Exact Bayesian Gaussian Cox Processes Using Random Integrals*

Bingjing Tang† and Julia Palacios‡

Abstract. The Gaussian Cox process is a popular model for point process data, in which the intensity function is a transformation of a Gaussian process. Posterior inference of this intensity function involves an intractable integral (i.e., the cumulative intensity function) in the likelihood resulting in doubly intractable posterior distribution. Here, we propose a nonparametric Bayesian approach for estimating the intensity function of an inhomogeneous Poisson process without reliance on large data augmentation or approximations of the likelihood function. We propose to jointly model the intensity and the cumulative intensity function, allowing us to directly bypass the need of approximating the cumulative intensity function in the likelihood. In this sense, our model is exact. We propose an MCMC sampler for posterior inference, test its performance on simulated data, and show that our method outperforms other exact and approximate methods. Finally, we demonstrate the utility of our method in three real-world scenarios including temporal and spatial event data, as well as aggregated time count data collected at multiple resolutions.

Key words. doubly intractable, Gaussian Cox processes, random integral, aggregated count data

MSC codes. 62G05, 62M20, 62M30

1. Introduction.

Inhomogeneous Poisson process plays a fundamental role in modeling both temporal and spatial data with a variety of applications in many scientific fields [22]. Over the past decades, several approaches for inference have been proposed, including kernel smoothing methods [40, 14], reproducing kernel Hilbert space (RKHS)–based methods [7, 19], spline-based approaches [33, 10], and Bayesian frameworks [15, 2]. The classical nonparametric approach is the kernel smoothing estimator [14], which is closely related to kernel density estimation and offers computational simplicity and efficiency. In this work, our focus is on Bayesian inference of Gaussian Cox processes in which the intensity function is, a priori, modeled as a transformed Gaussian process.

Cox processes [11, 20], provide useful tools for modeling point process phenomena in a variety of fields of science and engineering including biology [26], astronomy [25], and many others [9]. It has been proved that many other classes of point processes such as the Gamma renewal process and the Weibull process are particular types of Cox processes [54]. Cox processes are also known as doubly stochastic Poisson processes, arising from inhomogeneous Poisson processes with a random intensity measure, which is in turn a realization from a second random process. In particular, in the context of Bayesian inference of the intensity function of an inhomogeneous process in which the intensity is, a priori, modeled as a transformed Gaussian process, the model is called Gaussian Cox process.

An important class of Gaussian Cox processes is the log Gaussian Cox processes [32], in which the log intensity function is a Gaussian process ensuring positivity of the intensity function. When covariates are available, the intensity of this model $\lambda(x)$ is formulated as $\lambda(x) = \exp\{X(t)\boldsymbol{\beta} + g(s)\}$, a transformation of a linear combination of covariates $X(t)\boldsymbol{\beta}$ and Gaussian process $g(s)$. In this work, we will consider the no-covariates case. This transformed Gaussian process is a convenient flexible nonparametric prior model of the intensity function without restricting it to a particular functional form. Though the class of log Gaussian Cox processes possesses abundant appealing properties and it is widely used as a model for temporal and spatio-temporal point process data [15], the corresponding posterior inference of the intensity function is doubly intractable due to the intractability of the cumulative intensity function (the integral of the intensity function over data

*Submitted to the editors 11/21/2024.

Funding: This work was supported in part by the NSF Career award 2143242 and NIH R35 GM14833801.

†Department of Statistics, Stanford University (bingjing@stanford.edu).

‡Department of Statistics, Stanford University (juliapr@stanford.edu).

domain) in the likelihood (Equation (2.1)) [31, 37]. Therefore, inference methods either involve approximations in the likelihood or rely on computationally expensive data augmentation.

A classical approach assumes the intensity function is a finite-dimensional piecewise constant log-Gaussian function [32, 41, 43, 49, 51]. Bayesian inference of the intensity function is then performed by MCMC [29, 32, 41, 47, 48, 51], variational Bayes [51, 52], or integrated nested Laplace approximation (INLA) [43, 49]. While Variational Bayes is often the method of choice in Machine Learning, these methods have shown to be less computationally efficient than INLA [51, 52]. Among the three computational approaches, MCMC brings the most accuracy while INLA usually leads to fastest computation [49, 51]. An important aspect in the discretization methods is that the choice of change points (grids) controls the trade-off between numerical accuracy and computational efficiency [44].

Apart from those discretization methods, nonparametric Bayesian approaches have been proposed for exact MCMC or variational inference for Cox processes [2, 23, 24, 44]. Here 'exact' means all terms in the likelihood can be computed without approximations. [2] eliminates the need for approximating the cumulative intensity function by estimating an augmented posterior distribution of the intensity function and latent points via MCMC. Specifically, stimulated by the thinning algorithm of [27], a point-process variant of rejection sampling, [2] introduced latent (thinned) points in such a way that the joint distribution of observed and latent points follows a homogeneous Poisson process. Therefore the joint posterior inference of the intensity function and latent points could be easily implemented via standard MCMC samplers, e.g., a block Gibbs sampler. Since this approach does not involve any approximation error in the likelihood, its performance is more accurate than competing discretization methods. However, it has several limitations. First, its complexity is cubic in the number of all data points including both observed and latent points. Moreover, it is not applicable to high-dimensional input data since the expected number of latent points grows proportionally with the volume of the data domain. For instance, an n-dimensional hypercube with sides of length s has volume s^n.

Variational Bayesian approximations for Gaussian Cox processes have been recently proposed [3, 16, 28]. These methods employ inducing points to reduce computation costs, albeit usually yielding less precise outcomes compared to the aforementioned exact MCMC inference approaches. [16] target an augmented posterior in which in addition to introducing thinning points (as in [2]), the authors add Pólya-Gamma marks to those points. This allows them to derive analytical expressions of variational posterior mean and covariance. However, their results depend on finite dimensional integrals which cannot be computed analytically. In [28], the authors restrict the intensity function to a quadratic function of a Gaussian process.

An alternative approach is to treat the cumulative intensity function as a latent random variable. [23] and [24] factor the intensity function as the product of the normalized intensity function and the cumulative intensity function, each with independent priors. The authors then place a Beta Dirichlet process mixture prior on the normalized intensity function, and a Jeffreys prior on the cumulative intensity function. Though this model is appealing theoretically, its MCMC posterior computation requires truncating the number of mixture components to a finite value, as is typically done in Dirichlet process mixture models. Similarly, [44] treats the cumulative intensity function as a latent variable with a Gamma prior, however the authors estimate the posterior distribution of the intensity function only at inducing points via MCMC. Motivated by improving scalability, the authors select the set of inducing points by optimizing an utility function in a similar fashion as [39]. Instead of placing a functional prior on the intensity function, they construct a finite-dimensional prior on intensity function values at inducing and observed points together with the cumulative intensity function.

In this paper, in addition to placing a transformed Gaussian process prior on the intensity function, we a priori model the cumulative intensity function as a latent random variable. We then perform MCMC inference on the augmented posterior distribution of intensity function values at

observed and predicted locations together with the cumulative intensity function. Our proposed method involves neither discretization nor heavy data augmentation. In this sense, our method is exact as the likelihood is not approximated. Quite different from [23] and [44], though we also conceive the cumulative intensity function as random, we employ a transformed Gaussian process as the functional prior, which fortunately implies an analytical joint prior over the cumulative intensity function and intensity function values at a finite set of locations. Moreover, we will show that our proposed method can handle mixed data of recurrent events and binned counts, a prevalent situation in applications in biology, ecology and survival analysis. To our knowledge, our method is the first exact Gaussian Cox Processes method that infers intensity function from recurrent event data, count data, and a mixture of both types. In particular, handling count data can be very useful for analyzing large event datasets by grouping events into bins. In Section 2 we describe the inferential problem and our proposed methods. In Section 3 we test the performance of our mehtod and compare it to state-of-art methods. We then demonstrate the utility of our method in three real-world scenarios including temporal and spatial event data, as well as aggregated time count data collected at multiple resolutions. Finally, we discuss extensions of our proposed method to other point processes. Code implementation of our methods and documentation are available at https://github.com/bingjingle/Exact-Bayesian-Gaussian-Cox-Processes-Using-Random-Integrals.

2. Methods.

We assume two possible data scenarios. One in which we observe a set of N recurrent events $\{s_n\}_{n=1}^N \in \mathcal{S}$, and one in which we allow data to come as a mixture of recurrent events $\{s_n\}_{n=1}^N \in B_0 \subset \mathcal{S}$ and binned count data $\{c_j\}_{j=1}^J$ that records the number of observed events in $\{B_j\}_{j=1}^J$. Here $\{B_j\}_{j=0}^J$ is a partition of $\mathcal{S}$. In both cases, we assume they are realizations from an inhomogeneous Poisson process on $\mathcal{S} \subset \mathcal{R}^D$ with intensity function $\lambda(\cdot) : \mathcal{S} \to \mathbb{R}^+$. In the first case, the likelihood is

$$p(\{s_n\}_{n=1}^N|\lambda) = \exp\{-\Lambda(\mathcal{S})\} \prod_{n=1}^N \lambda(s_n), \tag{2.1}$$

where $\Lambda(\mathcal{S}) = \int_{\mathcal{S}} \lambda(s)ds$, and in the second case, the likelihood is

$$p(\{s_n\}_{n=1}^N, \{c_j\}_{j=1}^J|\lambda) = \exp\{-\Lambda(\mathcal{S})\} \prod_{n=1}^N \lambda(s_n) \prod_{j=1}^J \frac{\Lambda(B_j)^{c_j}}{c_j!}. \tag{2.2}$$

Typically, $\lambda(s)$ is a priori modeled as a transformed Gaussian process. A Gaussian process is a stochastic process such that function values at every finite set of locations follow a multivariate Gaussian distribution [53]. The transformation is chosen to ensure $\lambda(s) > 0 \ \forall s \in \mathcal{S}$. Popular choices for transformations are exponential [32] and sigmoid [2]. An important consequence of this nonparametric prior is that $\lambda(s)$ is a nonlinear transformation of a Gaussian process that is not analytically tractable for integration and so the cumulative intensity functions $\Lambda(\mathcal{S})$ and $\{\Lambda(B_j)\}_{j=1}^J$ in Equations (2.1) and (2.2) become intractable. Moreover, the distribution of $\Lambda(\mathcal{S})$ is not available in closed form. To avoid this problem, we use an alternative transformation and a priori model $\lambda(s)$, at a finite set of points, and $\Lambda(\mathcal{S})$ jointly. Similarly for the setting of Equation (2.2), we a priori jointly model $\left[\lambda(x_1), \ldots, \lambda(x_M), \{\Lambda(B_j)\}_{j=0}^J\right]'$. Before we define our prior, we will review some properties of Gaussian processes in the next section.

2.1. Unconstrained Gaussian processes.

It is a well known fact in the literature that if the function $f(x)$ is an unconstrained Gaussian process on $\mathcal{X}$, $\int_{\mathcal{X}} f(x)dx$ is a Gaussian functional [21, 45, 53]. If $\mathcal{X}$ is a fixed set, then $\int_{\mathcal{X}} f(x)dx$ is a Gaussian random variable and the joint vector of the Gaussian process evaluated at a finite set of locations and $\int_{\mathcal{X}} f(x)dx$ is a multivariate Gaussian vector. We present the detailed statement in Theorem 2.1 and its proof can be found in Appendix A.

Theorem 2.1. *Suppose the Gaussian process $f(\cdot)$ on the compact space $\mathcal{X}$ satisfies the assumption that its mean function $\mu(\cdot)$ and covariance kernel $k(\cdot,\cdot)$ are integrable, i.e., $\int_{\mathcal{X}}\mu(s)ds$,$\int_{\mathcal{X}}k(s,t)dt$ and $\int\int_{\mathcal{X}\times\mathcal{X}}k(s,t)dsdt$ exist. For every finite set of vectors $s_1,\ldots,s_p\in\mathcal{X}$, the vector $\mathbf{f}\coloneqq\left[f(s_1),\ldots,f(s_p),\int_{\mathcal{X}}f(s)ds\right]'$ follows a Gaussian distribution and*

$$\mathbf{f}\sim\mathcal{N}\left(\boldsymbol{\mu},\begin{pmatrix}\boldsymbol{V}_{SS} & \boldsymbol{V}_{SI}\\ \boldsymbol{V}_{SI}' & \boldsymbol{V}_{II}\end{pmatrix}\right),$$

where $\boldsymbol{\mu}\coloneqq\left[\mu(s_1),\ldots,\mu(s_p),\int_{\mathcal{X}}\mu(s)ds\right]'$, $\boldsymbol{V}_{SS}$ is the $p\times p$ kernel matrix containing covariance terms for all pairs of function values $\{f(s_i)\}_{i=1}^p$, $\boldsymbol{V}_{SI}$ is a p-dimensional vector formed by covariance terms between function values $\{f(s_i)\}_{i=1}^p$ and $\int_{\mathcal{X}}f(s)ds$ with i-th term being $\int_{\mathcal{X}}k(s_i,t)dt$, and $\boldsymbol{V}_{II}$ is the covariance of $\int_{\mathcal{X}}f(s)ds$, whose value is $\int\int_{\mathcal{X}\times\mathcal{X}}k(s,t)dsdt$.

Theorem 2.1 can be extended for a Gaussian process evaluated at any finite set of locations and integrals over subsets of $\mathcal{X}$. That is, for subsets $\{\mathcal{X}_i\}_{i=1}^q$ where $\mathcal{X}_i\subset\mathcal{X}$, the vector $\mathbf{e}\coloneqq\left[f(s_1),\ldots,f(s_p),\int_{\mathcal{X}_1}f(s)ds,\ldots,\int_{\mathcal{X}_q}f(s)ds\right]'$ follows a Gaussian distribution

$$\mathbf{e}\sim\mathcal{N}\left(\tilde{\boldsymbol{\mu}},\begin{pmatrix}\boldsymbol{V}_{SS} & \tilde{\boldsymbol{V}}_{SI}\\ \tilde{\boldsymbol{V}}_{SI}' & \tilde{\boldsymbol{V}}_{II}\end{pmatrix}\right),$$

where $\boldsymbol{\mu}\coloneqq[\mu(s_1),\ldots,\mu(s_p),\int_{\mathcal{X}_1}\mu(s)ds,\int_{\mathcal{X}_q}\mu(s)ds]'$, $\tilde{\boldsymbol{V}}_{SI}$ is a $p\times q$ matrix formed by covariance terms between function values $\{f(s_i)\}_{i=1}^p$ and integral values $\{\int_{\mathcal{X}_j}f(s)ds\}_{j=1}^q$ with ij-th term being $\int_{\mathcal{X}_j}k(s_i,t)dt$, and $\tilde{\boldsymbol{V}}_{II}$ is a $q\times q$ matrix containing covariance terms for all pairs of integral values $\{\int_{\mathcal{X}_j}f(s)ds\}_{j=1}^q$ with ij-th term being $\int\int_{\mathcal{X}_i\times\mathcal{X}_j}k(s,t)dsdt$.

