Journal of the American Statistical Association

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/uasa20

Generalized Bayesian Additive Regression Trees Models: Beyond Conditional Conjugacy

Antonio R. Linero

To cite this article: Antonio R. Linero (14 May 2024): Generalized Bayesian Additive Regression Trees Models: Beyond Conditional Conjugacy, Journal of the American Statistical Association, DOI: 10.1080/01621459.2024.2337156

To link to this article: https://doi.org/10.1080/01621459.2024.2337156

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2024, VOL. 00, NO. 0, 1–14: Theory and Methods
https://doi.org/10.1080/01621459.2024.2337156

Generalized Bayesian Additive Regression Trees Models: Beyond Conditional Conjugacy

Antonio R. Linero

Department of Statistics and Data Sciences, University of Texas at Austin, Austin, TX

ABSTRACT

Bayesian additive regression trees have seen increased interest in recent years due to their ability to combine machine learning techniques with principled uncertainty quantification. The Bayesian backfitting algorithm used to fit BART models, however, limits their application to a small class of models for which conditional conjugacy exists. In this article, we greatly expand the domain of applicability of BART to arbitrary *generalized BART* models by introducing a very simple, tuning-parameter-free, reversible jump Markov chain Monte Carlo algorithm. Our algorithm requires only that the user be able to compute the likelihood and (optionally) its gradient and Fisher information. The potential applications are very broad; we consider examples in survival analysis, structured heteroscedastic regression, and gamma shape regression. Supplementary materials for this article are available online, including a standardized description of the materials available for reproducing the work.

ARTICLE HISTORY
Received January 2022
Accepted March 2024

KEYWORDS
Bayesian methods; Bayesian nonparametrics; Decision Trees; Markov chain Monte Carlo; Random forests; Smoothing and nonparametric regression

1. Introduction

Since the introduction of boosting (Freund, Schapire, and Abe 1999), algorithms that ensemble shallow decision trees have become a fundamental part of the data science toolkit. A Bayesian framework for ensembling shallow decision trees is the Bayesian additive regression trees (BART) framework of Chipman, George, and McCulloch (2010). Some advantages of BART over other machine learning algorithms are that it provides direct uncertainty quantification and can naturally be incorporated into hierarchical models; while there are currently no theoretical guarantees regarding uncertainty quantification, it has been observed that BART performs surprisingly well in practice relative to other attempts at combining machine learning with statistical inference (Dorie et al. 2019). BART also bears a resemblance to the *random forests* algorithm (Breiman 2001): both BART and random forests average predictions over a distribution of trees (the posterior distribution and the bootstrap distribution, respectively). In terms of predictive performance, BART is competitive with, and sometimes outperforms, random forests (Chipman, George, and McCulloch 2010; Linero and Yang 2018). BART has several advantages over random forests, including (i) the ability to adapt to *additive* structures in the regression function (Ročková and van der Pas 2020), and (ii) the ability to use the posterior distribution to quantify uncertainty.

A drawback of BART is that one usually needs to tailor it to the problem at hand. Since the initial work of Chipman, George, and McCulloch (2010), which developed methods for semiparametric regression and classification, there have been substantial efforts to extend BART to other settings; a limited set of examples include survival analysis (Sparapani et al. 2016; Linero et al. 2022), Poisson regression (Murray 2021), and gamma regression (Linero, Sinha, and Lipsitz 2020). These developments have required either (i) the model to be such that software for normal or probit models can be adapted or (ii) the involvement of experts in BART methodology.

The difficulty of implementing new BART models stands in stark contrast with the difficulty of implementing new decision tree boosting algorithms, which can be done with very minimal expertise. In particular, given outcomes $\boldsymbol{Y} = (Y_1, \ldots, Y_N)$, covariate vectors $\boldsymbol{X} = (X_1, \ldots, X_N)$, and any utility function $R(\boldsymbol{Y} \mid \boldsymbol{X}, r, \eta) = \sum_{i=1}^N R_\eta\big(Y_i \mid r(X_i)\big)$ with nuisance parameter vector η, one can construct a *gradient boosting* algorithm (Friedman 2001) for estimating the function $r(x)$ that only requires users to provide the functions $R_\eta(y \mid \lambda)$, $U_\eta(y \mid \lambda) = \frac{\partial}{\partial\lambda} R_\eta(y \mid \lambda)$, and, optionally, $\mathcal{J}_\eta(y \mid \lambda) = -\frac{\partial}{\partial\lambda} U_\eta(y \mid \lambda)$; for model-based inference with a parametric family $\{f_\eta(\cdot \mid \lambda)\}$, note that we can take $R_\eta(y \mid \lambda) = \log f_\eta(y \mid \lambda)$. Software such as the `R` packages `xgboost` and `mboost` make it straight-forward for users to supply these functions manually, allowing boosting to be applied with arbitrary models and loss functions. This difference between BART and boosting is not because gradient boosting is an intrinsically simpler algorithm, but rather because the complex parts of the algorithm can be abstracted away from the user; in addition to boosting, abstracting away the implementation of complex inference algorithms has been key to the success of applied Bayesian modeling, which has been fueled by packages such as `Stan` and `JAGS`.

The contributions of this work are as follows.

1. Our primary aim is to introduce a framework for fitting *generalized BART* models with likelihoods of the form $\prod_i f_\eta\{Y_i \mid$

CONTACT Antonio R. Linero antonio.linero@austin.utexas.edu Department of Statistics and Data Sciences, University of Texas at Austin, Austin, TX
Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JASA.

$r(X_i)\}$ that, like gradient boosting, allows us to automate the application of BART to new settings.
2. We show that generalized BART is capable of addressing a wide variety of problems, and often outperforms gradient boosting for likelihood-based inference, while also providing natural uncertainty quantification.
3. We address the main obstacle to fitting generalized BART models: the reliance of BART on the generalized Bayesian backfitting algorithm of Hill, Linero, and Murray (2020), which requires users to be able to specify a prior $\pi_\mu(\mu)$ such that the integrated likelihood $\Lambda = \int \prod_i f_\eta(Y_i \mid \theta_i + \mu)\,\pi_\mu(\mu)\,d\mu$ can be computed in closed form for any collection of *backfit parameters* $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_N)$. Existing algorithms require this to compute a Metropolis-Hastings acceptance probability for modifying the structure of a tree.

Rather than starting from the assumption that Λ is analytically tractable, we instead assume (like boosting) that $\log f_\eta(y \mid \lambda)$ and its derivatives have been provided; strictly speaking even the derivatives need not be provided, as our algorithm can also be applied by approximating the derivatives with finite differences. Using only this assumption, we construct a generic reversible jump Markov chain Monte Carlo (RJMCMC, Green 1995) algorithm to sample new tree structures. The jump between dimensions is constructed using a Laplace approximation to ensure that the proposal has a high probability of being accepted. Importantly, our proposal is agnostic to the choice of model and completely free of tuning parameters.

We implement several models to illustrate both the accuracy and flexibility of our approach. We benchmark our algorithm on both semiparametric regression and classification problems, which are handled by existing algorithms; as a bonus, the RJMCMC algorithm avoids any data augmentation (Albert and Chib 1993). We then move on to previously intractable models such as structured variance modeling, accelerated failure time modeling with the log-logistic and generalized gamma distributions, and modeling of the shape parameter in gamma regression. In all cases we find that our RJMCMC algorithm works well.

In Section 2 we review the BART models which can currently be fit using existing Bayesian backfitting algorithms. In Section 3 we develop our RJMCMC algorithm for arbitrary generalized BART models. In Section 4, we study the posterior concentration properties of some generalized BART models. In Section 5 we illustrate our approach on a variety of both real and simulated problems. We close in Section 6 with a discussion.

2. Bayesian Additive Regression Trees

2.1. A Brief Review of BART

Suppose we have outcome data $\boldsymbol{Y} = (Y_1, \ldots, Y_N)$ and covariates $\boldsymbol{X} = (X_1, \ldots, X_N)$ where, for simplicity, we assume that X_i takes values in $[0,1]^P$. The Bayesian additive regression trees (BART) model as originally proposed by Chipman, George, and McCulloch (2010) is a semiparametric regression model of the form

$$Y_i \sim \text{Normal}\{r(X_i), \sigma^2\}$$
$$\text{where} \qquad r(x) = \sum_{t=1}^{T} g(x; \mathcal{T}_t, \mathcal{M}_t) \quad (i = 1, \ldots, N), \quad (1)$$

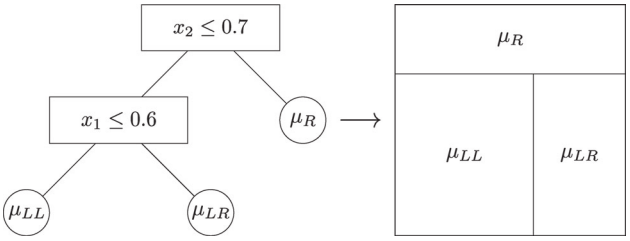

Figure 1. Schematic showing how a regression tree (left) gives rise to a step function of the predictors (right).

where N is the sample size. The functions $g(x; \mathcal{T}_t, \mathcal{M}_t)$ are *regression trees* parameterized by a *decision tree* $\mathcal{T}_t$ and a collection of predictions for the leaf nodes $\mathcal{M}_t$. Formally, we define a (binary) decision tree $\mathcal{T}$ as a collection of nodes $n \in \mathcal{N}(\mathcal{T})$ where n is a finite (potentially empty) string of the symbols L (left) and R (right). We say that $\ell \in \mathcal{N}(\mathcal{T})$ is a *leaf node* of $\mathcal{T}$ if both $\ell L \notin \mathcal{N}(\mathcal{T})$ and $\ell R \notin \mathcal{N}(\mathcal{T})$. Any node b which is not a leaf node is called a *branch node*, and we require that both $bL \in \mathcal{N}(\mathcal{T})$ and $bR \in \mathcal{N}(\mathcal{T})$ for every branch b. We let $\mathcal{L}(\mathcal{T})$ and $\mathcal{B}(\mathcal{T})$ denote the leaf and branch nodes of $\mathcal{T}$ respectively. It will also be convenient for us to define $\text{NOG}(\mathcal{T})$ to be the set of *non-grandparent* branches, that is, $\text{NOG}(\mathcal{T}) = \{b \in \mathcal{B}(\mathcal{T}) : bR \in \mathcal{L}(\mathcal{T}) \text{ and } bL \in \mathcal{L}(\mathcal{T})\}$; for example, the only non-grandparent branch in the tree in Figure 1 is the branch $b = L$.

