U(\phi) + \log p(\phi)} \\ U(\phi) &= \frac{1}{2}\int A(t)\phi^2(t)dt - \int b(t)\phi(t)dt \\ A(t) &= \sum_n \langle \omega_n \rangle_{q_2^\star} \delta(t - t_n) + \langle \omega(t) \rangle_{q_2^\star} \Lambda_{q_2^\star}(t) \\ b(t) &= \sum_n \frac{1}{2}\delta(t - t_n) - \frac{1}{2}\Lambda_{q_2^\star}(t). \end{aligned}$$

Generally, the integrals above cannot be evaluated analytically. Thus, we resort to another variational approximation, where we approximate the likelihood term, by a distribution that depends only on a finite set of inducing point $\{c\}$, $\tilde{q}(\phi_c, \phi) = p(\phi|\phi_c)q(\phi_c)$, the ELBO is

$$\left\langle \log \frac{e^{-\log\langle U(\phi)\rangle_{p(\phi|\phi_c)}} p(\phi_c)}{\tilde{q}(\phi_c)} \right\rangle_{\tilde{q}}$$

and we use the notation $\langle p \rangle_q = \mathbb{E}_q(p)$. The optimal $\tilde{q}(\phi_c)$ is given by

$$\tilde{q}^\star(\phi_c) \propto e^{-\log\langle U(\phi)\rangle_{p(\phi|\phi_c)}} p(\phi_c).$$

From here, using the known results of conditional GPs and sparse variational GPs [25,26], we have

$$\begin{aligned} \tilde{q}^\star(\phi_c) &= \mathcal{N}(\phi_c|\mu_c, \Sigma_c) \\ \Sigma_c &= \left[\int \kappa(t)^\top A(t)\kappa(t)dt + K_c^{-1}\right]^{-1} \\ \mu_c &= \Sigma_c\left(\int b(t)\kappa(t)dt\right) \end{aligned} \tag{21}$$

with K_c the covariance kernel between the inducing points, $\kappa(t) = k_c(t)^\top K_c^{-1}$ and $k_c(t)$ is the kernel between the inducing points and another set of points (either the real data or the integration points), such that $k_c(t) = \left(\tilde{K}(t, t_1), \ldots, \tilde{K}(t, t_L)\right)$ with $t_1, \ldots, t_l, \ldots, t_L$ are the inducing points. The mean and the variance of the sparse approximated GP are

$$\langle \phi(t) \rangle = \kappa(t)\mu_c \tag{22}$$

$$\sigma^2(\phi(t)) = K(t,t) - \kappa(t)^\top k_c(t) + \kappa(t)^\top \Sigma_c \kappa(t) \tag{23}$$

3.4. Optimal Q_2

Similarly to the previous section, we find that the optimal q_2 is factorized as

$$q_2\left(\{w_n\}_{n=1}^N, \Pi\right) = q_2\left(\{w_n\}_{n=1}^N\right) q_2\left(\{\hat{t}_m, \hat{w}_m\}\right)$$

Given Equation (17), we define the first factor as

$$q_2^\star(w_n) \propto \exp\left(-\frac{\langle \phi_n^2 \rangle_{q_1^\star}}{2} w_n\right) PG(w_n|1, 0),$$

which corresponds to a tilted PG distribution

$$q_2^\star(w_n) = PG\left(w_n|1, \sqrt{\langle \phi_n^2 \rangle_{q_1^\star}}\right). \tag{24}$$

with known expectations [21].

The second factor takes the form

$$\begin{aligned} &q_2^\star(\{\hat{t}_m, \hat{w}_m\}_{m=1}^M) \\ &\propto \prod_{m=1}^M \exp\left(-\frac{\langle \phi_m \rangle_{q_1^\star}}{2} - \frac{\langle \phi_m^2 \rangle_{q_1^\star}}{2} w_m\right) \cdot \exp\left(\langle \ln \lambda^\star \rangle_{q_1^\star}\right). \end{aligned}$$

It can be shown that this distribution corresponds to a Poisson process with intensity function

$$\begin{aligned} &\Lambda_{q_2}(\hat{t}, \hat{w}) \\ &= \exp\left(\langle \ln \lambda \rangle_{q_1^\star}\right) \frac{\exp\left(-\frac{\langle \phi \rangle_{q_1^\star}}{2}\right)}{2\cosh\left(\langle \phi^2 \rangle_{q_1^\star}\right)} PG\left(w_m | 1, \sqrt{\langle \phi^2 \rangle_{q_1^\star}}\right) \end{aligned} \tag{25}$$

where to simplify the notation we write ϕ instead of $\phi(\hat{t})$.

The VI algorithm is summarized in Algorithm 1.

Algorithm 1: NH–GPS Variational Inference.

Input: Observed Events $\{t_n\}_{n=1}^N$, hyperparameters$\{\sigma_g, \alpha, a_s, \sigma_s, \alpha_0, \beta_0\}$
Output: Mean and variance of ϕ

1 Initialize. **while** $\mathcal{L}$ *not converged* **do**
2 Estimate the mean and covariance of the GP over the inducing points as in Equation (21)
3 Estimate the mean and variance of the GP over the observations as in Equations (22) and (23)
4 Estimate the mean intensity bound as in Equation (20)
5 Estimate the mean and variance of the augmenting PG variables as in Equation (24)
6 Estimate the intensity function over the augmenting events as in Equation (25)
7 **end**

Hyperparameters Tuning

The Variational Inference does not tune the hyperparameters of the model. In order to achieve a better model, we wish to tune the hyperparameters of the model and to improve the likelihood of the model. Thus, we perform one step of gradient descent with respect to the ELBO at each iteration. The derivatives of the ELBO are given in Donner and Opper [19] (Appendix F). Notice that hyperparameter training, or tuning, is an implicit form of model selection, in that we are selecting among different models through the evaluation of the likelihoods.

3.5. Identifying the Background Rate and the Self-Effects Function

For some applications we may be interested in the specific shape of the background rate function s and the self-effects function g. At a first glance it is not entirely clear how to recover them from the inference, and we next describe how to do so.

Upon the convergence of the variational inference algorithm, we have an expression for the posterior mean and variance of the linear intensity function ϕ, as described in Equations (22) and (23). It is useful to define

$$\tilde{g}(t) = g(t) \exp(-\alpha t) \tag{26}$$

and we similarly define

$$\tilde{K}^g(t, t') = K^g(t, t') \exp\left(-\alpha(t - t')\right). \tag{27}$$

We can rewrite the posterior mean in Equation (22) as

$$\langle \phi(t) \rangle = k_c^s(t)^\top \tilde{\mu}_c + \sum_l \sum_{t_i < t} \sum_{t_j < t_l} \tilde{K}^g(t - t_i, t_l - t_j) \tilde{\mu}_c^l \tag{28}$$

where we defined $\tilde{\mu_c} = K_c^{-1}\mu_c$ and used the fact that $k_c(t)$ can be written explicitly as

$$k_c^s(t) + \sum_{t_i<t}\sum_{t_j<t_l} \tilde{K}^g(t-t_i, t_l-t_j)$$

with $k_c^s(t)$ being the kernel K^s between data point t and all the inducing points.

