BART-based inference for Poisson processes

Stamatina Lamprinakou *, Mauricio Barahona, Seth Flaxman, Sarah Filippi, Axel Gandy, Emma J. McCoy

Department of Mathematics, Imperial College London, London, United Kingdom

A B S T R A C T

The effectiveness of Bayesian Additive Regression Trees (BART) has been demonstrated in a variety of contexts including non-parametric regression and classification. A BART scheme for estimating the intensity of inhomogeneous Poisson processes is introduced. Poisson intensity estimation is a vital task in various applications including medical imaging, astrophysics and network traffic analysis. The new approach enables full posterior inference of the intensity in a non-parametric regression setting. The performance of the novel scheme is demonstrated through simulation studies on synthetic and real datasets up to five dimensions, and the new scheme is compared with alternative approaches.

© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

The Bayesian Additive Regression Trees (BART) model is a Bayesian framework, which uses a sum of trees to predict the posterior distribution of a response y given a p-dimensional covariate X and priors on the function relating the covariates to the response. Chipman et al. (2010) proposed an inference procedure using Metropolis Hastings within a Gibbs Sampler, whereas Lakshminarayanan et al. (2015) used a Particle Gibbs Sampler to increase mixing when the true posterior consists of deep trees or when the dimensionality of the data is high. Several theoretical studies of BART models (Rockova and van der Pas, 2017; Rockova and Saha, 2018; Linero and Yang, 2018) have recently established optimal posterior convergence rates. The BART model has been applied in various contexts including non-parametric mean regression (Chipman et al., 2010), classification (Chipman et al., 2010; Zhang and Härdle, 2010; Kindo et al., 2016), variable selection (Chipman et al., 2010; Bleich et al., 2014; Linero, 2018), estimation of monotone functions (Chipman et al., 2021), causal inference (Hill, 2011), survival analysis (Sparapani et al., 2016), and heteroskedasticity (Bleich and Kapelner, 2014; Pratola et al., 2016). Linero and Yang (2018) illustrated how the BART model suffers from a lack of smoothness and the curse of dimensionality, and overcome both potential shortcomings by considering a sparsity assumption similar to (Linero, 2018) and treating decisions at branches probabilistically.

The original BART model (Chipman et al., 2010) assumes that the response has a Gaussian distribution and the majority of applications have used this framework. Murray (2017) adapted the BART model to count data and categorical data via a log-linear transformation, and provided an efficient MCMC sampler. Our focus is on extending this methodology to estimate the intensity function of inhomogeneous Poisson processes.

---

* Corresponding author.
  E-mail address: s.lamprinakou18@imperial.ac.uk (S. Lamprinakou).

The question of estimating the intensity of Poisson processes has a long history, including both frequentist and Bayesian methods. Frequentist methods include fixed-bandwidth and adaptive bandwidth kernel estimators with edge correction (Diggle et al., 2003), and wavelet-based methods (e.g. Fryzlewicz and Nason, 2004; Patil et al., 2004). Bayesian methods include using a sigmoidal Gaussian Cox process model for intensity inference (Adams et al., 2009), a Markov random field (MRF) with Laplace prior (Sardy and Tseng, 2004), variational Bayesian intensity inference (Lloyd et al., 2015), and non-parametric Bayesian estimations of the intensity via piecewise functions with either random or fixed partitions of constant intensity (Arjas and Gasbarra, 1994; Heikkinen and Arjas, 1998; Gugushvili et al., 2018).

In this paper, we introduce an extension of the BART model (Chipman et al., 2010) for Poisson Processes whose intensity at each point is estimated via a tiny ensemble of trees. Specifically, the logarithm of the intensity at each point is modelled via a sum of trees (and hence the intensity is a product of trees). This approach enables full posterior inference of the intensity in a non-parametric regression setting. Our main contribution is a novel BART scheme for estimating the intensity of an inhomogeneous Poisson process. The simulation studies demonstrate that our algorithm is competitive with the Haar-Fisz algorithm in one dimension, kernel smoothing in two dimensions, and outperforms the kernel approach for multidimensional intensities. The simulation analysis also demonstrates that our proposed algorithm is competitive with the inference via spatial log-Gaussian Cox processes. We also demonstrate its ability to track varying intensity in synthetic and real data.

The outline of the article is as follows. Section 2 introduces our approach for estimating the intensity of a Poisson process through the BART model, and Section 3 presents the proposed inference algorithm. Sections 4 and 5 present the application of the algorithm to synthetic data and real data sets, respectively. Section 6 provides our conclusions and plans for future work.

2. The BART model for Poisson processes

Consider an inhomogeneous Poisson process defined on a d-dimensional domain $S \subset \mathbb{R}^d$, $d \geq 1$, with intensity $\lambda : S \to \mathbb{R}^+$. For such a process, the number of points within a subregion $B \subset S$ has a Poisson distribution with mean $\lambda_B = \int_B \lambda(s)\, ds$, and the number of points in disjoint subregions are independent (Daley and Vere-Jones, 2003). The homogeneous Poisson process is a special case with constant intensity $\lambda(s) = \lambda_0$, $\forall s \in S$.

To estimate the intensity of the inhomogeneous Poisson process, we use m partitions of the domain S, each associated with a tree T_h, $h = 1, \ldots, m$. The partitions are denoted $T_h = \{\Omega_{ht}\}_{t=1}^{b_h}$, where b_h is the number of terminal nodes in the corresponding tree T_h, and each leaf node t corresponds to one of the subregions Ω_{ht} of the partition T_h. Being a partition, every tree covers the full domain, i.e. $S = \cup_{t=1}^{b_h} \Omega_{ht}$ for every h. Each subregion Ω_{ht} has an associated parameter λ_{ht}, and hence each tree T_h has an associated vector of leaf intensities $\Lambda_h = (\lambda_{h1}, \lambda_{h2}, .., \lambda_{hb_h})$.

We model the intensity of $s \in S$ as:

$$\log(\lambda(s)) = \sum_{h=1}^{m} \sum_{t=1}^{b_h} \log(\lambda_{ht})\, I(s \in \Omega_{ht}) \tag{1}$$

$$T_h \sim \text{heterogeneous Galton-Watson process for a partition of } S \tag{2}$$

$$\lambda_{ht} | T_h \overset{\text{iid}}{\sim} \text{Gamma}(\alpha, \beta) \tag{3}$$

where $I(\cdot)$ denotes the indicator function. Equivalently, (1) can be expressed as

$$\lambda(s) = \prod_{h=1}^{m} \prod_{t=1}^{b_h} \lambda_{ht}^{I(s \in \Omega_{ht})}. \tag{4}$$

Given a fixed number of trees, m, the parameters of the model are thus the regression trees $T = \{T_h\}_{h=1}^{m}$ and their corresponding intensities $\Lambda = \{\Lambda_h\}_{h=1}^{m}$. Following Chipman et al. (2010), we assume that the tree components (T_h, Λ_h) are independent of each other, and that the terminal node parameters of every tree are independent, so that the prior can be factorized as:

$$P(\Lambda, T) = \prod_{h=1}^{m} P(\Lambda_h, T_h) = \prod_{h=1}^{m} P(\Lambda_h | T_h) P(T_h) = \prod_{h=1}^{m} \left[\prod_{t=1}^{b_h} P(\lambda_{ht} | T_h) \right] P(T_h). \tag{5}$$

Prior on the trees The trees T_h of the BART model are stochastic regression trees generated through a heterogeneous Galton-Watson (GW) process (Harris et al., 1963; Rockova and Saha, 2018). The GW process is the simplest branching process concerning the evolution of a population in discrete time. Individuals (tree nodes) of a generation (tree depth) give birth to a random number of individuals (tree nodes), called offspring, mutually independent and all with the same offspring distribution that may vary from generation (depth) to generation (depth). In our case, we use the prior introduced by

Chipman et al. (1998), that is a GW process in which each node has either zero or two offspring and the probability of a node splitting depends on its depth in the tree. Specifically, a node $\eta \in T_h$ splits into two offsprings with probability

$$p_{\text{split}}(\eta) = \frac{\gamma}{(1 + d(\eta))^\delta}, \tag{6}$$

where $d(\eta)$ is the depth of node η in the tree, and $\gamma \in (0, 1)$ and $\delta \geq 0$ are parameters of the model. Classic results from the theory of branching processes show that $\gamma \leq 0.5$ guarantees that the expected depth of the tree is finite. In our construction, each tree T_h is associated with a partition of S. Namely, if node η splits, we select uniformly at random one of the d dimensions of the space of the Poisson process, followed by uniform selection from the available split values associated with that dimension respecting the splitting rules higher in the tree.

Prior on the leaf intensities   Our choice of a Gamma prior for the leaf parameters λ_{ht} builds upon previous work by Murray (2017), who used a mixture of Generalized Inverse Gaussian (GIG) distributions as the prior on leaf parameters in a BART model for count regression. Here we impose a Gamma prior (a special case of GIG) on the leaf parameters, which simplifies the model and leads to a closed form of the conditional integrated likelihood below (see Section 3) as the Gamma distribution is the conjugate prior for the Poisson likelihood. We discuss the selection of it hyperparameters α and β in Section 3.1.

3. The inference algorithm

Given a finite realization of an inhomogeneous Poisson process with n sample points $\boldsymbol{s} = s_1, \ldots, s_n \in S \subset \mathbb{R}^d$, we seek to infer the parameters of the model (Λ, T) by sampling from the posterior $P(\Lambda, T | \boldsymbol{s})$.

Before presenting the sampling algorithm we summarize a preliminary result. To simplify our notation, let us define

$$g(s_i; T_h, \Lambda_h) = \prod_{t=1}^{b_h} \lambda_{ht}^{I(s_i \in \Omega_{ht})},$$

so that Eq. (4) becomes $\lambda(s_i) = \prod_{h=1}^{m} g(s_i; T_h, \Lambda_h)$.

Let us choose any arbitrary tree T_h in our ensemble T, and let us denote the set with the rest of the trees as $T_{(h)} = \{T_j\}_{j=1, j \neq h}^m$ and their leaf parameters as $\Lambda_{(h)} = \{\Lambda_j\}_{j=1, j \neq h}^m$. The intersection of all the partitions associated with the trees in $T_{(h)}$ gives us a global partition $\{\overline{\Omega}_k^{(h)}\}_{k=1}^{K(T_{(h)})}$ with $K(T_{(h)})$ subregions (Rockova and van der Pas, 2017).

Then we have the following result.

Remark 1.

(i) The conditional likelihood of the realization is given by

$$P(\boldsymbol{s} | \Lambda, T) = c_h \prod_{t=1}^{b_h} \lambda_{ht}^{n_{ht}} e^{-\lambda_{ht} c_{ht}}, \tag{7}$$

$$\text{with} \quad c_h = \prod_{i=1}^{n} \prod_{j=1, j \neq h}^{m} g(s_i; T_j, \Lambda_j),$$

$$c_{ht} = \sum_{k=1}^{K(T_{(h)})} \overline{\lambda}_k^{(h)} |\overline{\Omega}_k^{(h)} \cap \Omega_{ht}|,$$

where $\overline{\lambda}_k^{(h)} = \prod_{t=1, t \neq h}^{m} \prod_{l=1}^{b_t} \lambda_{tl}^{I(\Omega_{tl} \cap \overline{\Omega}_k^{(h)} \neq 0)}$, n_{ht} is the cardinality of the set $\{i : s_i \in \Omega_{ht}\}$, and $|\overline{\Omega}_k^{(h)} \cap \Omega_{ht}|$ is the volume of the region $\overline{\Omega}_k^{(h)} \cap \Omega_{ht}$.

(ii) For a tree h, the conditional integrated likelihood obtained by integrating out Λ_h is

$$P(\boldsymbol{s} | T_h, T_{(h)}, \Lambda_{(h)}) = c_h \left(\frac{\beta^\alpha}{\Gamma(\alpha)} \right)^{b_h} \prod_{t=1}^{b_h} \frac{\Gamma(n_{ht} + \alpha)}{(c_{ht} + \beta)^{n_{ht} + \alpha}}. \tag{8}$$

A proof can be found in Appendix B and Appendix C.

We now summarize our sampling algorithm. To sample from $P(\Lambda, T | \boldsymbol{s})$, we implement a Metropolis-Hastings within block Gibbs sampler (Algorithm 1), which requires m successive draws from $(T_h, \Lambda_h) | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}$. Note that

$$P(T_h, \Lambda_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) = P(T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) \, P(\Lambda_h | T_h, T_{(h)}, \Lambda_{(h)}, \boldsymbol{s})$$

$$\propto P(T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) \, P(\boldsymbol{s} | \Lambda, T) P(\Lambda_h | T_h)$$

$$= P(T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) \, P(\boldsymbol{s} | \Lambda, T) \prod_{t=1}^{b_h} P(\lambda_{ht} | T_h)$$

$$= P(T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) \, c_h \prod_{t=1}^{b_h} \lambda_{ht}^{n_{ht}} e^{-\lambda_{ht} c_{ht}} \prod_{t=1}^{b_h} \frac{\beta^{\alpha}}{\Gamma(\alpha)} \lambda_{ht}^{\alpha-1} e^{-\beta \lambda_{ht}}$$

$$\propto P(T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) \prod_{t=1}^{b_h} \lambda_{ht}^{n_{ht}+\alpha-1} e^{-(c_{ht}+\beta)\lambda_{ht}} \qquad (9)$$

which follows directly from Bayes' rule and Eqs. (5) and (3).

From (9), it is clear that a draw from $(T_h, \Lambda_h) | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}$ can be achieved in (b_h+1) successive steps consisting of:

- sampling $T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}$ using Metropolis-Hastings (Algorithm 2)
- sampling $\lambda_{ht} | T_h, T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}$ from a Gamma distribution with shape $n_{ht} + \alpha$ and rate $c_{ht} + \beta$ for $t = 1, .., b_h$.