2.2. Positive Gaussian prior. For simplicity, we will focus on the setting in Equation (2.1) and aim to define a joint prior on $\boldsymbol{\lambda}\coloneqq[\lambda(x_1),\ldots,\lambda(x_M),\Lambda(\mathcal{S})]'$ at locations of interest, which include both observed points $\{s_n\}_{n=1}^N$, and prediction (test) locations $\{t_l\}_{l=1}^{M-N}$, that is, $\{x_i\}_{i=1}^M\coloneqq\{t_l\}_{l=1}^{M-N}\cup\{s_n\}_{n=1}^N$. Unfortunately, there is no equivalent Theorem 2.1 for truncated positive Gaussian distributions. Generally, a linear function of truncated Gaussians is not truncated Gaussian, and if $[f(s_1),\ldots,f(s_p)]'$ follows a truncated positive Gaussian distribution, $\left[f(s_1),\ldots,f(s_p),\int_{\mathcal{X}}f(s)ds\right]'$ neither follows a truncated multivariate Gaussian distribution nor has a nice analytic form of distribution. However, we can first specify an (unrestricted) Gaussian vector $\mathbf{f}$ using Theorem 2.1, and then define $\boldsymbol{\lambda}=\mathbf{f}\cdot\mathbb{1}(\mathbf{f}>\mathbf{0})$, imposing an additional positive constraint to $\mathbf{f}$. To be precise, $\boldsymbol{\lambda}$ is a truncated positive Gaussian vector, i.e, $\boldsymbol{\lambda}\sim\mathcal{TN}(\boldsymbol{\mu},\boldsymbol{V})$, with density:

$$p(\boldsymbol{\lambda})=\frac{\exp\left\{-\frac{1}{2}(\boldsymbol{\lambda}-\boldsymbol{\mu})'\boldsymbol{V}^{-1}(\boldsymbol{\lambda}-\boldsymbol{\mu})\right\}}{\int_{\mathcal{F}}\exp\left\{-\frac{1}{2}(\boldsymbol{\lambda}-\boldsymbol{\mu})'\boldsymbol{V}^{-1}(\boldsymbol{\lambda}-\boldsymbol{\mu})\right\}d\boldsymbol{\lambda}}\cdot\mathbb{1}(\boldsymbol{\lambda}>\mathbf{0}) \tag{2.3}$$

where $\mathcal{F}=\otimes_{i=1}^{M+1}[0,+\infty)$ and $\mathbb{1}(\boldsymbol{\lambda}>\mathbf{0})$ is an indicator function that takes 1 if all elements of $\boldsymbol{\lambda}$ are positive. In this work, we set mean $\boldsymbol{\mu}$ to be zero, and define $\boldsymbol{V}$ according to Theorem 2.1, and the following two covariance kernels: the squared exponential kernel with hyperparameters $\theta=(\theta_0,\theta_1)$, i.e., $k_{SE}(x,x')=\theta_0\exp\left(-\frac{\theta_1\|x-x'\|^2}{2}\right)$, and the Brownian motion covariance kernel with a hyperparameter θ denoting the precision parameter, i.e., $k_{BM}(x,x')=\frac{1}{\theta}\min(x,x')$ (see their integrals in Appendix C). We also assume $\mathcal{S}=[0,T]$ and T is known. In general, one can select covariance kernels with analytic integrals, such as linear, squared exponential, and Brownian motion covariance kernels [53]. To explore the expressive power of this positive prior, we simulate 100 samples and compare them to those from log-Gaussian and sigmoid-Gaussian processes, as shown in Figure 10.

2.2.1. Prior defined on observed and predicted locations. The marginal prior on observed locations induced by Equation (2.3) is not a truncated Gaussian. We could have specified our prior on $\boldsymbol{\lambda}_N := [\lambda(s_1), \ldots, \lambda(s_N), \Lambda(\mathcal{S})]'$ only on observed points as in Equation (2.3). However, it is usually of interest to sample from the predictive posterior distribution at unobserved points. One could obtain the predictive distribution by defining a joint posterior on $\boldsymbol{\lambda}$, on both observed and unobserved locations, and integrating out the posterior over intensity function values (and cumulative intensities) at observed locations and hyperparameters. For this, we would need to define a joint prior on $\boldsymbol{\lambda}$, normally via $q(\boldsymbol{\lambda}) = q(\lambda(t_1), \ldots, \lambda(t_{M-N}) \mid \boldsymbol{\lambda}_N)p(\boldsymbol{\lambda}_N)$, where $q(\lambda(t_1), \ldots, \lambda(t_{M-N}) \mid \boldsymbol{\lambda}_N)$ can be conveniently defined as a truncated Gaussian. However, $q(\boldsymbol{\lambda}) \neq p(\boldsymbol{\lambda})$ (with $p(\cdot)$ as defined in Equation (2.3)). Moreover, simulating from $q(\lambda(t_1), \ldots, \lambda(t_{M-N}) \mid \boldsymbol{\lambda}_N)$ (i.e., truncated Gaussian) is often time consuming. Instead, we picked $p(\boldsymbol{\lambda})$ as our prior. Although this distribution is also truncated Gaussian, we will later show that we actually do not need to sample from this prior directly. For our posterior inference, using the $p(\cdot)$ prior results in a posterior that is easier to sample from than using the $q(\cdot)$ prior. We conveniently define the joint prior with prediction points as it is all that is needed for posterior inference.

A nuisance in the definition of the induced marginal prior on observed locations is that its definition changes as we change the prediction set. This is not the case for Gaussian priors as marginals remain in the same family. One potential concern would be the posterior sensitivity to the choice of prediction set when the number of observed locations is small. In our experiments, we observe that the induced marginal prior exhibits asymptotic convergence as described in Appendix B, implying that the induced marginal prior on observed locations becomes effectively stable as the size of the prediction set exceeds some threshold N. For example, in one-dimensional space, $N \approx 20$ as suggested by our experiment results. When the size of users' selected prediction is very small, to avoid the sensitivity, we suggest that users add additional locations to the prediction set, for example, in one-dimensional space, we suggest at least 20 prediction points.

2.3. Posterior inference. We are interested in estimating the posterior

$$p(\boldsymbol{\lambda}, \theta | \{x_i\}_{i=1}^M) \propto \; p_\theta(\theta) \cdot \mathcal{TN}(\boldsymbol{\lambda}; \mathbf{0}, V_\theta) \cdot \exp\{-\Lambda(\mathcal{S})\} \cdot \prod_{n=1}^{N} \lambda(s_n), \tag{2.4}$$

where the covariance V_θ is constructed from the kernel function $k_\theta(\cdot, \cdot)$ as described in Theorem 2.1, and the mean of the GP prior is assumed to be zero. For the Brownian Motion covariance kernel, we estimate the posterior distribution via Metropolis-within-Gibbs sampling in two steps, alternating between $\boldsymbol{\lambda}$ and θ. For the squared exponential kernel we take an empirical Bayes approach : we estimate the hyperparameter θ from the data, then plug the estimate into the prior before doing Bayesian inference.

To sample from the full conditional

$$p(\boldsymbol{\lambda} | \theta, \{x_i\}_{i=1}^M) \propto \mathcal{N}(\boldsymbol{\lambda}; \mathbf{0}, V_\theta) \, \mathbb{1}(\Lambda(\mathcal{S}) > 0) \exp\{-\Lambda(\mathcal{S})\} \prod_{i=1}^{M} \mathbb{1}(\lambda(x_i) > 0) \prod_{n=1}^{N} \lambda(s_n). \tag{2.5}$$

via a Metropolis-Hastings algorithm with Gaussian proposal would lead to rare acceptance due to the positivity constraint. A Metropolis-Hastings algorithm with truncated Gaussian proposals such as the one proposed in [29] would lead to both inefficiency in sampling from truncated multivariate normal distribution (especially for high-dimensional) and inaccuracy in acceptance rate estimation. Fortunately, we found that a routine elliptical slice sampler (ESS)[36] works well due to its adaptive bracket size and the fact that it changes the original high dimensional sampling problem to a one-dimensional sampling problem of the angle ϕ. ESS targets $p(\psi) = \frac{1}{Z}\mathcal{N}(\psi; 0, \Sigma)L(\psi)$ which matches the factorization in Equation (2.5). ESS proposes $\psi' = \psi sin\phi + \nu cos\phi$ that lies on the ellipse based on ψ and $\nu \sim \mathcal{N}(0, \Sigma)$, and its location depends on the angle ϕ. ϕ is then proposed uniformly from

a bracket $[\phi_{min}, \phi_{max}]$ which is shrunk exponentially quickly until an acceptable state is found (i.e., the proposal ψ' lying in the likelihood threshold).

To sample from

$$p(\theta|\boldsymbol{\lambda}, \{x_i\}_{i=1}^M) \propto \; p_\theta(\theta) \cdot \frac{\exp\left\{-\frac{1}{2}\boldsymbol{\lambda}' V_\theta^{-1}\boldsymbol{\lambda}\right\}}{\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\boldsymbol{\lambda}' V_\theta^{-1}\boldsymbol{\lambda}\right\} d\boldsymbol{\lambda}}, \tag{2.6}$$

where $\mathcal{F} = [0, +\infty)^{M+1}$, we note that the term $\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\boldsymbol{\lambda}' V_\theta^{-1}\boldsymbol{\lambda}\right\} d\boldsymbol{\lambda}$ in Equation (2.6) needs to be numerically approximated, therefore simulating from this conditional posterior distribution is a nontrivial problem. For example, HMC sampling from the full conditional of θ is not directly applicable here. However, in the case of Brownian motion covariance kernel with Gamma prior on θ, we obtain conjugacy and it is possible to sample from Equation (2.6) directly. To show this, consider the Brownian motion covariance kernel: $V_\theta = \frac{1}{\theta}\, C$, where

$$C = \begin{pmatrix} x_1 & \dots & \min(x_1, x_M) & x_1 T - \frac{1}{2}x_1^2 \\ \vdots & \ddots & \vdots & \vdots \\ \min(x_M, x_1) & \dots & x_M & x_M T - \frac{1}{2}x_M^2 \\ x_1 T - \frac{1}{2}x_1^2 & \dots & x_M T - \frac{1}{2}x_M^2 & \frac{1}{3}T^3 \end{pmatrix}. \tag{2.7}$$

The marginal posterior now becomes

$$\begin{aligned} p(\theta|\boldsymbol{\lambda}, \{x_i\}_{i=1}^M) &\propto p_\theta(\theta) \cdot \frac{\exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}\right\}}{\int_{\mathcal{F}} \exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}\right\} d\boldsymbol{\lambda}} \\ &= p_\theta(\theta) \cdot \frac{\exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}\right\}}{\sqrt{\theta}^{-(M+1)} \int_{\mathcal{F}} \exp\left\{-\frac{1}{2}(\sqrt{\theta}\boldsymbol{\lambda})' C^{-1}(\sqrt{\theta}\boldsymbol{\lambda})\right\} d\sqrt{\theta}\boldsymbol{\lambda}} \\ &= p_\theta(\theta) \cdot \frac{\exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}\right\}}{\sqrt{\theta}^{-(M+1)} \int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\mathbf{z}' C^{-1}\mathbf{z}\right\} d\mathbf{z}} \\ &\propto p_\theta(\theta) \cdot \sqrt{\theta}^{M+1} \exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}\right\}, \end{aligned} \tag{2.8}$$

where $\mathbf{z} = \sqrt{\theta}\boldsymbol{\lambda}$. The last step simplifies by noticing that the integral in the denominator no longer depends on θ. Setting $p_\theta(\theta) = \Gamma(\alpha, \beta)$, Equation (2.8) becomes a Gamma distribution with parameters $\tilde{\alpha} = \alpha + \frac{M+1}{2}$ and $\tilde{\beta} = \beta + \frac{1}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}$. In fact, this will also be the case for variance-covariance matrices of the form $\frac{1}{\theta}H_x$, where H_x is a symmetric and positive definite matrix that does not depend on θ.

Generally, a Brownian motion kernel brings three computational advantages: (1) the integral denominator now becomes analytical with respect to θ, namely, a product of a constant and $\sqrt{\theta}^{-(M+1)}$; (2) the conjugacy leads to a parametric marginal posterior distribution amenable for Gibbs sampling; (3) we do not need to compute V_θ, V_θ^{-1}, nor $chol(V_\theta)$ for each iterative update of θ, and instead we just need to compute C, C^{-1}, and $chol(C)$ only once. Moreover, C^{-1} is a tri-diagonal matrix [42] amenable to fast sparse matrix computations of matrix inverse and Cholesky decomposition.

A drawback of mean zero Brownian motion is that for function values at locations near the origin, the prior places very small variance at these locations, and a strong influence in the posterior, leading to low posterior values. To fix this problem, [42] proposed to use intrinsic Gaussian Markov random fields priors with a proper correction at the boundary of the precision matrix C^{-1}. This is equivalent to placing a noninformative prior on $\lambda(0)$ and then marginalizing $\lambda(0)$ out. In the rest of this section we will show how to apply this boundary correction technique to derive a corrected covariance.

First note that the distribution of $\boldsymbol{\lambda}$ conditioned on $\lambda(0) = y$ can be expressed as:

$$p\left(\boldsymbol{\lambda} \mid \lambda(0) = y, \theta\right) \propto \exp\left\{-\frac{\theta}{2}\left(\boldsymbol{\lambda} - y\boldsymbol{l}\right)' C^{-1}\left(\boldsymbol{\lambda} - y\boldsymbol{l}\right)\right\} \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}), \tag{2.9}$$

where $\boldsymbol{l} = (1, \ldots, 1, T)'$. Even though we conditioned on the initial value at y, i.e., $\lambda(0) = y$, the random walk density of $\boldsymbol{\lambda}$ is expressed in terms of function differences $\lambda(x_{i+1}) - \lambda(x_i) = \lambda(x_{i+1}) - y - (\lambda(x_i) - y)$. This allows us to express the conditional density as a multivariate Gaussian distribution with shifted mean. Details are derived in Appendix D. We then place a flat Gaussian prior on $\lambda(0)$, $\mathcal{N}(y; 0, \sigma^2)$, with a large value of σ, and integrate $\lambda(0)$ out to obtain the following modified random walk density:

$$p\left(\boldsymbol{\lambda} \mid \theta\right) = \int_{-\infty}^{+\infty} p\left(\boldsymbol{\lambda} \mid \lambda(0) = y, \theta\right) \mathcal{N}(y; 0, \sigma^2) dy \approx \exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}'\tilde{Q}\boldsymbol{\lambda}\right\} \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \tag{2.10}$$

where $\tilde{Q} = C^{-1} - \dfrac{C^{-1}\boldsymbol{l}\boldsymbol{l}'C^{-1}}{\boldsymbol{l}'C^{-1}\boldsymbol{l}}$. Since $\tilde{Q}$ is rank deficient, we usually add a small perturbation to its diagonal elements to obtain its inverse, i.e., $\tilde{C} = \left(\tilde{Q} + \epsilon\mathbf{I}\right)^{-1}$. Posterior sensitivity to different values of ϵ is shown in Appendix E. Although Equation (2.10) is obtained by taking the limit when $\sigma \to \infty$, this serves as our motivation to define a new prior (not an approximation) with variance matrix $\frac{1}{\theta}\tilde{C}$ instead of $\frac{1}{\theta}C$.