From Equation (28) we can identify the posterior mean of s and g as

$$\langle s(t)\rangle = k_c^s(t)^\top \tilde{\mu}_c \tag{29}$$

$$\langle g(t)\rangle = \sum_l \sum_{t_j<t_l} \tilde{K}^g(t, t_l-t_j)\tilde{\mu}_c^l. \tag{30}$$

To identify the covariance of s and g we start with the posterior covariance of ϕ

$$\begin{aligned}\operatorname{cov}(\phi(t), \phi(t')) = \Sigma(t,t') &= \tilde{K}(t,t') - k_c(t)^\top B k_c(t') \\ B &= K_c^{-1} - K_c^{-1}\Sigma_c K_c^{-1}.\end{aligned} \tag{31}$$

Using again the explicit form of $k_c(t)$ we rewrite the equation above as

$$\Sigma(t,t') = K_t^s + \hat{K}_t^g - {K_{tc}^s}^\top B K_{tc}^s - 2{K_{tc}^s}^\top B \hat{K}_{tc}^g - \hat{K}_{tc}^{g\top} B \hat{K}_{tc}^g \tag{32}$$

where $K_t^{s/g}$ is the kernel matrix between the data points, $K_{tc}^{s/g}$ is the kernel between the data points and the set of inducing points, and we defined

$$\hat{K}_g(t,t') = \sum_{t_i<t}\sum_{t_j<t'} \tilde{K}_g(t-t_i, t'-t_j).$$

From the expression above we can identify the marginal covariances of s and g separately, as well as their joint covariance. To sample s and $\tilde{g}$ from their posterior distribution we would need the full expression of the covariance, including both the marginal and cross covariances. For analyzing the model's ability to recover s and $\tilde{g}$ we are interested in the marginal covariances, which can be expressed as

$$\begin{aligned}\operatorname{cov}(s(t), s(t')) &= K_t^s - {K_{tc}^s}^\top B K_{t'c}^s \\ \operatorname{cov}(\tilde{g}(t), \tilde{g}(t')) &= \tilde{K}_t^g - \sum_{l,m}\sum_{\substack{t_j<t_l \\ t_i<t_m}} \tilde{K}_g(t, t_l-t_j) B_{l,m} \tilde{K}_g(t', t_m-t_i).\end{aligned} \tag{33}$$

4. Related Work

Before presenting the results of the experiment for our model, we go through recent developments in the field. We start with a short overview of Bayesian inference for Cox and Hawkes processes and then focus on the three models to which we later compare our approach.

Bayesian approaches to Cox processes model the intensity with a Gaussian process prior, which is then passed through a link function to ensure its positivity. A common choice of the link function is the exponential or the quadratic functions [27,28]. Another choice, which is more relevant to our work, is the sigmoid link function, resulting in the *sigmoidal Gaussian Cox process*. Inference in this model was first done empirically in study [29], as well as with moment-based parameters estimators [30]. Markov chain Monte Carlo methods were also developed [31], as well as variational inference [19].

In our work, we use a Bayesian semi-parametric inference approach. Earlier work, which introduces Bayesian nonparametric approaches to point processes, includes, among others, Ishwaran and James [32], who define kernel mixtures of Gamma measures for the intensity, Wolpert and Ickstadt [33], who define inhomogeneous Gamma random fields, and Taddy and Kottas [34], where joint nonparametric mixtures are introduced.

As for the Hawkes process, first attempts to perform Bayesian inference relied on the definition in terms of a marked Poisson cluster process and identifying the branching structure of the self-excitation [16]. One model that uses this approach is the mutually regressive point process (MRPP) [20]. MRPP is designed to model neuronal spike trains. In this work, the classical self-excitatory Hawkes Process intensity function is augmented by a probability term. This term induces inhibition when it is close to zero. In a sense, this model includes two memory kernels—one excitatory only, which appears in the intensity function, and another which can also induce inhibition in the augmenting probability term. Different from the MRPP, in our work, we achieve such flexibility of the self-effects in a simpler fashion by assuming the GP prior on the self-effects. As mentioned before, this also allows for the type of effect to change over time, which does not appear in the work of Apostolopoulou et al. [20].

A highly flexible approach to estimating the intensity function of the Hawkes process relies on GP priors [35–37]. A recent adaptation of this approach is the model described in Zhou et al. [17]. Similar to the model described in our work, the authors avoid the limiting parameterization of the memory kernel by using GPs. Different from our work, Zhou et al. [17] remain in the linear Hawkes process regime and assume that the effects of past events are only excitatory, whereas our approach allows both excitatory and inhibitory effects.

The last variation we describe is a sigmoid nonlinear multivariate Hawkes process (SNMHP) [38]. In this work, Zhou et al. describe a multivariate nonlinear HP, where, similarly to our work, the chosen link function is a sigmoid; however, we chose a nonparametric approach to model the causal influence function with a weighted mixture of basis functions from a certain family. Similar to the MRPP model, SNMHP was designed to model neuronal activity. A common assumption in this field is that each neuron is affected only by a subset of the other neurons in the network. Zhou et al. incorporate this assumption directly to their model by including a sparsity-inducing prior over the weights. In terms of inference, Zhou et al. proposed an expectation maximization algorithm, whereas we propose a fully Bayesian approach.

5. Experiments

In this section, we demonstrate the performance of our model and inference algorithm in different fields. First, we establish that the variational inference algorithm described in Section 3 is reasonable, using synthetic data as ground truth. Next, we apply the model and the inference algorithm to real-world datasets in the field of neuroscience and crime prediction. In these examples, the data are time series of events. We feed these events to the model and perform inference to estimate the underlying intensity function. The inferred intensity function can be used in different ways. For one, we use it to estimate the performance of the model (by calculating the log-likelihood or other metrics) and compare it to competing models. We also use it to simulate data from the model, which helps us assess whether the model is a reasonable candidate to describe the data (see Section 5.2.2). In the case of multivariate data, we use the inferred intensity function to assess the interaction between the different components of the data (see Section 5.2.3).

All the algorithms and experiments for this work are implemented in Python and are available online. To parallelize the computation over the available computing resources, we used the JAX package [39]. In the Gibbs sampler, the sampling of the PG variables was done using the PyPólyaGamma package [40]. The code and data are publicly available. See the Data Availability Statement.

5.1. Synthetic Data

To assess the performance of the inference algorithms presented in Section 3, we learn the parameters of the data generated by the model, namely the intensity function and the intensity bound, and compare the learned parameters to the ground truth. To generate data, we start by sampling the memory GP and the background GP, based on

Equations (5) and (6). We generate events from the model using Poisson thinning [41]. First, we sample the number of candidates $J \sim Poisson(\lambda T)$, and sample candidate events $\{t_1, \ldots t_J\}$ uniformly. Next, we chronologically iterate through the candidates and accept them with probability $\frac{\Lambda(t_j|\{t_1,\ldots t_{j-1}\})}{\lambda}$.

The results for the synthetic data are included in Figure 1. The time window used was one second, and the dataset includes 91 events. In panel (a), the comparison between the ground truth predictive intensity and the one inferred by the learning algorithms demonstrates the accuracy of the inference methods. We compare the ground truth to the mean of the Gibbs samples, and the mean of the approximating distribution of the VI.