These steps are implemented through Metropolis-Hastings in Algorithm 1. Note also that

$$P(T_h | T_{(h)}, \Lambda_{(h)}, \boldsymbol{s}) \propto P(\boldsymbol{s} | T_h, T_{(h)}, \Lambda_{(h)}) \, P(T_h),$$

so that the conditional integrated likelihood (8) is required to compute the Hastings ratio.

Algorithm 1 Metropolis-Hastings within Gibbs sampler.

---

for $v = 1, 2, 3, ..$ **do**
  for $h = 1$ to m **do**
    Sample $T_h^{(v+1)} | \boldsymbol{s}, \{T_j^{(v+1)}\}_{j=1}^{h-1}, \{T_j^{(v)}\}_{j=h+1}^{m}, \{\Lambda_j^{(v+1)}\}_{j=1}^{h-1}, \{\Lambda_j^{(v)}\}_{j=h+1}^{m}$
      using Algorithm 2
    for $t = 1$ to b_h **do**
      Sample $\lambda_{ht}^{(v+1)} | \boldsymbol{s}, \{T_j^{(v+1)}\}_{j=1}^{h}, \{T_j^{(v)}\}_{j=h+1}^{m}, \{\Lambda_j^{(v+1)}\}_{j=1}^{h-1}, \{\Lambda_j^{(v)}\}_{j=h+1}^{m}$ from Gamma$(n_{ht} + \alpha, c_{ht} + \beta)$
    end for
  end for
end for

Algorithm 2 Metropolis-Hastings Algorithm for sampling from the posterior $P(T_j | \boldsymbol{s}, T_{(j)}, \Lambda_{(j)})$.

Generate a candidate value T_j^* with probability $q(T_j^* | T_j^{(v)})$.

Set $T_j^{(v+1)} = T_j^*$ with probability

$$\alpha(T_j^{(v)}, T_j^*) = \min \left\{ 1, \frac{q(T_j^{(v)} | T_j^*)}{q(T_j^* | T_j^{(v)})} \frac{P(\boldsymbol{s} | T_j^*, T_{(j)}, \Lambda_{(j)})}{P(\boldsymbol{s} | T_j^{(v)}, T_{(j)}, \Lambda_{(j)})} \frac{P(T_j^*)}{P(T_j^{(v)})} \right\}$$

Otherwise, set $T_j^{(v+1)} = T_j^{(v)}$.

The transition kernel q in Algorithm 2 is chosen from the three proposals: GROW, PRUNE, CHANGE (Chipman et al., 2010; Kapelner and Bleich, 2013). The GROW proposal randomly picks a terminal node, splits the chosen terminal into two new nodes and assigns a decision rule to it. The PRUNE proposal randomly picks a parent of two terminal nodes and turns it into a terminal node by collapsing the nodes below it. The CHANGE proposal randomly picks an internal node and randomly reassigns to it a splitting rule. We describe the implementation of the proposals in Appendix A.

For completeness, in the supplementary material, we present the full development of the algorithm for inference of the intensity of inhomogeneous Poisson processes via only one tree.

3.1. Fixing the hyperparameters of the model

Hyperparameters of the Gamma distribution for the leaf intensities We use a simple data-informed approach to fix the hyperparameters α and β of the Gamma distribution (3). We discretize the domain into N_G subregions of equal volume ($N_G = (\lceil 100^{1/d} \rceil)^d$ works well in practice up to 5 dimensions) and count the number of samples s_i per subregion. We thus obtain the empirical densities in each of the subregions: ξ_i, $i = 1, \ldots, N_G$. Given the form of the intensity (4) as a product of m trees, we consider the m-th roots $\Xi = \{\xi_i^{1/m}\}_{i=1}^{N_G}$ as candidates for the intensity of each tree. Taking the

sample mean $\widehat{\mu}_\Xi$ and sample variance $\widehat{\sigma}_\Xi^2$, we choose the model hyperparameters α and β to correspond to those of a Gamma distribution with the same mean and variance, i.e., $\alpha = \widehat{\mu}_\Xi^2 / \widehat{\sigma}_\Xi^2$ and $\beta = \widehat{\mu}_\Xi / \widehat{\sigma}_\Xi^2$, although fixing $\beta = 1$ can also give good estimates of the intensity. Although setting $N_G = (\lceil 100^{1/d} \rceil)^d$ leads to convergence and good estimates of the intensity in our simulation studies below, there are other possibilities. Alternatively, we can bin the data based on a criterion that takes into account the number of samples, n, and the number of dimensions, d. For example, the number of bins per dimension, n_b, can be computed as (Scott, 2008; Wand, 1997): (i) $n_b = \lceil n^{1/(d+1)} \rceil$, (ii) $n_b = \lceil n^{1/(d+2)} \rceil$, or (iii) $n_b = \max_{k \in \{1,2,\dots,d\}}[\lceil DR_k \cdot n^{1/(d+2)} / (2 \cdot \mathrm{IQR}(\{s_{i,k}\})) \rceil]$, where IQR denotes the interquartile range of the sample, DR_k is the range of the domain in dimension k (here we scale the initial domain to a unit hypercube so that $DR_k = 1$, $\forall k$), and by extension $N_G = n_b^d$. In our simulation scenarios below, all these approaches lead to comparable convergence times and estimates of the intensity.

Hyperparameters of the stochastic ensemble of regression trees The GW stochastic process that generates our tree ensemble has several hyperparameters. The parameters (γ, δ) control the shape of trees. The parameter $\gamma > 0$ controls the probability that the root of a tree will split into two offspring, while the parameter $\delta > 0$ penalizes against deep trees. As noted in (Chipman et al., 2010), for a sum-of-trees model, we want to keep the depth of the tree small whilst ensuring non-trivial trees, hence, in our simulation study we fix $\gamma = 0.98$ and $\delta = 2$. Second, each of the d dimensions has to be assigned a grid of split values, from which the subregions of the partition are randomly chosen, yet always respecting the consistency of the ancestors in the tree (that is respecting the splitting rules higher in the tree). Here, we use a simple uniform grid for each of the d-dimensions (Pratola et al., 2016): we normalize each dimension of the space from (0,1) and discretize each dimension into N_d segments. ($N_d = 100$ works well in practice and is used throughout our examples.) More sophisticated, data-informed grids are also possible, although using, e.g., the sample points as split values does not improve noticeably the performance in our examples. Finally, the number of trees m also needs to be fixed as in Chipman et al. (2010). In our examples below, we have checked the performance of our algorithm with varying number of trees m between 2 and 50. We find that good performance can be achieved with a moderate number of trees, m, between 3 and 10 depending on the particular example.

4. Simulation study on synthetic data

We carried out a simulation study on synthetic data to illustrate the performance of Algorithm 1 to estimate first the intensity of one dimensional and two dimensional inhomogeneous Poisson processes and finally the intensity of multidimensional Poisson processes.

We simulate realizations of Poisson processes on the domain $[0, 1)^d$ for $d \in \{1, 2, 3, 4, 5\}$ via thinning (Lewis and Shedler, 1979). The hyperparameters of the model (for the trees and the leaf intensities) are fixed as described in Section 3.1. We initially randomly generate m trees of zero depth. The probabilities of the proposals in Algorithm 2 are set to: $P(\mathrm{GROW}) = P(\mathrm{PRUNE}) = 0.4$ and $P(\mathrm{CHANGE}) = 0.2$. A set $\{z_i\}$ is defined by uniformly sampling points in the domain $[0, 1)^d$.

We run 3 parallel chains of the same length. We discard their first halves treating the second halves as a sample from the target distribution. We assess chain convergence using the Gelman-Rubin convergence diagnostic (Gelman et al., 1992) applied to the estimated intensity for each point of the set $\{z_i\}$, as well as trace plots and autocorrelation plots for some points of the testing set.

At each state t of a simulated chain we estimate the intensity for each point z_i by a product of trees denoted as

$$\widehat{\lambda}^{(t)}(z_i) = \prod_{j=1}^{m} g(z_i; T_j^{(t)}, \Lambda_j^{(t)}).$$

The induced sequence $\{\widehat{\lambda}^{(t)}(\cdot)\}_{t=1}^{\infty}$ for the sequence of draws $\{(T_1^{(t)}, \Lambda_1^{(t)}), \dots, (T_m^{(t)}, \Lambda_m^{(t)})\}_{t=1}^{\infty}$ converges to $P(\widehat{\lambda}|\boldsymbol{s})$. We estimate the posterior mean $E[\widehat{\lambda}(\cdot)|s_1, \dots s_n]$, the posterior median of $\widehat{\lambda}(\cdot)$, and the highest density interval (hdi) using the function *hdi* provided by the **R** package **bayestestR** (Makowski et al., 2019). To assess the performance of our algorithm, we compute the Average Absolute Error (AAE) of the computed estimate:

$$\mathrm{AAE}(\widehat{\lambda}) = \frac{1}{N_z} \sum_{i=1}^{N_z} |\widehat{\lambda}(z_i) - \lambda(z_i)| \tag{10}$$

and the Root Integrated Square Error (RISE):

$$\mathrm{RISE}(\widehat{\lambda}) = \left(\frac{1}{N_z} \sum_{i=1}^{N_z} (\widehat{\lambda}(z_i) - \lambda(z_i))^2 \right)^{1/2} \tag{11}$$

where N_z is the number of test points.

In the spirit of Akaike information criterion (AIC) (Loader, 1999), we also introduce two diagnostics targeting the likelihood function to evaluate if increasing the number of trees leads to better intensity estimation:

$$D_g = 2 \left(\log P(s_1, .., s_n) - k_g \right),$$

and

$$D_l = 2 \left(\log P(s_1, .., s_n) - k_l \right),$$

where k_g is the number of global cells, and k_l is the overall number of leaves in the ensemble. We estimate both diagnostics using the sequence of the draws $(T^{(w)}, \Lambda^{(w)}) = \left\{ \left(T_1^{(w)}, \Lambda_1^{(w)} \right), ..., \left(T_m^{(w)}, \Lambda_m^{(w)} \right) \right\}$ after the burn-in period as

$$D_g \approx 2 \frac{1}{N_w} \sum_{w=1}^{N_w} \left(\log P \left(s_1, .., s_n | \left(T^{(w)}, \Lambda^{(w)} \right) \right) - k_g^{(w)} \right),$$

and

$$D_l \approx 2 \frac{1}{N_w} \sum_{w=1}^{N_w} \left(\log P \left(s_1, .., s_n | \left(T^{(w)}, \Lambda^{(w)} \right) \right) - k_l^{(w)} \right),$$

where $k_g^{(w)}$ and $k_l^{(w)}$ are the number of global cells and the overall number of leaves in the ensemble associated to the w_{th} draw, respectively.

AIC has been shown to be asymptotically equal to leave-one-out cross validation (LOO-CV) (Stone, 1977; Gelman et al., 2014). According to Leininger and Gelfand (2017), the computational burden required for leave-one-out cross validation considering a point pattern data is impractical. We introduce a leave-partition-out (LPO) method, assuming that the initial process $N(t)$ is obtained by combining independent processes $\{N_i(t)\}_{i=1}^{N_p}$, as follows

$$D_{LPO} = \sum_{i=1}^{N_P} \log P \left(N_i(t) | N(t) - \{N_i(t)\} \right) \tag{12}$$

where $P \left(N_i(t) | N(t) - \{N_i(t)\} \right)$ is the leave-partition-out predictive intensity given the process $N(t)$ without the i_{th} partition, $N_i(t)$. We can evaluate (12) as follows,

$$D_{LPO} = \sum_{i=1}^{N_P} \log \left(\frac{1}{N_w} \sum_{w=1}^{N_w} P \left(N_i(t) | \left(T^{(w,i)}, \Lambda^{(w,i)} \right) \right) \right)$$

where $(T^{(w,i)}, \Lambda^{(w,i)})$ is the sequence of draws $\left\{ \left(T_1^{(w,i)}, \Lambda_1^{(w,i)} \right), ..., \left(T_m^{(w,i)}, \Lambda_m^{(w,i)} \right) \right\}$ after the burn-in period leaving out the partition $N_i(t)$. We assume that each event of $N(t)$ is coming from $N_i(t)$ with probability p_i. The bias of the method is introduced by randomly splitting the process into individual processes. We can get the LOO-CV by LPO, defining appropriately the parameter N_p. As higher the number N_p is, as less biased the method is. In the simulation scenarios, we consider that $p_i = 0.1$, $i = 1, ..., N_p$ and $N_p = 10$ for computational reasons. The diagnostics show that tiny ensembles of trees provide good estimates in our simulation scenarios.