In our implementations, we simply replace C with $\tilde{C}$ in Equation (2.8) and obtain

$$p(\theta|\boldsymbol{\lambda}) \propto p_\theta(\theta) \cdot \sqrt{\theta}^{M+1} \exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}'\tilde{C}^{-1}\boldsymbol{\lambda}\right\} = \Gamma\left(\alpha + \frac{M+1}{2}, \beta + \frac{1}{2}\boldsymbol{\lambda}'\tilde{C}^{-1}\boldsymbol{\lambda}\right) \tag{2.11}$$

To clarify, this intrinsic random walk with precision matrix boundary correction also corresponds to the first-order random walk utilized in the popular integrated nested Laplace approximation model (INLA) [43], a baseline model we use for comparisons in Section 3.

For other covariance kernels, we fix the value of θ to an estimated value first and then we perform posterior sampling of $\boldsymbol{\lambda}$ at locations of interest conditioned on θ as in Equation (2.5).

Estimation of θ. We first assume the intensity function is a piecewise constant function according to a regular grid of $m-1$ points. Denote the regular interval length as $\Delta = \frac{T}{m-1}$. The grid points are located at $\left\{\frac{2k-1}{2}\Delta\right\}_{k=1}^{m-1}$ and therefore we have m intervals in total, among which both the first and last intervals have a length of $\frac{\Delta}{2}$. The function values at each interval is denoted by $\boldsymbol{\lambda_m}^* = (\lambda_1^*, \ldots, \lambda_m^*)$ and Λ_m^* denotes its corresponding cumulative intensity function. We then optimize the following:

$$\underset{m,\theta,\boldsymbol{\lambda_m}^*}{\arg\max}(1-c) \cdot \log(p\left(s_1, \ldots, s_N|\boldsymbol{\lambda_m}^*\right)) + c \cdot \log(\mathcal{TN}\left(\boldsymbol{\lambda_m}^*, \Lambda_m^*; \mathbf{0}, V_\theta\right)) \tag{2.12}$$

where c is a constant (set to be 0.2 in this work based on empirical observations). We name this procedure weighted MAP method. For a fixed value of m, we optimize over $\boldsymbol{\lambda_m^*}$ and θ using R-DEoptim [35]. We repeat this optimization procedure with different choices for m (e.g, $m \in \{1, \ldots, 10\}$) and end with the optimal value of Equation (2.12).

Extending kernels to multiple dimensions. For point processes observed in higher dimensions, [17] suggests two main kernel constructions, which can be obtained multiplying or adding unidimensional kernels. For 2-dimesional data $(x_1, x_2), (x_1', x_2') \in \mathbb{R}^2$, the product-kernel is defined as $k_1(x_1, x_1') \times k_2(x_2, x_2')$ and the additive-kernel is defined as $k_1(x_1, x_1') + k_2(x_2, x_2')$. Since a product-kernel offers more flexibility, we use it to model the intensity function for spatial data in the second real example in Subsection 3.3. For a Brownian motion kernel, the product-kernel we use is the two-dimensional Brownian Sheet [38] with covariance kernel:

$$k_{BS}((x_1, x_2), (x_1', x_2')) = \frac{1}{\theta}\min(x_1, x_1') \cdot \min(x_2, x_2') \tag{2.13}$$

A squared exponential kernel is directly applicable to high dimensional input spaces and categorized as a product-kernel. The hyperparameters estimation and the posterior inference described above for uni-dimensional input space is directly applicable to multi-dimensional input space (see the two-dimensional case in Subsection 3.3). However, for higher-dimensional spaces, our proposed MAP method becomes significantly more computationally intensive. Specifically, if we use $m-1$ grid points per dimension, then the size of the subspace (i.e., the size of λ_m^*) scales as m^d, making optimization over λ_m^* increasingly costly.

3. Experiments.

In this section, we present two synthetic examples and three real examples that consist of recurrent events, spatial point data, and mixed recurrent and count data. For each synthetic example, we compare our proposed random integral (RI) method with the Sigmoid Gaussian Cox Process method (SGCP) proposed by [2] and the INLA method [43]. Among the three exact MCMC methods described in the introduction [2, 23, 44], only SGCP [2] provides open-source code. In terms of computational efficiency, variational Bayes (VB) approaches and INLA are popular choices, with INLA dominating in terms of citations and faster in terms of running time. For example, Table 2 in [3] shows the running time of SGCP is 9 times longer than the variational method STVB, while Table 2 in our paper shows the running time of SGCP is 5800 times longer than INLA.

We implement our algorithm in Python and run it on a shared computing cluster consisting of 24 CPU cores and 191 GB of memory and set a time limit of 7 days. For the RI method on temporal data, we run a total of 50,000 iterations after 10,000 iterations of burn-in. For the SGCP model, each iteration took more than 10,000 times longer than the RI method and for this reason, we were only able to obtain about 10,000 iterations after 10,000 iterations of burn-in. However, some results of the SGCP method shown in Appendix E could not be reported. For the second spatial data real example in Subsection 3.3 we run a total of 100,000 iterations after 100,000 iterations of burn-in.

3.1. Simulations.

We simulated 100 realizations from a Poisson processes with each of the following two intensities:

$$\lambda_1(s) = 2\exp\{-s/15\} + \exp\left\{-((s-25)/10)^2\right\}, \quad s \in [0, 50]. \tag{3.1}$$

$$\lambda_2(s) = 10, \quad s \in [0, 5]. \tag{3.2}$$

For each realization and for all the three Bayesian inference methods (RI/SGCP/INLA), we assumed a Brownian motion covariance kernel and placed a noninformative conjugate gamma prior on the precision parameter θ with hyperparameters $\alpha = \beta = 0.1$. Additionally, we ran both the RI and SGCP methods with a squared exponential kernel and estimated hyperparamters with two methods: (1) via the weighted MAP method (RI/SGCP-MAP), as described in Subsection 2.3; (2) the oracle MLE method (RI/SGCP-MLE) where hyperparameters of the squared exponential kernel are estimated using the true intenstiy function values and observed events. We evaluated the performance of these methods according to sum of squared errors (SSE), coverage, credible intervals width and average running time among each 100 datasets (see their definitions in Appendix E). We compared our Bayesian methods with the classical kernel smoothing (KS) approach of [14, 4]. Specifically, we applied edge-corrected kernel smoothing using a squared-exponential kernel and selected the bandwidth through several commonly used techniques: Poisson likelihood cross-validation (PPL), Scott's rule (SCOTT), Diggle and Berman's [8] mean square error cross-validation method (DIGGLE), and adaptive bandwidth (ADAPTIVE)[1, 12], all based on the R code implementation spatstat [5] and sparr [13]. In addition to the simulations under the trajectories of Equations (3.1) and (3.2), we tested our method on simulations with the intensities of Equations (3.1) and (3.2) scaled by a factor of 2 and 3. Corresponding results can be found in Appendix E.

Results for one simulated dataset with intensities $\lambda_1(s)$ and $\lambda_2(s)$ (Equation (3.1)) are depicted in Figure 1, with squared exponential kernels (Figure 1A) and Brownian motion kernels (Figure 1B).

Although all Bayesian methods have comparable performance, INLA-BM (green shaded region in Figure 1B) shows higher uncertainty than any other method. This is consistent across 100 simulations, where INLA's median credible interval width is 1.44 versus 1.20 with our method for $\lambda_1(s)$ and similarly for $\lambda_2(s)$ (see Table 1, first 3 rows, last column). Performance statistics based on 100 simulations shown in Table 1 indicate that our method is the best performing method among those based on Brownian motion kernels in terms of median SSE. Similarly, for squared exponential kernels with MAP hyperparameter estimation, our method outperforms SGCP in median SSE. Based on comparisons among the three Bayesian methods over the two synthetic examples, it is concluded that both RI and SGCP outperforms INLA, RI outperforming SGCP for most cases. Nonparametric methods based on kernel smoothing with PPL and Scott bandwidth selection perform well in terms of SSE, while ADAPTIVE bandwidth kernel smoothing performs well in terms of coverage in the first synthetic example; however, they perform significantly worse than all Bayesian methods in the second synthetic example.

In all simulations, methods using Brownian motion kernels display larger uncertainty compared to those using squared exponential kernels. This may be due to the fact that hyperparameters of Brownian motion kernels are estimated within the MCMC procedure, while hyperparameters of squared exponential kernels are fixed with estimates. However, the median coverage of most methods are larger than 95%, suggesting those methods are producing conservative credible intervals. Further, we report average running times among 100 simulated datasets per 10,000 iterations for all MCMC algorithms in both RI and SGCP methods for both synthetic examples in Table 2. Even though the cluster used has 24 CPU cores, we parallelized by datasets and so the running times reported in Table 2 correspond to time averages per dataset using a single CPU core. In addition, for INLA-BM, the reported running times are averages per dataset (not per iteration). Though INLA runs the fastest, RI achieved a large improvement over SGCP in terms of running time, which is comparable with INLA. For example, for intensity $\lambda_1(s)$, on average, INLA-BM takes 4.80 s in total and RI-BM takes 15.84 s in total, whereas SGCP-BM takes 13.97 hours in total.

3.2. Earthquakes in Japan. We are interested in estimating the intensity rate of earthquakes in Japan during the year 2019 with magnitude of at least 2.5. The dataset is gathered from the U.S. Geological Survey (2020)[1], including 901 time points. Analogous to the synthetic examples, we run the RI model with both a Brownian motion covariance kernel and a squared exponential kernel and present the corresponding results in the top two panels of Figure 2 (blue dashed lines). We observe that this earthquake temporal point process has a rate around 2.5, with a modest increase observed in the later time period.

3.3. Redwoods. We reanalyze the redwoods spatial dataset that consists of the locations of seedlings and saplings of California Giant Redwoods in a square sampling region (approximately 130 feet across). It was first described and analysed by [46]. The R dataset 'redwoodfull' contains the full point pattern of 195 trees and the space area has been rescaled to the unit square $[0,1] \times [0,1]$. Here we use both the Brownian motion covariance kernel and the squared exponential kernel for GP priors on the intensity function. For a Brownian motion covariance kernel, in order to inherit its computation efficiency we resort to the two-dimensional Brownian Sheet with boundary correction as described in Subsection 2.3. When we perform the boundary correction procedure, there exists a numerical problem related to the computation of C^{-1}, since the definition of the two-dimensional Brownian Sheet leads to very small diagonal elements in C. We have modified our code to first compute $(aC)^{-1}$, where a is a constant (e.g., 10^{12}), then multiply by a.

Results are presented in the bottom two panels of Figure 2. We observe that both methods capture spatial concentration patterns, though there is disagreement in some of the concentrations. In this example, we actually prefer squared exponential kernel as it allows for longer range dependencies.

[1] from the website https://earthquake.usgs.gov/fdsnws/event/1/

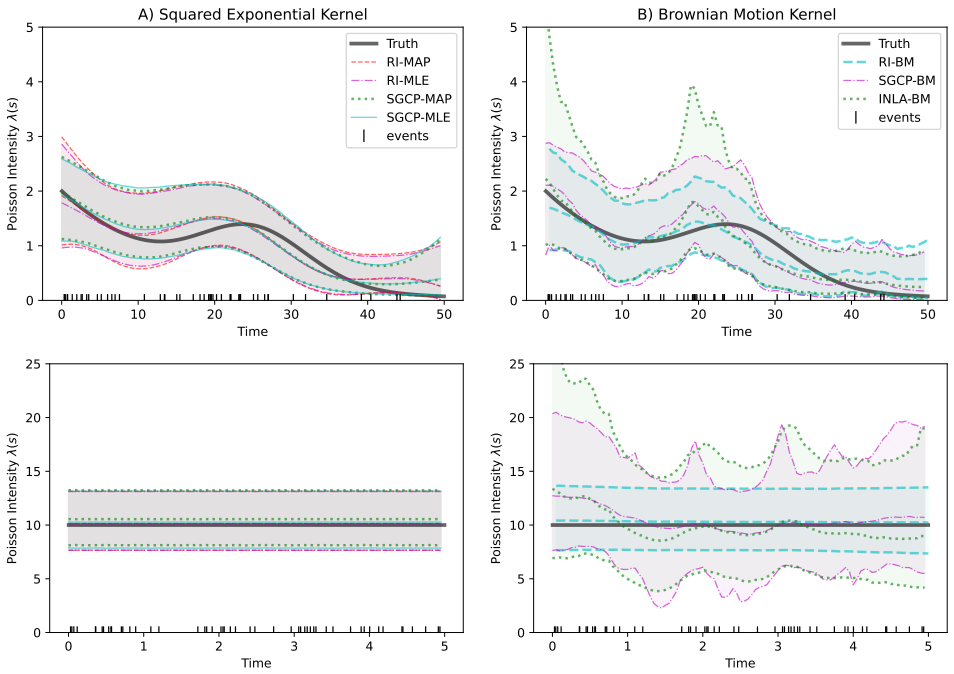

Figure 1: Top row corresponds to posterior inference of intensity function $\lambda_1(s)$ from one simulated dataset of 50 events, while bottom row corresponds to posterior inference of intensity function $\lambda_2(s)$ from one simulated dataset of 51 events. True trajectory is depicted by solid black curve, posterior medians by dashed central curves and 95% CI by shaded regions. Time of simulated events are shown as tick marks at the bottom of each plot. Different methods are distinguished by colors as described in legend boxes, and images in the same column share the same legend box. Methods used in panel A assume squared exponential kernels and those in panel B assume Brownian motion covariance kernels.

3.4. Mixed recurrent and count data . To exemplify the applicability of our methodology to a mixture of recurrent event and count data, we modified the earthquake data previously analyzed in Subsection 3.2. Specifically, we assume the exact recurrent event time points are observed within the first 281 days and the rest of the year (84 days) is reported as weekly counts (12 weeks). In this case, we a priori model the vector $\tilde{\boldsymbol{\lambda}} := \left[\lambda(x_1), \ldots, \lambda(x_M), \int_{B_1} \lambda(s)ds, \ldots, \int_{B_{12}} \lambda(s)ds\right]'$ as positive multivariate Gaussian whose mean and covariance are described in Subsection 2.1. As before, $\{x_i\}_{i=1}^M$ are the locations of interest, and the B_js correspond to the weekly time intervals (from '00:00:00' of day $281+7(j-1)$ to '00:00:00' of day $288+7(j-1)$). See its likelihood in Equation (2.2). Posterior results are also depicted as red curves in the top panel of Figure 2. Remarkably, we obtain very similar results to those obtained from the higher resolution recurrent event data (blue curves) analyzed in Subsection 3.2.