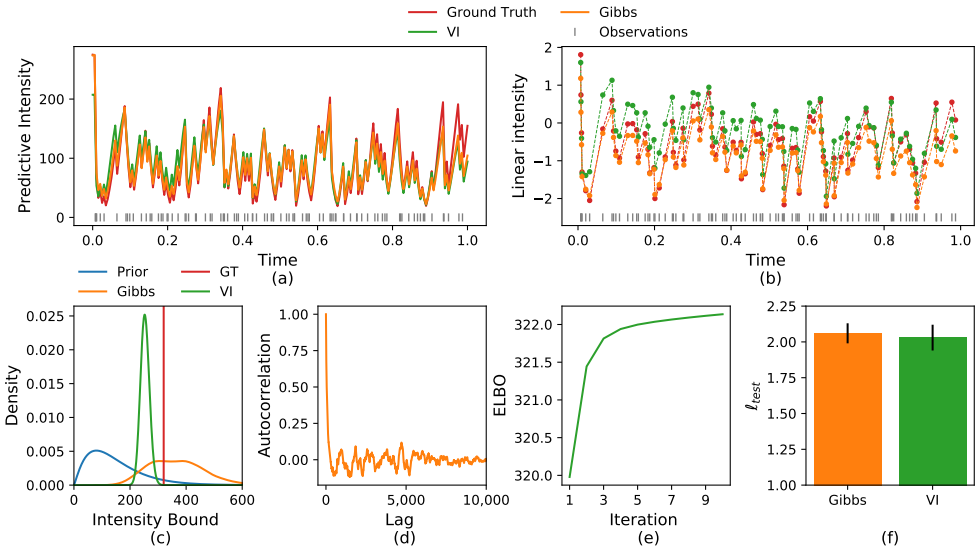

Figure 1. (**a**) Comparison of the ground truth predictive intensity and the one sampled from the VI and Gibbs inference. (**b**) Comparison of the ground truth linear intensity $\phi(\cdot)$ and the one learned by the VI and Gibbs sampler. (**c**) Comparison of the ground truth intensity bound and the one learned by the inference, and the prior distribution. (**d**) The autocorrelation of the intensity bound Gibbs samples. (**e**) The variational lower bound as a function of the algorithm iteration. (**f**) Comparison of the test log-likelihood of the Gibbs sampler and the VI.

Panel (b) compares the ground truth value of the linear intensity and the one inferred by the two learning algorithms. In this example, the linear intensity is negative in some of the time points. Unlike for the predictive density, the Gibbs sampler is more accurate than the VI. Panel (c) presents the inference results for the intensity bound. As expected, the approximated distribution by the VI is much more narrow than the distribution of the Gibbs samples.

Panels (d) and (e) show the autocorrelation of the intensity bound and the ELBO through the Gibbs samples and VI iterations. In this example the convergence of the ELBO is very fast, and the autocorrelation of the Gibbs sample vanishes only after a few thousand iterations. We use the test log-likelihood per data point, averaged over ten datasets, to quantify the performance of the two inference algorithms. The Gibbs sampler and the VI achieve very similar results.

In the experiment presented above, the data were generated from the model, where s and g were sampled from a GP. When dealing with real data, we cannot expect that the data follow the model exactly. To demonstrate that our model can fit data that were not sampled from it directly, we infer the intensity when the data is generated from a slightly different model. In this case we take $s(t) = \beta_1 \cos(\theta_1 t)$ and $g(t) = \beta_2 \cos(\theta_2 t)$. The results can be found in Figure 2. We perform the same analysis presented in Figure 1. Similarly, both the Gibbs sampler and the VI algorithm recover the underlying intensity very well. Both inference methods achieve comparable results in terms of log-likelihood averaged over 10 test datasets.

As the VI algorithm achieves similar results to the Gibbs sampler in a much faster computation time, in the next sections we present the results only for the VI algorithm.

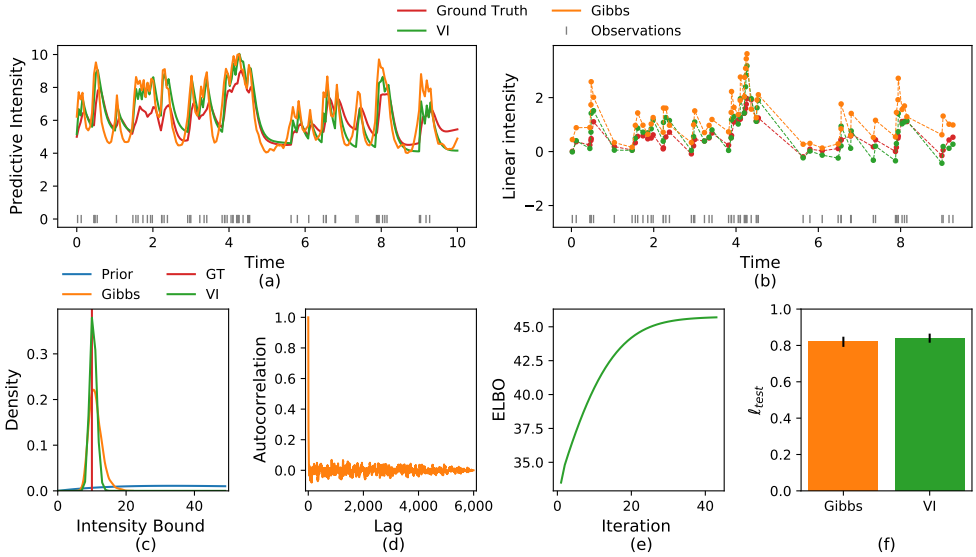

Figure 2. (**a**) Comparison of the ground truth predictive intensity and the one sampled from the VI and Gibbs inference. (**b**) Comparison of the ground truth linear intensity $\phi(\cdot)$ and the one learned by the VI and Gibbs sampler. (**c**) Comparison of the ground truth intensity bound and the one learned by the inference, and the prior distribution. (**d**) The autocorrelation of the intensity bound Gibbs samples. (**e**) The variational lower bound as a function of the algorithm iteration. (**f**) Comparison of the test log-likelihood of the Gibbs sampler and the VI.

In Section 3.5, we discussed the identifiability of the background rate function and the self-effects function from the inferred linear intensity. In Figure 3, we present the results of the inference of these functions from synthetic data. Both functions are recovered from the inference results of the linear intensity. In panel c, we can see that $\tilde{g}$ goes to zero for longer time differences, as expected from the integration of the exponential decay into the self-effects kernel.

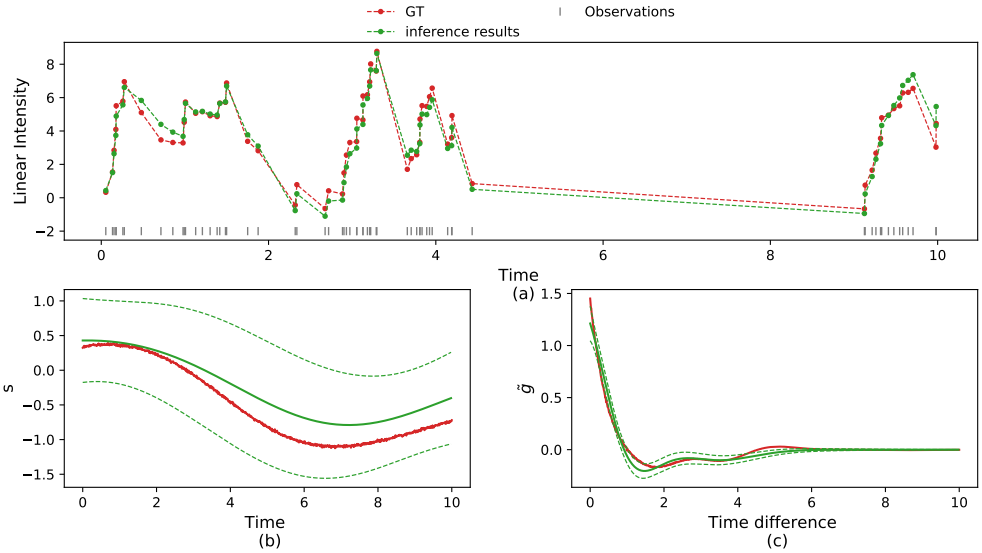

Figure 3. (**a**) Comparison of the ground truth linear intensity and the one inferred by the VI algorithm. (**b**) Comparison of the ground truth background rate function s and the one inferred by the VI algorithm. (**c**) Comparison of the ground truth self-effects function g and the one inferred by the VI algorithm.