To confirm the proposed diagnostics, we use p-thinning (Illian et al., 2008, Chapter 6) with $p = 0.8$ to create training and test datasets in two of the simulation scenarios. We employ Root Standardized Mean Square Error (RSMSE) and Rank Probability Score (RPS) with the test data set comparing observed counts in disjoint equal volume subregions $\{S_i\}_{i=1}^{N_s}$ as follows:

$$\text{RSMSE}(\widehat{N}) = \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \frac{(\widehat{N}(S_i) - N(S_i))^2}{\widehat{N}(S_i)} \right)^{1/2} \tag{13}$$

and

$$\text{RPS}(N(S_j)) = \sum_{u=0}^{N(S_j)-1} F(u)^2 + \sum_{u=N(S_j)}^{\infty} (F(u) - 1)^2, \tag{14}$$

where F is the Poisson distribution with parameter $m = \int_{S_j} \hat{\lambda}(s) ds$, $N(S_i)$ the actual number of testing points in S_i and $\widehat{N}(S_i)$ the estimated number of testing points in S_i given by

$$\widehat{N}(S_i) = \int_{S_i} \frac{1-p}{p} \widehat{\lambda}(s) ds \simeq \frac{1-p}{p} \frac{1}{N_z^i} \sum_{z_j \in S_i} \widehat{\lambda}(z_j) |S_i| \tag{15}$$
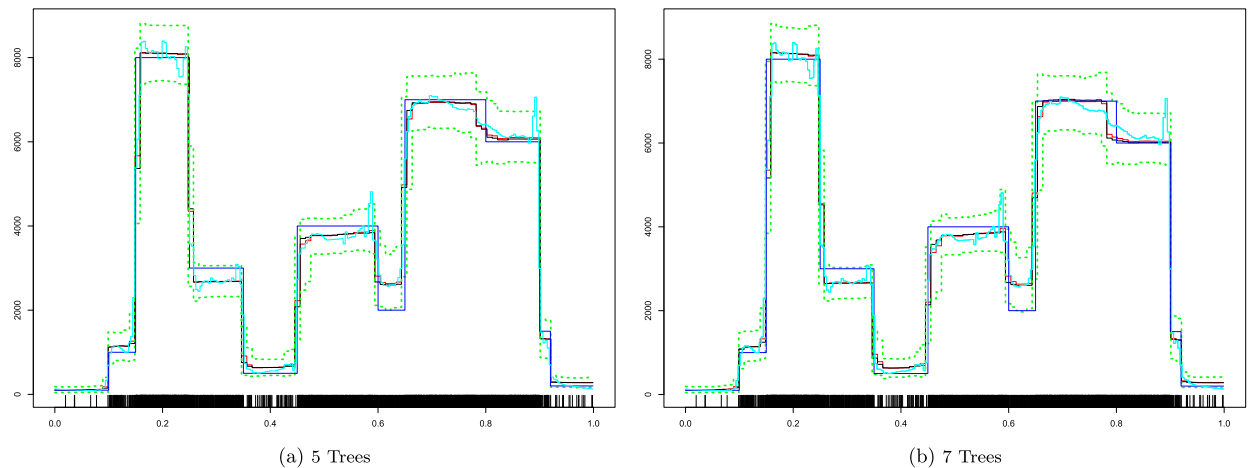
(a) 5 Trees   (b) 7 Trees

Fig. 1. The original intensity (blue curve), the posterior mean (red curve), the posterior median (black curve), the 95% hdi interval of the estimated intensity illustrated by the dotted green lines and the Haar-Fisz estimator (cyan curve). The rug plot on the bottom displays the 3590 event times.

with N_z^i being the number of points $\{z_j\}$ falling in S_i and estimating the intensity at each points s, $\widehat{\lambda}(s)$, via the posterior mean $E[\widehat{\lambda}(\cdot)|s_1, ..s_n]$.

For one dimensional processes, we compare the results of Algorithm 1 to the Haar-Fisz algorithm (Fryzlewicz and Nason, 2004), a wavelet based method for estimating the intensity of one dimensional Poisson Processes that outperforms well known competitors. We apply the Haar-Fisz algorithm to the counts of points falling into 256 consecutive intervals using the **R** package **haarfisz** (Fryzlewicz, 2010). Our algorithm is competitive with the Haar algorithm for smooth intensity functions and is not strongly out-performed by the Haar-Fisz algorithm when the underlying intensity is a stepwise function.

For two-dimensional processes, we compare the results of our algorithm with fixed-bandwidth estimators and log-Gaussian Cox processes (LGCP) with intensity $\lambda(\boldsymbol{s}) = \exp(a + u(\boldsymbol{s}))$ where u is a Gaussian process with exponential co-variance function. We used a discretization version of the LGCP model defined on a regular grid over space which we implemented using Stan-code (Gelman et al., 2015). As noted in Davies and Baddeley (2018), the choice of the kernel is not of primary importance, we choose a Gaussian kernel for its wide applicability. In our tables of results, the smoothing bandwidth, sigma, selected using likelihood cross-validation (Loader, 1999) denoted by (LCV), and we have also included other values of sigma to demonstrate the sensitivity to bandwidth choice. The kernel estimators, and the bandwidth value given by likelihood cross-validation, were computed using the **R** package **spatstat** (Baddeley and Turner, 2005). Our algorithm outperforms the maximum likelihood approach using linear conditional intensity, as expected. Our algorithm outperforms kernel smoothing and LGCP for stepwise functions and is competitive with them for a smooth intensity.

Finally, we examine the performance of our algorithm for multidimensional intensities by generating realizations of Poisson Processes on the domain $[0, 1)^d$ for $d \in \{3, 5\}$ via thinning. Future work includes the study of intensities in higher dimensions ($d > 5$). We compare our intensity estimates with kernel smoothing estimators having isotropic standard deviation matrices with diagonal elements equal to h and the methodology for applying maximum likelihood to point process models with linear conditional intensity (Peng, 2003). We select the bandwidth h using likelihood cross-validation (Loader, 1999) denoted by (LCV).

4.1. One dimensional Poisson process with stepwise intensity

Our first example is a one dimensional Poisson Process with piecewise constant intensity with several steps (Fig. 1). We run 3 parallel chains of the same length for 200000 iterations for 2-10 trees, 100000 for 12 trees, 50000 iterations for 15 trees and 30000 iterations for 20 trees.

Our algorithm detects the change points and provides good estimates of the intensity and is competitive in terms of AAE with the Haar-Fisz algorithm, but does not perform as well in terms of RISE (see Fig. 1 and Tables 3-6). We have found the metrics and convergence diagnostics in a set of uniformly chosen points without excluding the points close to jumps. Due to inferring the intensity via a product of stepwise functions, it is expected that the proposed algorithm will provide estimates with higher variability close to jumps. The proposed algorithm outperforms the Haar-Fisz algorithm without considering the points close to jumps. Tables 4-5 show the metrics for various number of trees without considering the points in a distance $= \pm 0.02$ from the jumps.

The diagnostics D_g, D_l and D_{LPO} obtain their highest values for 7, 4 and 8 trees, respectively. The analysis demonstrates only small differences between log-likelihood values as the number of trees increases, supporting results found in previous BART studies that the method is robust to the choice of m. The average RSMSE and RPS on testing points over 7 different splits of the original data set (Tables 1-2) provide evidence that ensembles with more than seven trees do not improve the fit of the proposed algorithm.

Table 1

The average RPS on testing points over 7 different splits of the original data set in Fig. 1.

Proposed BART Algorithm						
Number of trees	$N_s = 1$	$N_s = 10$	$N_s = 25$	$N_s = 50$	$N_s = 75$	$N_s = 75$
2	17.05	5.24	3.10	2.08	1.66	1.44
3	17.12	5.25	3.11	2.08	1.65	1.43
4	16.98	5.28	3.09	2.07	1.64	1.42
5	16.94	5.26	3.04	2.06	1.63	1.41
7	17.01	5.22	3.00	2.04	1.62	1.40
8	17.09	5.20	2.99	2.03	1.62	1.40
9	17.07	5.20	2.98	2.02	1.61	1.39
10	20.10	5.78	3.12	2.12	1.63	1.43
12	17.74	5.37	2.99	2.06	1.63	1.39
15	17.03	5.15	2.94	2.01	1.60	1.38
20	17.08	5.16	2.93	2.00	1.60	1.38

Table 2

The average RSMSE on testing points over 7 different splits of the original data set in Fig. 1.

Proposed BART Algorithm						
Number of trees	$N_s = 1$	$N_s = 10$	$N_s = 25$	$N_s = 50$	$N_s = 75$	$N_s = 75$
2	0.95	1.13	1.06	1.02	0.99	1.00
3	0.95	1.13	1.06	1.02	0.98	0.99
4	0.95	1.14	1.06	1.02	0.98	0.98
5	0.94	1.13	1.04	1.02	0.98	0.98
7	0.95	1.13	1.04	1.01	0.97	0.97
8	0.95	1.12	1.03	1.01	0.97	0.96
9	0.95	1.13	1.03	1.01	0.97	0.97
10	1.10	1.20	1.06	1.04	0.98	0.98
12	0.98	1.18	1.04	1.02	0.98	0.96
15	0.95	1.12	1.02	1.00	0.97	0.96
20	0.95	1.12	1.02	1.00	0.97	0.96

Table 3

Average Absolute Error and Root Integrated Square Error for various number of trees for the data in Fig. 1.

Proposed BART Algorithm							
Number of trees	AAE for Posterior Mean	AAE for Posterior Median	RISE for Posterior Mean	RISE for Posterior Median	D_g	D_l	D_{LPO}
3	308.87	320.84	603.54	633.48	54095.1	54090	-339.7
4	287.89	283.03	580.69	587.08	54096.5	54090	-368
5	289.27	281.13	580.55	586.24	54098	54088.4	-352.5
7	281.59	274.88	588.7	592.11	54098	54082.7	-263.5
8	280.62	274.07	588.73	591.29	54097.9	54079.5	-261.5
9	282.78	276.99	593.93	595.23	54096.9	54075.2	-327.9
10	283.79	279.07	593.95	595.41	54095.6	54071.6	-322.6
20	297.21	287.86	599.77	595.04	54082.9	54029.7	-436

Table 4

Average Absolute Error and Root Integrated Square Error for the data in Fig. 1 without considering points close to steps.

Proposed BART Algorithm				
Number of trees	AAE for Posterior Mean	AAE for Posterior Median	RISE for Posterior Mean	RISE for Posterior Median
4	144.48	139.58	181.21	174.82
5	144.55	139.02	180.74	176.19
7	124.53	123.2	175.74	172.4

Table 5
Average Absolute Error and Root Integrated Square Error for Haar-Fisz estimator for the data in Fig. 1 without considering points close to steps.

Haar-Fisz Algorithm	
AAE	RISE
141.95	192.6

Table 6
Average Absolute Error and Root Integrated Square Error for Haar-Fisz estimator for the data in Fig. 1.

Haar-Fisz Algorithm	
AAE	RISE
272.3	476.9

Table 7
Average Absolute Error, Root Integrated Square Error and diagnostics for various trees for the data in Fig. 2.

Proposed BART Algorithm							
Number of trees	AAE for Posterior Mean	AAE for Posterior Median	RISE for Posterior Mean	RISE for Posterior Median	D_g	D_l	D_{LPO}
3	224.1	230.3	419.2	453.7	87227.2	87232.3	505.1
4	208.7	213	410.2	447.9	87223.7	87230.5	491.2
5	216.8	212.9	389.5	410.9	87211.6	87220.6	406
6	228.9	221.9	395.8	412.8	87197.5	87214.7	463.9

Table 8
Average Absolute Error and Root Integrated Square Error for fixed bandwidth estimators for the data in Fig. 2.

Kernel Smoothing		
Bandwidth (sigma)	AAE	RISE
0.027	763.8	1041.3
0.038	662.7	956.8
0.047 (LCV)	636.7	960.6
0.067	672.8	1042.5

4.2. Two-dimensional Poisson process with stepwise intensity function

To demonstrate the applicability of our algorithm in a two-dimensional setting, Figs. 2-3 and Tables 7-9 reveal that our algorithm outperforms kernel smoothing and inference with spatial log-Gaussian Cox processes for stepwise intensity functions. We run 3 parallel chains of the same length for 100000 iterations for 3-6 trees. The convergence criteria indicate convergence of the simulated chains for the majority of points. As may be expected, the simulation study shows that points close to jumps are estimated with less reliability. The algorithm converges less well at these points, as demonstrated by the Gelman-Rubin diagnostic (see supplementary material). The diagnostics D_g, D_l and D_{LPO} obtain their highest values for three trees, respectively. The diagnostics indicate that small ensembles of trees can provide a good estimate of the intensity.

4.3. Inhomogeneous three-dimensional Poisson process with Gaussian intensity

Our first example for multidimensional intensities is a three-dimensional Poisson process with intensity $\lambda(x) = 500e^{x^T x}$ for $x \in [0,1)^3$. We generated a realization of 1616 points via thinning. We run 3 parallel chains of the same length for 100000 iterations for 3-10 trees and 30000 iterations for 12 trees. Tables 10 and 11 illustrate the statistics of our algorithm and kernel smoothing. Figs. 4 and 5 show our estimators and the kernel estimator with h=0.073 for 8 Trees and 10 Trees with fixed third dimension (x[3]) at 0.4 and 0.8, respectively.
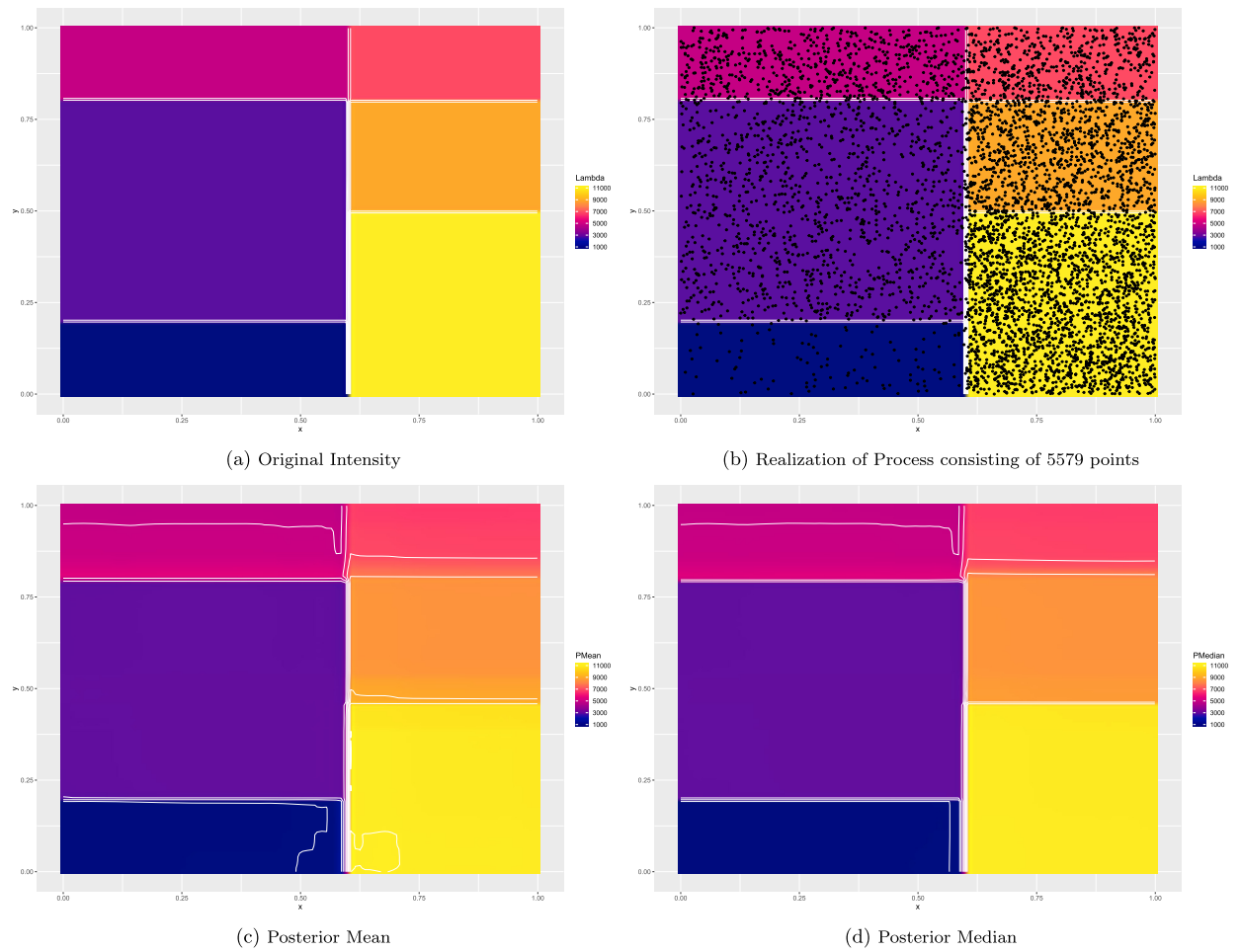
(a) Original Intensity

(b) Realization of Process consisting of 5579 points

(c) Posterior Mean

(d) Posterior Median

Fig. 2. Original Intensity, posterior mean and posterior median for 4 trees.