4. Discussion. In this work, we propose an exact nonparametric Bayesian approach to estimate the intensity function of an inhomogeneous Poisson process. The key insight in our method is the realization that if the intensity function is a priori modeled as a Gaussian process, then the Poisson likelihood is a function of a latent Gaussian vector consisting of intensity function values and the cumulative intensity function over the observed set. Since the intensity function needs to be positive, we further restrict the latent vector to being positive. This positive restriction, however, does not increase the sampling complexity of our MCMC approach.

The greatest advantage of our method is that it does not require discretization, expensive data augmentation, or variational approximations. Quite different from [29], which approximates the cumulative intensity function via discretization, our method is exact and mainly focuses on solving the doubly intractable posterior problem [37], though both utilize truncated Gaussian distributions.

Intensity	Methods	SSE at 100 Grids	Coverage at 100 Grids	Credible Interval Width
λ_1	RI-BM	**7.30**(5.22, 9.29)	98% (95%, 100%)	1.20 (1.15, 1.26)
	SGCP-BM	8.87 (**5.19**, 12.54)	100% (100%, 100%)	1.35 (1.24, 1.43)
	INLA-BM	9.01 (6.18, 13.17)	100% (98%, 100%)	1.44 (1.31, 1.59)
	RI-MAP	**7.97**(**5.60**, 11.51)	88% (72%, 99%)	0.84 (0.71, 1.03)
	SGCP-MAP	8.76 (6.73, **11.02**)	77% (59%, 89%)	0.84 (0.73, 0.99)
	RI-MLE	7.17 (4.49, **9.62**)	100% (94%, 100%)	1.04 (0.98, 1.07)
	SGCP-MLE	**6.33** (**4.14**, 9.72)	100% (88%, 100%)	0.97 (0.93, 1.03)
	KS-PPL	7.70 (4.66, 11.01)	31% (19%, 51%)	0.25 (0.19, 0.39)
	KS-SCOTT	**6.18** (**3.76**, **9.17**)	32% (24%, 46%)	0.25 (0.23, 0.26)
	KS-DIGGLE	3178.84 (2684.54, 3596.24)	29% (26%, 31%)	10.30 (9.34, 11.36)
	KS-ADAPTIVE	10.35 (7.00, 14.49)	92% (83%, 98%)	1.05 (0.95, 1.12)
λ_2	RI-BM	**76.63** (**28.65**, **220.65**)	100% (100%, 100%)	6.29 (5.95, 6.74)
	SGCP-BM	231.96 (106.33, 429.64)	100% (100%, 100%)	9.13 (8.56, 10.30)
	INLA-BM	328.58 (173.47, 570.52)	100% (100%, 100%)	12.04 (11.36, 12.90)
	RI-MAP	**121.56** (36.37, **298.73**)	100% (100%, 100%)	5.60 (5.26, 5.87)
	SGCP-MAP	145.73 (**33.50**, 444.92)	100% (100%, 100%)	5.82 (5.43, 6.30)
	RI-MLE	**70.82** (**21.04**, **216.55**)	100% (100%, 100%)	5.84 (5.41, 6.88)
	SGCP-MLE	105.56 (**37.58**, 265.23)	100% (100%, 100%)	5.64 (5.30, 5.95)
	KS-PPL	500053.20 (267875.20, 847638.70)	88% (84%, 92%)	155.13 (118.26, 203.99)
	KS-SCOTT	**470.70** (**252.59**, **784.66**)	100% (81%, 100%)	7.79 (7.17, 8.15)
	KS-DIGGLE	868245.30 (450177.00, 1693187.20)	87% (82%, 91%)	202.95 (153.87, 279.27)
	KS-ADAPTIVE	728.72 (468.86, 1178.17)	100% (91%, 100%)	12.28 (11.30, 13.15)

Table 1: x-BM refers to method x using a Brownian motion kernel. x-MAP denotes method x with a squared exponential kernel, where the hyperparameters are estimated via a weighted MAP approach. x-MLE denotes method x with a squared exponential kernel, where the hyperparameters are estimated using an oracle MLE approach—that is, based on the true intensity function values and observed events. KS-x denotes the kernel smoothing approach with a squared-exponential kernel, where x indicates the method used for bandwidth selection. Performance comparison over 100 grids (test points) on 100 simulations with intensities $\lambda_1(s)$ and $\lambda_2(s)$. The last three columns present quantities in the format: 0.50 quantile (0.25 quantile, 0.75 quantile). Bold is the best among methods with the same kernel. Details of metric definitions and calculations are in Appendix E.

Discretization results in poor approximation, especially in high dimensions, as the choice of the number of knots controls the trade-off between numerical accuracy and computational efficiency. This problem is exacerbated when data consists of mixed recurrent event and count data as many integrals would need to be approximated. Further, as mentioned earlier in Subsection 2.3, we use a different but more efficient MCMC scheme than [29]. Our proposed method is most similar to both [23] and [44], in that we treat the cumulative intensity function as a latent random variable, and target a posterior distribution augmented by the cumulative intensity function. Both [23] and [44] reduce time complexity from cubic in the number of both thinning and observed data points in [2] to linear in the number of observed data points for each MCMC iteration. In our proposed method, attributed to the special structure of Brownian motion kernel covariance, we only need to compute all covariance matrices once during the MCMC procedure, bringing improvement in time efficiency. Specifically, [2] has a time complexity of $O(t(M+S)^3)$, [44] scales computationally in $O(tMk^2)$, while the time complexity under our method is $\max(O(M^3), O(tM^2))$, where t denotes the number of iterations, S denotes the expected number of latent thinning points, and k denotes the number of inducing points. On the other hand, one limitation of our approach is the requirement of integrable covariance kernels.

Data augmentation methods via Poisson thinning [2, 16] require a finite bound and a tractable acceptance probability, restricting its applicability. For example, [50] mentions the bottleneck of applying the data augmentation via thinning method to renewal processes with unbounded hazard functions. In contrast, the random integral method can handle both unbounded and more complex intractable cases. A future direction consists in extending the random integral method to more general point processes such as those used in survival analysis [6, 18, 30]. We anticipate our method will be able to naturally accommodate censored data and spatio-temporally varying covariates.

Methods	Average Time ± Standard Deviation	
	λ_1	λ_2
RI-BM	2.64 ± 0.11 s	2.28 ± 0.90 s
SGCP-BM	28146.07 ± 4862.67 s	5724.21 ± 1025.27 s
INLA-BM	4.80 ± 0.19 s	4.51 ± 1.02 s
RI-MAP	3.17 ± 0.44 s	3.58 ± 0.98 s
SGCP-MAP	127598.22 ± 36866.47 s	41339.5 ± 12600.63 s
RI-MLE	3.49 ± 0.33 s	2.79± 0.27 s
SGCP-MLE	169883.37 ± 28625.97 s	41057.43 ± 8801.7 s
KS-PPL	0.09 ± 0.04 s	0.09 ± 0.05 s
KS-SCOTT	0.06 ± 0.04 s	0.07 ± 0.04 s
KS-DIGGLE	0.11 ± 0.06 s	0.08 ± 0.04 s
KS-ADAPTIVE	537.62 ± 190.47 s	319.52 ± 166.12 s

Table 2: Average running times across 100 simulated datasets for estimating $\lambda_1(s)$ and $\lambda_2(s)$. We note that, among all KS methods, KS-ADAPTIVE incurs substantially higher computational cost than the other KS methods, since the construction of confidence intervals requires bootstrap resampling.

Finally, we showed that our method can be extended to model mixed data consisting of recurrent event data and count data over binned intervals. Surprisingly, we showed an example in which binning some of the observations resulted in no loss of information about the intensity function. This feature not only increases the applicability of nonparametric Bayesian Cox processes to more realistic scenarios, but it can also be exploited for increasing scalability of our method. We anticipate that it is possible to devise a strategy for binning observations in order to increase the applicability of our method to large datasets.

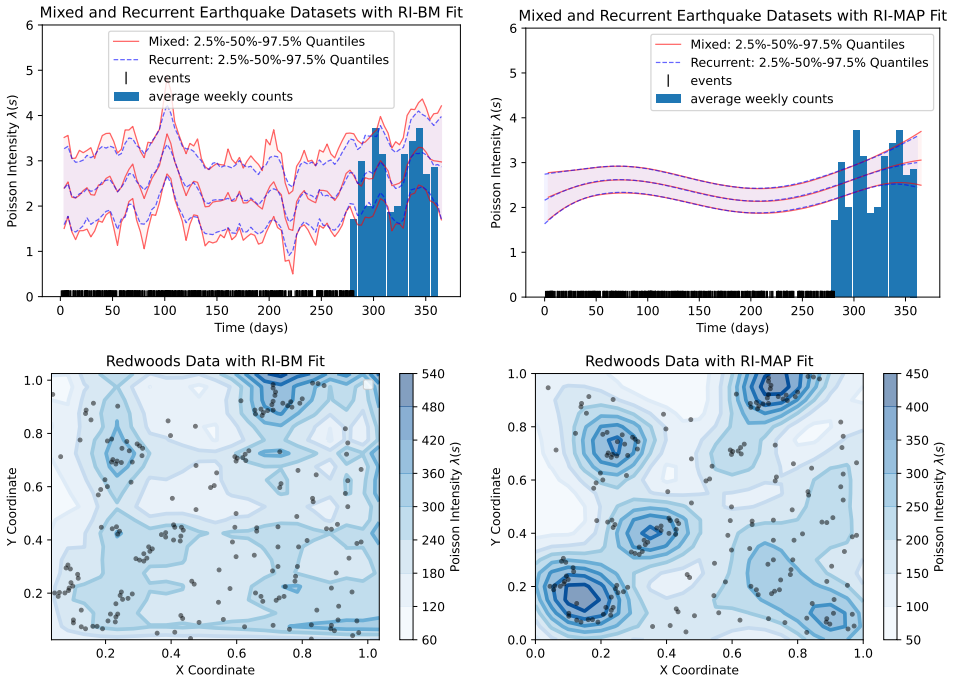

Figure 2: Posterior inferences for real examples from Subsections 3.2 to 3.4. Bottom panels report median among posterior samples of the latent GP for the redwoods spatial dataset and locations of observed events are represented by black dots, while top panels present $2.5\% - 50\% - 97.5\%$ posterior quantiles of the intensity function for both recurrent and mixed earthquake data in Japan and time of observed events are depicted as tick marks at the bottom of each plot. Among each row, the left panel corresponds to results using a Brownian motion covariance kernel and the right panel corresponds to results using a squared exponential kernel with MAP.

Appendix A. joint distribution of function values and integral for Gaussian processes. First, we state a key intermediate result from [34] that we will use to prove Theorem 2.1:

Proposition A.1. *[Proposition 12.15 in [34]] Suppose* $\{X_n : n \in N\}$ *is a sequence of Gaussian random vectors and* $X_n \xrightarrow{a.s.} X$. *If* $b := lim_{n\to\infty} E[X_n]$ *and* $C := lim_{n\to\infty} Cov[X_n]$ *exist, then* X *is Gaussian with mean* b *and covariance matrix* C.

Proof. Let μ_n denote $E[X_n]$, Σ_n denote $Cov[X_n]$, and ψ denote the characteristic function. Since $X_n \xrightarrow{a.s.} X$ implies that $\psi_{X_n}(t) \to \psi_X(t)$ for each t, the rest work is to derive $\lim_{n\to\infty} \psi_{X_n}(t)$.

$$\lim_{n\to\infty} \psi_{X_n}(t) = \lim_{n\to\infty} \exp\left\{it'\mu_n - \frac{1}{2}t'\Sigma_n t\right\}$$
$$= \exp\left\{it'b - \frac{1}{2}t'Ct\right\}$$

The last step is due to the continuity of $\exp\left\{it'\mu_n - \frac{1}{2}t'\Sigma_n t\right\}$ w.r.t. components of μ_n and Σ_n. Now we have

$$\psi_X(t) = \exp\left\{it'b - \frac{1}{2}t'Ct\right\}.$$

The result then follows. ∎

Theorem 2.1. *Suppose the Gaussian process* $f(\cdot)$ *on the compact space* $\mathcal{X}$ *satisfies the assumption that its mean function* $\mu(\cdot)$ *and covariance kernel* $k(\cdot,\cdot)$ *are integrable, i.e.,* $\int_{\mathcal{X}} \mu(s)ds, \int_{\mathcal{X}} k(s,t)dt$ *and* $\int\int_{\mathcal{X}\times\mathcal{X}} k(s,t)dsdt$ *exist. For every finite set of vectors* $s_1, \ldots, s_p \in \mathcal{X}$, *the vector* $\mathbf{f} :=$

$\left[f(s_1), \ldots, f(s_p), \int_{\mathcal{X}} f(s)ds\right]'$ *follows a Gaussian distribution and*

$$\mathbf{f} \sim \mathcal{N}\left(\boldsymbol{\mu}, \begin{pmatrix} \boldsymbol{V}_{SS} & \boldsymbol{V}_{SI} \\ \boldsymbol{V}'_{SI} & \boldsymbol{V}_{II} \end{pmatrix}\right),$$

where $\boldsymbol{\mu} \coloneqq \left[\mu(s_1), \ldots, \mu(s_p), \int_{\mathcal{X}} \mu(s)ds\right]'$, $\boldsymbol{V}_{SS}$ *is the* $p \times p$ *kernel matrix containing covariance terms for all pairs of function values* $\{f(s_i)\}_{i=1}^p$, $\boldsymbol{V}_{SI}$ *is a* p*-dimensional vector formed by covariance terms between function values* $\{f(s_i)\}_{i=1}^p$ *and* $\int_{\mathcal{X}} f(s)ds$ *with* i*-th term being* $\int_{\mathcal{X}} k(s_i, t)dt$*, and* $\boldsymbol{V}_{II}$ *is the covariance of* $\int_{\mathcal{X}} f(s)ds$*, whose value is* $\int \int_{\mathcal{X}\times\mathcal{X}} k(s,t)dsdt$*.*

Proof. It is well known that any affine transformation of a multivariate normal distribution is still a normal distribution. That is to say, if $X \sim \mathcal{N}(X; \mu, \Sigma)$, then $BX \sim \mathcal{N}(BX; B\mu, B\Sigma B')$, where X is a n-dimensional random vector and B is a $m \times n$ constant matrix.
For presentation simplicity, here we only focus on the case when $\mathcal{X} = [0, T]$, i.e., an uni-dimensional space. The proof can be easily extended to a higher finite dimension. Because $f(s)$ is a continuous function, we can define the integral term $\int_{\mathcal{X}} f(s)ds$ as the limit of the Riemann sum:

$$\int_0^T f(s)ds \coloneqq \lim_{n\to\infty} \frac{T}{n} \sum_{i=1}^{n} f\left(\frac{i}{n}T\right).$$

Now, define $X_n = \left(f\left(s_1\right), \ldots, f\left(s_p\right), f\left(\frac{1}{n}T\right), \ldots, f\left(T\right)\right)'$, then

$$X_n \sim \mathcal{N}\left(\tilde{\mu}_n = \begin{pmatrix} \mu(s_1) \\ \vdots \\ \mu(s_p) \\ \mu(\frac{1}{n}T) \\ \vdots \\ \mu(T) \end{pmatrix}, \tilde{\Sigma}_n = \begin{pmatrix} k\left(s_1, s_1\right) & \ldots & k\left(s_1, s_p\right) & k\left(s_1, \frac{1}{n}T\right) & \ldots & k\left(s_1, T\right) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ k\left(s_p, s_1\right) & \ldots & k\left(s_p, s_p\right) & k\left(s_p, \frac{1}{n}T\right) & \ldots & k\left(s_p, T\right) \\ k\left(\frac{1}{n}T, s_1\right) & \ldots & k\left(\frac{1}{n}T, s_p\right) & k\left(\frac{1}{n}T, \frac{1}{n}T\right) & \ldots & k\left(\frac{1}{n}T, T\right) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ k\left(T, s_1\right) & \ldots & k\left(T, s_p\right) & k\left(T, \frac{1}{n}T\right) & \ldots & k\left(T, T\right) \end{pmatrix}\right)$$

Let $A = \begin{pmatrix} \mathbf{I}_{p\times p} & \mathbf{0}_{p\times n} \\ \mathbf{0}_{1\times p} & \left(\frac{\mathbf{T}}{\mathbf{n}}\right)_{1\times n} \end{pmatrix}$, then $AX_n = \begin{pmatrix} f\left(s_1\right) \\ \vdots \\ f\left(s_p\right) \\ \frac{T}{n}\sum_{i=1}^{n} f\left(\frac{i}{n}T\right) \end{pmatrix} \xrightarrow{a.s.} \begin{pmatrix} f\left(s_1\right) \\ \vdots \\ f\left(s_p\right) \\ \int_0^T f(s)ds \end{pmatrix}$.