It is important to notice that the stationarity conditions for nonlinear Hawkes processes are not sustained in our simulation procedure, as we ignore the past of the process, and time $t < 0$. There are no events prior to the beginning of the simulation that will have self-effects. In this case, we generate a transient version rather than the stationary version of the process.

When t is big enough, we would expect to attain a stationary version of the process. However, the full conditions of stationarity for our process are not well-defined, and they are out of the scope of the current work. Nonetheless, conditioned on a given sample from the GP, to attain stationarity one only needs to ensure that $\int_0^\infty g(t)dt < \infty$ [8], as the intensity is bounded by λ.

Notably, our inference procedure is independent of the stationary nature of the driving GP, which might itself not be stationary, depending on the kernels used to define it. This implies that for these conditions to hold, we must study the process in a per-kernel basis.

5.2. Real Data

5.2.1. Crime Report Data

Our model assumes both inhibitory and excitatory self-effects, but it should also be able to capture phenomena where only one of the two types of effects exist. To test this, we fit our model to crime report data, where it is assumed that past events have an excitatory effect on future events [42].

In criminology, it is known that crimes are clustered events. For example, burglars will repeatedly attack nearby targets in order to take advantage of known vulnerabilities, and gang-conflict shootings may incite or encourage retaliatory violence from the opposing gang in the local territory of the rivals. The nature of such retaliatory events, as well as the exploitation of resources by the burglars, are highly random and context-based phenomena, since this will depend on the criminals, location, or gang at hand. Hence, a highly flexible approach is required, and our methodology is able to express the inhomogeneity of each crime pattern through the unknown function g and s.

We use the same two datasets described in Zhou et al. [17], and follow their data processing procedure. Each dataset contains one type of crime, and so we use the univariate version of the model.

An example of the results of the inference process is presented in Figure 4. It includes the inferred linear intensity of the fitted model over the first 80 events in the Vancouver crime dataset.

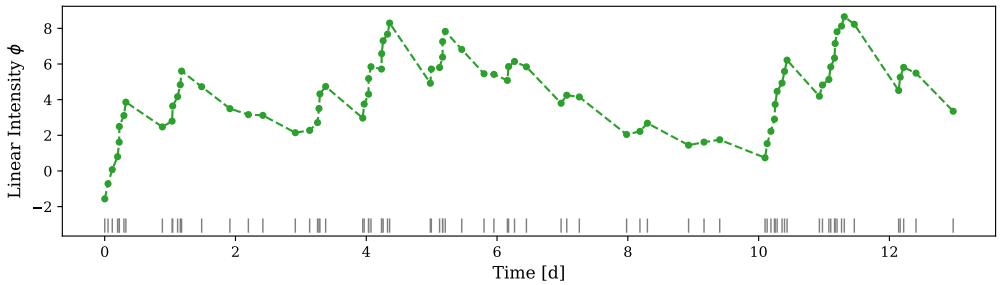

Figure 4. Inferred linear intensity ϕ for the first 80 events in the Vancouver crime dataset. The linear intensity is estimated only for the observed events and the dashed line between the dots is merely linear interpolation.

Table 1 compares the test log-likelihood of our NH–GPS model to the one reported in Zhou et al. [17]. The work of Zhou et al. [17] includes several inference methods and we compare our results to the results of their reported mean-field variational inference approach as it is the closest to our inference procedure. We perform the experiment five times and report the mean and variance of the test log-likelihood. As expected, our model performs similarly to the semi-parametric Hawkes process presented by Zhou et al. [17].

Table 1. Crime report data test log-likelihood.

Dataset	Zhou et al. (2020) [17]	NH–GPS
Vancouver	453.11 ± 8.94	453.8 ± 12.2
NYPD	-200.7 ± 3.32	-202.8 ± 7.54

5.2.2. Neuronal Activity Data

One of the motivating real-world phenomena behind our work is the spiking activity of neurons, where it is known that the process has both self-excitatory and self-inhibitory effects. Having a reliable model that captures the data accurately is useful to identify and analyze different physiological mechanisms that drive the neuronal activity.

As an example for our model's ability to capture neuronal activity we use the datasets that were first presented in Gerhard et al. [43] (Figure 2b,c). One dataset includes ten recordings from a single neuron in a monkey cortex, with a duration of one second each, and the other includes ten recordings from a single neuron in a human cortex for a duration of ten seconds each. In this work, point process generalized linear models were used to fit the data, and the data generated from it yielded unrealistic spiking patterns. This hints the need for a nonlinear model to describe the data.

The dataset described above were further analyzed in Apostolopoulou et al. [20] (Figure 5), where the mutually regressive point process (MR–PP) is introduced. The fitted MR–PP model produced realistic spike trains and we use it as a comparison for our model.

We fit the model to the multitrial single neuron datasets and infer the intensity of the assumed underlying point process. Figure 5 presents the results for one trial from each dataset. In both cases, the inferred linear intensity obtains both positive and negative values, which implies both excitatory and inhibitory spiking patterns.

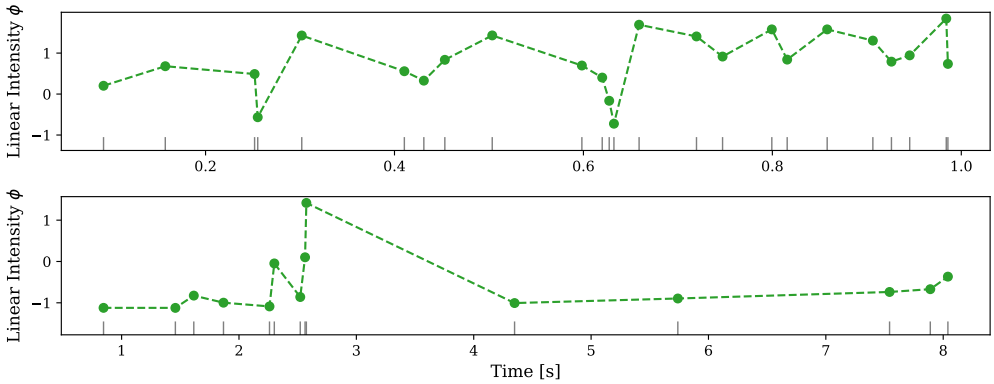

Figure 5. Inferred intensity for single neuron data. Upper panel—results for trial number 9 from the dataset recorded from monkey cortex. Lower panel—results for trial number 7 from the dataset recorded from human cortex. The linear intensity is estimated only for the observed events and the dashed line between the dots is merely linear interpolation.

In Figure 6, we assess the ability of the model to capture the data. The left column includes the raster plot of the real data and the middle plot the raster plot generated from the fitted model. Similarly to the real data, the generated data displays both excitation, in the form of clustered events, and inhibition.