Associated to each $b \in \mathcal{B}(\mathcal{T})$ is a *splitting rule* of the form $[x_{j_b} \le C_b]$. If x is associated to b and x satisfies b's splitting rule then we associate x to bL; otherwise, we associate x to bR. We write $x \overset{\mathcal{T}}{\rightsquigarrow} n$ (or $x \rightsquigarrow n$ when $\mathcal{T}$ is clear from context) to denote that x is associated to node n of tree $\mathcal{T}$. The collection of predictions can then be defined by $\mathcal{M}_t = \{\mu_{t\ell} : \ell \in \mathcal{L}(\mathcal{T}_t)\}$. By design, the leaf nodes partition the predictor space so that $x \rightsquigarrow \ell$ for exactly one $\ell \in \mathcal{L}(\mathcal{T}_t)$. Given $(\mathcal{T}_t, \mathcal{M}_t)$ the decision tree outputs the prediction $g(x; \mathcal{T}_t, \mathcal{M}_t) = \mu_{t\ell}$ if-and-only-if $x \overset{\mathcal{T}_t}{\rightsquigarrow} \ell$. A schematic showing how predictions are generated from a regression tree is given in Figure 1.

The BART model places independent priors on the regression trees $(\mathcal{T}_t, \mathcal{M}_t) \overset{\text{iid}}{\sim} \pi_{\mathcal{T}}(\mathcal{T}_t)\,\pi_{\mathcal{M}}(\mathcal{M}_t \mid \mathcal{T}_t)$. We assume independence across the leaf node parameters, that is, $\pi_{\mathcal{M}}(\mathcal{M}_t \mid \mathcal{T}_t) = \prod_{\ell \in \mathcal{L}(\mathcal{T}_t)} \pi_\mu(\mu_{t\ell})$. When possible, π_μ is chosen so that it is conditionally conjugate; for the model (1) we take $\pi_\mu(\mu) = \text{Normal}(\mu \mid 0, \sigma_\mu^2)$.

The most common choice of prior for $\pi_{\mathcal{T}}(\mathcal{T})$ is a *branching process*: starting at depth $d = 0$, each node of depth d is made a branch node with probability $\rho_d = \gamma(1+d)^{-\beta}$ and is made a leaf otherwise. This process iterates until all nodes at depth d are leaves. After the shape of the tree is generated, Chipman, George, and McCulloch (2010) propose generating the splitting rules $[x_{j_b} \le C_b]$ for each $b \in \mathcal{B}(\mathcal{T})$ by (i) sampling a decision rule j_b from $\{1, \ldots, P\}$ such that j_b can produce a "valid" splitting rule and (ii) sampling $C_b \sim \text{Uniform}(X_{ij} : X_i \rightsquigarrow b)$ such that the splitting rule is "valid"; if no such valid X_{ij} exists, we instead force the node to be a leaf. For a rule to be valid, Chipman, George, and McCulloch (2010) require that the rule associate some minimum number of X_i's to each child node (say, 5). A simple alternative, which we use here, is to simply take $j_b = j$ with some probability $s = (s_1, \ldots, s_P)$ (the simplest option being

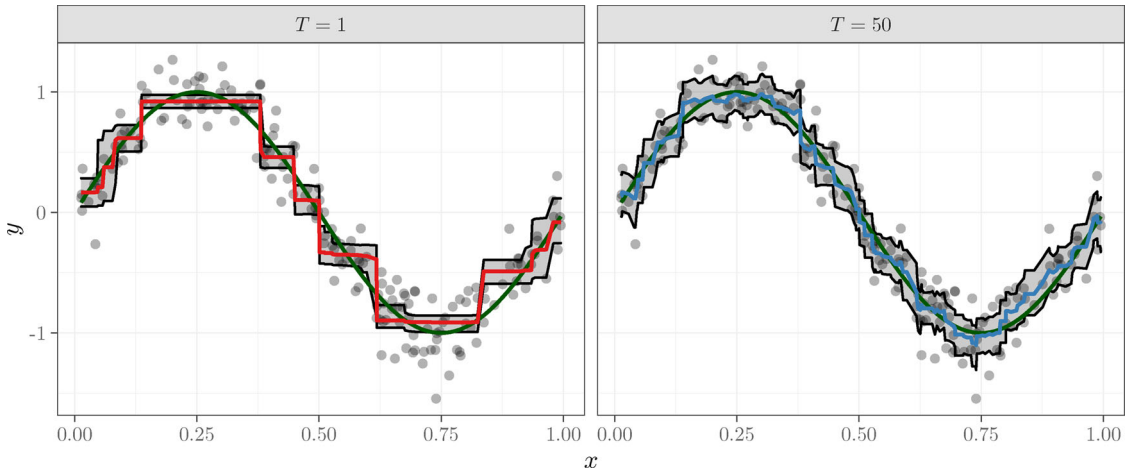

Figure 2. Comparison of the fit of BART with $T = 1$ (left) and $T = 50$ (right) to $Y_i \sim \text{Normal}\{\sin(2\pi X_i), 0.2^2\}$. The function $\sin(2\pi x)$ is given by the dark green line; bands correspond to posterior 95% credible bands.

$s_j = 1/P$) and then sample $C_b \sim \text{Uniform}(L_{bj}, U_{bj})$ where $\prod_{k=1}^{P}[L_{bk}, U_{bk}]$ is the hyperrectangle in $[0,1]^P$ of points x with $x \rightsquigarrow b$.

BART improves upon using a single decision tree $r(x) = g(x; \mathcal{T}, \mathcal{M})$ in several ways. First, as seen in Figure 2, the addition of many decision trees together can *smooth* the estimates of a function; this results in both more accurate predictions and uncertainty quantification. Second, the posterior tends to be easier to explore when many trees are used. Third, as argued heuristically by Chipman, George, and McCulloch (2010) and rigorously by Linero and Yang (2018); Ročková and van der Pas (2020), BART models induce a "shrinkage toward approximately additive models:" samples of BART from the prior tend to involve, at most, lower-order interactions in the covariates. Outside of highly structured problems (e.g., image or speech recognition), this structure is representative of what one often expects to see in practice; for this reason, BART has been seen to perform very well across many problems in prediction (Chipman, George, and McCulloch 2010), survival analysis (Sparapani et al. 2016), and causal inference (Hahn, Murray, and Carvalho 2020; Hill 2011).

2.2. Generalized BART Models

In this article we consider BART models in which the function $r(x)$ enters the model in an arbitrary form. Our approach is applicable to any posterior of the form

$$\pi(r, \eta \mid \mathcal{D}) \propto \exp\left\{\sum_{i=1}^{N} \log f_\eta\big(Y_i \mid r(X_i)\big)\right\} \pi(r)\,\pi(\eta), \qquad (2)$$

where $\log f_\eta(y \mid \lambda)$ is the log-likelihood of some parametric model $\{f_\eta(\cdot \mid \lambda) : \eta \in \mathcal{H}, \lambda \in \mathbb{R}\}$ and η is a vector of nuisance parameters. We note, however, that it is straightforward to replace $\log f_\eta(y \mid \lambda)$ with an arbitrary utility function $R_\eta(y \mid \lambda)$ in our framework. We say that the model is a *generalized BART model* if r has a BART prior.

The seminal work of Chipman, George, and McCulloch (2010) develops the semiparametric regression model $f_\sigma(y \mid \lambda) = \text{Normal}(y \mid \lambda, \sigma^2)$ (for continuous outcome data) and the Binomial probit regression model $f_n(y \mid \lambda) = \text{Binomial}\big(y \mid n, \Phi(\lambda)\big)$ (for binary data). Several other models have also been developed in this framework, such as the Poisson model $f(y \mid \lambda) = \text{Poisson}(y \mid e^\lambda)$ (Murray 2021) for count outcomes and the gamma regression model $f_\alpha(y \mid \lambda) = \text{Gam}(y \mid \alpha, e^\lambda)$ (Linero, Sinha, and Lipsitz 2020) for nonnegative outcomes. Taking the nuisance parameter η to be infinite-dimensional, this also includes several recently proposed BART models for fully-nonparametric regression and survival analysis (George et al. 2019; Henderson et al. 2020; Linero et al. 2022; Li, Linero, and Murray 2023).

The need for generic algorithms for fitting generalized BART models is evinced by the fact that, in some cases, the theoretical development of generalized BART has preceded our ability to implement it. For example, Saha (2023) proposes and studies BART models in the exponential family $f(y \mid \lambda) = \exp\{\lambda\, T(y) - b(\lambda) + c(y)\}$ without providing algorithms for fitting these models. Instead, prior to this work, implementing new instances of the generalized BART model required researchers to either find clever ways of adapting existing Bayesian backfitting algorithms (e.g., by introducing latent variables as in Kindo et al. 2016) or find novel setups for leveraging conjugacy (Murray 2021); both options generally require extensively modifying existing software.

2.3. Bayesian Backfitting in Generalized BART Models

Inference in the semiparametric model (1) proceeds by means of a *Bayesian backfitting* algorithm, which iteratively updates the pairs $(\mathcal{T}_t, \mathcal{M}_t)$ for $t = 1, \ldots, T$. To facilitate forthcoming comparisons with our RJMCMC algorithm, we describe the original Bayesian backfitting algorithm of Chipman, George, and McCulloch (2010) in a slightly unconventional way. In order to update $(\mathcal{T}_t, \mathcal{M}_t)$, we first define $\theta_i = \sum_{k \neq t} g(X_i; \mathcal{T}_k, \mathcal{M}_k)$ so that $Y_i \sim \text{Normal}(\theta_i + \mu_\ell, \sigma^2)$ where ℓ is the leaf such that $X_i \overset{\mathcal{T}_t}{\rightsquigarrow} \ell$. Let $\mathcal{T}_{-t} = \{\mathcal{T}_k : k \neq t\}$ denote the set of all trees except for tree $\mathcal{T}_t$, $\mathcal{M}_{-t} = \{\mu_{k\ell} : k \neq t\}$ denote the set of all leaf node parameters except those associated to tree t, and

$R_i = Y_i - \theta_i$ denote the *backfit residual*. The full conditional $\pi(\mathcal{T}_t \mid \mathcal{T}_{-t}, \mathcal{M}_{-t}, \boldsymbol{X}, \boldsymbol{Y}, \sigma^2)$ of $\mathcal{T}_t$ with $\mathcal{M}_t$ marginalized out is then proportional to

$$\begin{aligned}\pi_{\mathcal{T}}(\mathcal{T}_t) \prod_{\ell \in \mathcal{L}(\mathcal{T}_t)} \int \pi_\mu(\mu) \prod_{i: X_i \overset{\mathcal{T}_t}{\rightsquigarrow} \ell} \text{Normal}(Y_i \mid \theta_i + \mu)\, d\mu \\ = \pi_{\mathcal{T}}(\mathcal{T}_t) \prod_{\ell \in \mathcal{L}(\mathcal{T}_t)} \int \pi_\mu(\mu) \prod_{i: X_i \overset{\mathcal{T}_t}{\rightsquigarrow} \ell} \text{Normal}(R_i \mid \mu, \sigma^2)\, d\mu,\end{aligned} \tag{3}$$

Importantly, when $\pi_\mu(\mu) = \text{Normal}(\mu \mid 0, \sigma_\mu^2)$ this marginal likelihood can be computed in closed-form due to the conjugacy properties of the normal distribution (see Kapelner and Bleich 2016 for details). This allows us to update $\mathcal{T}_t$ using a Metropolis-Hastings algorithm: we sample $\mathcal{T}' \sim q(\mathcal{T}' \mid \mathcal{T}_t)$ from some proposal distribution $q(\cdot \mid \cdot)$ and accept or reject it according to a Metropolis-Hastings ratio based on (3). Generally, the `BIRTH`, `DEATH`, and `CHANGE` proposals of Chipman, George, and McCulloch (1998) (or the more advanced versions of these moves proposed by Pratola 2016) are used for $q(\cdot \mid \cdot)$; we discuss variants of these moves in Section 3.1. While BART was initially developed for semiparametric regression, Chipman, George, and McCulloch (2010) show how to adapt (1) to classification settings using a probit model $Y_i \sim \text{Bernoulli}[\Phi\{r(X_i)\}]$. Inference then proceeds by combining the above Metropolis-Hastings approach with the data augmentation procedure of Albert and Chib (1993).