Since AX_n is an affine transformation of a multivariate normal distribution,

$$AX_n \sim \mathcal{N}(A\tilde{\mu}_n, A\tilde{\Sigma}_n A'),$$

where $A\tilde{\mu}_n = \begin{pmatrix} \mu\left(s_1\right) \\ \vdots \\ \mu\left(s_p\right) \\ \frac{T}{n}\sum_{i=1}^{n} \mu\left(\frac{i}{n}T\right) \end{pmatrix} \xrightarrow{n\to\infty} \begin{pmatrix} \mu\left(s_1\right) \\ \vdots \\ \mu\left(s_p\right) \\ \int_0^T \mu(s)ds \end{pmatrix}$,

and $A\tilde{\Sigma}_n A' = \begin{pmatrix} k\left(s_1, s_1\right) & \ldots & k\left(s_1, s_p\right) & \frac{T}{n}\sum_{i=1}^{n} k\left(s_1, \frac{i}{n}T\right) \\ \vdots & \ddots & \vdots & \vdots \\ k\left(s_p, s_1\right) & \ldots & k\left(s_p, s_p\right) & \frac{T}{n}\sum_{i=1}^{n} k\left(s_p, \frac{i}{n}T\right) \\ \frac{T}{n}\sum_{i=1}^{n} k\left(\frac{i}{n}T, s_1\right) & \ldots & \frac{T}{n}\sum_{i=1}^{n} k\left(\frac{i}{n}T, s_p\right) & \left(\frac{T}{n}\right)^2 \sum_{j=1}^{n}\sum_{i=1}^{n} k\left(\frac{i}{n}T, \frac{j}{n}T\right) \end{pmatrix}$

$$\xrightarrow{n\to\infty} \begin{pmatrix} k\left(s_1, s_1\right) & \ldots & k\left(s_1, s_p\right) & \int_0^T k(s_1, t)dt \\ \vdots & \ddots & \vdots & \vdots \\ k\left(s_p, s_1\right) & \ldots & k\left(s_p, s_p\right) & \int_0^T k(s_p, t)dt \\ \int_0^T k(t, s_1)dt & \ldots & \int_0^T k(t, s_p)dt & \int_0^T \int_0^T k(s,t)dsdt \end{pmatrix}.$$

The result then follows from Proposition A.1. ∎

Pairs of samples	KL Divergence
L=20 vs. L=100	2.159529e-07
L=100 vs. L=200	1.715119e-06
L=200 vs. L=300	3.163985e-07

Table 3: KL divergence for each pair of marginal density estimates on $\lambda(s_1)$.

Appendix B. empirical asymptotic convergence of the positive Gaussian prior. To evaluate how sensitive our induced marginal prior on $\boldsymbol{\lambda}_N$ is to the prediction set, we estimated different marginal priors via Monte Carlo. We assumed $\left(\lambda(s_1), \{\lambda(t_1)\}_{l=1}^L\right) \sim \mathcal{TN}(\mathbf{0}, \Sigma)$, where $s_1 = 0.31$ and $\{\lambda(t_1)\}_{l=1}^L$ at evenly spaced grids in $[0,1]$. We assumed squared exponential kernel for Σ and set $L = 20, 100, 200, 300$. The estimated marginal densities of $\lambda(s_1)$ and KL divergences for each pair of them are presented in Figure 3 and Table 3. We conclude from both qualitative and quantitative results that, if $\{t_l\}_{l=1}^L$ is dense in $\mathcal{S}$ and $p(\{\lambda(s_i)\}_{i=1}^N, \{\lambda(t_l)\}_{l=1}^L)$ follows a truncated positive Gaussian distribution, then the limiting induced marginal exists, that is

$$\lim_{L\to\infty} \int_{\otimes_{i=1}^L [0,+\infty)} \mathcal{TN}\left(\{\lambda(s_i)\}_{i=1}^N, \{\lambda(t_l)\}_{l=1}^L\right) d\{\lambda(t_l)\}_{l=1}^L$$

exists.

Appendix C. kernel integral. In this section, we would show how to derive the integral terms in the covariance matrix in Theorem 2.1 for two specific kernel functions.

C.1. Squared exponential kernels . For a Gaussian process with a squared exponential kernel, here we show the derivation for $\int_0^T k_{SE}(s,t)dt$ and $\int_0^T \int_0^T k_{SE}(s,t)dsdt$. We first claim three facts about the Gauss error function $erf(z) := \frac{2}{\sqrt{\pi}} \int_0^z \exp\{-t^2\}\, dt$.

FACT 1:. $\int_p^q \exp\left\{-\frac{x^2}{2}\right\} dx = \sqrt{\frac{\pi}{2}} \left(erf\left(q/\sqrt{2}\right) - erf\left(p/\sqrt{2}\right)\right)$

FACT 2:. $\int erf(z)dz = z\, erf(z) + \frac{\exp\{-z^2\}}{\sqrt{\pi}} + c$

FACT 3:. $erf(-z) = -erf(z)$

Utilizing the three facts, we compute both single and double integral terms as below.

$$\begin{aligned}
\int_0^T k_{SE}(s,t)dt &= \int_0^T \theta_0 \exp\left(-\frac{\theta_1(s-t)^2}{2}\right) dt \\
&= \frac{\theta_0}{\sqrt{\theta_1}} \int_0^T \exp\left(-\frac{\theta_1(t-s)^2}{2}\right) d\sqrt{\theta_1}(t-s) \\
&= \frac{\theta_0}{\sqrt{\theta_1}} \int_{-\sqrt{\theta_1}s}^{\sqrt{\theta_1}(T-s)} \exp\left\{-\frac{v^2}{2}\right\} dv \\
&= \frac{\theta_0}{\sqrt{\theta_1}} \sqrt{\frac{\pi}{2}} \left[erf\left(\sqrt{\frac{\theta_1}{2}}(T-s)\right) - erf\left(-\sqrt{\frac{\theta_1}{2}}s\right)\right] \\
&= \theta_0 \sqrt{\frac{\pi}{2\theta_1}} \left[erf\left(\sqrt{\frac{\theta_1}{2}}(T-s)\right) + erf\left(\sqrt{\frac{\theta_1}{2}}s\right)\right]
\end{aligned}$$

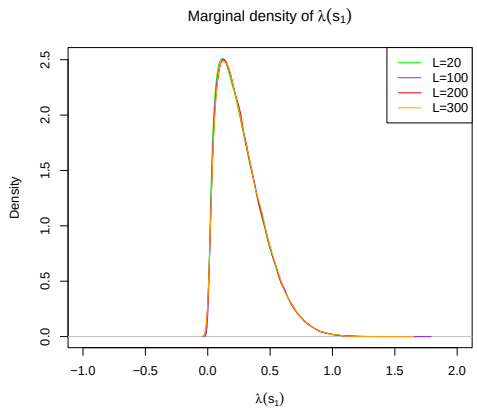

Figure 3: Marginal density estimates for $\lambda(s_1)$ corresponding to $L = 20, 100, 200, 300$.

where $v \coloneqq \sqrt{\theta_1}(t-s)$.

$$\int_0^T \int_0^T k_{SE}(s,t)dtds = \int_0^T \theta_0 \sqrt{\frac{\pi}{2\theta_1}} \left[erf\left(\sqrt{\frac{\theta_1}{2}}(T-s)\right) + erf\left(\sqrt{\frac{\theta_1}{2}}s\right)\right] ds$$

$$= \theta_0 \sqrt{\frac{\pi}{2\theta_1}} \left[\int_0^T erf\left(\sqrt{\frac{\theta_1}{2}}(T-s)\right) ds + \int_0^T erf\left(\sqrt{\frac{\theta_1}{2}}s\right) ds \right]$$

$$= \theta_0 \sqrt{\frac{\pi}{2\theta_1}} \left[-\sqrt{\frac{2}{\theta_1}} \int_0^T erf\left(\sqrt{\frac{\theta_1}{2}}(T-s)\right) d\sqrt{\frac{\theta_1}{2}}(T-s) + \sqrt{\frac{2}{\theta_1}} \int_0^T erf\left(\sqrt{\frac{\theta_1}{2}}s\right) d\sqrt{\frac{\theta_1}{2}}s \right]$$

$$= \theta_0 \sqrt{\frac{\pi}{2\theta_1}} \left[-\sqrt{\frac{2}{\theta_1}} \int_0^T erf\left(\sqrt{\frac{\theta_1}{2}}(T-s)\right) d\sqrt{\frac{\theta_1}{2}}(T-s) + \sqrt{\frac{2}{\theta_1}} \int_0^T erf\left(\sqrt{\frac{\theta_1}{2}}s\right) d\sqrt{\frac{\theta_1}{2}}s \right]$$

$$= \frac{\sqrt{\pi}\theta_0}{\theta_1} \left[\int_0^{\sqrt{\frac{\theta_1}{2}}T} erf\,(u)\,du + \int_0^{\sqrt{\frac{\theta_1}{2}}T} erf\,(v)\,dv \right]$$

$$= \frac{2\sqrt{\pi}\theta_0}{\theta_1} \left[u\ erf(u) + \frac{\exp\left\{-u^2\right\}}{\sqrt{\pi}} \Bigg|_0^{\sqrt{\frac{\theta_1}{2}}T} \right]$$

$$= \frac{2\sqrt{\pi}\theta_0}{\theta_1} \left[\sqrt{\frac{\theta_1}{2}}T\ erf\left(\sqrt{\frac{\theta_1}{2}}T\right) + \frac{\exp\left\{-\frac{\theta_1}{2}T^2\right\}}{\sqrt{\pi}} - \frac{1}{\sqrt{\pi}} \right]$$

$$= \frac{2\theta_0}{\theta_1} \left[\sqrt{\frac{\pi\theta_1}{2}}T\ erf\left(\sqrt{\frac{\theta_1}{2}}T\right) + \exp\left\{-\frac{\theta_1}{2}T^2\right\} - 1 \right]$$

where $u \coloneqq \sqrt{\frac{\theta_1}{2}}(T-s)$ and $v \coloneqq \sqrt{\frac{\theta_1}{2}}s$.

C.2. Brownian motion covariance kernels. Next, we would show a similar derivation for a Gaussian process with a Brownian motion covariance kernel.

$$\begin{aligned}\int_0^T k_{BM}(s,t)dt &= \int_0^T \frac{1}{\theta}\min(s,t)\,dt = \frac{1}{\theta}\left[\int_0^s tdt + \int_s^T sdt\right] \\ &= \frac{1}{\theta}\left[\frac{1}{2}t^2\Big|_0^s + st\Big|_s^T\right] = \frac{1}{\theta}\left(sT - \frac{1}{2}s^2\right)\end{aligned}$$

$$\begin{aligned}\int_0^T\int_0^T k_{BM}(s,t)dtds &= \int_0^T \frac{1}{\theta}\left(sT - \frac{1}{2}s^2\right)ds \\ &= \frac{1}{\theta}\left(\frac{1}{2}Ts^2 - \frac{1}{6}s^3\right)\Big|_0^T = \frac{1}{3\theta}T^3\end{aligned}$$

Appendix D. Brownian motion precision matrix boundary correction. We first show how to derive $p(\boldsymbol{\lambda} \mid \lambda(0) = y, \theta)$ in Equation (2.9). For a Brownian motion starting at y, the conditional distribution of $(\lambda(x_1), \ldots, \lambda(x_M))$ given $\lambda(0) = y$ is:

$$\begin{aligned}p(\lambda(x_1),\ldots,\lambda(x_M)|\lambda(0)=y,\theta) &\propto \exp\left\{-\frac{\theta}{2}\left[\frac{\left(\lambda(x_{(1)})-y\right)^2}{x_{(1)}} + \sum_{i=2}^{M}\frac{\left[\lambda(x_{(i)})-\lambda(x_{(i-1)})\right]^2}{x_{(i)}-x_{(i-1)}}\right]\right\} \\ &= \exp\left\{-\frac{\theta}{2}\begin{pmatrix}\lambda(x_{(1)})-y\\ \vdots \\ \lambda(x_{(M)})-y\end{pmatrix}' Q \begin{pmatrix}\lambda(x_{(1)})-y\\ \vdots \\ \lambda(x_{(M)})-y\end{pmatrix}\right\} \\ &= \exp\left\{-\frac{\theta}{2}\begin{pmatrix}\lambda(x_1)-y\\ \vdots \\ \lambda(x_M)-y\end{pmatrix}' \Sigma^{-1} \begin{pmatrix}\lambda(x_1)-y\\ \vdots \\ \lambda(x_M)-y\end{pmatrix}\right\} \qquad \text{(D.1)}\end{aligned}$$

where $x_{(1)}, \ldots, x_{(M)}$ are the increasingly ordered locations of $x_1, \ldots, x_M$, Q is a symmetric matrix with components

$$Q_{ij} = \begin{cases}\dfrac{1}{x_{(1)}} + \dfrac{1}{x_{(2)}-x_{(1)}} & \text{if } i=j=1 \\ \dfrac{1}{x_{(i+1)}-x_{(i)}} + \dfrac{1}{x_{(i)}-x_{(i-1)}} & \text{if } 1<i=j<M \\ \dfrac{1}{x_{(M)}-x_{(M-1)}} & \text{if } i=j=M \\ -\dfrac{1}{x_{(i+1)}-x_{(i)}} & \text{if } j=i+1 \\ 0 & \text{otherwise}\end{cases},$$

and $\Sigma = \begin{pmatrix} x_1 & \ldots & \min(x_1, x_M) \\ \vdots & \ddots & \vdots \\ \min(x_M, x_1) & \ldots & x_M \end{pmatrix}$.