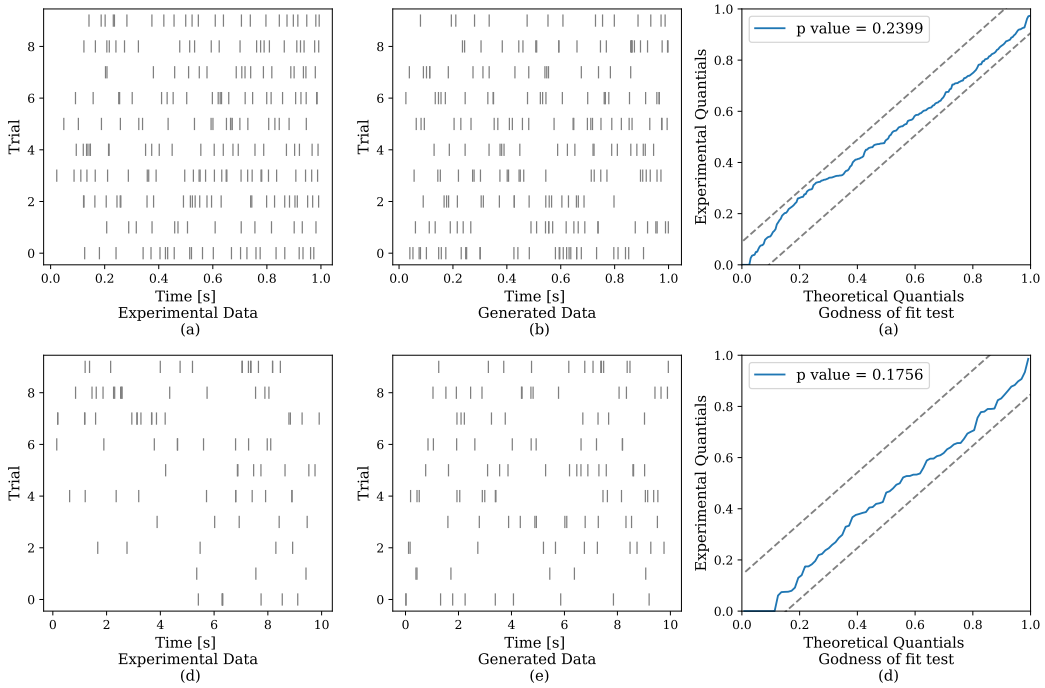

Figure 6. Upper row—single neuron recordings from monkey cortex. **Lower row**—single neuron recordings from human cortex. **Left column**—recorded data. **Middle column**—data generated from the learned models. **Right column**—results of the Kolmogorov–Smirnov test. The NH–GPS generates data that resembles the real data, and passes the goodness of fit test.

To quantify how suitable the model is to the data, we apply the random time change theorem [18] to the inferred intensity and the experimental data. The theorem states that realizations from a general point process can be transformed to realizations from a homogeneous Poisson process with a unit rate. Similarly to the work of Apostolopoulou et al. [20], we further transform the exponential realizations to those from a uniform distribution, following Brown et al. [44]. We then use the Kolmogorov–Smirnov test to compare the quantiles of the distribution of the transformed realizations to the quantiles of the uniform distribution. The results of this test are displayed in Figure 6 in the right column. The comparison relies on 95% confidence bounds, which are indicated by the dashed lines. The model passes the goodness-of-fit test (p-value > 0.05), and the p-value is higher than the one achieved by the MR–PP. We compare the reported p-value achieved by the MR–PP model for the two datasets in Table 2. For both datasets, our model achieves higher a p-value than the MR–PP.

Table 2. p-Value of the KS Test on neuronal activity data.

Dataset	MR–PP	NH–GPS
Monkey Cortex	0.103	0.23
Human Cortex	0.096	0.175

5.2.3. Multi-Neurons Data

Last, we demonstrate the performance of the multivariate version of our model. We use the data presented in Zhou et al. [38]. This dataset includes spike trains simultaneously recorded from 25 neurons in the primary visual cortex of an anesthetized cat.

One application of our model for the use-case of multi-neuron recording, is the analysis of the interactions between the different neurons. As presented in Section 3.5, after fitting the model to the data we can recover the function g, which describes the effects of past events on future events. In the case of a multivariate model, this function is defined for

every pair of dimensions, and describes the interaction between them. An example of this can be found in Figure 7. On the left is the recorded data, and on the right is the interactions between two neurons from the dataset.

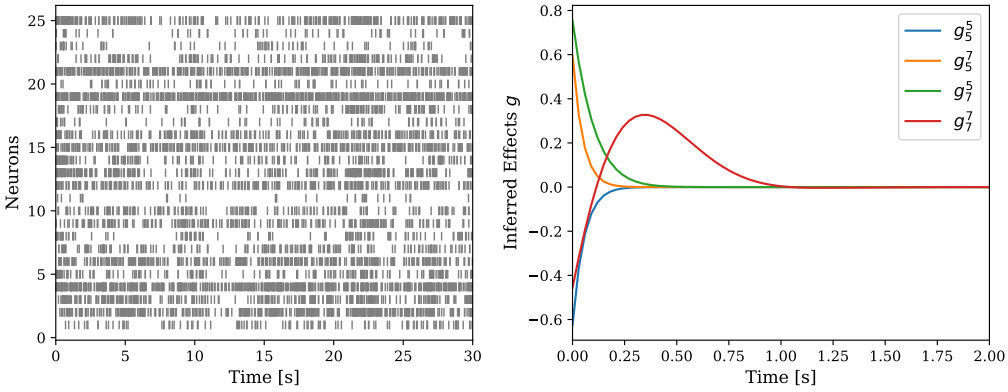

Figure 7. (**Left**)—the recorded activity of 25 neurons. (**Right**)—example of the recovery of the influence function g between neuron number 5 and neuron number 7. Both excitatory and inhibitory influence is observed.

Next, we compare the performance of our model to the model described in Zhou et al. [38], by looking at the test log-likelihood.. We follow the same train-test split as Zhou et al. [38], where the first 30 seconds of the recording are used as the training set and the last 30 seconds are used as the test set.

The test log-likelihood can be found in Table 3. We compare our model the the one from Apostolopoulou et al. [20] (MR–PP) and the model presented in Zhou et al. [38] (SNMHP). Our model achieves a higher test log-likelihood score than the competing models. This demonstrates the power in the flexibility of our model and showcases its applicability also to multivariate data.

Table 3. Test log-likelihood for the multi-neurons datasets for different models.

SNMHP	MR–PP	NH–GPS
-6.13×10^3	-5.9×10^3	-5.29×10^3

6. Discussion

In this work, we present the nonlinear Hawkes model with Gaussian process self-effects (NH–GPS) and a Bayesian variational inference scheme to infer its parameters. We motivated the development of the new model with the need for a flexible model that can capture both exciting and inhibiting interactions between events, while maintaining the ability to learn also when data are scarce. The model includes both a univariate and a multivariate version.

Due to the structure of the model, we derive an inference algorithm without the branching structure that is commonly used for Bayesian inference in Hawkes processes. We propose a mean-field variational inference algorithm, which relies on a data augmenting scheme. We show that the results of the variational inference are comparable with those of a Gibbs sampler.

We demonstrate the performance of our model in four different real-world applications. Due to the flexibility of our model, it achieves good results on data where events have only excitatory effects and on data where events have both excitatory and inhibitory effects.

Different from recent work on nonlinear Hawkes process inference [38], our work presents a two-fold advantage. The introduction of the Gaussian Process allows for a wider range of applications as well as the Bayesian perspective, which lends itself to uncertainty quantification, model selection, and regularization. All thought that nonlinear Hawkes processes were ubiquitous in applications for finance, social dynamics, and neuroscience.