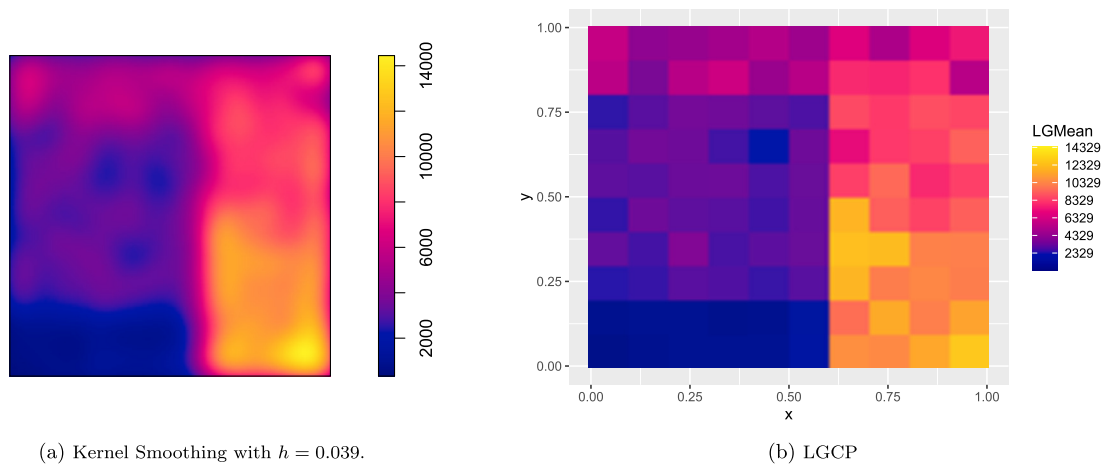(a) Kernel Smoothing with $h = 0.039$.

(b) LGCP

Fig. 3. Kernel estimator and inference with spatial log-Gaussian Cox processes.

The diagnostics D_g, D_l and D_{LPO} get their highest values with 4 trees, respectively. We observe that the diagnostic D_l slightly differs between 4 and 8 trees. The diagnostic D_g is similar between 4 and 5 trees. The estimate of the average logarithm of Poisson process likelihood does not change significantly from 4 trees to 12 trees. Specifically, we observe its maximum equal to 10536.3 at 12 trees, while its minimum to 10531.9 at 4 trees. In addition, the estimated average number of leaves in a tree of an ensemble is about 3 for $4 - 12$ trees. That explains why we observe higher values of diagnostics

Table 9
Average Absolute Error and Root Integrated Square Error with LGCP for the data in Fig. 2.

Inference with spatial log-Gaussian Cox processes		
grid	AAE	RISE
10×10	568	751
20×20	678	953

Table 10
Average Absolute Error, Root Integrated Square Error and diagnostics for various number of trees.

Proposed BART Algorithm							
Number of trees	AAE for Mean	AAE for Median	RISE for Mean	RISE for Median	D_g	D_l	D_{LPO}
4	247.6	254.9	360.7	376.3	20993.7	21040.6	-1409
5	250.2	258.3	364.3	380.4	20992.1	21039.8	-1492
6	247.6	254.7	360.8	375.4	20979.5	21038.6	-1529
8	234.8	239.4	341	352.4	20938.6	21032.8	-1515
10	226.8	229.4	330.5	338.5	20883	21026.9	-1539
12	221.6	222.3	320.6	326.4	20810.4	21020.7	-1609

Table 11
Average Absolute Error and Root Integrated Square Error for various isotropic variance matrices.

Kernel Smoothing		
h	AAE	RISE
0.053	480.8	667.5
0.073 (LCV)	415.86	645.16
0.08	417.7	661.4
0.085	423.2	676.2
0.1	450.3	727.6
0.3	890.4	1236

for a small number of trees. The metrics AAE and $RISE$ are optimised with 12 trees. However, it should be noted that only small variations in the metrics are seen between 4 and 12 trees. The diagnostics provide evidence that increasing the number of trees does not improve the fit of the proposed model.

4.4. Inhomogeneous five dimensional Poisson process with sparsity assumption

Here, we demonstrate the performance of our algorithm to detect the dimensions that contribute most in the intensity of $s \in \mathcal{S}$ in a noisy environment. Consider a five dimensional inhomogeneous Poisson process with intensity function of $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5) \in [0, 1]^5$ depending on 3 of 5 dimensions:

$$\lambda(\mathbf{x}) = (2 \mathbb{1}\,(x_1 < 0.2) + 10 \mathbb{1}\,(x_1 \geq 0.2)) * (3 \mathbb{1}\,(x_2 < 0.5) + 15 \mathbb{1}\,(x_2 \geq 0.5))$$
$$* (3 \mathbb{1}\,(x_3 < 0.8) + 30 \mathbb{1}\,(x_3 \geq 0.8))$$

We generate a realization of 669 points via thinning. We run 3 parallel chains of the same length for 100000 iterations for 4-8 trees, 50000 iterations for 10 trees, 30000 iterations for 12 trees and 10000 iterations for 15 trees. The convergence criterion is smaller than 1.1 for the majority of testing points.

Table 12 shows the metrics and diagnostics D_g and D_l of the estimated intensity over various numbers of trees. The diagnostics D_g and D_l obtain their highest values with 4 trees, and the diagnostic D_l shows only small differences between 4 and 5 trees. We note that (i) the average number of leaves in a tree of the ensemble is about 2.2 for 4-5 trees, and (ii) the estimated logarithm of Poisson process likelihood for 4 and 5 trees are 4271.5 and 4271.8, respectively. The diagnostic D_{LPO} gets its highest value with 5 trees. The p-thinning approach confirms the diagnostics, and indicates that increasing the number of trees does not improve the fit of the proposed model to the data. (See Tables 17 and 18.)

Table 16 demonstrates the frequency of times we meet each dimension in the decision rules of a tree. Table 15 shows how likely each dimension is to be involved in the root's decision rule. The results illustrate that the important covariates x_1, x_2 and x_3 are more likely to be involved in the decision rules of a tree than the noisy dimensions x_4 and x_5. That indicates the algorithm prioritizes the dimensions that contribute most to the intensity. Fig. 6 shows that the means of the posterior marginal intensities are similar to the expected marginal intensities given that $\{x_i\}_{i=1}^5$ are uniform independent covariates.
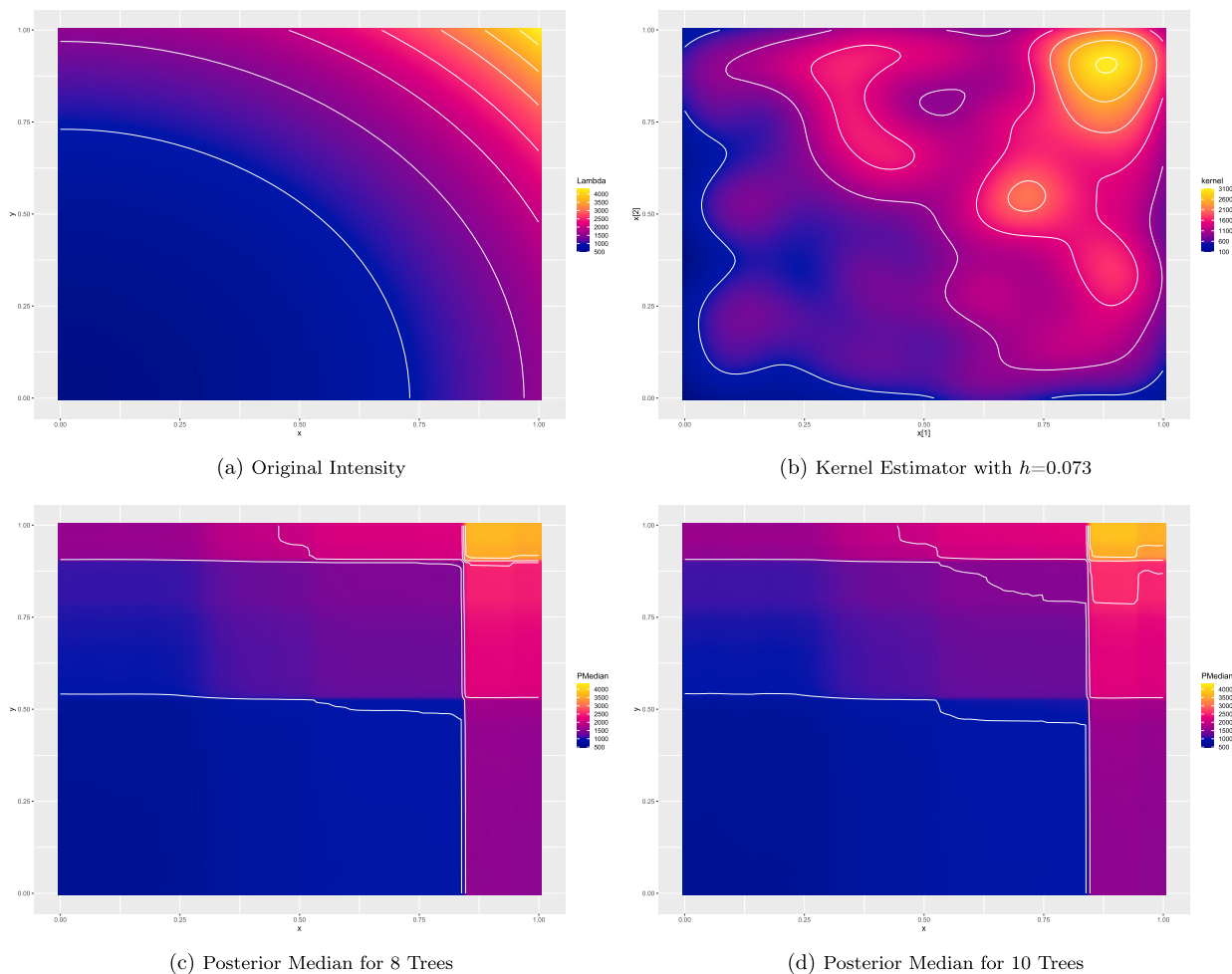
(a) Original Intensity

(b) Kernel Estimator with h=0.073

(c) Posterior Median for 8 Trees

(d) Posterior Median for 10 Trees

Fig. 4. Kernel estimator and Posterior Median for 8 and 10 Trees with x[3] = 0.4.

Table 12
Average Absolute Error, Root Integrated Square Error and diagnostics for various number of trees in the case of Inhomogeneous five dimensional Poisson Process with sparsity assumption.

Proposed BART Algorithm						
Number of trees	AAE for Mean	AAE for Median	RISE for Mean	RISE for Median	D_g	D_l	D_{LPO}
4	48.36	45.47	159.95	170.35	8510.1	8525.4	-485.9
5	49.18	44.54	158.82	169.07	8486.1	8520.9	-467.1
6	50.59	45.05	161.36	170.61	8462.1	8519	-477.4
8	56.06	47.94	162.56	164.46	8349	8511.8	-490.8
10	61.55	52.23	169.72	166.62	8141.8	8505.5	-503.1
12	67.01	57.06	180.53	175.23	7774.7	8499.8	-522.2
15	75	65.06	192.88	181.03	6813.1	8490	-500.2

Tables 12, 13 and 14 show that our algorithm outperforms kernel smoothing and the maximum likelihood approach considering linear conditional intensity as expected. The ability of our method to identify important features demonstrates an important advantage over other procedures.