Derived from Equation (D.1) and Theorem 2.1, we obtain $p(\boldsymbol{\lambda} \mid \lambda(0) = y, \theta)$ as below.

$$\begin{aligned}p(\boldsymbol{\lambda} \mid \lambda(0)=y,\theta) &\propto \exp\left\{-\frac{\theta}{2}\begin{pmatrix}\lambda(x_1)-y\\ \vdots \\ \lambda(x_M)-y \\ \int_0^T\lambda(s)ds - Ty\end{pmatrix}' C^{-1} \begin{pmatrix}\lambda(x_1)-y\\ \vdots \\ \lambda(x_M)-y \\ \int_0^T\lambda(s)ds - Ty\end{pmatrix}\right\} \\ &= \exp\left\{-\frac{\theta}{2}(\boldsymbol{\lambda}-y\boldsymbol{l})' C^{-1}(\boldsymbol{\lambda}-y\boldsymbol{l})\right\}\end{aligned}$$

where $\boldsymbol{l} = (1, \ldots, 1, T)'$ and C is the same as Equation (2.7).

Next, we will show how to derive the distribution of a positive truncated random walk, after boundary correction, corresponding to Equation (2.10).

$$
\begin{aligned}
& p(\boldsymbol{\lambda} \mid \theta) \\
& = \int_{-\infty}^{+\infty} p(\boldsymbol{\lambda} \mid \lambda(0) = y, \theta) \mathcal{N}(y; 0, \sigma^2) dy \\
& \propto \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \int_{-\infty}^{+\infty} \exp\left\{ -\frac{\theta}{2} \left[(y\boldsymbol{l} - \boldsymbol{\lambda})' C^{-1} (y\boldsymbol{l} - \boldsymbol{\lambda}) + \frac{y^2}{\theta\sigma^2} \right] \right\} dy \\
& = \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \int_{-\infty}^{+\infty} \exp\left\{ -\frac{\theta}{2} \left[y^2 \left(\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2} \right) - 2y \boldsymbol{l}' C^{-1} \boldsymbol{\lambda} + \boldsymbol{\lambda}' C^{-1} \boldsymbol{\lambda} \right] \right\} dy \\
& = \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \int_{-\infty}^{+\infty} \exp\left\{ -\frac{\theta}{2} \left[\left(\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2} \right) \left(y - \frac{\boldsymbol{l}' C^{-1} \boldsymbol{\lambda}}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right)^2 + \boldsymbol{\lambda}' C^{-1} \boldsymbol{\lambda} - \frac{(\boldsymbol{l}' C^{-1} \boldsymbol{\lambda})^2}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right] \right\} dy \\
& = \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \exp\left\{ -\frac{\theta}{2} \left[\boldsymbol{\lambda}' C^{-1} \boldsymbol{\lambda} - \frac{(\boldsymbol{l}' C^{-1} \boldsymbol{\lambda})^2}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right] \right\} \int_{-\infty}^{+\infty} \exp\left\{ -\frac{\theta \left(\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2} \right)}{2} \left(y - \frac{\boldsymbol{l}' C^{-1} \boldsymbol{\lambda}}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right)^2 \right\} dy \\
& = \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \exp\left\{ -\frac{\theta}{2} \left[\boldsymbol{\lambda}' C^{-1} \boldsymbol{\lambda} - \frac{(\boldsymbol{l}' C^{-1} \boldsymbol{\lambda})^2}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right] \right\} \left(\theta \boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\sigma^2} \right)^{-\frac{1}{2}} \int_{-\infty}^{+\infty} \exp\left\{ -\frac{z^2}{2} \right\} dz \\
& \propto \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \exp\left\{ -\frac{\theta}{2} \left[\boldsymbol{\lambda}' C^{-1} \boldsymbol{\lambda} - \frac{(\boldsymbol{l}' C^{-1} \boldsymbol{\lambda})^2}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right] \right\} \\
& = \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \exp\left\{ -\frac{\theta}{2} \boldsymbol{\lambda}' \left[C^{-1} - \frac{C^{-1} \boldsymbol{l} \boldsymbol{l}' C^{-1}}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right] \boldsymbol{\lambda} \right\} \\
& \coloneqq\approx \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0}) \exp\left\{ -\frac{\theta}{2} \boldsymbol{\lambda}' \tilde{Q} \boldsymbol{\lambda} \right\}
\end{aligned}
$$

where $z \coloneqq \sqrt{\theta \boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\sigma^2}} \left(y - \frac{\boldsymbol{l}' C^{-1} \boldsymbol{\lambda}}{\boldsymbol{l}' C^{-1} \boldsymbol{l} + \frac{1}{\theta\sigma^2}} \right)$, and $\tilde{Q} \coloneqq C^{-1} - \frac{C^{-1} \boldsymbol{l} \boldsymbol{l}' C^{-1}}{\boldsymbol{l}' C^{-1} \boldsymbol{l}}$.

Appendix E. additional experiments results.

E.1. Evaluation metrics. **Sum of square errors at grid points :** is computed by summing up squared differences between the median of predicted intensity function values and ground truth values.

Coverage at grid points : is the percentage of points at which the true intensity lies within the corresponding 95% credible intervals.

Credible interval width : is computed as the average width of the 95% posterior credible intervals across all grid points.

These three statistics are evaluated on a regular grid of 100 points per dataset. We report the 25th, 50th (median), and 75th percentiles of the resulting values across 100 datasets.

E.2. Simulations. Here, we present additional qualitative and quantitative results for the two Poisson processes with intensities given in Equations (3.1) and (3.2), as well as posterior sensitivity to different values of ϵ, as shown in Figure 9. In addition to Figure 1, the trace plots and histograms in Figures 4 and 5 illustrate traceplots of the latent GP at midpoint and histograms of the cumulative intensity function for the two simulated datasets with intensities $\lambda_1(s)$ and $\lambda_2(s)$. All results demonstrate that the MCMC algorithm converges well and modes of histograms approach the ground truth values for the cumulative intensity function.

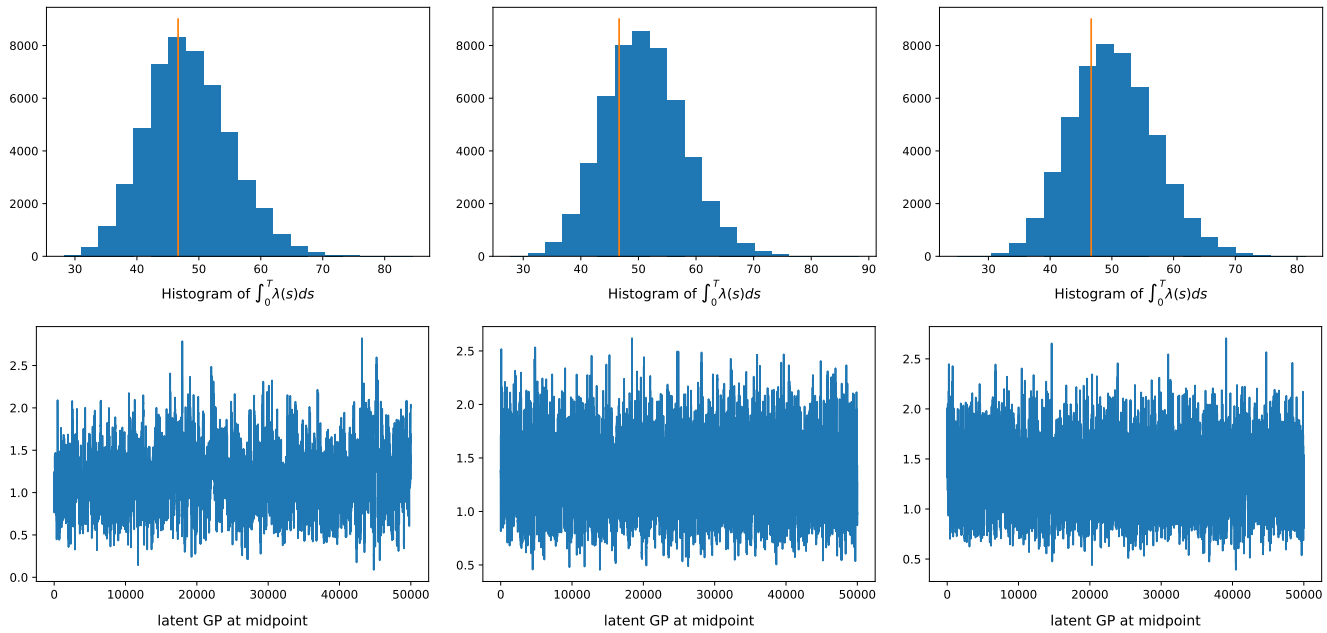

Figure 4: MCMC traceplots for posterior samples of latent GP at midpoint along with histograms of the cumulative intensity function for the same simulated dataset from $\lambda_1(s)$ as that in top row of Figure 1. On the top row, the red vertical line in each histogram represents the ground truth value for $\int_0^T \lambda_1(s)ds$. The three columns from left to right correspond to results using methods RI-BM, RI-MAP and RI-MLE respectively.

We also include samples from the truncated Gaussian, log-Gaussian, and sigmoid-Gaussian processes in Figure 10. Tables 4 to 7 report quantiles of the same performance evaluation metrics as those inTables 1 and 2, evaluated across 100 simulated datasets from the original, doubled, and tripled versions of both intensities. In addition to reporting results at grid points, we also include metrics computed at the observed locations.

E.3. Earthquakes in Japan & redwoods. Figures 6 and 8 mainly present traceplots of latent GP and histograms of the cumulative intensity function for real examples in Subsections 3.2 and 3.3. Additionally, posterior results obtained using INLA for Subsection 3.2 are shown in Figure 6. Figure 7 displays the posterior results for the mixed data described in Subsection 3.4, presented separately from the posterior results for the recurrent data in Subsection 3.2. In Figure 8, we also include the posterior median intensity estimated via the kernel smoothing method for Subsection 3.3.

Table 8 reports the effective sample size at the midpoint, along with the minimum and maximum values across all grid points for the earthquake example in Subsection 3.2. Similarly, Table 9 reports these metrics for the example in Subsection 3.3.

Expected Event Counts	Methods	SSE at Observations	Coverage at Observations	Credible Interval Width
46.65	RI-BM	**3.21** (**2.25**, **4.16**)	98% (95%, 100%)	1.32 (1.25, 1.39)
	SGCP-BM	4.07 (2.86, 7.57)	100% (100%, 100%)	1.60 (1.46, 1.73)
	INLA-BM	4.19 (3.18, 6.32)	100% (100%, 100%)	2.02 (1.69, 2.30)
	RI-MAP	4.23 (3.08, 5.46)	90% (77%, 100%)	0.96 (0.80, 1.18)
	SGCP-MAP	**3.63** (**3.04**, **4.59**)	88% (74%, 100%)	0.94 (0.80, 1.11)
	RI-MLE	3.55 (2.40, **4.41**)	100% (94%, 100%)	1.19 (1.14, 1.24)
	SGCP-MLE	**3.17** (**2.08**, 4.52)	100% (96%, 100%)	1.15 (1.09, 1.18)
93.3	RI-BM	**17.70** (13.72, **23.45**)	99% (94%, 100%)	1.83 (1.74, 1.93)
	SGCP-BM	19.42 (**13.21**, 33.42)	100% (100%, 100%)	2.58 (2.43, 2.75)
	INLA-BM	21.77 (15.63, 35.66)	100% (100%, 100%)	3.06 (2.83, 3.42)
	RI-MAP	24.20 (17.56, 29.65)	93% (77%, 100%)	1.61 (1.44, 1.79)
	RI-MLE	**14.66** (**10.76**, 25.12)	100% (98%, 100%)	1.81 (1.76, 1.88)
139.95	RI-BM	**49.32** (**38.53**, **63.82**)	99% (93%, 100%)	2.32 (2.20, 2.50)
	INLA-BM	65.31 (42.13, 101.72)	100% (99%, 100%)	4.14 (3.77, 4.51)
	RI-MAP	63.46 (35.73, 89.16)	94% (77%, 100%)	2.09 (1.88, 2.34)
	RI-MLE	**37.26** (**20.94**, 66.00)	100% (96%, 100%)	2.35 (2.28, 2.41)
Expected Event Counts	**Methods**	**SSE at 100 Grids**	**Coverage at 100 Grids**	**Credible Interval Width at 100 Grids**
93.3	RI-BM	**16.20** (**12.58**, **20.97**)	99% (96%, 100%)	1.68 (1.57, 1.74)
	SGCP-BM	17.84 (13.54, 28.46)	100% (100%, 100%)	2.14 (2.02, 2.28)
	INLA-BM	19.95 (15.12, 27.41)	100% (99%, 100%)	2.33 (2.21, 2.51)
	RI-MAP	20.60 (15.11, 26.20)	92% (80%, 100%)	1.39 (1.25, 1.55)
	RI-MLE	**13.35** (**9.95**, **19.11**)	100% (95%, 100%)	1.55 (1.50, 1.60)
139.95	RI-BM	**27.91** (**19.76**, **37.28**)	99% (93%, 100%)	2.07 (1.95, 2.21)
	INLA-BM	35.65 (24.77, 53.81)	100% (99%, 100%)	3.15 (2.91, 3.34)
	RI-MAP	35.86 (18.92, 48.65)	94% (81%, 100%)	1.79 (1.62, 1.94)
	RI-MLE	**21.44** (**11.74**, **37.21**)	100% (96%, 100%)	1.98 (1.92, 2.03)

Table 4: Performance comparison on simulations with intensities $\lambda_1(s)$, $2\lambda_1(s)$ and $3\lambda_1(s)$. The last three columns present quantities in the format: 0.50 quantile (0.25 quantile, 0.75 quantile). Bold is the best among methods with the same kernel.

Expected Event Counts	Methods	Average Time ± Standard Deviation
93.3	RI-BM	3.68 ± 1.52 s
	SGCP-BM	234392.72 ± 21852.78 s
	INLA-BM	4.81 ± 0.85 s
	RI-MAP	3.67 ± 0.29 s
	RI-MLE	3.81 ± 0.39 s
139.95	RI-BM	3.22± 0.60 s
	INLA-BM	4.79 ± 0.12 s
	RI-MAP	3.97 ± 0.30 s
	RI-MLE	4.22 ± 0.28 s

Table 5: Running time for estimating intensities $2\lambda_1(s)$ and $3\lambda_1(s)$.