Previous works are tailored with specific applications in mind, whereas our methodology is of a general nature.

More importantly, the introduction of priors, as well as the GP, allows the practitioner to easily introduce expert knowledge, as one is able to modify the behavior of the self-effects by introducing different kernels.

In this work, we did not include results regarding the prediction abilities of the model, as we leave it for future work. We do believe it is relatively simple to acquire in our model and we briefly describe how to do so. The inference over the aggregated history function ϕ, which includes both the sum of the Gaussian process associated with the exogenous arrival rate and the endogenous event rate, allows us to generate estimates for predictions. Once we sample the value of ϕ conditioned on the previous observed events, we can estimate the mean arrival time of the next event by estimating the integral $\mathbb{E}(t_{n+1}) = \int_0^\infty tP(t|\mathcal{H})$ via numerical methods, such as Monte Carlo integration.

We would also like to expand our model to describe its spatio-temporal processes. Our model can be directly extended to capture events in time and space, as the memory kernel is reduced to a kernel function of a GP, which can be applied also to multi-dimensional data.

Author Contributions: Conceptualization, N.M.-S. and M.O.; methodology, N.M.-S. and C.O.; software, N.M.-S.; validation, N.M.-S. and C.O., M.O.; formal analysis, N.M.-S.; investigation, N.M.-S.; writing—original draft preparation, N.M.-S. and C.O.; writing—review and editing, M.O.; visualization, N.M.-S. All authors have read and agreed to the published version of the manuscript.

Funding: The research of N.M.-S. and M.O. was partially funded by the Deutsche Forschungsgemeinschaft (DFG)—Project-ID 318763901–SFB1294. The research of C.O. was funded by the BIFOLD Berlin Institute for the Foundations of Learning and Data (ref. 01IS18025A and ref 01IS18037A).

Data Availability Statement: All the code and data included in this paper can be found at https://github.com/noashin/NHGPS (accessed on 28 February 2022).

Acknowledgments: The authors would like to thank Christian Donner for topical discussions which contributed to the research presented in this paper.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A. Gibbs Sampler

We use a blocked Gibbs sampler, which groups two or more variables and samples them at once. Thus, we need to identify the conditional posterior distribution of all the relevant groups.

Appendix A.1. Conditional Distribution of the Upper Intensity Bound

The conditional distribution of the upper intensity bound is

$$p\left(\lambda|\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \{t_n\}\right) \propto e^{-\lambda T}\lambda^{N+M}p(\lambda)$$

which we identify as a Gamma distribution

$$\begin{aligned} &p\left(\lambda|\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \{t_n\}\right) \propto Gamma(\alpha, \beta) \\ &\alpha = \alpha_0 + N + M \\ &\beta = \beta_0 + T. \end{aligned} \tag{A1}$$

Appendix A.2. Conditional Distribution of the Linear Intensity Function

The conditional distribution of the linear intensity function in the observed and augmenting events is

$$p(\phi_{N+M}|\{\hat{t}_m, \hat{w}_m\}, \{w_n\}, \phi, \{t_n\}, \lambda) \propto \exp\left(\sum_{n=1}^{N} f(w_n, \phi_n) + \sum_{m=1}^{M} f(\hat{w}_m, -\phi_m)\right) p(\phi_{N+M}),$$

where we use the shortened notation ϕ_n instead of $\phi(t_n)$ and ϕ_{N+M} instead of $\{\{\phi(t_n)\}_{n=1}^{N}, \{\phi(\hat{t}_m)\}_{m=1}^{M}\}$. Given Equation (13) in the main text, the likelihood term in the posterior is a GP with mean

$$\mu = \left(\frac{1}{2w_1}, \ldots, \frac{1}{2w_N}, -\frac{1}{2\hat{w}_1}, -\frac{1}{2\hat{w}_M}\right)^\top$$

and diagonal covariance matrix

$$\Sigma^{-1} = Diag(w_1, \ldots, w_N, \hat{w}_1, \hat{w}_M).$$

Given the GP prior over ϕ the conditional posterior is also a GP

$$\begin{aligned} \phi_{N+M} &\sim GP(\mu_{M+N}, \Sigma_{M+N}) \\ \mu_{M+N} &= \Sigma_{N+M}\Sigma^{-1}\mu \\ \Sigma_{N+M}^{-1} &= \Sigma^{-1} + \tilde{K}^{-1}. \end{aligned} \tag{A2}$$

Appendix A.3. Conditional Distribution of the PG Variables

The conditional posterior distribution of the augmenting PG variables is

$$\begin{aligned} &p(\{w_n\}, \{\hat{w}_m\}) \\ &\propto \prod_{n=1}^{N} \exp\left(-\frac{\phi_n^2}{2} w_n\right) \\ &\times PG(w_n; 1, 0) \prod_{m=1}^{M} \exp\left(-\frac{\phi_m^2}{2} \hat{w}_m\right) PG(w_m; 1, 0). \end{aligned}$$

Using the definition of the tilted PG distribution [21], the posterior distribution for these parameters is

$$\begin{aligned} w_n &\propto PG(1, \phi_n) \\ \hat{w}_m &\propto PG(1, \phi_m). \end{aligned}$$

Appendix A.4. Conditional Distribution of the Augmenting Events

The conditional posterior of the augmenting events is proportional to

$$\begin{aligned} p\left(\{\hat{t}_m\}_{m=1}^{M} | \{t_n\}_{n=1}^{N}, \{\hat{w}_m\}_{m=1}^{M}, \{w_n\}_{n=1}^{N}, \phi, \lambda\right) \\ \propto \prod_{m=1}^{M} \lambda e^{f(\hat{w}_m, -\phi_m)} PG(\hat{w}_m; 1, 0). \end{aligned}$$

We recognize the conditional posterior with the unnormalized density of a marked inhomogeneous Poisson process with intensity

$$\Pi_0(\hat{w}, \hat{t}) = \lambda e^{f(\hat{w}_m, -\phi_m)} PG(\hat{w}_m; 1, 0). \tag{A3}$$

The underlying density of the inhomogeneous Poisson process in the realizations space $\mathcal{T}$ is given by

$$\int_{\mathcal{W}} \Pi_0(\hat{w}, \hat{t}) d\hat{w} = \lambda \sigma(-\phi_m).$$

To sample the augmenting realizations, we use the thinning algorithm [45]. First, we sample the expected number of events

$$J \sim Poisson(\lambda T).$$

Next, J candidates are sampled uniformly over the realizations space $\mathcal{T}$. To evaluate the intensity at the candidate points we need to evaluate the linear intensity ϕ in these points, given its values in the real events and the previously sampled augmenting events. This can be done using results from GP regression [46]

$$\begin{aligned}
\phi_{J|N+M} &\sim GP(\mu, \tilde{K}) \\
\mu &= \tilde{K}_{J,N+M} \tilde{K}_{N+M}^{-1} \phi_{N+M} \\
\tilde{K} &= \tilde{K}_J - \tilde{K}_{J,N+M} \tilde{K}_{N+M}^{-1} \tilde{K}_{J,N+M}^{\top}.
\end{aligned}$$

Once we have ϕ_J, we perform thinning—for each candidate $\hat{t}_j$ we generate a random number r_j between 0 and 1. If $r_j < \sigma(-\phi_j)$ we accept the candidate $\hat{t}_j$, otherwise we discard it.