While convenient and intuitive, the process of going from Y_i to R_i masks a more general expression which allows the Bayesian backfitting algorithm to be generalized; specifically, for a generic parametric model $f_\eta(y \mid \lambda)$ the relevant conditional distribution is

$$\begin{aligned}&\pi(\mathcal{T}_t \mid \mathcal{T}_{-t}, \mathcal{M}_{-t}, \boldsymbol{Y}, \boldsymbol{X}, \sigma^2) \\ &\quad \propto \pi_{\mathcal{T}}(\mathcal{T}_t) \prod_{\ell \in \mathcal{L}(\mathcal{T}_t)} \int \pi_\mu(\mu) \prod_{i: X_i \overset{\mathcal{T}_t}{\rightsquigarrow} \ell} f_\eta(Y_i \mid \theta_i + \mu)\, d\mu.\end{aligned}$$

We can therefore generalize the Bayesian backfitting algorithm if we can make $\pi_\mu(\mu)$ conjugate to $\prod_i f_\eta(Y_i \mid \theta_i + \mu)$. A *generalized Bayesian backfitting* algorithm based on this expression is given in Algorithm 1. Beyond the normal-normal model, this Bayesian backfitting algorithm has been used to implement (i) Poisson loglinear models and multinomial logistic regression (Murray 2021), (ii) gamma regression (Linero, Sinha, and Lipsitz 2020), (iii) nonparametric variance models (Pratola et al. 2020), and (iv) the Cox proportional hazards model (Linero et al. 2022). For example, the Poisson loglinear model takes $f(Y_i \mid \theta_i + \mu) = \text{Poisson}(Y_i \mid e^{\theta_i + \mu})$, for which the log-gamma distribution $\mu_{t\ell} \sim \log \text{Gam}(a_0, b_0)$ is a conditionally conjugate prior; specifically, we have

$$\begin{aligned}\pi_\mu(d\mu) \prod_{X_i \overset{\mathcal{T}}{\rightsquigarrow} \ell} f_\eta(Y_i \mid \theta_i + \mu) = \frac{b_0^{a_0} \exp(\sum_{X_i \rightsquigarrow \ell} Y_i \theta_i)}{\Gamma(a_0) \prod_{X_i \rightsquigarrow \ell} Y_i!} \\ \times \exp\{\mu(a_0 + \sum_{X_i \rightsquigarrow \ell} Y_i) - e^\mu (b_0 + \sum_{X_i \rightsquigarrow \ell} e^{\theta_i})\},\end{aligned}$$

Algorithm 1 One iteration of a generalized Bayesian backfitting algorithm for updating $(\mathcal{T}_t, \mathcal{M}_t)$

Input: $\{\mathcal{T}_t, \mathcal{M}_t : t = 1, \ldots, T\}$, $\boldsymbol{Y}, \boldsymbol{X}, \eta, q(\cdot \mid \cdot)$

1: **for** $t = 1, \ldots, T$ **do**
2: Compute $\theta_i \leftarrow \sum_{k \neq t} g(X_i; \mathcal{T}_k, \mathcal{M}_k)$ for $i = 1, \ldots, N$.
3: Propose a new tree structure $\mathcal{T}' \sim q(\mathcal{T}' \mid \mathcal{T}_t)$.
4: Compute the integrated likelihoods $\Lambda(\mathcal{T}_t)$ and $\Lambda(\mathcal{T}')$ where

$$\Lambda(\mathcal{T}) = \prod_{\ell \in \mathcal{L}(\mathcal{T})} \int \pi_\mu(\mu) \prod_{i: X_i \overset{\mathcal{T}}{\rightsquigarrow} \ell} f_\eta(Y_i \mid \theta_i + \mu)\, d\mu.$$

5: Compute the acceptance probability

$$A = \min\left\{\frac{\Lambda(\mathcal{T}')\, \pi_{\mathcal{T}}(\mathcal{T}')\, q(\mathcal{T}_t \mid \mathcal{T}')}{\Lambda(\mathcal{T}_t)\, \pi_{\mathcal{T}}(\mathcal{T}_t)\, q(\mathcal{T}' \mid \mathcal{T}_t)}, 1\right\}.$$

6: With probability A, set $\mathcal{T}_t \leftarrow \mathcal{T}'$; otherwise, leave $\mathcal{T}_t$ unchanged.
7: Sample $\mathcal{M}_t$ from its full conditional distribution.
8: **end for**

which we recognize as proportional to a $\log \text{Gam}(a_0 + \sum_{i: X_i \rightsquigarrow \ell} Y_i, b_0 + \sum_{i: X_i \rightsquigarrow \ell} e^{\theta_i})$ density.

Unfortunately, for many models of interest it will not be possible to find a π_μ which is conjugate to $\prod_i f_\eta(Y_i \mid \theta_i + \mu)$. The class of models for which this is feasible is, in fact, surprisingly narrow: for example, one cannot leverage the conjugacy of the beta distribution to the binomial likelihood to construct a generalized Bayesian backfitting algorithm. One possible solution, which was used by Chipman et al. (2021) to implement a *monotone* variant of BART, is to compute $\int \pi_\mu(\mu) \prod_{i: X_i \overset{\mathcal{T}}{\rightsquigarrow} \ell} f_\eta(Y_i \mid \theta_i + \mu)\, d\mu$ numerically and then sample $\mathcal{M}_t$ using a discrete approximation to the posterior; this introduces new problems, as it requires both approximating the posterior on a grid and evaluating the likelihood at a large number of grid points. In the following section, we show how to bypass the need for conjugacy via RJMCMC.

3. Implementing Generalized BART with RJMCMC

We now show how to implement the generalized BART model using a generic reversible jump Markov chain Monte Carlo (RJMCMC) algorithm. Because RJMCMC has a reputation for being difficult to implement, and given the breadth of applications we want to consider, it is essential that the algorithms we propose depend on neither tuning parameters nor the details of a given problem.

We also provide a "default" prior for routine use which works well across many problems. This is essential for widespread adoption of our approach, as prior specification is a barrier to the use of Bayesian nonparametric methods by nonexperts.

3.1. Reversible Jump Markov Chain Monte Carlo on Trees

Throughout this section, we consider updating a regression tree $(\mathcal{T}_t, \mathcal{M}_t)$ with the quantities $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_N)$ and η fixed, where

$\theta_i = \sum_{k \neq t} g(X_i; \mathcal{T}_k, \mathcal{M}_k)$. To lighten notation, we will suppress dependence of most quantities in this section on $(\boldsymbol{Y}, \boldsymbol{X}, \eta, \boldsymbol{\theta})$, and we will drop the index t from $(\mathcal{T}_t, \mathcal{M}_t)$. Conditional on the θ_i's and the nuisance parameter vector η, the model for the data is $Y_i \sim f_\eta\{y \mid \theta_i + g(X_i; \mathcal{T}, \mathcal{M})\}$. The likelihood is then given by

$$\mathscr{L}(\mathcal{T}, \mathcal{M}) = \prod_{\ell \in \mathcal{L}(\mathcal{T})} \prod_{i: X_i \rightsquigarrow \ell} f_\eta(Y_i \mid \theta_i + \mu_\ell). \tag{4}$$

This quantity plays the same role in our RJMCMC scheme as the integrated likelihood $\Lambda(\mathcal{T})$ does in the generalized Bayesian backfitting algorithm of Section 2.2.

We consider the following Metropolis-Hastings proposals, which are directly analogous to standard proposals for the Bayesian CART of Chipman, George, and McCulloch (1998); our proposals operate on $(\mathcal{T}, \mathcal{M})$ rather than just $\mathcal{T}$.

BIRTH Randomly choose a leaf node $\ell \in \mathcal{L}(\mathcal{T})$ and sample a splitting rule $[x_{j\ell} \leq C_\ell]$. Convert ℓ from a leaf to a branch with two leaf children and sample $(\mu'_{\ell L}, \mu'_{\ell R}) \sim G_{\text{BIRTH}}(\mu'_{\ell L}, \mu'_{\ell R})$ where $G_{\text{BIRTH}}(\cdot, \cdot)$ is a proposal distribution to be described in Section 3.2.

DEATH Randomly choose a branch node $b \in \text{NOG}(\mathcal{T})$ and convert b from a branch to a leaf (deleting its children). Then sample $\mu'_b \sim G_{\text{DEATH}}(\mu')$ where $G_{\text{DEATH}}(\cdot)$ is a proposal distribution to be described in Section 3.2.

CHANGE Randomly choose a branch node $b \in \text{NOG}(\mathcal{T})$ and sample a new splitting rule $[x_{j'_b} \leq C'_\ell]$ from the prior. Then sample new leaf node predictions $(\mu'_{bL}, \mu'_{bR}) \sim G_{\text{CHANGE}}(\mu'_{bL}, \mu'_{bR})$ where $G_{\text{CHANGE}}(\cdot, \cdot)$ is a proposal distribution to be described in Section 3.2.

We now give a valid Metropolis-Hastings acceptance ratio for the BIRTH, DEATH, and CHANGE moves. It is useful to define, for a given node n (not necessarily a leaf), the quantity

$$\mathscr{F}(n \mid \mathcal{T}, \mu) = \pi_\mu(\mu) \prod_{i: X_i \rightsquigarrow n} f_\eta(Y_i \mid \theta_i + \mu). \tag{5}$$

Using (5), the likelihood (4) is given by $\mathscr{L}(\mathcal{T}, \mathcal{M}) = \prod_{\ell \in \mathcal{L}(\mathcal{T})} \mathscr{F}(\ell \mid \mathcal{T}, \mu_\ell)/\pi_\mu(\mu_\ell)$.