5. Intensity estimation for real data

In this section, we first apply our algorithm to real data sets when modelled as realizations of inhomogeneous Poisson processes in one and two dimensions. To assess the performance of our algorithm, we break the domain $[0, 1)^d$ into equal
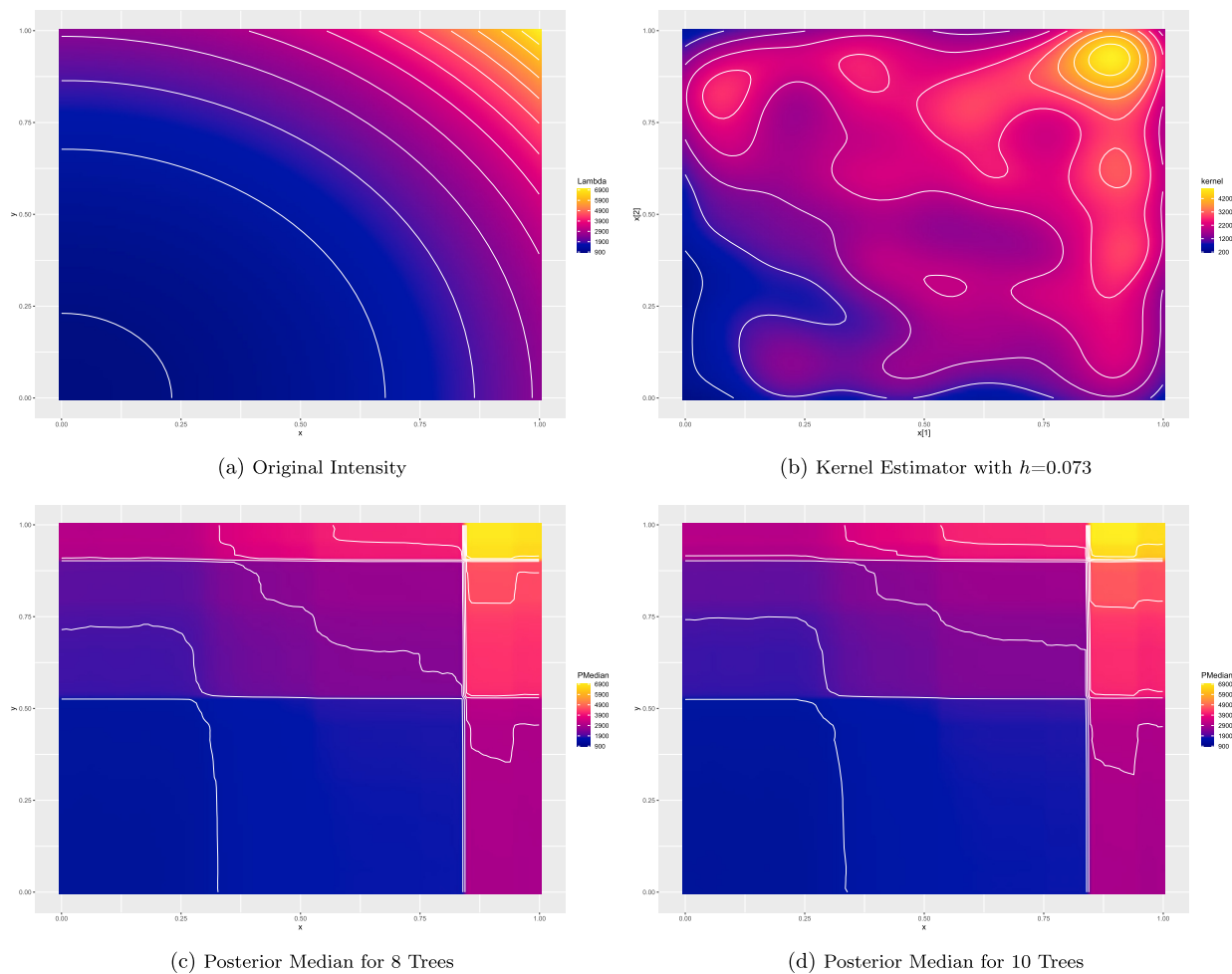
(a) Original Intensity

(b) Kernel Estimator with $h=0.073$

(c) Posterior Median for 8 Trees

(d) Posterior Median for 10 Trees

Fig. 5. Kernel Estimator and Posterior Median for 8 and 10 Trees with $x[3] = 0.8$.

Table 13
Average Absolute Error and Root Integrated Square Error for fixed bandwidth estimators in the case of Inhomogeneous five dimensional Poisson Process with sparsity assumption.

Kernel Smoothing		
Bandwidth (sigma)	AAE	RISE
0.121 (LCV)	407.1	888.1

Table 14
Average Absolute Error and Root Integrated Square Error for linear conditional intensity in the case of Inhomogeneous five dimensional Poisson Process with sparsity assumption.

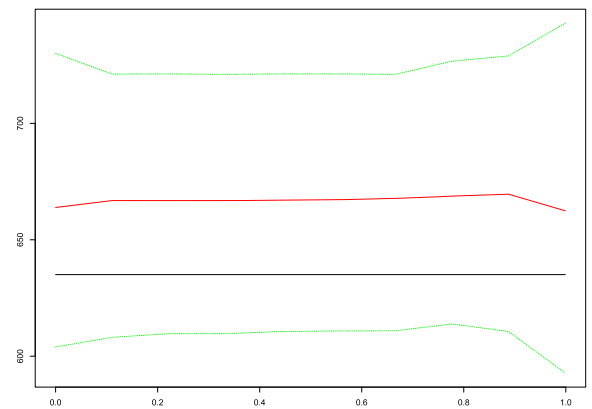| Linear conditional intensity	
AAE	RISE
654.2	1076.5
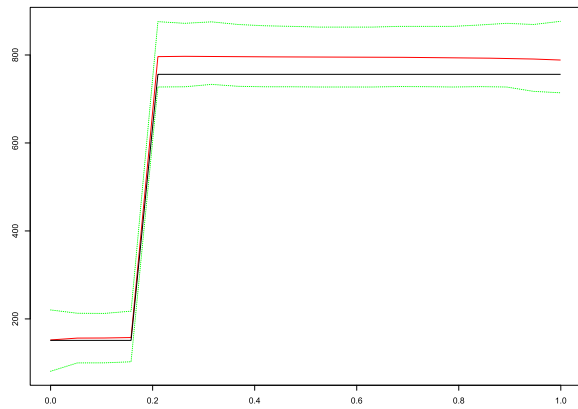
(a) The posterior mean of $\lambda(x_1)$ (red line; 95% CI (green line)) and the true expected value of $\lambda(x_1)$ (black line).
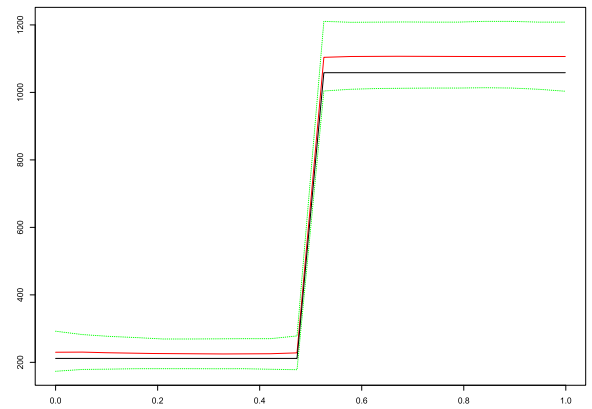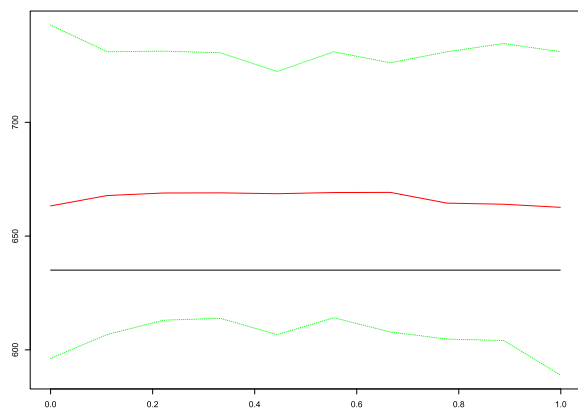
(b) The posterior mean of $\lambda(x_2)$ (red line; 95% CI (green line)) and the true expected value of $\lambda(x_2)$ (black line).
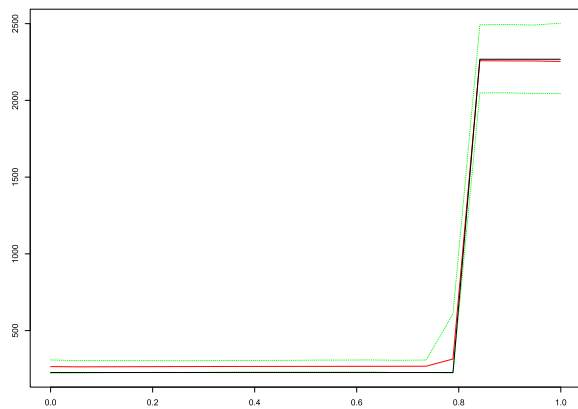
(c) The posterior mean of $\lambda(x_3)$ (red line; 95% CI (green line)) and the true expected value of $\lambda(x_3)$ (black line).

(d) The posterior mean of $\lambda(x_4)$ (red line; 95% CI (green line)) and the true expected value of $\lambda(x_4)$ (black line).

(e) The posterior mean of $\lambda(x_5)$ (red line; 95% CI (green line)) and the true expected value of $\lambda(x_5)$ (black line).

Fig. 6. Posterior marginal intensities considering 4 trees.

Table 15
How likely each dimension is to be involved in the root's decision rule.

Proposed BART Algorithm					
Number of trees	x_1	x_2	x_3	x_4	x_5
4	0.31	0.29	0.34	0.03	0.03
5	0.35	0.29	0.26	0.05	0.06

Table 16
The frequency of times we meet each dimension in the decision rules of a tree.

Proposed BART Algorithm					
Number of trees	x_1	x_2	x_3	x_4	x_5
4	0.35	0.36	0.37	0.06	0.07
5	0.39	0.34	0.37	0.09	0.10

Table 17
The average RPS on testing points over 7 different splits of the original data set in the case of Inhomogeneous five dimensional Poisson Process with sparsity assumption.

Proposed BART Algorithm			
Number of trees	$N_s = 1$	$N_s = 32$	$N_s = 243$
4	5.40	0.99	0.71
5	5.40	1	0.71
6	5.42	1	0.71
8	5.42	1	0.71
10	5.42	1	0.71
15	5.43	1	0.71

Table 18
The average RSMSE on testing points over 7 different splits of the original data set in the case of Inhomogeneous five dimensional Poisson Process with sparsity assumption.

Proposed BART Algorithm			
Number of trees	$N_s = 1$	$N_s = 32$	$N_s = 243$
4	0.64	0.95	1
5	0.64	0.95	1
6	0.65	0.95	1
8	0.65	0.96	1.01
10	0.65	0.96	1.01
15	0.65	0.96	1.01

volume subareas $\{S_i\}_{i=1}^{N_S}$ and consider a set $\{z_i\}$ by uniformly sampling points in the domain $[0, 1)^d$. We compute the AAE of the estimated expected number of points falling into each of the subareas:

$$\text{AAE}(\widehat{N}) = \frac{1}{N_S} \sum_{i=1}^{N_S} |\widehat{N}(S_i) - N(S_i)| \tag{16}$$

and Root Integrated Square Error (RISE):

$$\text{RISE}(\widehat{N}) = \left(\frac{1}{N_S} \sum_{i=1}^{N_S} (\widehat{N}(S_i) - N(S_i))^2 \right)^{1/2}, \tag{17}$$

where $N(S_i)$ is the actual number of points in S_i and

(a) Aggregated Earthquakes in 128 consecutive intervals
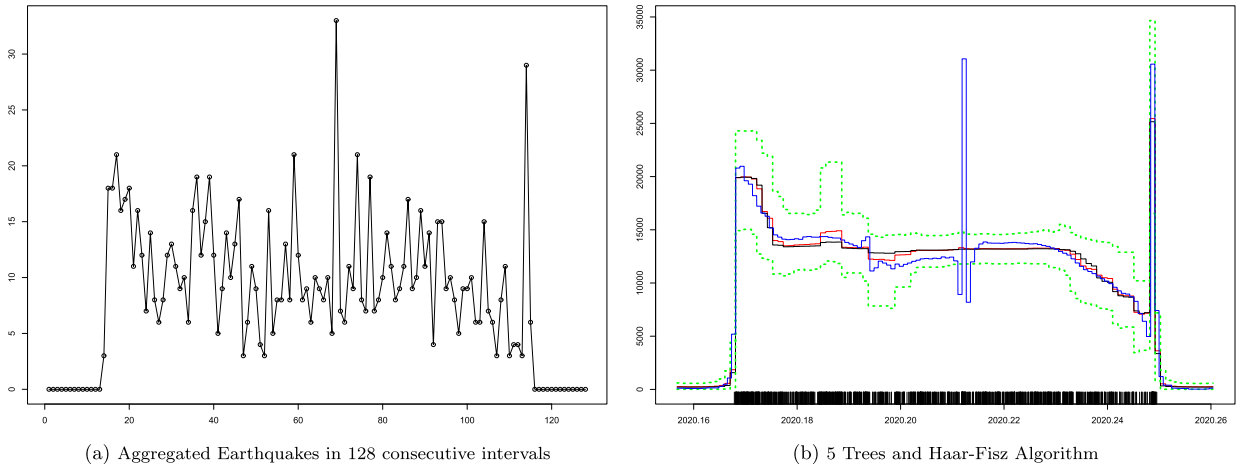

(b) 5 Trees and Haar-Fisz Algorithm

Fig. 7. Earthquakes Data: The posterior mean (red curve), the posterior median (black curve), the 95% hdi interval of the estimated intensity illustrated by the dotted green lines and the intensity estimator of the Haar-Fisz Algorithm illustrated by the blue line. The rug plot on the bottom displays the event times.

$$\widehat{N}(S_i) = \int_{S_i} \widehat{\lambda}(s)ds \simeq \frac{1}{N_{S_i}} \sum_{z_j \in S_i} |S_i| \widehat{\lambda}(z_j) \tag{18}$$

with N_{S_i} being the number of testing points $\{z_j\}$ falling in S_i. We apply the metrics AAE and RISE to compare our intensity estimates of one dimensional processes with those obtained by applying the Haar-Fisz algorithm for one dimensional data; and with kernel estimators for two-dimensional data. We observe that our algorithm, the Haar-Fisz algorithm and the kernel smoothing lead to similar results. As expected, the reconstructions of the intensity function are less smooth than those derived with kernel smoothing. The kernel estimator, as well as the bandwidth value given by likelihood cross-validation were computed using the **R** package **spatstat** (Baddeley and Turner, 2005). We provide more simulation results in the supplementary material.