Algorithm 2: NH–GPS Gibbs Sampler

Input: Observed events $\{t_n\}_{n=1}^{N}$, hyperparameters $\{\sigma_g, \alpha, a_s, \sigma_s, \alpha_0, \beta_0\}$
Output: R samples of $\{\hat{t}_m\}_{m=1}^{M_r}$, λ, ϕ_{N+M_r}

Initialize—$M, \phi_{N+M}, \{\hat{t}_m\}_{m=1}^{M_r}$ randomly. **for** $r \leftarrow 0$ **to** R **do**
  Sample $w \sim PG(1, \phi_{N+M})$
  Sample $\lambda \sim \text{Gamma}(\alpha, \beta)$ as in Equation (A1)
  Sample $\phi_{N+M} \sim GP(\mu_{N+M}, \Sigma_{N+M})$ as in Equation (A2)
  Sample $\{\hat{t}\}_{m=1}^{M_r}$ as in Section A.4
end

Appendix B. Hyperparameters Learning for the Gibbs Sampler

The augmented model is not conditionally conjugated with respect to the kernel hyperparameters. This is usually solved by using MCMC within the Gibbs sampler approach [47,48]. This method applies rejection sampling, such as Metropolis–Hastings (MH) [49] and Hamiltonian Monte Carlo (HMC) [50], to sample the hyperparameters, and relies heavily on design choice. A wrong choice of the proposal distribution (for MH) or the mass matrix (for HMC) may result in a very slow convergence, or prevent the sampler from converging at all.

We choose the less traditional approach of taking a gradient step within the Gibbs sampler. This is implemented in the following way—after sampling all the model parameters from the conditional posterior distributions described above, we derive the negative model log posterior with respect to the hyperparameters and take a step in the direction of the negative gradient.

This approach can be developed further in the spirit of stochastic gradient descent (SGD), meaning, rather than updating the hyperparameters after each iteration of the Gibbs sampler, we perform several steps of sampling, take the gradient of the averaged posterior, and update the hyperparameters following the averaged gradient.

We include bellow the derivatives with respect for the hyperparameters of the model $\theta = \sigma_g, \alpha, a_g, \sigma_s$.

To learn the hyperparameters, we derive the posterior of the model, which appears in Equation (17) in the main text. First, we notice that all of the hyperparameters appear in the prior over the linear intensity

$$\log p(\phi) \propto -\frac{1}{2}\log\det(\tilde{K}) - \frac{1}{2}\phi^\top \tilde{K}^{-1}\phi$$

and all of the hyperparameters appear in the prior kernel.

We next derive an entry in the kernel with respect to the different hyperparameters.

$$\begin{aligned}
\frac{\partial \tilde{K_{l,k}}}{\partial a_s} &= \exp\left(-\frac{\| t_l - t_k \|^2}{\sigma_s^2}\right) \\
\frac{\partial \tilde{K_{l,k}}}{\partial \sigma_s} &= a_s \exp\left(-\frac{\| t_l - t_k \|^2}{\sigma_s^2}\right)\frac{\| t_l - t_k \|^2}{\sigma_s^3} \\
\frac{\partial \tilde{K_{l,k}}}{\partial \alpha} &= \sum_{t_i<t_l}\sum_{t_j<t_k} K^g_{t_i-t_l,t_j-t_k} \\
&\quad \times \exp\big(-\alpha\big(t_i - t_l + t_j - t_k\big)\big)\big(t_l - t_i + t_k - t_j\big) \\
\frac{\partial \tilde{K_{l,k}}}{\partial \sigma_g} &= \sum_{t_i<t_l}\sum_{t_j<t_k} K^g_{t_i-t_l,t_j-t_k} \\
&\quad \times \exp\big(-\alpha\big(t_i - t_l + t_j - t_k\big)\big)\frac{\| \left(t_l - t_i\right) - \left(t_k - t_j\right) \|^2}{\sigma_g^3}.
\end{aligned}$$

We can plug these results to the chain rule, and we get

$$\nabla \log p(\phi) = -\frac{1}{2}\mathrm{trace}\left(\tilde{K}^{-1}\nabla\tilde{K}\right) + \frac{1}{2}\phi^\top \tilde{K}^{-1}\nabla\tilde{K}\tilde{K}^{-1}\phi.$$

References

1. Ogata, Y. Statistical models for earthquake occurrences and residual analysis for point processes. *J. Am. Stat. Assoc.* **1988**, *83*, 9–27. [CrossRef]
2. Zhao, Q.; Erdogdu, M.A.; He, H.Y.; Rajaraman, A.; Leskovec, J. SEISMIC: A Self-Exciting Point Process Model for Predicting Tweet Popularity. In Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; Association for Computing Machinery (KDD '15), New York, NY, USA, 10–13 August 2015; pp. 1513–1522.
3. Dayan, P.; Abbott, L.F. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*; MIT Press: Cambridge, MA, USA, 2001; Chapter 7.5; pp. 265–273.
4. Cox, D.R. Some statistical methods connected with series of events. *J. R. Stat. Soc. Ser. B* **1955**, *17*, 129–157. [CrossRef]
5. Hawkes, A.G.; Oakes, D. A cluster process representation of a self-exciting process. *J. Appl. Probab.* **1974**, *11*, 493–503. [CrossRef]
6. Maffei, A.; Nelson, S.B.; Turrigiano, G.G. Selective reconfiguration of layer 4 visual cortical circuitry by visual deprivation. *Nat. Neurosci.* **2004**, *7*, 1353–1359. [CrossRef]
7. Smith, T.C.; Jahr, C.E. Self-inhibition of olfactory bulb neurons. *Nat. Neurosci.* **2002**, *5*, 760–766. [CrossRef]
8. Brémaud, P.; Massoulié, L. Stability of nonlinear Hawkes processes. *Ann. Probab.* **1996**, 1563–1588. [CrossRef]
9. Zhu, L. Central limit theorem for nonlinear Hawkes processes. *J. Appl. Probab.* **2013**, *50*, 760–771. [CrossRef]
10. Truccolo, W. From point process observations to collective neural dynamics: Nonlinear Hawkes process GLMs, low-dimensional dynamics and coarse graining. *J. Physiol.* **2016**, *110*, 336–347. [CrossRef]
11. Sulem, D.; Rivoirard, V.; Rousseau, J. Bayesian estimation of nonlinear Hawkes process. *arXiv* **2021**, arXiv:2103.17164.
12. Jia, J.; Benson, A.R. Neural jump stochastic differential equations. *arXiv* **2019**, arXiv:1905.10403.
13. Xiao, S.; Farajtabar, M.; Ye, X.; Yan, J.; Song, L.; Zha, H. Wasserstein Learning of Deep Generative Point Process Models. In *Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Long Beach, CA, USA, 2017; Volume 30.
14. Kingman, J.F.C. *Poisson Processes*; Oxford Studies in Probability Volume 3; The Clarendon Press Oxford University Press: New York, NY, USA, 1993.
15. Daley, D.J.; Vere-Jones, D. *An Introduction to the Theory of Point Processes: Volume II: General Theory and Structure*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2007.
16. Rasmussen, J.G. Bayesian inference for Hawkes processes. *Methodol. Comput. Appl. Probab.* **2013**, *15*, 623–642. [CrossRef]