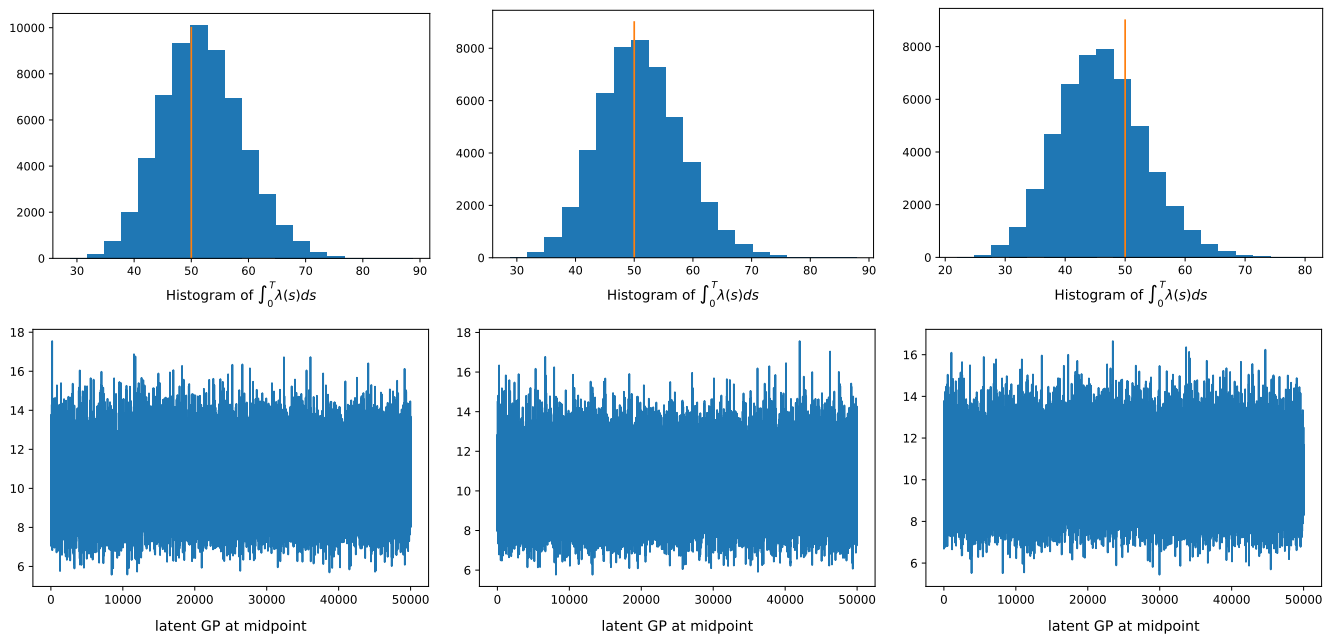

Figure 5: MCMC traceplots for posterior samples of latent GP at midpoint along with histograms of the cumulative intensity function for the same simulated dataset from $\lambda_2(s)$ as that in bottom row of Figure 1. On the top row, the red vertical line in each histogram represents the ground truth value for $\int_0^T \lambda_2(s)ds$. The three columns from left to right correspond to results using methods RI-BM, RI-MAP and RI-MLE respectively.

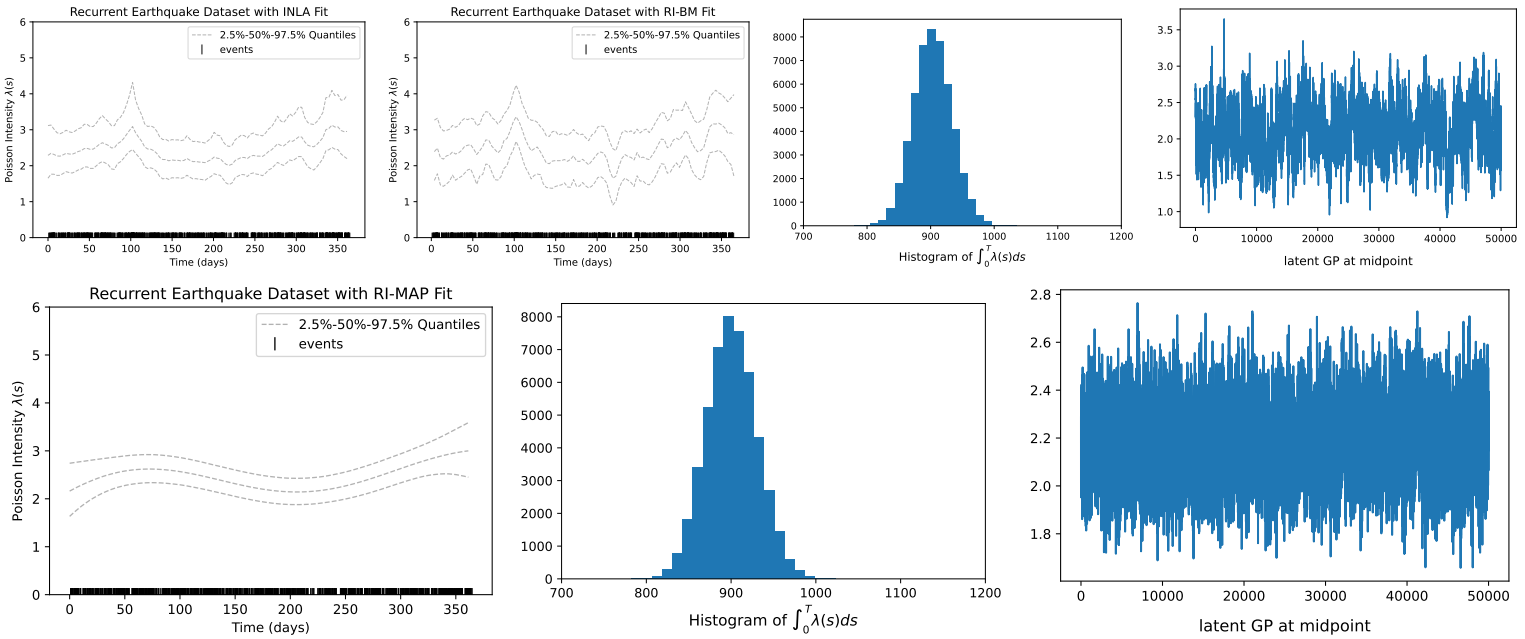

Figure 6: MCMC traceplots for posterior samples of latent GP at midpoint along with histograms of the cumulative intensity function for the recurrent earthquake data in Subsection 3.2. The panel in the top-left corner presents the posterior results using INLA-BM. The remaining panels in the top row show results obtained using our proposed method, RI-BM. The bottom row displays results from our alternative approach, RI-MAP.

Expected Event Counts	Methods	SSE at Observations	Coverage at Observations	Credible Interval Width
50	RI-BM	**40.57** (**11.68**, **107.33**)	100% (100%,100%)	6.28 (5.95,6.73)
	SGCP-BM	112.96 (55.14,236.44)	100% (100%,100%)	9.10 (8.45,10.08)
	INLA-BM	146.70 (86.12,251.31)	100% (100%,100%)	12.71 (11.79,14.13)
	RI-MAP	63.88 (**18.20**,**152.75**)	100% (100%,100%)	5.60 (5.26,5.87)
	SGCP-MAP	65.93 (18.21,213.01)	100% (100%,100%)	5.82 (5.43,6.29)
	RI-MLE	**43.51** (**13.69**,**133.90**)	100% (100%,100%)	5.84 (5.41,6.84)
	SGCP-MLE	53.96 (17.70,136.95)	100% (100%,100%)	5.65 (5.29,5.95)
100	RI-BM	**282.11** (**72.87**,**585.80**)	100% (100%,100%)	9.10 (8.52,9.77)
	SGCP-BM	778.82 (267.56,1.65e+3)	100% (100%,100%)	14.72 (13.37,16.01)
	INLA-BM	801.15 (450.33,1.40e+3)	100% (100%,100%)	18.92 (18.03,20.66)
	RI-MAP	**213.48** (82.61,693.65)	100% (100%,100%)	7.75 (7.46,8.16)
	RI-MLE	677.97 (143.01,4.38e+3)	100% (100%,100%)	8.15 (7.51,36.28)
150	RI-BM	**1.19e+3** (**205.51**, **2.44e+3**)	100% (100%,100%)	12.06 (11.11,13.60)
	INLA-BM	2.16e+3 (1.30e+3,3.70e+3)	100% (100%,100%)	24.53 (23.13,26.05)
	RI-MAP	**605.16** (218.68,1.52e+3)	100% (100%,100%)	9.42 (9.20,9.85)
	RI-MLE	4.21e+5 (1.36e+5,2.62e+6)	0% (0%,48%)	36.90 (16.19,110.35)
Expected Event Counts	**Methods**	**SSE at 100 Grids**	**Coverage at 100 Grids**	**Credible Interval Width at 100 Grids**
100	RI-BM	**298.82** (**75.56**,**571.92**)	100% (100%,100%)	9.05 (8.49,9.73)
	SGCP-BM	884.86 (303.59,1.47e+3)	100% (100%,100%)	14.94 (13.60,16.19)
	INLA-BM	850.71 (512.17,1.48e+3)	100% (100%,100%)	18.15 (17.43,19.73)
	RI-MAP	**213.53** (79.55,675.76)	100% (100%,100%)	7.75 (7.46,8.16)
	RI-MLE	396.97 (131.65,1.49e+3)	100% (100%,100%)	8.15 (7.51,37.86)
150	RI-BM	**830.65** (**138.58**,**1.74e+3**)	100% (100%,100%)	12.04 (11.09,13.52)
	INLA-BM	1.61e+3 (947.39,2.61e+3)	100% (100%,100%)	23.59 (22.42,24.99)
	RI-MAP	**434.85** (151.44,999.74)	100% (100%,100%)	9.42 (9.20,9.85)
	RI-MLE	1.94e+5 (6.97e+4,1.69e+6)	0% (0%,96%)	36.90 (16.24,117.17)

Table 6: Performance comparison on simulations with intensities $\lambda_2(s)$, $2\lambda_2(s)$ and $3\lambda_2(s)$. The last three columns present quantities in the format: 0.50 quantile (0.25 quantile, 0.75 quantile). Bold is the best among methods with the same kernel.

Expected Event Counts	Methods	Average Time ± Standard Deviation
100	RI-BM	1.83 ± 0.36 s
	SGCP-BM	43811.94 ± 9316.78 s
	INLA-BM	4.47 ± 0.20 s
	RI-MAP	3.02± 0.24 s
	RI-MLE	3.82 ± 0.54 s
150	RI-BM	2.45 ± 0.89 s
	INLA-BM	4.34 ± 0.07 s
	RI-MAP	3.27 ± 0.18 s
	RI-MLE	5.28 ± 0.61 s

Table 7: Running time for estimating intensities $2\lambda_2(s)$ and $3\lambda_2(s)$.

Table 8: Effective sample size per 10000 iterations for the recurrent earthquake data in Subsection 3.2.

Methods	Minimum ESS	Maximum ESS	ESS at midpoint
RI-MAP	123.31	621.00	218.79
RI-BM	13.80	52.87	35.28

Table 9: Effective sample size per 10000 iterations for thinned chains in Subsection 3.3.

Methods	Minimum ESS	Maximum ESS	ESS at midpoint
RI-MAP	59.99	376.55	155.54
RI-BM	5.88	148.57	86.52

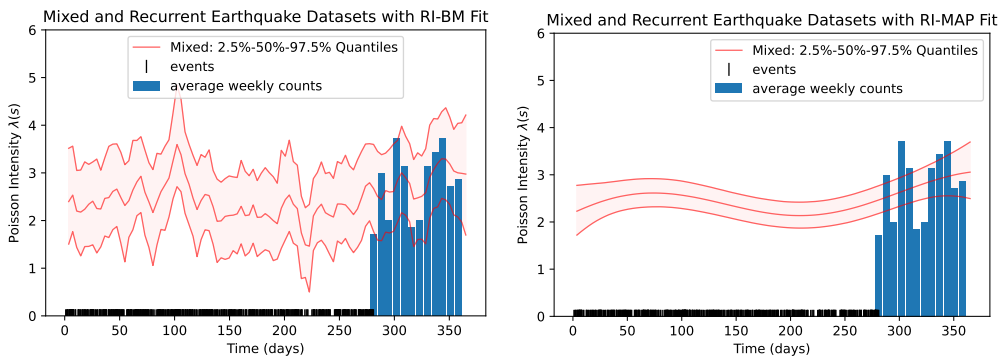

Figure 7: $2.5\% - 50\% - 97.5\%$ posterior quantiles of the intensity function for mixed earthquake data in Subsection 3.4. Left panel corresponds to results using a Brownian motion covariance kernel and the right panel corresponds to results using a squared exponential kernel with MAP.

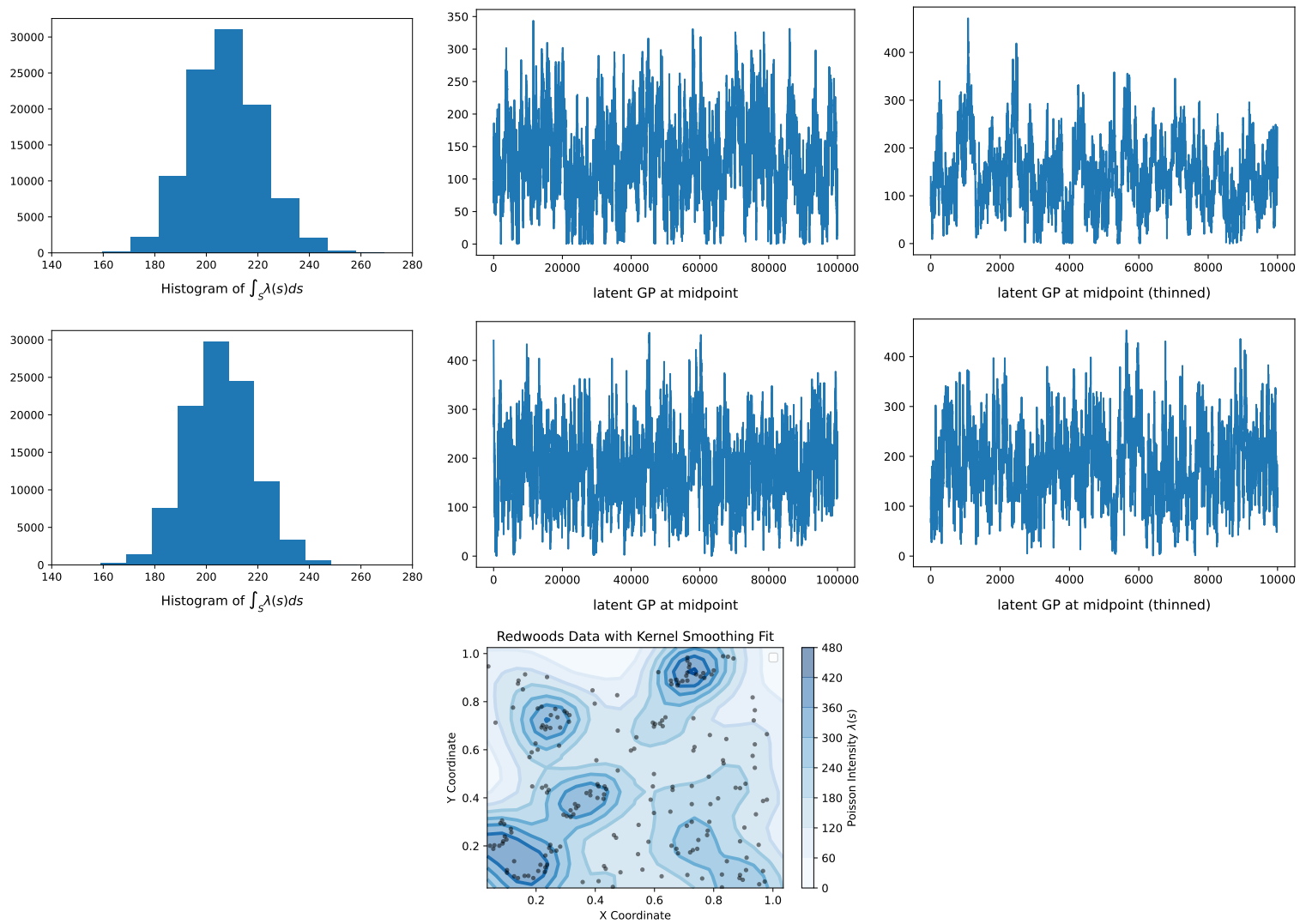

Figure 8: MCMC traceplots for posterior samples of latent GP at midpoint along with histograms of the cumulative intensity function for the redwoods data in Subsection 3.3. The top row presents results obtained using our proposed method, RI-BM, while the middle row corresponds to results from RI-MAP. The bottom row shows the posterior median of the intensity function estimated via kernel smoothing, using a bandwidth of 0.08.