Proposition 1. Let $p_{\text{BIRTH}}(\mathcal{T})$ and $p_{\text{DEATH}}(\mathcal{T})$ denote the probability of proposing BIRTH and DEATH moves to modify $\mathcal{T}$, respectively and let $|A|$ denote the size of a finite set A. For the BIRTH, DEATH, and CHANGE moves, accepting the proposed change with probability $1 \wedge R$ leaves the posterior invariant, where

$$R_{\text{BIRTH}} = \frac{\rho_d(1-\rho_{d+1})^2}{(1-\rho_d)} \cdot \frac{\mathscr{F}(\ell L \mid \mathcal{T}', \mu'_{\ell L})\, \mathscr{F}(\ell R \mid \mathcal{T}', \mu'_{\ell R})}{\mathscr{F}(\ell \mid \mathcal{T}, \mu_\ell)} \cdot \frac{p_{\text{DEATH}}(\mathcal{T}') \,|\text{NOG}(\mathcal{T}')|^{-1}}{p_{\text{BIRTH}}(\mathcal{T}) \,|\mathcal{L}(\mathcal{T})|^{-1}} \cdot \frac{G_{\text{DEATH}}(\mu_\ell)}{G_{\text{BIRTH}}(\mu'_{\ell L}, \mu'_{\ell R})}$$

$$R_{\text{DEATH}} = \frac{(1-\rho_d)}{\rho_d(1-\rho_{d+1})^2} \cdot \frac{\mathscr{F}(b \mid \mathcal{T}', \mu'_b)}{\mathscr{F}(bL \mid \mathcal{T}, \mu_{bL})\, \mathscr{F}(bR \mid \mathcal{T}, \mu_{bR})} \cdot \frac{p_{\text{BIRTH}}(\mathcal{T}') \,|\mathcal{L}(\mathcal{T})|^{-1}}{p_{\text{DEATH}}(\mathcal{T}) \,|\text{NOG}(\mathcal{T}')|^{-1}} \cdot \frac{G_{\text{BIRTH}}(\mu_{bL}, \mu_{bR})}{G_{\text{DEATH}}(\mu'_b)}$$

and

Algorithm 2 One iteration of reversible jump Bayesian backfitting

Input: $\boldsymbol{Y}, \boldsymbol{X}, \eta, \{\mathcal{T}_t, \mathcal{M}_t\}_{t=1}^T$

1: Set $\theta_i \leftarrow \sum_{t=1}^T g(X_i; \mathcal{T}_t, \mathcal{M}_t)$ for $i = 1, \ldots, N$.
2: **for** $i = 1, \ldots, T$ **do**
3: Set $\theta_i \leftarrow \theta_i - g(X_i; \mathcal{T}_t, \mathcal{M}_t)$ for $i = 1, \ldots, N$.
4: Sample $(\mathcal{T}', \mathcal{M}')$ by randomly choosing between the BIRTH, DEATH, and CHANGE steps.
5: Compute the associated acceptance probability from Proposition 1 with $(\mathcal{T}_t, \mathcal{M}_t)$ in place of $(\mathcal{T}, \mathcal{M})$ and accept $(\mathcal{T}', \mathcal{M}')$ with that probability.
6: Sample $\mathcal{M}_t$ targeting its full conditional using an MH step.
7: Set $\theta_i \leftarrow \theta_i + g(X_i; \mathcal{T}_t, \mathcal{M}_t)$ for $i = 1, \ldots, N$.
8: **end for**

$$R_{\text{CHANGE}} = \frac{\mathscr{F}(bL \mid \mathcal{T}', \mu'_{bL})\, \mathscr{F}(bR \mid \mathcal{T}', \mu'_{bR})}{\mathscr{F}(bL \mid \mathcal{T}, \mu_{bL})\, \mathscr{F}(bR \mid \mathcal{T}, \mu_{bR})} \cdot \frac{G_{\text{CHANGE}}(\mu_{bL}, \mu_{bR})}{G_{\text{CHANGE}}(\mu'_{bL}, \mu'_{bR})}.$$

Proposition 1 can be established by applying the results of Green (1995) after introducing a suitable dimension-matching transformation. In the supplementary material we give a derivation of R_{BIRTH} (R_{DEATH} being the inverse move and R_{CHANGE} not requiring RJMCMC). Algorithm 2 summarizes the proposed approach.

3.2. Choice of the Proposal Distribution

The success of Algorithm 2 depends crucially on the quality of the proposal mechanisms $G_{\text{DEATH}}(\mu_\ell)$, $G_{\text{CHANGE}}(\mu_\ell)$, and $G_{\text{BIRTH}}(\mu_{\ell L}, \mu_{\ell R})$. As part of the joint proposal for $(\mathcal{T}, \mathcal{M})$, these proposals are allowed to depend on $\mathcal{T}$ as well as $\boldsymbol{Y}$ and $\boldsymbol{\theta}$. An effective proposal should be both accurate and applicable to arbitrary models. To meet this need, we choose the proposal to be a Normal(m, v^2) distribution constructed using the *Laplace approximation* (see, e.g., Gelman et al. 2013, chap. 13), which requires only that we have access to the first and second derivatives of $\log f_\eta(y \mid \lambda)$. Recall that we define $U_\eta(y \mid \lambda) = \frac{\partial}{\partial \lambda} \log f_\eta(y \mid \lambda)$ and $\mathcal{J}_\eta(y \mid \lambda) = -\frac{\partial}{\partial \lambda} U_\eta(y \mid \lambda)$. Then, for example, in the BIRTH step we propose $\mu'_{\ell L} \sim \text{Normal}(m_{\ell L}, v^2_{\ell L})$ where

$$m_{\ell L} = \arg\max_\mu \sum_{i: X_i \rightsquigarrow \ell L} \log f_\eta(Y_i \mid \theta_i + \mu) + \log \pi_\mu(\mu) \quad \text{and}$$

$$v_{\ell L}^{-2} = \sum_{i: X_i \rightsquigarrow \ell L} \mathcal{J}_\eta(Y_i \mid \theta_i + m_{\ell L}) - \frac{d^2}{d\mu^2} \log \pi_\mu(\mu)|_{\mu = m_{\ell L}}. \tag{6}$$

The values $m_{\ell L}$ and $v_{\ell L}$ can be computed using, for example, Newton's method: starting from $\mu_{\ell L}$, we perform the update

$$m_{\ell L} \leftarrow m_{\ell L} + \frac{\sum_i U_\eta(Y_i \mid \theta_i + m_{\ell L}) + \frac{d}{d\mu} \log \pi_\mu(\mu)|_{\mu = m_{\ell L}}}{\sum_i \mathcal{J}_\eta(Y_i \mid \theta_i + m_{\ell L}) - \frac{d^2}{d\mu^2} \log \pi_\mu(\mu)|_{\mu = m_{\ell L}}},$$

until some stopping criterion is reached. Alternatively, the *Fisher scoring* algorithm replaces $\mathcal{J}_\eta(Y_i \mid \theta_i + \mu)$ with $\mathcal{I}_\eta(\theta_i + \mu) =$

Algorithm 3 Fisher scoring for computing m_ℓ and v_ℓ for the proposal distribution of μ_ℓ

Input: $\ell, \mathcal{T}, \boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{\theta}, \eta, \sigma^2_\mu$

Let: $U_\eta(\mu, \ell) = \sum_{i:X_i \rightsquigarrow \ell} U_\eta(Y_i \mid \theta_i + \mu) - \mu/\sigma^2_\mu$ and $\mathcal{I}_\eta(\mu, \ell) = \sum_{i:X_i \rightsquigarrow \ell} \mathcal{I}(\theta_i + \mu) + 1/\sigma^2_\mu$.

1: Initialize m_ℓ: for `BIRTH` moves, use the value of μ from the parent node; for `DEATH` moves use $(\mu_{\ell L} + \mu_{\ell R})/2$
2: **while** $|U_\eta(m_\ell, \ell)| > \mathcal{I}_\eta(m_\ell, \ell)^{1/2}/10$ **do**
3: $\quad m_\ell \leftarrow m_\ell + U_\eta(m_\ell, \ell)/\mathcal{I}_\eta(m_\ell, \ell)$
4: **end while**
5: $v \leftarrow \mathcal{I}_\eta(m_\ell, \ell)^{-1/2}$
6: **return** (m_ℓ, v_ℓ)

$\mathbb{E}\{\mathcal{J}_\eta(Y_i \mid \theta_i + \mu) \mid \boldsymbol{\theta}, \eta, \mu\}$; in our experience Fisher scoring tends to be more robust than Newton's method, and we will use Fisher scoring whenever it is feasible. Note also that we do not need to compute (6) exactly, as we just want reasonable Gaussian approximations to the full conditional distributions of the leaf node parameters; any inaccuracies are naturally corrected for by their effect on the Metropolis-Hastings acceptance probability. Algorithm 3 gives the Fisher scoring algorithm we used in our illustrations, assuming $\pi_\mu(\mu) = \text{Normal}(\mu \mid 0, \sigma^2_\mu)$; to use Newton's method instead, simply replace $\mathcal{I}_\eta(\lambda)$ with $\mathcal{J}_\eta(Y_i \mid \lambda)$ where appropriate.

Conveniently, the use of a highly informative Gaussian prior for the leaf parameters has benefits for the accuracy of the Laplace approximation. First, the likelihood is encouraged to be nearly Gaussian even if little data is associated to a particular node. Second, because the prior shrinks the μ's toward zero, Newton's method and Fisher scoring generally converge very quickly even if we initialize the algorithm naively at $\mu = 0$.

Remark 1. Note that we are using the Laplace approximation only to generate a reasonable proposal distribution, and that our Metropolis-Hastings algorithm leaves the posterior invariant *exactly*. This approach can be contrasted with using the Laplace approximation to the marginal likelihood to approximate the marginal likelihood of a given tree in a Metropolis-Hastings algorithm, which does *not* leave the posterior invariant. Note also that it is important that the same Laplace approximation is used for both the `BIRTH` and `DEATH` steps, as this is essential for the `BIRTH` move to be the inverse of the `DEATH` move and vice-versa.

Remark 2. We have found that, depending on the model, our RJMCMC scheme can be sensitive to the choice of σ_μ: if a value of σ^2_μ is chosen which is too large, the algorithm gets "stuck" early on and does not progress toward the stationary distribution. To fix this, we have found that it is essential to slowly increase σ_μ from 0 to its desired value for some fraction of the burn-in phase of the MCMC (we used 25% of the burn-in iterations for this).

3.3. Choice of the Prior Distribution

There are three prior specifications we must make: the prior for the trees ($\pi_\mathcal{T}$), the leaf node parameters ($\pi_\mathcal{M}$), and the nuisance parameter η. For $\pi_\mathcal{T}$, we use Chipman, George, and McCulloch's (2010) defaults ($\gamma = 0.95$, $\beta = 2$, and $T \in 50, 200$), though choosing T via cross-validation may be beneficial. Additionally, as suggested by Linero (2018), we recommend replacing the uniform distribution for j_b described in Section 2.1 with $j_b \sim \text{Categorical}(s)$ where the hyperparameter s is given a Dirichlet$(\xi/P, \ldots, \xi/P)$ hyperprior; this allows the model to filter out irrelevant variables much more effectively than the original BART prior. As η is problem specific, we offer no general guidelines on the selection of its prior.