17. Zhou, F.; Li, Z.; Fan, X.; Wang, Y.; Sowmya, A.; Chen, F. Efficient Inference for Nonparametric Hawkes Processes Using Auxiliary Latent Variables. *J. Mach. Learn. Res.* **2020**, *21*, 1–31.
18. Daley, D.J.; Vere-Jones, D. *An Introduction to the Theory of Point Processes: Volume I*, 2nd ed.; Probability and Its Applications; Springer: New York, NY, USA, 2003.
19. Donner, C.; Opper, M. Efficient Bayesian inference of sigmoidal Gaussian Cox processes. *J. Mach. Learn. Res.* **2018**, *19*, 2710–2743.
20. Apostolopoulou, I.; Linderman, S.; Miller, K.; Dubrawski, A. Mutually Regressive Point Processes. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Vancouver, BC, Canada, 2019; Volume 32.
21. Polson, N.G.; Scott, J.G.; Windle, J. Bayesian inference for logistic models using Pólya–Gamma latent variables. *J. Am. Stat. Assoc.* **2013**, *108*, 1339–1349. [CrossRef]
22. Donner, C.; Opper, M. Efficient Bayesian Inference for a Gaussian Process Density Model. In Proceedings of the Conference on Uncertainty in Artificial Intelligence, Monterey, CA, USA, 6–10 August 2018; Globerson, A., Silva, R., Eds.; PMLR, AUAI Press: Monterey, CA, USA, 2018.
23. Jordan, M.I.; Ghahramani, Z.; Jaakkola, T.S.; Saul, L.K. An introduction to variational methods for graphical models. *Mach. Learn.* **1999**, *37*, 183–233. [CrossRef]
24. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006.
25. Csató, L.; Opper, M.; Winther, O. TAP Gibbs Free Energy, Belief Propagation and Sparsity. In Proceedings of the Neural Information Processing Systems: Natural and Synthetic (NIPS), Vancouver, BC, Canada, 3–8 December 2001; pp. 657–663.
26. Titsias, M.K. Variational Learning of Inducing Variables in Sparse Gaussian Processes. In *JMLR Proceedings, Proceedings of the International Conference on Artificial Intelligence and Statistics, Clearwater Beach, CA, USA, 16–18 April 2009*; Dyk, D.A.V., Welling, M., Eds.; PMLR: Birmingham, UK, 2009; Volume 5, pp. 567–574.
27. Hensman, J.; Matthews, A.G.d.G.; Filippone, M.; Ghahramani, Z. MCMC for variationally sparse Gaussian processes. *arXiv* **2015**, arXiv:1506.04000.
28. Lloyd, C.M.; Gunter, T.; Osborne, M.A.; Roberts, S.J. Variational Inference for Gaussian Process Modulated Poisson Processes. In *JMLR Workshop and Conference Proceedings, Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015*; Bach, F.R., Blei, D.M., Eds.; PMLR: Birmingham, UK, 2015; Volume 37, pp. 1814–1822.
29. Møller, J.; Syversveen, A.R.; Waagepetersen, R.P. Log gaussian cox processes. *Scand. J. Stat.* **1998**, *25*, 451–482. [CrossRef]
30. Brix, A.; Diggle, P.J. Spatiotemporal prediction for log-Gaussian Cox processes. *J. R. Stat. Soc. Ser. B* **2001**, *63*, 823–841. [CrossRef]
31. Adams, R.P.; Murray, I.; MacKay, D.J.C. Tractable nonparametric Bayesian inference in Poisson processes with Gaussian process intensities. In *ACM International Conference Proceeding Series, Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009*; Danyluk, A.P., Bottou, L., Littman, M.L., Eds.; ACM: New York, NY, USA, 2009; Volume 382, pp. 9–16.
32. Ishwaran, H.; James, L.F. Computational methods for multiplicative intensity models using weighted gamma processes: Proportional hazards, marked point processes, and panel count data. *J. Am. Stat. Assoc.* **2004**, *99*, 175–190. [CrossRef]
33. Wolpert, R.L.; Ickstadt, K. Poisson/gamma random field models for spatial statistics. *Biometrika* **1998**, *85*, 251–267. [CrossRef]
34. Taddy, M.A.; Kottas, A. Mixture modeling for marked Poisson processes. *Bayesian Anal.* **2012**, *7*, 335–362. [CrossRef]
35. Zhang, R.; Walder, C.; Rizoiu, M.A.; Xie, L. Efficient non-parametric Bayesian Hawkes processes. *arXiv* **2018**, arXiv:1810.03730.
36. Zhou, F.; Li, Z.; Fan, X.; Wang, Y.; Sowmya, A.; Chen, F. Efficient EM-Variational Inference for Hawkes Process. *arXiv* **2019**, arXiv:1905.12251.
37. Zhang, R.; Walder, C.J.; Rizoiu, M.A. Variational Inference for Sparse Gaussian Process Modulated Hawkes Process. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; AAAI Press: Palo Alto, CA, USA, 2020; pp. 6803–6810.
38. Zhou, F.; Zhang, Y.; Zhu, J. Efficient Inference of Flexible Interaction in Spiking-neuron Networks. In Proceedings of the 9th International Conference on Learning Representations (ICLR 2021), Vienna, Austria, 3–7 May 2021.
39. Bradbury, J.; Frostig, R.; Hawkins, P.; Johnson, M.J.; Leary, C.; Maclaurin, D.; Necula, G.; Paszke, A.; VanderPlas, J.; Wanderman-Milne, S.; et al. JAX: Composable Transformations of Python+NumPy Programs. 2018. Available online: https://github.com/google/jax (accessed on 28 January 2022).
40. Linderman, S. PyPólyaGamma. GitHub. 2017. Available online: https://github.com/slinderman/pypolyagamma (accessed on 28 January 2022).
41. Lewis, P.W.; Shedler, G.S. Simulation of nonhomogeneous Poisson processes by thinning. *Nav. Res. Logist. Q.* **1979**, *26*, 403–413. [CrossRef]
42. Mohler, G.O.; Short, M.B.; Brantingham, P.J.; Schoenberg, F.P.; Tita, G.E. Self-exciting point process modeling of crime. *J. Am. Stat. Assoc.* **2011**, *106*, 100–108. [CrossRef]
43. Gerhard, F.; Deger, M.; Truccolo, W. On the stability and dynamics of stochastic spiking neuron models: Nonlinear Hawkes process and point process GLMs. *PLoS Comput. Biol.* **2017**, *13*, 1–31. [CrossRef]
44. Brown, E.N.; Barbieri, R.; Ventura, V.; Kass, R.E.; Frank, L.M. The time-rescaling theorem and its application to neural spike train data analysis. *Neural Comput.* **2002**, *14*, 325–346. [CrossRef]
45. Ogata, Y. On Lewis' simulation method for point processes. *IEEE Trans. Inf. Theory* **1981**, *27*, 23–31. [CrossRef]

46. Rasmussen, C.E.; Williams, C.K.I. Adaptive computation and machine learning. In *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.
47. Gilks, W.R.; Wild, P. Adaptive Rejection Sampling for Gibbs Sampling. *J. R. Stat. Soc. Ser. C* **1992**, *41*, 337–348. [CrossRef]
48. Martino, L.; Yang, H.; Luengo, D.; Kanniainen, J.; Corander, J. A fast universal self-tuned sampler within Gibbs sampling. *Digit. Signal Process.* **2015**, *47*, 68–83. [CrossRef]
49. Hastings, W.K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **1970**, *57*, 97–109. [CrossRef]
50. Duane, S.; Kennedy, A.D.; Pendleton, B.J.; Roweth, D. Hybrid Monte Carlo. *Phys. Lett. B* **1987**, *195*, 216–222. [CrossRef]