5.1. Earthquakes data

This data set is available online from the Earthquake Hazards Program and consists of the times of 1088 earthquakes from 2-3-2020 to 1-4-2020. We consider the period from 27-2-2020 to 5-4-2020 to avoid edges. We run 3 parallel chains of the same length for 100000 iterations for 3-10 trees. The convergence criteria included in the supplementary material indicate that the considered chains have converged.

Fig. 7 presents the Posterior Mean and the Posterior Median for 5 Trees, as well as the intensity estimate of the Haar-Fisz algorithm applied to the counts in 128 consecutive intervals of equal length. The deterministic discretized intensity of the **R** package **haarfisz** is divided by the duration of an interval. The differences between both algorithms are due to different assumptions; the Haar-Fisz algorithm considers the aggregated counts into disjoint subintervals of the domain, while the proposed algorithm the times of individual events. The most noticeable difference is observed between 2020.212 and 2020.213 (69th interval) where we see a jump in earthquakes from 5 to 33 and again to 7. The Haar-Fisz algorithm detects that peak as we feed it with that information, while the proposed algorithm does not indicate a sharp rise in the intensity in that period, treating it as an outlier. The intensity estimate of the Haar-Fitz algorithm applied in 64 consecutive intervals is closer to the proposed algorithm (see Fig. 8), as expected. Similar to coarser binning, the proposed algorithm is less prone to overfitting to spikes in the data, which get filtered out. The estimated AAE and RISE demonstrate good performance compared to the Haar-Fisz method. The simulation results illustrate that our algorithm can track the varying intensity of earthquakes. (See Tables 19 and 20.)

The diagnostics D_g, D_l and D_{LPO} obtain their highest values at 9, 3 and 8 trees, respectively (see Table 19). The AIC diagnostics values between 3 and 9 trees show only small variations, we choose 5 trees for the analysis, noting that the results will not vary significantly for other choices of m in this region.

5.2. Lansing data

The **lansing** data set included in the **R** package **spatstat** describes the locations of different types of trees in the Lansing woods forest. Our attention is restricted to the locations of 514 maples that are presented with dots in Figs. 9-10. We run 3 parallel chains of the same length for 200000 iterations for 3-10 trees and 100000 iterations for 12 trees. The diagnostic criteria included in the supplementary material indicate that the considered chains have converged for the majority of testing points.
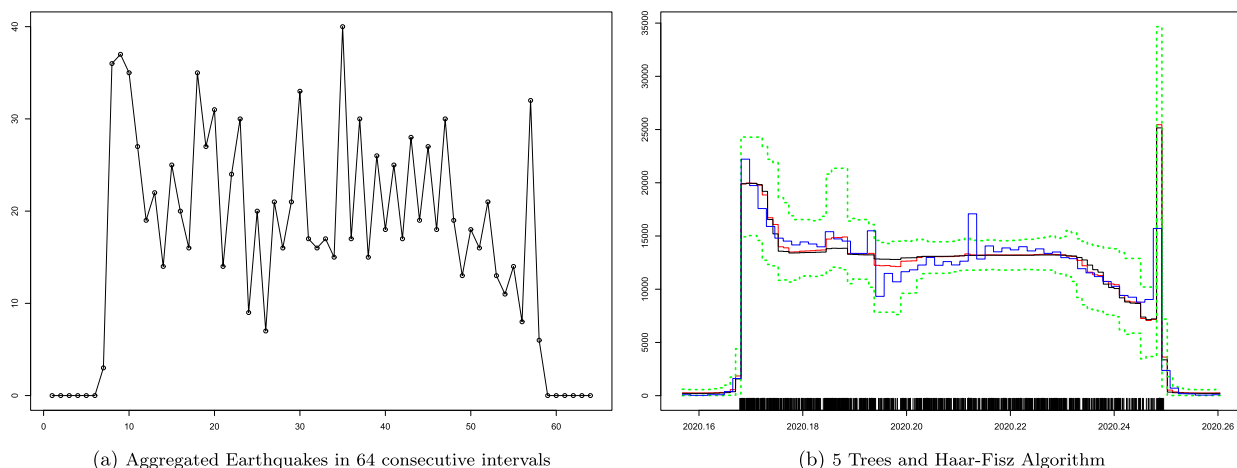
(a) Aggregated Earthquakes in 64 consecutive intervals

(b) 5 Trees and Haar-Fisz Algorithm

Fig. 8. Earthquakes Data: The posterior mean (red curve), the posterior median (black curve), the 95% hdi interval of the estimated intensity illustrated by the dotted green lines and the intensity estimator of the Haar-Fisz Algorithm illustrated by the blue line. The rug plot on the bottom displays the event times.

Table 19
Average Absolute Error, Root Integrated Square Error and diagnostics for the data in Fig. 7.

Proposed BART Algorithm						
Number of trees	AAE for Posterior Mean	AAE for Posterior Median	RISE for Posterior Mean	RISE for Posterior Median	D_g	D_l	D_{LPO}
3	93.8	94.1	106.9	107.1	13570.1	13565.4	-1194.7
4	94	94.1	106.8	107	13570.6	13563.6	-1163.8
5	93.6	94	106.7	107	13570.1	13560.7	-1150.5
6	93.8	94	106.9	107	13571.6	13559.6	-1169.7
8	93.5	94	106.6	107.1	13571.8	13554.9	-1140.2
9	93.4	94	106.7	107.3	13572.1	13552.6	-1192.5
10	93.4	94	106.8	107.3	13571.9	13549.8	-1184.4

Table 20
Average Absolute Error and Root Mean Square Error for Haar-Fisz estimator for the data in Fig. 7.

Haar-Fisz Algorithm		
Subintervals	AAE	RMSE
128	94.1	107.8
64	94	107

We compare our algorithm to a fixed bandwidth estimator using a Gaussian kernel. Our algorithm and the kernel estimator are consistent in the overall structure. The differences are due to the different nature of the methods. Given the tree locations, our algorithm recovers the spatial pattern of trees as rectangular regions of different intensities (Fig. 9), whereas the kernel method produces a continuum with more localized peaks in space. As expected, the kernel estimator presented in Fig. 10 consists of smoother subregions with various intensities. Tables 21-23 show that our algorithm is competitive to kernel smoothing with fixed bandwidth chosen with likelihood cross-validation. In contrast to our method, kernel methods are highly sensitive to parameter (bandwidth) choice.

The diagnostics D_g and D_l obtain their highest values at 4 and 10 trees, respectively.

6. Discussion and future work

In this article, we have studied how the Bayesian Additive Regression Trees (BART) model can be applied to estimating the intensity of Poisson processes. The BART framework provides a flexible non-parametric approach to capturing non-linear and additive effects in the underlying functional form of the intensity. Our numerical experiments show that our algorithm provides good approximations of the intensity with ensembles of less than 10 trees. This enables our algorithm to detect the dimensions contributing most to the intensity. The ability of our method to identify important features demonstrates an important advantage over other procedures.
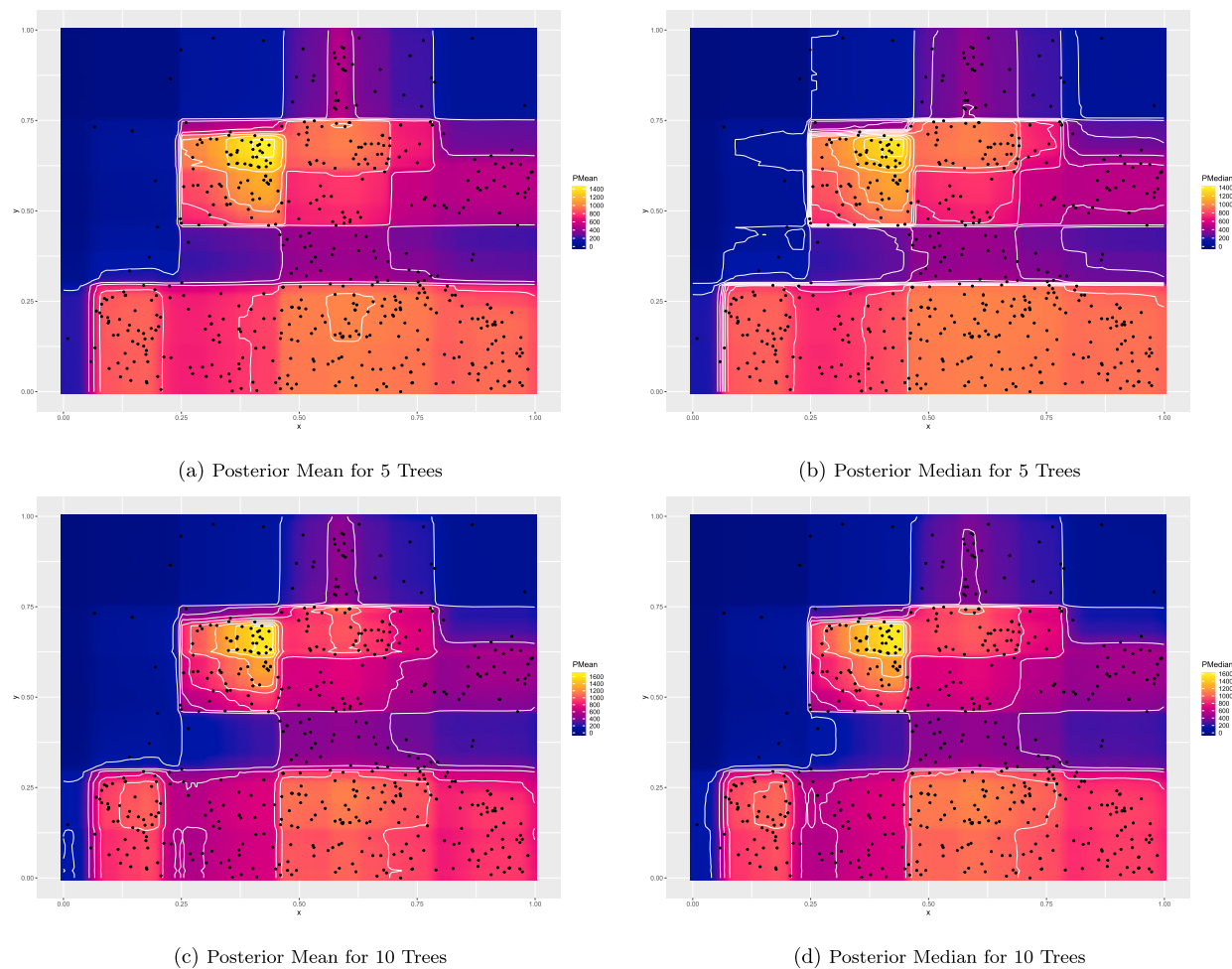
(a) Posterior Mean for 5 Trees

(b) Posterior Median for 5 Trees

(c) Posterior Mean for 10 Trees

(d) Posterior Median for 10 Trees

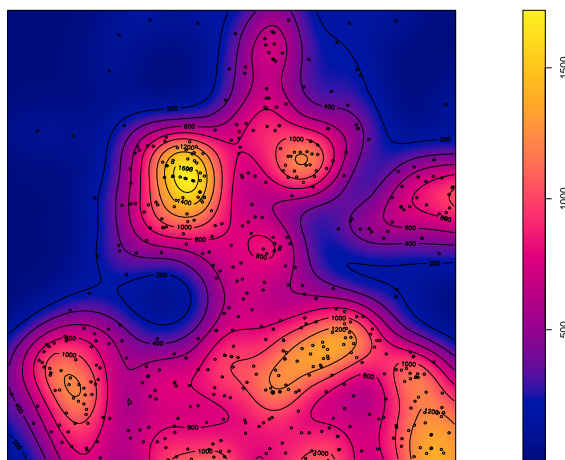**Fig. 9.** Posterior Mean and Posterior Median for 5 and 10 Trees.

Fig. 10. Fixed-bandwidth chosen using likelihood cross-validation.

Our approach enables full posterior inference of the intensity in a non-parametric regression setting. In addition, the method extends easily to higher dimensional settings. The simulation study on synthetic data sets shows that our algorithm can detect change points and provides good estimates of the intensity via either the posterior mean or the posterior median. Our algorithm is competitive with the Haar-Fisz algorithm and kernel methods in one and two dimensions and in-

Table 21

Average Absolute Error, Root Integrated Square Error with $N_S = 225$ and diagnostics for the data in Fig. 9.

	Proposed BART Algorithm					
Number of trees	AAE for Posterior Mean	AAE for Posterior Median	RMSE for Posterior Mean	RMSE for Posterior Median	D_g	D_l
3	1.3	1.2	1.7	1.8	5686.5	5705.3
4	1.2	1.2	1.7	1.8	5683.8	5709.5
5	1.2	1.2	1.7	1.7	5672.4	5705.4
7	1.2	1.2	1.7	1.71	5643.5	5702
8	1.2	1.2	1.7	1.7	5634	5707.2
9	1.2	1.2	1.7	1.7	5614.3	5698.1
10	1.2	1.2	1.6	1.7	5596.6	5699.8
12	1.2	1.2	1.7	1.7	5558.2	5692.5

Table 22

Average Absolute Error and Root Integrated Square Error with $N_S = 400$ for the data in Fig. 9.

	Proposed BART Algorithm			
Number of trees	AAE for Posterior Mean	AAE for Posterior Median	RMSE for Posterior Mean	RMSE for Posterior Median
3	0.9	0.9	1.3	1.3
4	0.9	0.9	1.2	1.3
5	0.9	0.9	1.2	1.3
7	0.9	0.9	1.2	1.2
8	0.9	0.9	1.2	1.2
9	0.9	0.9	1.2	1.2
10	0.9	0.9	1.2	1.2
12	0.9	0.9	1.2	1.2

Table 23

Average Absolute Error and Root Integrated Square Error for fixed bandwidth estimators for data in Fig. 10.