The choice of π_μ is less straight-forward. We are no longer constrained by the conjugacy requirements of the generalized Bayesian backfitting algorithm, and so for simplicity we take $\pi_\mu(\mu) = \text{Normal}(\mu \mid 0, \sigma^2_\mu)$. The appropriate scale for σ^2_μ will typically be problem specific, making it difficult to make a general recommendation. One strategy we have found to work well is to use a half-Cauchy prior $\pi(\sigma_\mu) \propto (1 + \sigma^2_\mu/c^2)^{-1}$ for some small c (say, $c = k/\sqrt{T}$ where T is the number of trees and k is 1 or 0.1). We use a Metropolis-within-Gibbs strategy to update σ_μ under the half-Cauchy prior; specifically, we propose σ_μ from the full conditional associated with an improper (conditionally conjugate) Uniform$(0, \infty)$ prior on σ_μ.

4. Posterior Contraction in Generalized BART Models

There is a growing literature that studies the rate at which the posterior distribution of a BART model contracts (Linero and Yang 2018; Ročková and van der Pas 2020; Orlandi et al. 2021; Linero et al. 2022; Li, Linero, and Murray 2023; Saha 2023), with the theory spanning exponential families to nonparametric density estimation. We add to this literature by establishing posterior concentration rates for a class of generalized BART models that contains most of the illustrations given in this manuscript, under the simplifying assumption that the nuisance parameter η is known. To link prior work on the theoretical properties of BART to the generalized BART model, we assume a bound on the Kullback-Leibler divergence between densities $f_\eta(y \mid \mu)$ and $f_\eta(y \mid \mu + \Delta)$ of the form:

$$KL_p\{f_\eta(\cdot \mid \mu) \| f_\eta(\cdot \mid \mu + \Delta)\} = \int f_\eta(y \mid \mu) \left(\log \frac{f_\eta(y \mid \mu)}{f_\eta(y \mid \mu + \Delta)} \right)^p \nu(dy) \leq C_\eta \, \Delta^2, \tag{A1}$$

for all $|\Delta| \leq 1$, where C_η is a universal constant depending on the model, $p = 1, 2$, and $\nu(dy)$ denotes an appropriate dominating measure.

We make the following assumptions about the unknown function of interest $r_0(x)$.

Condition F. The function $r_0 : \mathbb{R}^P \to \mathbb{R}$ is α-Hölder smooth for some $\alpha \in (0, 1]$ on $\mathbb{R}^P$, and depends on only D of the coordinates. Additionally, the X_i's are supported on $[0, 1]^P$.

In other words, $r_0(x)$ is a smooth function of P predictors, with only D predictors relevant. Condition F can be relaxed in several ways: For example, we can replace Condition F with a *low-order interaction* assumption $r_0(x) = \sum_{v=1}^V r_{0v}(x)$, with improved rates if the individual functions $r_{0v}(x)$ are themselves sparse. Alternatively, we can weaken the assumption that $\alpha \leq 1$ by using *soft* decision trees (Linero and Yang 2018; Ročková and

van der Pas 2020). Our proof strategy can also accommodate P diverging nearly-exponentially in N, provided that D remains bounded, at the expense of an additional variable selection term $\sqrt{\frac{D \log P}{N}}$ term in the following error rates. Our results are summarized in the following theorem, which is proven in the supplementary material.

Theorem 1. Suppose $(Y_i, X_i : i = 1, \ldots, n)$ are iid pairs with $X_i \sim F_X$ and $[Y_i \mid X_i = x] \sim f_\eta\{y \mid r_0(x)\}$ where $r_0(x)$ satisfies Condition F, the family $\{f_\eta(\cdot \mid \mu) : \mu \in \mathbb{R}\}$ satisfies Condition A1, and the prior $r \sim \Pi$ satisfies Condition P in the supplementary material. Then there exists a constant $M > 0$ such that

$$\Pi\{d(r, r_0) \geq M\epsilon_n \mid (Y_i, X_i : i = 1, \ldots, n)\} \to 0 \quad \text{in probability}$$

as $n \to \infty$, where $d(r, r_0) = \int\int |f_\eta\{y \mid r(x)\} - f_\eta\{y \mid r_0(x)\}| \, \nu(dy) \, F_X(dx)$ is the F_X-integrated total variation distance, $\Pi\{\cdot \mid (Y_i, X_i : i = 1, \ldots, n)\}$ is the posterior distribution of r given $(Y_i, X_i : 1 = 1, \ldots, n)$, and $\epsilon_n = (\log n / n)^{\alpha/(2\alpha + D)}$.

In the supplementary material, we verify Condition A1 for several of the families examined in this article, leading to the following corollary.

Corollary 1. Condition A1 holds for the following models:

- Exponential dispersion families $f_\eta(y \mid \mu) = \exp\left\{\frac{y\mu - b(\mu)}{\eta} + c(y; \eta)\right\}$ such that the $V(\mu) = b''(\mu)$ is bounded (e.g., normal, Bernoulli-logit, and beta regression models).
- Location-scale families $f_\eta(y \mid \mu) = \frac{1}{\eta} f\left(\frac{y - \mu}{\eta}\right)$ with $f(x)$ twice continuously-differentiable and satisfying Condition T in the supplementary material (e.g., the normal, logistic, and Student's t_ν location-scale families).
- The Bayesian quantile regression forests of Kindo et al. (2016), where $f_\eta(y \mid \mu)$ is an asymmetric Laplace distribution with scale τ, quantile q, and $\eta = (\tau, q)$.

Hence, under Condition F and Condition P, Theorem 1 applies to these models.

Remark 3. Theorem 1 prioritizes conciseness and simplicity of the proof rather than generality. For many applications, such as Poisson regression with the canonical link or the structured variance model used in Section 5.2, assumption A1 is too strong. In particular, we would need to replace C_η with $C_\eta(\mu)$ where $C_\eta(\mu)$ is not bounded, which makes it difficult to construct an appropriate high-mass low-entropy sieve; one can bypass this by either using a bounded link function (Saha 2023) or by appealing to fractional posteriors (Linero and Yang 2018; Bhattacharya, Pati, and Yang 2019). One might also consider more relevant metrics than the L_1-integrated total variation $d(r, r_0)$, such as the empirical L_2 distance defined by $\|r - r_0\|_n^2 = \frac{1}{n}\sum_{i=1}^n \int \{r_0(X_i) - r(X_i)\}^2$. The conclusion of Theorem 1 holds with $d(r, r_0)$ replaced by $\|r - r_0\|_n$ provided, roughly speaking, that there exist sufficiently powerful tests of the hypothesis $H_0 : r = r_0$ against the alternative $H_1 : \|r - r_0\|_n > C\epsilon_n$ for some $C > 0$ (see chap. 8 of Ghosal and van der Vaart 2017), which is the case for both the Bernoulli-Logit and normal regression problems. We note that, while the metrics above are defined in terms of F_X, we do not require that F_X be known to the analyst.

5. Illustrations

5.1. Sanity Checks: Nonparametric Regression and Classification

To understand if there are any striking limitations of the RJMCMC approach, we apply it to two problems for which there are existing algorithms: the semiparametric regression problem (1) and nonparametric classification with the logistic link. Going in, we should expect that the RJMCMC algorithm should be inferior in terms of mixing to the algorithm of Chipman, George, and McCulloch (2010), as RJMCMC does not use the integrated likelihood (which is available in closed form) to propose changes. For logistic regression it is less clear what to expect, as the algorithm of Sparapani, Spanbauer, and McCulloch (2021), which we compare to, makes use of a data augmentation strategy of Holmes and Held (2006) that itself can substantially slow down mixing. We consider a typical benchmark function for BART methods which takes

$$r_F(x) = 10 \sin(\pi \, x_1 \, x_2) + 20(x_3 - 0.5)^2 + 10 \, x_4 + 5 \, x_5, \quad (7)$$

with X_{ij} irrelevant for all $j > 5$.

We compare our RJMCMC algorithm to the methods implemented in the R package `BART`. We choose this package specifically because, to the best of our knowledge, it is the only publicly available package that implements BART with the logistic link; like our default prior, it also implements the sparsity-inducing Dirichlet hyperprior of Linero (2018). We remark that `BART` differs slightly in how the prior is specified, and for this reason we do not expect that the predictive performance will be precisely the same between the two methods; if RJMCMC performs better, however, this gives us some assurance that the algorithm is correct and mixes well enough to produce reasonable predictions. In all cases we compare RJMCMC and `BART` on a single simulated dataset, however, the results we present are typical of all replications of the simulations we have performed.

We first consider the semiparametric regression problem (1) with nuisance parameter $\eta = \sigma$ and $r_0(x) = r_F(x)$ where $r_F(x)$ is given by (7) with $\sigma^2 = 1$, $N = 500$, and $P = 20$. For both methods we ran the Bayesian backfitting algorithm for 10,000 iterations, with the first 5000 discarded to burn-in. For each iteration, we computed the mean squared error MSE $= N_{\text{test}}^{-1} \sum_i \{Y_i^\star - r(X_i^\star)\}^2$ where $(Y_i^\star, X_i^\star)$ is a collection of 500 heldout samples.

Figure 3 displays the samples of MSE for both approaches. Both methods are similar in terms of mixing; in particular, the mixing of RJMCMC does not appear to be appreciably worse. We also see that RJMCMC results in a lower MSE on average.

For the classification problem we take $Y_i \sim \text{Bernoulli}[\mathfrak{s}\{r_0(X_i)\}]$ where $\eta = \emptyset$, $r_0(x) = \frac{r_F(x) - 14}{5}$, and $\mathfrak{s}(z) = (1 + e^{-z})^{-1}$ is the logistic function; this normalization of $r_F(x)$ was chosen so that $r_0(X_i)$ has approximately mean 0 and variance 1.