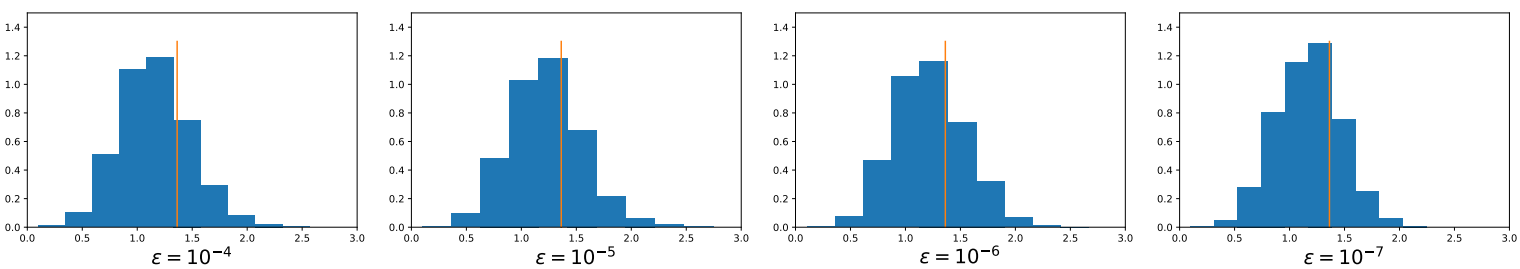

Figure 9: Histograms of posterior samples of latent GP at midpoint for a simulated dataset from $\lambda_1(s)$ when $\epsilon = 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}$ (from left to right) .

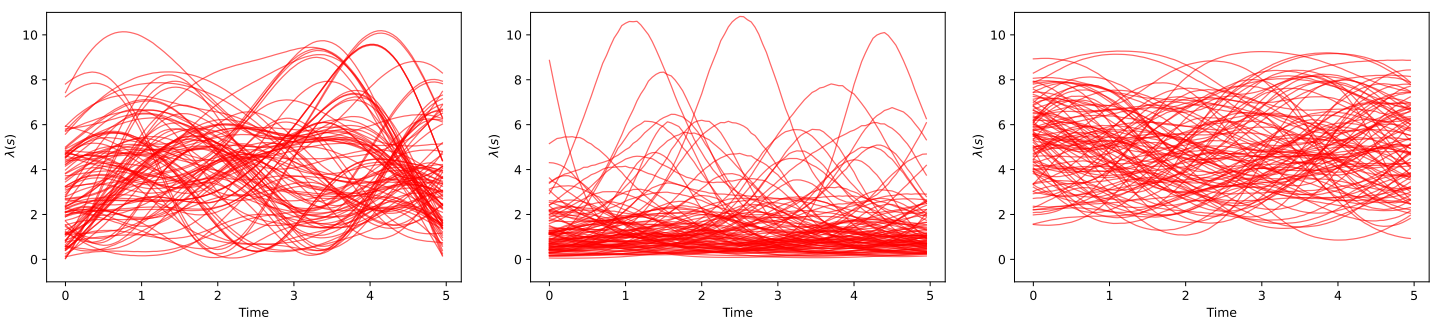

Figure 10: Left: samples from a squared-exponential covariance truncated GP with $\theta_0 = 10$ and $\theta_1 = 1$; Middle: samples from a squared-exponential covariance log GP with $\theta_0 = 0.7$ and $\theta_1 = 1$; Right: samples from a squared-exponential covariance sigmoid GP with $\theta_0 = 0.7$, $\theta_1 = 1$ and $\lambda_{sup} = 10$.

REFERENCES

[1] I. S. Abramson, *On bandwidth variation in kernel estimates–a square root law*, Ann. Statist., 10 (1982), pp. 1217–1223.

[2] R. P. Adams, I. Murray, and D. J. C. MacKay, *Tractable nonparametric Bayesian inference in Poisson processes with Gaussian process intensities*, in Proceedings of the 26th International Conference on Machine Learning, 2009, pp. 9–16.

[3] V. Aglietti, E. V. Bonilla, T. Damoulas, and S. Cripps, *Structured variational inference in continuous Cox process models*, in Advances in Neural Information Processing Systems, vol. 32, 2019.

[4] A. Baddeley, E. Rubak, and R. Turner, *Spatial point patterns: methodology and applications with R*, vol. 1, CRC press, Boca Raton, 2016.

[5] A. Baddeley and R. Turner, *Spatstat: An R package for analyzing spatial point patterns*, Journal of Statistical Software, 12 (2005), pp. 1–42.

[6] J. E. Barrett and A. C. C. Coolen, *Gaussian process regression for survival data with competing risks*, 2013, https://arxiv.org/abs/1312.1591. preprint.

[7] R. Bartoszyński, B. W. Brown, C. M. McBride, and J. R. Thompson, *Some nonparametric techniques for estimating the intensity function of a cancer related nonstationary Poisson process*, Ann. Statist., 9 (1981), pp. 1050–1060.

[8] M. Berman and P. Diggle, *Estimating weighted integrals of the second-order intensity of a spatial point process*, J. R. Stat. Soc. Ser. B Stat. Methodol., 51 (1989), pp. 81–92.

[9] R. T. Chen, B. Amos, and M. Nickel, *Neural Spatio-Temporal Point Processes*, in International Conference on Learning Representations, 2021.

[10] S. Chen, J. Y. S. Xi, and W. K. V. Chan, *Estimating Spline-Based Nonhomogeneous Poisson Intensities Using Constrained Quadratic Programming*, in 2023 Winter Simulation Conference (WSC), IEEE, 2023, pp. 339–350.

[11] D. R. Cox, *Some statistical methods connected with series of events*, J. R. Stat. Soc. Ser. B Stat. Methodol., 17 (1955), pp. 129–157.

[12] T. M. Davies and A. Baddeley, *Fast computation of spatially adaptive kernel estimates*, Statistics and Computing, 28 (2018), pp. 937–956.

[13] T. M. Davies, J. C. Marshall, and M. L. Hazelton, *Tutorial on kernel estimation of continuous spatial and spatiotemporal relative risk*, Statistics in Medicine, 37 (2018), pp. 1191–1221.

[14] P. Diggle, *A kernel method for smoothing point process data*, Journal of the Royal Statistical Society: Series C (Applied Statistics), 34 (1985), pp. 138–147.

[15] P. J. Diggle, P. Moraga, B. Rowlingson, and B. M. Taylor, *Spatial and Spatio-Temporal Log-Gaussian Cox Processes: Extending the Geostatistical Paradigm*, Statistical Science, (2013), pp. 542–563.

[16] C. Donner and M. Opper, *Efficient Bayesian inference of sigmoidal Gaussian Cox processes*, J. Mach. Learn. Res., 19 (2018), pp. 1–34.

[17] D. Duvenaud, *Automatic model construction with Gaussian processes*, PhD thesis, University of Cambridge, 2014.

[18] T. Fernández, N. Rivera, and Y. W. Teh, *Gaussian processes for survival analysis*, in Advances in Neural Information Processing Systems, vol. 29, 2016.

[19] S. Flaxman, Y. W. Teh, and D. Sejdinovic, *Poisson intensity estimation with reproducing kernels*, in Artificial Intelligence and Statistics, PMLR, 2017, pp. 270–279.

[20] P. Guttorp and T. L. Thorarinsdottir, *What happened to discrete chaos, the Quenouille process, and the sharp Markov property? Some history of stochastic point processes*, International Statistical Review, 80 (2012), pp. 253–268.

[21] P. Hennig, M. A. Osborne, and H. P. Kersting, *Probabilistic Numerics: Computation as Machine Learning*, Cambridge University Press, Cambridge, 2022.

[22] J. F. C. Kingman, *Poisson processes*, vol. 3, Clarendon Press, Oxford, 1992.

[23] A. Kottas, *Dirichlet process mixtures of Beta distributions, with applications to density and intensity estimation*, in Proceedings of the Workshop on Learning with Nonparametric Bayesian Methods, 23rd International Conference on Machine Learning, Pittsburgh, PA, 2006.

[24] A. Kottas and B. Sansó, *Bayesian mixture modeling for spatial Poisson process intensities, with applications to extreme value analysis*, Journal of Statistical Planning and Inference, 137 (2007), pp. 3151–3163.

[25] T. Lawrence, A. Baddeley, R. K. Milne, and G. Nair, *Point pattern analysis on a region of a sphere*, Stat, 5 (2016), pp. 144–157.

[26] M. Legg and M. Chitre, *Clustering of snapping shrimp snaps on long time scales: a simulation study*, Acoustics 2012, Fremantle, Australia, (2012).

[27] P. A. W. Lewis and G. S. Shedler, *Simulation of nonhomogeneous Poisson processes by thinning*, Naval Research Logistics Quarterly, 26 (1979), pp. 403–413.

[28] C. Lloyd, T. Gunter, M. Osborne, and S. Roberts, *Variational inference for Gaussian process modulated Poisson processes*, in International Conference on Machine Learning, PMLR, 2015, pp. 1814–1822.

[29] A. F. López-Lopera, S. John, and N. Durrande, *Gaussian process modulated Cox processes under linear inequality constraints*, in The 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 1997–2006.
[30] S. Martino, R. Akerkar, and H. Rue, *Approximate Bayesian inference for survival models*, Scand. J. Stat., 38 (2011), pp. 514–528.
[31] J. Møller, A. N. Pettitt, R. Reeves, and K. K. Berthelsen, *An efficient Markov chain Monte Carlo method for distributions with intractable normalising constants*, Biometrika, 93 (2006), pp. 451–458.
[32] J. Møller, A. R. Syversveen, and R. P. Waagepetersen, *Log Gaussian Cox processes*, Scand. J. Statist., 25 (1998), pp. 451–482.
[33] L. E. Morgan, B. L. Nelson, A. C. Titman, and D. J. Worthington, *A spline-based method for modelling and generating a nonhomogeneous Poisson process*, in 2019 Winter Simulation Conference (WSC), IEEE, 2019, pp. 356–367.
[34] P. Mörters and Y. Peres, *Brownian motion*, vol. 30, Cambridge University Press, Cambridge, 2010.
[35] K. Mullen, D. Ardia, D. L. Gil, D. Windover, and J. Cline, *DEoptim: An R package for global optimization by differential evolution*, Journal of Statistical Software, 40 (2011), pp. 1–26.
[36] I. Murray, R. Adams, and D. MacKay, *Elliptical slice sampling*, in Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 541–548.
[37] I. Murray, Z. Ghahramani, and D. J. C. MacKay, *MCMC for doubly-intractable distributions*, in Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI), 2006, pp. 359–366.
[38] R. Pyke, *Partial sums of matrix arrays, and Brownian sheets*, in Stochastic Analysis, Wiley, 1973, pp. 331–348.
[39] J. Quinonero-Candela and C. E. Rasmussen, *A unifying view of sparse approximate Gaussian process regression*, J. Mach. Learn. Res., 6 (2005), pp. 1939–1959.
[40] H. Ramlau-Hansen, *Smoothing counting process intensities by means of kernel functions*, Ann. Statist., 11 (1983), pp. 453–466.
[41] S. L. Rathbun and N. Cressie, *Asymptotic properties of estimators for the parameters of spatial inhomogeneous Poisson point processes*, Adv. in Appl. Probab., 26 (1994), pp. 122–154.
[42] H. Rue and L. Held, *Gaussian Markov random fields: theory and applications*, Chapman and Hall/CRC, Boca Raton, 2005.
[43] H. Rue, S. Martino, and N. Chopin, *Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations*, J. R. Stat. Soc. Ser. B Stat. Methodol., 71 (2009), pp. 319–392.
[44] Y.-L. K. Samo and S. Roberts, *Scalable nonparametric Bayesian inference on point processes with Gaussian processes*, in International conference on machine learning, PMLR, 2015, pp. 2227–2236.
[45] S. Särkkä, *Linear operators and stochastic partial differential equations in Gaussian process regression*, in International Conference on Artificial Neural Networks, Springer, 2011, pp. 151–158.
[46] D. J. Strauss, *A model for clustering*, Biometrika, 62 (1975), pp. 467–475.
[47] B. M. Taylor, T. M. Davies, B. S. Rowlingson, and P. J. Diggle, *lgcp: An R package for inference with spatial and spatio-temporal log-Gaussian Cox processes*, Journal of Statistical Software, 52 (2013), pp. 1–40.
[48] B. M. Taylor, T. M. Davies, B. S. Rowlingson, and P. J. Diggle, *Bayesian inference and data augmentation schemes for spatial, spatiotemporal and multivariate log-Gaussian Cox processes in R*, Journal of Statistical Software, 63 (2015), pp. 1–48.
[49] B. M. Taylor and P. J. Diggle, *INLA or MCMC? A tutorial and comparative evaluation for spatial prediction in log-Gaussian Cox processes*, Journal of Statistical Computation and Simulation, 84 (2014), pp. 2266–2284.
[50] Y. Teh and V. Rao, *Gaussian process modulated renewal processes*, Advances in Neural Information Processing Systems, 24 (2011).
[51] M. Teng, F. Nathoo, and T. D. Johnson, *Bayesian computation for log-Gaussian Cox processes: A comparative analysis of methods*, Journal of Statistical Computation and Simulation, 87 (2017), pp. 2227–2252.
[52] S. I. Watson, *Twenty ways to estimate the log-Gaussian Cox process model with point and aggregated case data: the rts2 package for R*, 2024, https://arxiv.org/abs/2403.09448. preprint.
[53] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2, MIT press, Cambridge, 2006.
[54] N. Yannaros, *Weibull renewal processes*, Ann. Inst. Statist. Math., 46 (1994), pp. 641–648.