Kernel Smoothing		
Bandwidth (sigma)	AAE	RISE
0.05 (LCV) for $N_S = 225$	1.03	1.42
0.05 (LCV) for $N_S = 400$	0.82	1.13

ference using spatial log-Gaussian Cox processes. The strength of our method is its performance in higher dimensions, and we demonstrate that it outperforms the kernel approach for multidimensional intensities. We also demonstrate that our inference for the intensity is consistent with the variability of the rate of events in real and synthetic data. The convergence criteria included in the supplementary material indicate good convergence of the considered chains. We ran each chain for at least 100000 iterations to increase our confidence in the results. However, our algorithm works well with considerably fewer iterations (around 10000). The BART model assumes independence of the underlying tree structure. The alternative method of (Sardy and Tseng, 2004) makes use of a locally dependent Markov Random Field, and one way of extending our model in this direction is to consider neighbouring intensities following Chipman et al. (2021).

Our method has only considered the standard priors commonly used in BART procedures, an interesting avenue of future research would be to implement different prior assumptions. In addition, we have fixed the parameters for the Galton-Watson prior on the trees, and further work on sensitivies to hyperparameter selection and alternative methods for inference of the hyperparameters is of interest. Currently, our model is limited to non-homogeneous Poisson Process and we believe the flexibility of the BART approach could be extended to more general point processes.

Appendix A. Metropolis Hastings proposals

We describe the proposals of Algorithm 2. The Hastings ratio can be expressed as the product of three terms (Kapelner and Bleich, 2016):

- Transition Ratio:

$$\text{TR} = \frac{q(T_j^{(t)}|T_j^*)}{q(T_j^*|T_j^{(t)})}$$

- Likelihood Ratio:

$$\text{LR} = \frac{P(\boldsymbol{s}|T_j^*, T_{(j)}, \Lambda_{(j)})}{P(\boldsymbol{s}|T_j^{(t)}, T_{(j)}, \Lambda_{(j)})}$$

- Tree Structure Ratio:

$$\text{TSR} = \frac{P(T_j^*)}{P(T_j^{(t)})}$$

A.1. GROW proposal

This proposal randomly picks a terminal node, splits the chosen terminal into two new nodes and assigns a decision rule to it.

Let η be the randomly picked terminal node in tree $T_j^{(t)}$. We denote the new nodes as η_L and η_R. We now derive the expressions for the transition ratio (TR), tree structure ratio (TSR) and likelihood ratio (LR).

Transition ratio   It holds that:

(i) $q(T_j^*|T_j^{(t)}) = \text{P(GROW)}$
  $\times$ P(selecting a leaf η to grow from)
  $\times$ P(selecting an available dimension j to split on)
  $\times$ P(selecting the slitting value given the chosen dimension to split on)
  $= \text{P(GROW)} \frac{1}{b_j} \frac{1}{\text{card}(k_\eta)} \frac{1}{\text{card}(\tau_\eta)}$

  where b_j is the number of terminal nodes in the tree $T_j^{(t)}$, k_h the set of all available dimensions to split the node η, τ_η the set of all available splitting values given the chosen dimension for splitting the node η and card(S) the cardinality of a set S.

(ii) $q(T_j^{(t)}|T_j^*) = \text{P(PRUNE)}$
  $\times$ P(selecting a node η having two terminal nodes to prune from)
  $= \text{P(PRUNE)} \frac{1}{w^*}$

  where w^* is the number of internal nodes with two terminal nodes as children in the tree T_j^*.

Hence the transition ratio is given by

$$\text{TR} = \frac{P(\text{PRUNE})\frac{1}{w^*}}{P(\text{GROW})\frac{1}{b_j}\frac{1}{\text{card}(k_\eta)}\frac{1}{\text{card}(\tau_\eta)}}.$$

Tree structure ratio: The difference between the structures of the proposed tree $T_j^{(t)}$ and the tree T_j^* is the two offsprings η_L and η_R. Thus the tree structure ratio is:

$$\text{TSR} = \frac{P(T_j^*)}{P(T_j^{(t)})} = \frac{(1 - p_{\text{SPLIT}}(\eta_L))(1 - p_{\text{SPLIT}}(\eta_R)) \, p_{\text{SPLIT}}(\eta) \, p_{RULE}(\eta)}{(1 - p_{\text{SPLIT}}(\eta))}$$

$$= \frac{\left(1 - \frac{\gamma}{(1+d(\eta_L))^\delta}\right)\left(1 - \frac{\gamma}{(1+d(\eta_R))^\delta}\right) \frac{\gamma}{(1+d(\eta))^\delta} \frac{1}{\text{card}(k_\eta)} \frac{1}{\text{card}(\tau_\eta)}}{1 - \frac{\gamma}{(1+d(\eta))^\delta}},$$

where $p_{\text{SPLIT}}(\eta)$ is the splitting probability for a node η and $p_{RULE}(\eta)$ the distribution of decision rule associated to node η.

Likelihood ratio   The likelihood ratio is an application of equation (8) twice, that is once considering the proposed tree, T_j^* (numerator) and the other considering the tree of the current iteration t, $T_j^{(t)}$ (denominator), which can be simplified as follows

$$\text{LR} = \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\frac{\Gamma(n_{j\eta_L}+\alpha)}{(c_{j\eta_L}+\beta)^{n_{j\eta_L}+\alpha}} \frac{\Gamma(n_{j\eta_R}+\alpha)}{(c_{j\eta_R}+\beta)^{n_{j\eta_R}+\alpha}}}{\frac{\Gamma(n_{j\eta}+\alpha)}{(c_{j\eta}+\beta)^{n_{j\eta}+\alpha}}}$$

$$= \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(n_{j\eta_L}+\alpha)\Gamma(n_{j\eta_R}+\alpha)}{\Gamma(n_{j\eta}+\alpha)} \frac{(c_{j\eta}+\beta)^{n_{j\eta}+\alpha}}{(c_{j\eta_L}+\beta)^{n_{j\eta_L}+\alpha}(c_{j\eta_R}+\beta)^{n_{j\eta_R}+\alpha}}$$

A.2. PRUNE proposal

This proposal randomly picks a parent of two terminal nodes and turns it into a terminal node by collapsing the nodes below it.

Let η be the picked parent of two terminal nodes, y and c the dimension and splitting value of the rule linked to the node η.

Transition ratio   It holds that:

(i) $q(T_j^*|T_j^{(t)}) = \text{P(PRUNE)}$
$\qquad\qquad \times \text{P(selecting a parent of two terminal nodes to prune from)}$
$\qquad\qquad = \text{P(PRUNE)} \frac{1}{w}$

where w is the number of nodes with two terminal nodes as children in the tree $T_j^{(t)}$.

(ii) $q(T_j^{(t)}|T_j^*) = \text{P(GROW)}$
$\qquad\qquad \times \text{P(selecting the node } \eta \text{ to grow from)}$
$\qquad\qquad \times \text{P(selecting the dimension } y)$
$\qquad\qquad \times \text{P(selecting the slitting value c given the chosen dimension } y)$
$\qquad\qquad = \text{P(GROW)} \frac{1}{w^*} \frac{1}{\text{card}(k_\eta)} \frac{1}{\text{card}(\tau_\eta)}$

where w^* is the number of terminal nodes in the tree T_j^*, k_h the set of all available dimensions to split the node η and τ_η the set of all available splitting values given the chosen dimension y for splitting the node η.

Hence the transition ratio is given by

$$\text{TR} = \frac{P(\text{GROW}) \frac{1}{w^*} \frac{1}{\text{card}(k_\eta)} \frac{1}{\text{card}(\tau_\eta)}}{P(\text{PRUNE}) \frac{1}{w}}.$$

Tree structure ratio   The proposed tree differs by not having the two children nodes η_L and η_R. Thus the tree structure ratio is:

$$\text{TSR} = \frac{P(T_j^*)}{P(T_j^{(t)})} = \frac{(1 - p_{\text{SPLIT}}(\eta))}{(1 - p_{\text{SPLIT}}(\eta_L))(1 - p_{\text{SPLIT}}(\eta_R)) \, p_{\text{SPLIT}}(\eta) \, p_{RULE}(\eta)}$$

$$= \frac{1 - \frac{\gamma}{(1+d(\eta))^\delta}}{\left(1 - \frac{\gamma}{(1+d(\eta_L))^\delta}\right) \left(1 - \frac{\gamma}{(1+d(\eta_R))^\delta}\right) \frac{\gamma}{(1+d(\eta))^\delta} \frac{1}{\text{card}(k_\eta)} \frac{1}{\text{card}(\tau_\eta)}}$$

Likelihood ratio   Similar to the GROW proposal, the likelihood ratio can be written as follows

$$\text{LR} = \left(\frac{\beta^\alpha}{\Gamma(\alpha)}\right)^{-1} \frac{\frac{\Gamma(n_{j\eta}+\alpha)}{(c_{j\eta}+\beta)^{n_{j\eta}+\alpha}}}{\frac{\Gamma(n_{j\eta_L}+\alpha)}{(c_{j\eta_L}+\beta)^{n_{j\eta_L}+\alpha}} \frac{\Gamma(n_{j\eta_R}+\alpha)}{(c_{j\eta_R}+\beta)^{n_{j\eta_R}+\alpha}}}$$

$$= \left(\frac{\beta^\alpha}{\Gamma(\alpha)}\right)^{-1} \frac{\Gamma(n_{j\eta}+\alpha)}{\Gamma(n_{j\eta_L}+\alpha)\Gamma(n_{j\eta_R}+\alpha)} \frac{(c_{j\eta_L}+\beta)^{n_{j\eta_L}+\alpha}(c_{j\eta_R}+\beta)^{n_{j\eta_R}+\alpha}}{(c_{j\eta}+\beta)^{n_{j\eta}+\alpha}}$$

A.3. CHANGE proposal

This proposal randomly picks an internal node and randomly reassigns to it a splitting rule.

Let η be the picked internal node having rule $y < c$ and children denoted as η_R and η_L. We assume that $\widetilde{y} < \widetilde{c}$ is its new assigned rule in the proposed tree, T_j^*. Following Kapelner and Bleich (2016), for simplicity we are restricted to picking an internal node having two terminal nodes as children.

Transition ratio   It holds that:

(i) $q(T_j^*|T_j^{(t)}) = $ P(CHANGE)

$\times$ P(selecting an internal node η to change)

$\times$ P(selecting the new available dimension $\widetilde{y}$ to split on)

$\times$ P(selecting the new splitting value $\widetilde{c}$ given the chosen dimension $\widetilde{y}$)

(ii) $q(T_j^{(t)}|T_j^*) = $ P(CHANGE)

$\times$ P(selecting the node η to change)

$\times$ P(selecting the dimension y to split on)

$\times$ P(selecting the splitting value c given the chosen dimension y)

Thus the Transition Ratio is

$$\text{TR} = \frac{P(\text{selecting } c \text{ to split on given the chosen dimension } y)}{P(\text{selecting } \widetilde{c} \text{ to split on given the chosen dimension } \widetilde{y})}$$

Tree structure ratio   The two trees differ in the splitting rule at node η. Thus we have that

$$\text{TSR} = \frac{P(T_j^*)}{P(T_j^{(t)})} = \frac{p_{\text{SPLIT}}(\eta)\, p_{\text{RULE}}(\eta|T_j^*)}{p_{\text{SPLIT}}(\eta)\, p_{\text{RULE}}(\eta|T_j^{(t)})}$$

$$= \frac{P(\text{selecting } \widetilde{y})\, P(\text{selecting } \widetilde{c} \text{ given } \widetilde{y})}{P(\text{selecting } y)\, P(\text{selecting } c \text{ given } y)}$$

$$= \frac{P(\text{selecting } \widetilde{c} \text{ given } \widetilde{y})}{P(\text{selecting } c \text{ given } y)}.$$

It then follows that $\text{TR} \times \text{TSR} = 1$, and hence only the likelihood ratio needs to be found to obtain the Hastings ratio.

Likelihood ratio   Let $n_L^* = n_{j\eta_L}^{(T_j^*)}$, $n_R^* = n_{j\eta_R}^{(T_j^*)}$, $c_L^* = c_{j\eta_L}^{(T_j^*)}$, $c_R^* = c_{j\eta_R}^{(T_j^*)}$, $n_L^{(t)} = n_{j\eta_L}^{(T_j^{(t)})}$, $n_R^{(t)} = n_{j\eta_R}^{(T_j^{(t)})}$, $c_L^{(t)} = c_{j\eta_L}^{(T_j^{(t)})}$ and $c_R^{(t)} = c_{j\eta_R}^{(T_j^{(t)})}$, where (T_j^*) and $(T_j^{(t)})$ indicate that the corresponding quantities are related to the tree T_j^* and $T_J^{(t)}$ respectively. Following the previous proposals, the likelihood ratio is

$$\text{LR} = \frac{\frac{\Gamma(n_L^*+\alpha)}{(c_L^*+\beta)^{n_L^*+\alpha}}\frac{\Gamma(n_R^*+\alpha)}{(c_R^*+\beta)^{n_R^*+\alpha}}}{\frac{\Gamma(n_L^{(t)}+\alpha)}{(c_L^{(t)}+\beta)^{n_L^{(t)}+\alpha}}\frac{\Gamma(n_R^{(t)}+\alpha)}{(c_R^{(t)}+\beta)^{n_R^{(t)}+\alpha}}} = \frac{(c_L^{(t)}+\beta)^{n_L^{(t)}+\alpha}\,(c_R^{(t)}+\beta)^{n_R^{(t)}+\alpha}}{(c_L^*+\beta)^{n_L^*+\alpha}\,(c_R^*+\beta)^{n_R^*+\alpha}}\,\frac{\Gamma(n_L^*+\alpha)\,\Gamma(n_R^*+\alpha)}{\Gamma(n_L^{(t)}+\alpha)\,\Gamma(n_R^{(t)}+\alpha)}.$$

Appendix B. The Poisson process conditional likelihood

Let us consider a finite realization of an inhomogeneous Poisson process with n points $\boldsymbol{s}$. Given the tree components (T, Λ), and approximating the intensity of a point $s_i \in S$ by a product of m trees $\lambda(s_i) = \prod_{j=1}^{m} g(s_i; T_j, \Lambda_j)$, the likelihood is:

$$P(\boldsymbol{s}|\Lambda, T) = \prod_{i=1}^{n} \lambda(s_i) \exp\left(-\int_S \lambda(s)ds\right)$$

$$= \prod_{i=1}^{n}\prod_{j=1}^{m} g(s_i; T_j, \Lambda_j) \exp\left(-\int_S \prod_{j=1}^{m} g(s; T_j, \Lambda_j)ds\right). \tag{B.1}$$

The first term of the above equation can be written as follows

$$\prod_{i=1}^{n}\prod_{j=1}^{m} g(s_i; T_j, \Lambda_j) = \prod_{i=1}^{n}\prod_{j=1, j\neq h}^{m} g(s_i; T_j, \Lambda_j)g(s_i; T_h, \Lambda_h)$$

$$= \prod_{i=1}^{n}\prod_{j=1, j\neq h}^{m} g(s_i; T_j, \Lambda_j)\left(\prod_{i=1}^{n} g(s_i; T_h, \Lambda_h)\right) = c_h \prod_{t=1}^{b_h} \lambda_{ht}^{n_{ht}}$$

where $c_h = \prod_{i=1}^{n}\prod_{j=1, j\neq h}^{m} g(s_i; T_j, \Lambda_j)$ and n_{ht} is the cardinality of the set $\{i : s_i \in \Omega_{ht}\}$.