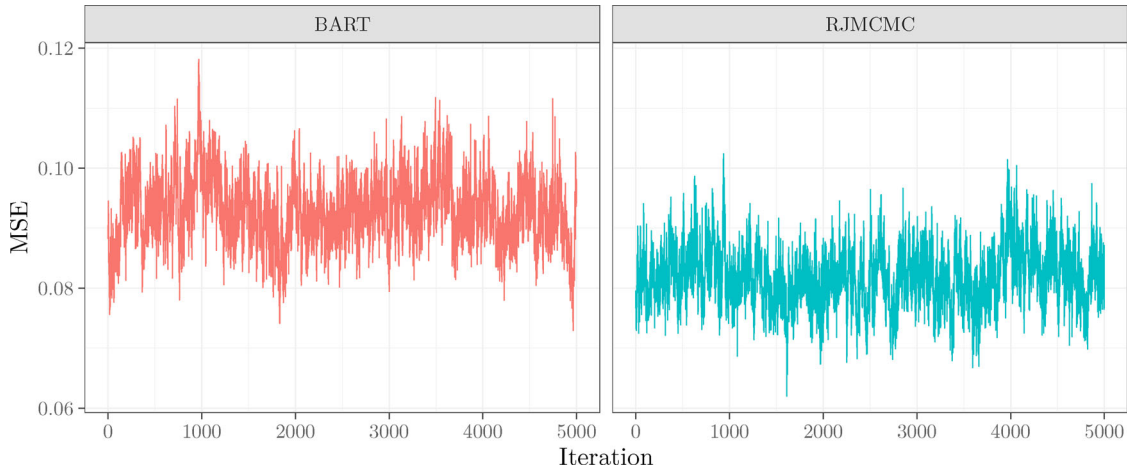

Figure 3. Traceplot of the heldout mean squared error for the semiparametric regression model. Left: results as implemented in the `BART` package. Right: results using our RJMCMC algorithm.

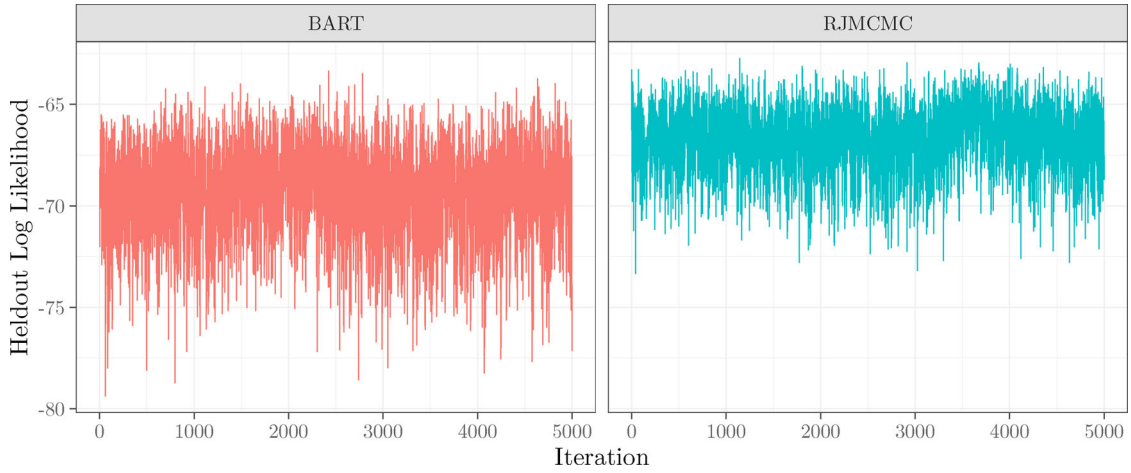

Figure 4. Traceplot of the heldout log-likelihood for the logistic link BART model. Left: results using the `BART` package. Right: results using our RJMCMC algorithm.

Data augmentation can be applied to fit BART classification models using the logistic link, using the scale-mixtures-of-normals approach of Holmes and Held (2006), the Pólya-gamma approach of Polson, Scott, and Windle (2013), or the gamma augmentation approach of Murray (2021). The downside of these approaches is that data augmentation can slow down mixing substantially, especially in cases where the outcome distribution is highly imbalanced (Johndrow et al. 2019).

Our RJMCMC algorithm removes the need for data augmentation entirely, and requires only that we plug in the likelihood, score, and Fisher information; for the Bernoulli-logit model $Y_i \sim \text{Bernoulli}\{\mathfrak{s}(\lambda)\}$, this is given by

$$\log f(y \mid \lambda) = y \log \mathfrak{s}(\lambda) + (1-y)\log\{1-\mathfrak{s}(\lambda)\},$$

$$U(y \mid \lambda) = y - \mathfrak{s}(\lambda), \qquad \text{and} \qquad \mathcal{I}(\lambda) = \mathfrak{s}(\lambda)\{1-\mathfrak{s}(\lambda)\}.$$

We fit the classification model using both the `BART` package (which uses the data augmentation scheme of Holmes and Held 2006) and our RJMCMC algorithm. For each iteration we record the heldout log-likelihood $\sum_i Y_i^\star \log \mathfrak{s}\{r(X_i^\star)\} + (1-Y_i^\star)\log[1-\mathfrak{s}\{r(X_i^\star)\}]$,where $(Y_i^\star, X_i^\star)$ are 500 heldout observations. In Figure 4 we give traceplots of the heldout log-likelihood for both methods, and we again observe that RJMCMC does not mix appreciably worse than `BART` while producing better predictions on the heldout data.

Finally, we compare the RJMCMC algorithm to the data augmentation approach in terms of computational efficiency. In Figure 5 we present the time to fit 20 replicate datasets using the `BART` package and our RJMCMC approach; both approaches use $T = 200$ trees and run 10,000 iterations of their respective Markov chains. We again stress that direct comparisons between RJMCMC and any given package should be taken with a grain of salt due to differences in the implementation details, but it is reassuring that the RJMCMC scheme is slightly faster on a per-iteration basis.

5.2. Variance Modeling

We now turn our attention to generalized BART models that cannot be fit with existing Bayesian backfitting algorithms. A common concern when constructing a regression model is *heteroscedasticity* of the error distribution. A selling point of generalized linear models, for example, is that they handle the mean-variance relationships inherent to proportion or count data.

In this section we consider BART models which allow for a specified (but essentially arbitrary) mean-variance relationship using a Gaussian working model. Specifically, we set

$$[Y_i \mid X_i] \sim \text{Normal}\{m_i, \phi\, V(m_i)\} \tag{8}$$

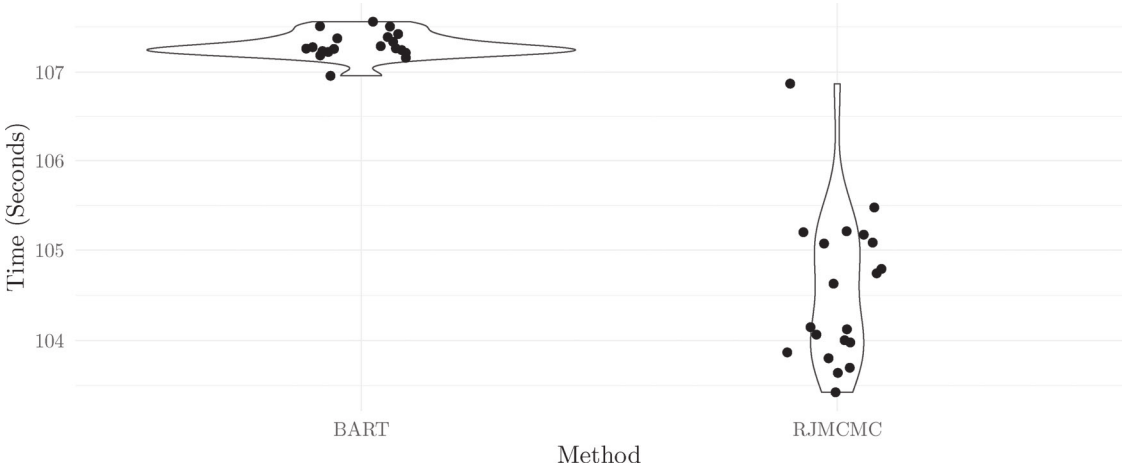

Figure 5. Comparison of time to collect 10,000 posterior samples for the logistic regression model using the `BART` package (left) and RJMCMC (right) for 20 synthetic datasets.

where $m_i = g\{r(X_i)\}$. Here, $g(\mu)$ and $V(m)$ are user-specified functions which relate $r(x)$, the mean, and the variance. In this case, $\eta = (\phi, V(\cdot))$. In the supplementary material we show that $U_\eta(y \mid \lambda)$ and $\mathcal{I}_\eta(\lambda)$ are given by

$$U_\eta(y \mid \lambda) = \left(-\frac{V'(m)}{2V(m)} + \frac{V'(m)(y-m)^2}{2\phi V(m)^2} + \frac{y-m}{\phi V(m)}\right) g'(\lambda)$$
$$\mathcal{I}_\eta(\lambda) = \left(\frac{V'(m)^2}{2V(m)^2} + \frac{1}{\phi V(m)}\right) g'(\lambda)^2$$

where $m = g(\lambda)$, $V'(m) = \frac{d}{dm}V(m)$, and $g'(\lambda) = \frac{d}{d\lambda}g(\lambda)$. Additionally, the full conditional of $\tau = \phi^{-1}$ is $\text{Gam}\{\tau \mid N/2, 1/2\sum_i (Y_i - m_i)^2/V(m_i)\} \times \pi(\tau)$. Plugging these expressions into our generic RJMCMC scheme, we can fit a BART model to any mean-variance relationship.

To illustrate, we generated $Y_i \sim \text{Poisson}(m_i)$ with $m_i = \exp\{r(X_i)\}$, which implies the mean-variance relation $V(m) = m$. We avoid fitting a Poisson loglinear model in order to (i) illustrate the ability of the structured variance model to work even when the outcome model is misspecified and (ii) because the Poisson model does not allow for overdispersion, which is a common concern in practice. We took $m_i = \exp\left\{2 + \left(\frac{r_F(X_i)-14}{5}\right)\right\}$ with $r_F(x)$ given by (7). We compare the following three BART implementations.

- **bartMachine**: A standard BART model which takes $Y_i \sim \text{Normal}\{r(X_i), \sigma^2\}$, fit using the `bartMachine` package.
- **rbart**: A heteroscedastic BART model of Pratola et al. (2020), which takes $Y_i \sim \text{Normal}\{r(X_i), \sigma^2(X_i)\}$. This model was fit using the `rbart` package.
- **RJMCMC**: The BART model (8) which takes $Y_i \sim \text{Normal}\{e^{r(X_i)}, \phi e^{r(X_i)}\}$.

The goal of this comparison is to determine (i) if our RJMCMC algorithm is capable of fitting (8), (ii) if there is a substantial gain in performance from modeling the variance, and (iii) if there is additional gain from correctly specifying the mean-variance relationship. We compare methods based on the root mean-squared error $\text{RMSE} = \sqrt{N_{\text{test}}^{-1}\sum_i (m_i^\star - \widehat{m}_i^\star)^2}$ on a collection of 500 heldout samples $(X_i^\star, Y_i^\star)$. Results are given in Figure 6.

We see from the traceplot of RMSE that the RJMCMC model results in a substantially lower RMSE on heldout data, and that the RMSE mixes well for all three methods; while `rbart` is able to account for heteroscedasticity, it gives only a modest improvement over `bartMachine`. For both `bartMachine` and `rbart` we see that the models tend to underestimate m_i when m_i is large. The overall RMSEs using the Bayes estimate for each method are 6.67 (`bartMachine`), 5.71 (`rbart`), and 3.25 (RJMCMC).

5.3. Accelerated Failure Time Models

We now illustrate our approach on several *accelerated failure time* (AFT) models for survival analysis (Wei 1992). Let T_i denote a survival time and let C_i denote the censoring time such that we observe $Y_i = \min\{T_i, C_i\}$ and $\delta_i = I(Y_i = T_i)$. The accelerated failure time model takes $\log T_i = r(X_i) + \sigma\,\epsilon_i$where ϵ_i belongs to some parametric family of distributions; common choices include the normal, logistic, and log-gamma distributions. Regarding $\log Y_i$ as the outcome, the log-likelihood of the AFT model is given by

$$\mathcal{L}(r, \eta) = \prod_i S_\epsilon\left(\frac{\log Y_i - r(X_i)}{\sigma}\right)\left\{\frac{h_\epsilon\left(\frac{\log Y_i - r(X_i)}{\sigma}\right)}{\sigma}\right\}^{\delta_i},$$

where $S_\epsilon(t)$ is the survival function of ϵ_i, $f_\epsilon(t)$ is the density of ϵ_i, and $h_\epsilon(t) = f_\epsilon(t)/S_\epsilon(t)$ is the hazard function of ϵ_i. We consider $\epsilon_i \sim \text{Logistic}(0,1)$ and $\epsilon_i \sim \log\text{Gam}(\alpha,\alpha)$. These models correspond to *log-logistic* ($\eta = \sigma$) and *generalized gamma* ($\eta = (\sigma,\alpha)$) AFT models for T_i respectively. For both models, we consider a ground truth of $r_0(x) = r_F(x)$ and $\sigma = 1$.

The log-logistic model, for which $f_\epsilon(t) = \mathfrak{s}(t)\{1 - \mathfrak{s}(t)\}$, is particularly convenient in that both the survival function $S_\epsilon(t) = 1 - \mathfrak{s}(t)$ and hazard function $h_\epsilon(t) = \mathfrak{s}(t)$ can be written in closed form. To this point, the generalized gamma model is the first model for which we cannot compute $\mathcal{I}_\eta(\lambda)$ in closed form. We therefore use this as an opportunity to show that our methodology works well even when we approximate the required derivatives numerically. Given a function $w(\mu)$, we use finite differences to approximate the first and

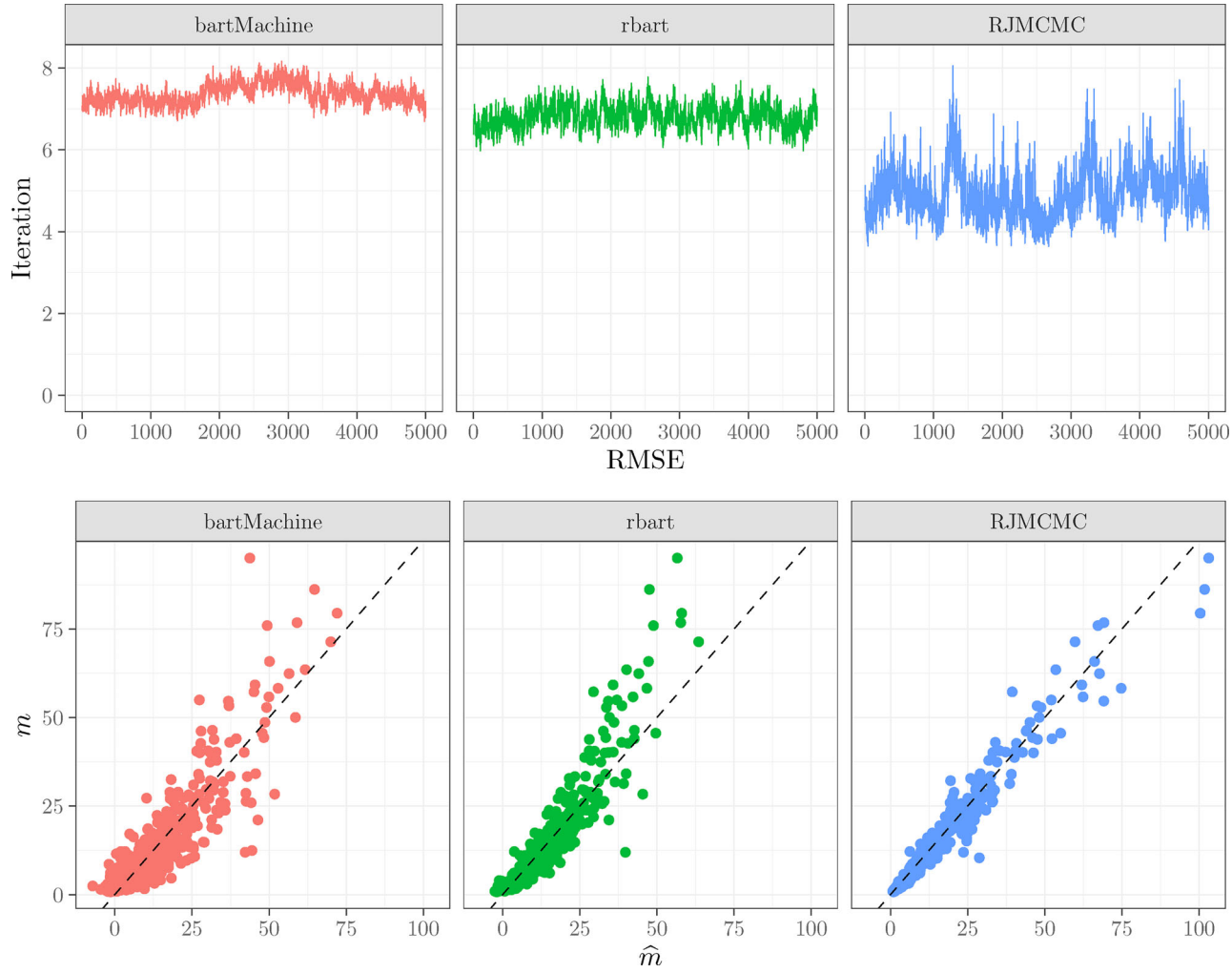

Figure 6. Top: traceplot of RMSE on heldout samples for each method. Bottom: Plot of the Bayes point estimate of $\mathbb{E}(Y_i \mid X_i)$ ($\widehat{m}_i$) against its true value (m); the relationship $m = \widehat{m}$ is given by the dashed line.

second derivatives as $w'(\mu) \approx \frac{w(\mu+\Delta)-w(\mu-\Delta)}{2\Delta}$ and $w''(\mu) \approx \frac{w(\mu+\Delta)-2w(\mu)+w(\mu-\Delta)}{\Delta^2}$ with $\Delta = 10^{-6}$.

We simulate data from both models with $r_0(x) = \frac{r_F(x)-14}{5}$ and $(N, P) = (500, 10)$. We censored the data at randomly by sampling C_i and T_i from the same distribution; by design, this results in roughly 50% of the samples being censored regardless of the value of X_i. For the generalized gamma model, we used the ground truth $\sigma_0 = \alpha_0 = 1$.

Overall, we found that both chains mixed well, with the exception that the mixing of σ and α was poor for the generalized gamma AFT model; this poor mixing occurs because σ and α are highly correlated in the posterior, and should be updated jointly rather than with the slice sampler we used. We also found that the log-logistic model took less time per iteration because the survival function of the log-logistic model is available in closed form. In general, σ and α are poorly identified due to the fact that both parameters are largely variance parameters for $\log T_i$, with 95% credible intervals being $\sigma \in (0.67, 3.72)$ and $\alpha \in (0.60, 7.45)$. Despite this, the chain mixes very well on the variance parameter $V = \text{var}(\log T_i \mid r, \sigma, \alpha) = \sigma^2 \psi'(\alpha)$, with the Bayes estimate $\widehat{V} = 1.72$ being very close to the true value $V_0 = 1.64$.

Plots like those in Figure 8 (right) and Figure 6 (bottom) are given in the Supplementary material; they show that both the log-logistic and generalized gamma models recover $r(x)$ effectively. Estimates of the survival curve, along with 95% credible bands, for some randomly sampled observations in a heldout test set are given in Figure 7. We see that the point estimates and credible bands provide accurate inference for the true survival curves

Application to Liver Disease Data

We apply the AFT log-logistic (AFTLL) and generalized gamma (AFTGG) models to a dataset from a randomized clinical trial on time to death for individuals suffering from primary biliary cirrhosis; this data is publicly available as the `pbc` dataset in the package `survival`. Our goal is to determine which of the parametric families provides the best description of this data. In addition to these models, we consider a semiparametric Weibull model with hazard function of the form $h(t \mid \lambda, k) = \frac{k}{e^{\lambda}}\left(\frac{t}{e^{\lambda}}\right)^{k-1}$,with the survival time modeled as $T_i \sim h\{t \mid r(X_i), k\}$ and $\eta = k$; a similar model is proposed by Linero et al. (2022).

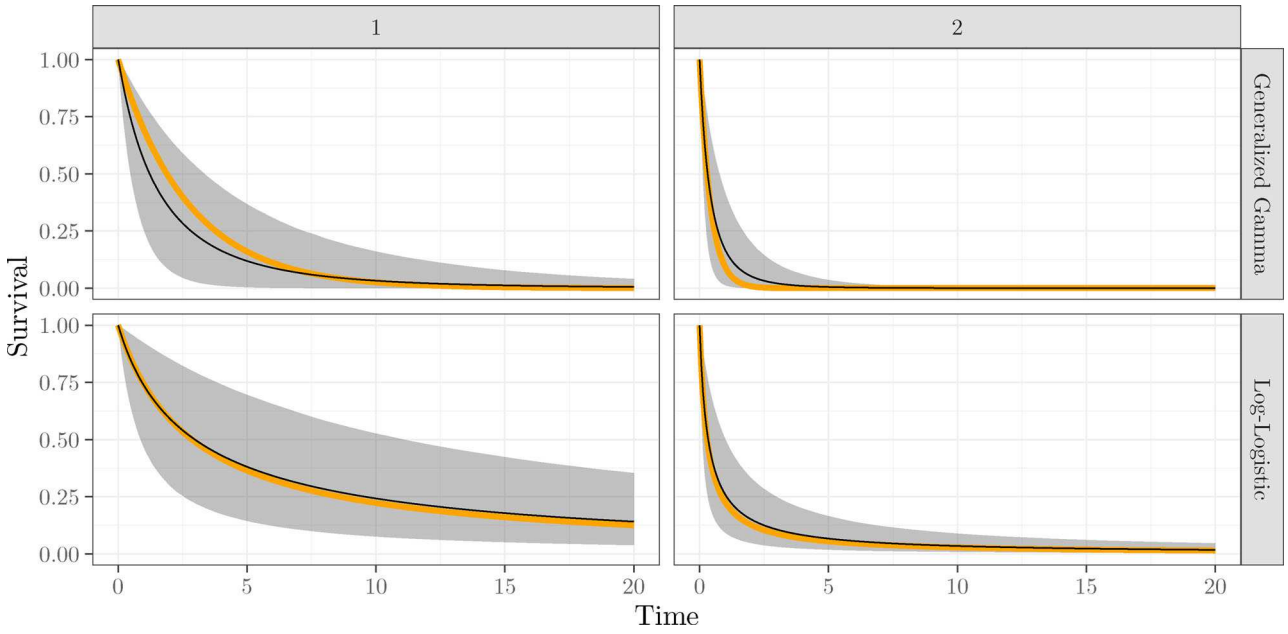

Figure 7. Estimated (black) and true (orange) survival curves, with 95% posterior credible bands, for two randomly selected observations for the generalized gamma and log-logistic AFT models.