The exponential term of (B.1) can be expressed as:

$$\exp\left(-\int_S \prod_{j=1}^{m} g(s; T_j, \Lambda_j)ds\right) = \exp\left(-\int_S \prod_{j=1, j\neq h}^{m} g(s; T_j, \Lambda_j)g(s; T_h, \Lambda_h)\right)$$

$$= \exp\left(-\int_S \prod_{j=1, j\neq h}^{m} g(s; T_j, \Lambda_j)\left(\sum_{t=1}^{b_h}\lambda_{ht}I(s \in \Omega_{ht})\right)ds\right)$$

$$= \exp\left(-\int_S \sum_{t=1}^{b_h}\lambda_{ht}\prod_{j=1, j\neq h}^{m} g(s; T_j, \Lambda_j)I(s \in \Omega_{ht})ds\right)$$

Tonelli's theorem allows the change of order between summation and integral.

$$\exp\left(-\int_S \prod_{j=1}^{m} g(s; T_j, \Lambda_j)ds\right) = \exp\left(-\sum_{t=1}^{b_h}\lambda_{ht}\int_S \prod_{j=1, j\neq h}^{m} g(s; T_j, \Lambda_j)I(s \in \Omega_{ht})ds\right)$$

$$= \exp\left(-\sum_{t=1}^{b_h}\lambda_{ht}c_{ht}\right)$$

where

$$c_{ht} = \int_S \left(\prod_{j=1, j\neq h}^{m} g(s; T_j, \Lambda_j)\right) I(s \in \Omega_{ht})ds.$$

Let $T_{(h)} = \{T_j\}_{j=1, j\neq h}^{m}$ be an ensemble of trees not including the tree T_h that defines the global partition $\{\overline{\Omega}_k^{(h)}\}_{k=1}^{K(T_{(h)})}$ by merging all cuts in $\{T_j\}_{j=1, j\neq h}^{m}$. Giving,

$$\prod_{j=1, j\neq h}^{m} g(s; T_j, \Lambda_j) = \sum_{k=1}^{K(T_h)} \overline{\lambda}_k^{(h)}I(s \in \overline{\Omega}_k^{(h)})$$

where

$$\overline{\lambda}_k^{(h)} = \prod_{t=1, t\neq h}^{m}\prod_{l=1}^{b_t}\lambda_{tl}^{I(\Omega_{tl}\cap\overline{\Omega}_k^{(h)}\neq 0)},$$

leading to the following expression for c_{ht},

$$c_{ht} = \int_S \left(\prod_{j=1, j\neq h}^{m} g(s, T_j, \Lambda_j)\right) I(s \in \Omega_{ht})ds = \int_S \left(\sum_{k=1}^{K(T_{(h)})} \overline{\lambda}_k^{(h)}I(s \in \overline{\Omega}_k^{(h)})\right) I(s \in \Omega_{ht})ds$$

$$= \sum_{k=1}^{K(T_{(h)})} \overline{\lambda}_k^{(h)} \int_S I(s \in \overline{\Omega}_k^{(h)} \cap \Omega_{ht})ds = \sum_{k=1}^{K(T_{(h)})} \overline{\lambda}_k^{(h)}|\overline{\Omega}_k^{(h)} \cap \Omega_{ht}|,$$

where $|\overline{\Omega}_k^{(h)} \cap \Omega_{ht}|$ is the volume of the region $\Omega_k^{(h)} \cap \Omega_{ht}$. Hence the conditional likelihood can be written as follows

$$P(\boldsymbol{s}|\Lambda, T) = c_h \prod_{t=1}^{b_h} \lambda_{ht}^{n_{ht}}e^{-\lambda_{ht}c_{ht}}.$$

Appendix C. The conditional integrated likelihood

The conditional integrated likelihood is given by

$$P(\boldsymbol{s}|T_h, T_{(h)}, \Lambda_{(h)}) = \int_0^{\infty} P(\boldsymbol{s}, \Lambda_h|T_h, T_{(h)}, \Lambda_{(h)})d\Lambda_h$$

$$= \int_0^\infty P(s|\Lambda, T)\, P(\Lambda_h|T_h, T_{(h)}, \Lambda_{(h)})\, d\Lambda_h$$

$$= c_h \int_0^\infty \cdots \int_0^\infty \prod_{t=1}^{b_h} \lambda_{ht}^{n_{ht}} e^{-\lambda_{ht} c_{ht}} \prod_{t=1}^{b_h} \frac{\beta^\alpha}{\Gamma(\alpha)} e^{-\beta \lambda_{ht}} \lambda_{ht}^{\alpha-1} d\lambda_{h1} \ldots d\lambda_{hb_h}$$

$$= c_h \left(\frac{\beta^\alpha}{\Gamma(\alpha)} \right)^{b_h} \prod_{t=1}^{b_h} \int_0^\infty \lambda_{ht}^{n_{ht}+\alpha-1} e^{-(c_{ht}+\beta)\lambda_{ht}} d\lambda_{ht}$$

$$= c_h \left(\frac{\beta^\alpha}{\Gamma(\alpha)} \right)^{b_h} \prod_{t=1}^{b_h} \frac{\Gamma(n_{ht}+\alpha)}{(c_{ht}+\beta)^{n_{ht}+\alpha}}$$

Appendix D. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csda.2022.107658.

References

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

Arjas, Elja, Gasbarra, Dario, 1994. Nonparametric Bayesian inference from right censored survival data, using the Gibbs sampler. Stat. Sin., 505–524.

Baddeley, Adrian, Turner, Rolf, 2005. spatstat: an R package for analyzing spatial point patterns. J. Stat. Softw. 12 (6), 1–42. http://www.jstatsoft.org/v12/i06/.

Bleich, Justin, Kapelner, Adam, 2014. Bayesian additive regression trees with parametric models of heteroskedasticity. Preprint, arXiv:1402.5397.

Bleich, Justin, Kapelner, Adam, George, Edward I., Jensen, Shane T., et al., 2014. Variable selection for BART: an application to gene regulation. Ann. Appl. Stat. 8 (3), 1750–1781.

Chipman, Hugh A., George, Edward I., McCulloch, Robert E., 1998. Bayesian CART model search. J. Am. Stat. Assoc. 93 (443), 935–948.

Chipman, Hugh A., George, Edward I., McCulloch, Robert E., et al., 2010. BART: Bayesian additive regression trees. Ann. Appl. Stat. 4 (1), 266–298.

Chipman, Hugh A., George, Edward I., McCulloch, Robert E., Shively, Thomas S., 2021. mBART: multidimensional monotone BART. Bayesian Anal., 1–30. https://doi.org/10.1214/21-BA1259.

Daley, Daryl J., Vere-Jones, David, 2003. Elementary Theory and Methods. Springer.

Davies, Tilman M., Baddeley, Adrian, 2018. Fast computation of spatially adaptive kernel estimates. Stat. Comput. 28 (4), 937–956.

Diggle, Peter J., et al., 2003. Statistical Analysis of Spatial Point Patterns, 2nd ed. Academic Press.

Fryzlewicz, Piotr, 2010. haarfisz: software to perform Haar Fisz transforms. R package version 4.5. https://CRAN.R-project.org/package=haarfisz.

Fryzlewicz, Piotr, Nason, Guy P., 2004. A Haar-Fisz algorithm for Poisson intensity estimation. J. Comput. Graph. Stat. 13 (3), 621–638.

Gelman, Andrew, Rubin, Donald B., et al., 1992. Inference from iterative simulation using multiple sequences. Stat. Sci. 7 (4), 457–472.

Gelman, Andrew, Hwang, Jessica, Vehtari, Aki, 2014. Understanding predictive information criteria for Bayesian models. Stat. Comput. 24 (6), 997–1016.

Gelman, Andrew, Lee, Daniel, Stan, Jiqiang Guo, 2015. A probabilistic programming language for Bayesian inference and optimization. J. Educ. Behav. Stat. 40 (5), 530–543.

Gugushvili, Shota, van der Meulen, Frank, Schauer, Moritz, Spreij, Peter, 2018. Fast and scalable non-parametric Bayesian inference for Poisson point processes. Preprint, arXiv:1804.03616.

Harris, Theodore Edward, et al., 1963. The Theory of Branching Processes. Springer, Berlin.

Heikkinen, Juha, Arjas, Elja, 1998. Non-parametric Bayesian estimation of a spatial Poisson intensity. Scand. J. Stat. 25 (3), 435–450.

Hill, Jennifer L., 2011. Bayesian nonparametric modeling for causal inference. J. Comput. Graph. Stat. 20 (1), 217–240.

Illian, Janine, Penttinen, Antti, Stoyan, Helga, Stoyan, Dietrich, 2008. Statistical Analysis and Modelling of Spatial Point Patterns, vol. 70. John Wiley & Sons.

Kapelner, Adam, Bleich, Justin, 2013. bartmachine: machine learning with Bayesian additive regression trees. Preprint, arXiv:1312.2171.

Kapelner, Adam, Bleich, Justin, 2016. bartMachine: machine learning with Bayesian additive regression trees. J. Stat. Softw. (ISSN 1548-7660) 70 (4), 1–40. https://doi.org/10.18637/jss.v070.i04.

Kindo, Bereket P., Wang, Hao, Peña, Edsel A., 2016. Multinomial probit Bayesian additive regression trees. Stat 5 (1), 119–131.

Lakshminarayanan, Balaji, Roy, Daniel, Teh, Yee Whye, 2015. Particle Gibbs for Bayesian additive regression trees. In: Artificial Intelligence and Statistics, pp. 553–561.

Leininger, Thomas J., Gelfand, Alan E., 2017. Bayesian inference and model assessment for spatial point patterns using posterior predictive samples. Bayesian Anal. 12 (1), 1–30.

Lewis, P.A.W., Shedler, Gerald S., 1979. Simulation of nonhomogeneous Poisson processes by thinning. Nav. Res. Logist. Q. 26 (3), 403–413.

Linero, Antonio R., 2018. Bayesian regression trees for high-dimensional prediction and variable selection. J. Am. Stat. Assoc. 113 (522), 626–636.

Linero, Antonio R., Yang, Yun, 2018. Bayesian regression tree ensembles that adapt to smoothness and sparsity. J. R. Stat. Soc., Ser. B, Stat. Methodol. 80 (5), 1087–1110.

Lloyd, Chris, Gunter, Tom, Osborne, Michael, Roberts, Stephen, 2015. Variational inference for Gaussian process modulated Poisson processes. In: International Conference on Machine Learning, pp. 1814–1822.

Loader, C., 1999. Local Regression and Likelihood. Springer, New York.

Makowski, Dominique, Ben-Shachar, Mattan, Lüdecke, Daniel, 2019. bayestestr: describing effects and their uncertainty, existence and significance within the Bayesian framework. J. Open Sour. Softw. 4 (40), 1541.

Murray, Jared S., 2017. Log-linear Bayesian additive regression trees for categorical and count responses. Preprint, arXiv:1701.01503.

Patil, Prakash N., Wood, Andrew T.A., et al., 2004. Counting process intensity estimation by orthogonal wavelet methods. Bernoulli 10 (1), 1–24.

Peng, Roger, 2003. Multi-dimensional point process models in r. J. Stat. Softw. 8, 1–27.

Pratola, Matthew T., et al., 2016. Efficient Metropolis–Hastings proposal mechanisms for Bayesian regression tree models. Bayesian Anal. 11 (3), 885–911.

Rockova, Veronika, Saha, Enakshi, 2018. On theory for Bart. Preprint, arXiv:1810.00787.

Rockova, Veronika, van der Pas, Stephanie, 2017. Posterior concentration for Bayesian regression trees and their ensembles. Preprint, arXiv:1708.08734.

Sardy, Sylvain, Tseng, Paul, 2004. On the statistical analysis of smoothing by maximizing dirty Markov random field posterior distributions. J. Am. Stat. Assoc. 99 (465), 191–204. https://doi.org/10.1198/016214504000000188.

Scott, David, 2008. Histograms: theory and practice, pp. 47–94. https://doi.org/10.1002/9780470316849.ch3.

Sparapani, Rodney A., Logan, Brent R., McCulloch, Robert E., Laud, Purushottam W., 2016. Nonparametric survival analysis using Bayesian additive regression trees (BART). Stat. Med. 35 (16), 2741–2753.

Stone, Mervyn, 1977. An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion. J. R. Stat. Soc., Ser. B, Methodol. 39 (1), 44–47.

Wand, M.P., 1997. Data-based choice of histogram bin width. Am. Stat. 51 (1), 59–64.

Zhang, Junni L., Härdle, Wolfgang K., 2010. The Bayesian additive classification tree applied to credit risk modelling. Comput. Stat. Data Anal. 54 (5), 1197–1205.