This Weibull model, which sits at the intersection of AFT and proportional hazards models, is a special case of the generalized gamma model with $\alpha = 1$. Additionally, the generalized gamma model includes the log-normal AFT model as a limiting case as $\alpha \to \infty$; this makes the generalized gamma model a potentially useful tool for deciding between different parametric families.

To gain insight into whether different models lead to different qualitative prognoses for patients, we compare the estimates of $r(X_i)$ for the different models in the supplementary material. We found that the models agreed remarkably well in their estimates of $r(X_i)$.

Conversely, we also found that the data did not distinguish well between the different models, particularly for large survival times—this is due to poor identifiability of the shape parameter of the different models, rather than any issue with the nonparametric part of the model. In the supplementary material, we plot the posterior distribution of the shape parameter α in the generalized gamma model under a Uniform$(0, 40)$ prior, and find that the data is consistent with both the Weibull model ($\alpha = 1$) and log-normal model ($\alpha \to \infty$). These models make quite different predictions for the hazard at later timepoints, with the Weibull model having a monotonically-increasing hazard ($k \approx 1.3$) and the log-normal and log-logistic models both having non-monotone hazards.

Finally, we evaluate the goodness of fit of the AFTLL, AFTGG, and Weibull models using the log-pseudo marginal likelihood (LPML) given by $\sum_i \log f(Y_i, \delta_i \mid \boldsymbol{Y}_{-i}, \boldsymbol{\delta}_{-i}, \boldsymbol{X})$where $\boldsymbol{Y}_{-i}$ and $\boldsymbol{\delta}_{-i}$ denote the vector of event times and censoring indicators with observation i removed, while $\boldsymbol{X} = (X_1, \ldots, X_N)$ and $f(Y_i, \delta_i \mid \boldsymbol{Y}_{-i}, \boldsymbol{\delta}_{-i}, \boldsymbol{X})$ is the *predictive density* given by $\int f_\eta(Y_i, \delta_i \mid r(X_i))\, \pi(r, \eta \mid \boldsymbol{Y}_{-i}, \boldsymbol{\delta}_{-i}, \boldsymbol{X})\, dr\, d\eta$. The LPML can be conveniently computed from the output of the MCMC sampler using the `loo` package in R. The fits of all three models are quite similar, with the estimated LPMLs being $(-350.6, -353.4, -350.1)$ for the Weibull, AFTGG, and AFTLL models, respectively. According to LPML, there is a slight preference for the log-normal model, which has a non-monotone hazard, although the Weibull model performs very similarly. This observation is consistent with our findings in the supplementary material, where we find that the posterior distribution of the AFTGG model is consistent with both the Weibull ($\alpha = 1$) and log-normal ($\alpha \to \infty$) models.

5.4. Gamma Shape Regression

An interesting extension of the accelerated failure time models discussed in Section 5.3 is to allow for the shape of the hazard function itself to depend on the covariates; this would allow some individuals to have monotonically increasing, decreasing, or non-monotone hazards depending on their covariates. One approach to doing this is to model the shape parameter α in the generalized gamma model in a covariate-dependent fashion as well. Toward this end, we consider a gamma regression model which takes $Y_i \sim \text{Gam}\{\alpha(X_i), \beta\}$ (where $\eta = \beta$). A BART model for the related gamma regression model $Y_i \sim \text{Gam}\{\alpha, \beta(X_i)\}$ was considered by Linero, Sinha, and Lipsitz (2020), who showed that this model can be fit using the generalized Bayesian backfitting algorithm; due to the fact that $\beta(X_i)$ is not a shape parameter for the gamma distribution, however, this model is not appropriate for modeling changes in the shape of the hazard.

We model the shape parameter on the log scale, taking $\alpha(X_i) = \exp\{r(X_i)\}$. It is then straight-forward to show that

$$U_\eta(y \mid \lambda) = e^\lambda\{\log\beta - \psi(e^\lambda) + \log y\} \quad \text{and}$$
$$\mathcal{I}_\eta(\lambda) = e^{2\lambda}\, \psi'(e^\lambda)$$

where $\psi(\alpha) = \frac{d}{d\alpha}\log\Gamma(\alpha)$ and $\psi'(\alpha) = \frac{d}{d\alpha}\psi(\alpha)$ are the digamma and trigamma functions, respectively. We simulate data from the model with $(N, \beta) = (100, 1)$ and $r(x) = 2 + \frac{r_F(x)-14}{5}$ so that $\log\alpha(X_i)$ has roughly mean 2 and variance 1,

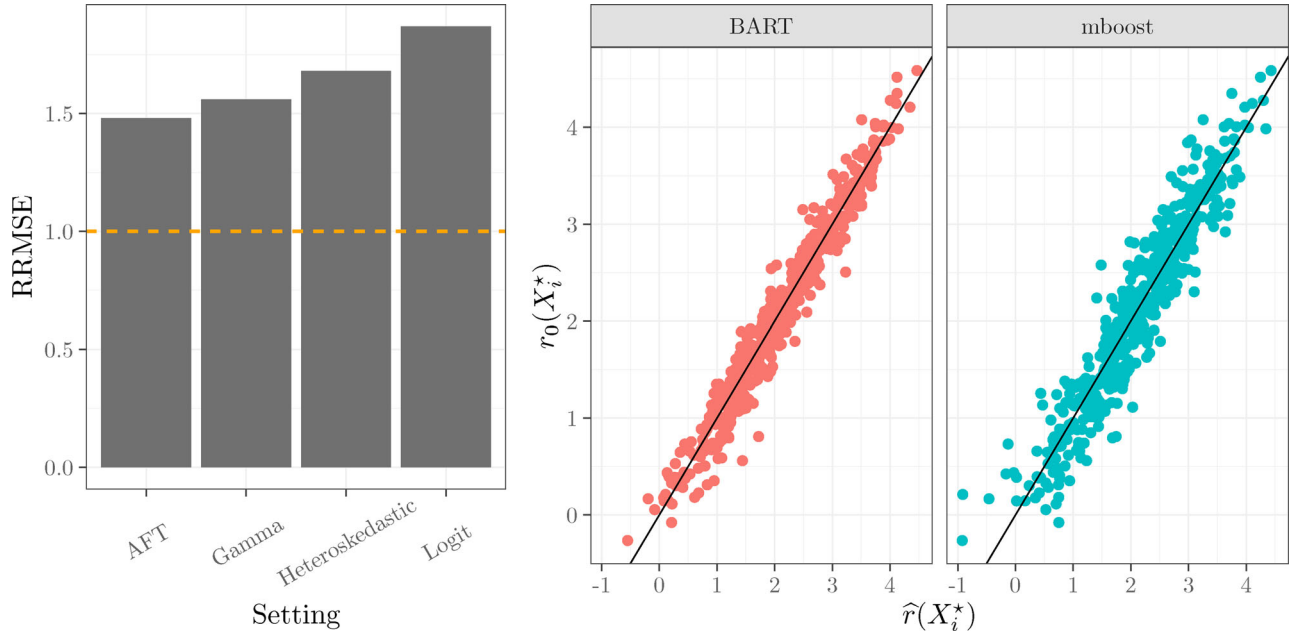

Figure 8. Left: root mean-squared error of optimally-tuned `mboost` relative to the root mean-squared error (RRMSE) of the default generalized BART prior for the logistic AFT (AFT), gamma shape (Gamma), structured variance (Heteroscedastic), and logistic regression (Logit) models. To aide visualization, the orange dashed line at 1 would denote a tie in performance with generalized BART. Right: plots of $\widehat{r}(X_i^\star)$ against $r_0(X_i^\star)$ for generalized BART and `mboost` for the gamma shape regression model.

and fit the model with the default prior. Mixing of the RJMCMC scheme is given in the Supplementary Material; summarizing, we found that the chain mixed well. As shown in the right panel of Figure 8, generalized BART is able to accurately recover $r_0(x)$ on a set of heldout covariates $(X_1^\star, \ldots, X_{N_{\text{test}}}^\star)$.

5.5. Comparison with Tree Boosting

We give a brief comparison of the generalized BART model with tree boosting as implemented in the `mboost` package in `R`. Our comparisons are biased *in favor* of the `mboost` package: for each comparison, we chose the `mboost` hyperparameters (the shrinkage parameter `nu` and the number of boosting iterations `mstop`) to minimize the error on the test set, and chose the maximal depth of the tree (`maxdepth`) equal to 2 to ensure that `mboost` does not include any spurious higher-order interactions. By contrast, the hyperparameters for generalized BART are either fixed a-priori or learned from the training data.

We compare `mboost` to generalized BART on the logistic regression problem, the log-logistic accelerated failure time (AFT) problem, the gamma shape regression problem, and the structure heteroscedastic regression problem. The `mboost` package implements logistic regression and log-logistic AFT models, and we used the functionality within `mboost` to build custom procedures for the gamma and heteroscedastic regression models. In each case, accuracy is measured through the mean squared error $\sqrt{N_{\text{test}}^{-1}\sum_i\{r_0(X_i^\star) - \widehat{r}(X_i^\star)\}^2}$ where $(X_1^\star, \ldots, X_{N_{\text{test}}}^\star)$ denotes a heldout test set of 500 points and $\widehat{r}(x)$ denotes the point estimate of $r_0(x)$ (for boosting) or the posterior mean of $r(x)$ (for BART).

Results are given in Figure 8, with $N = 500$ and $P = 10$; these results are representative of what occurs in repeated simulations. Despite the simulation settings here generally favorable to boosting (the hyperparameters were optimally tuned to the test set, there is little noise, and the number of nuisance predictors is small) the results are strongly in favor of generalized BART. Specifically, the RMSE of `mboost` ranges from 50% larger to 80% larger than the RMSE of generalized BART. The right panel of Figure 8, which focuses on the gamma shape regression problem, displays $r_0(X_i^\star)$ against $\widehat{r}(X_i^\star)$ for `mboost` and generalized BART, and provides a sanity check that both methods are working as intended. Both sets of predictions cluster around the 45 degree line, with `mboost` being less precise.

6. Discussion

The approach outlined in this article greatly expands the problems to which BART can be applied, and we emphasize that none of the models we applied BART to required any modifications to our algorithm. There are many directions for extending this framework in future work. For example, by modifying the approach to allow for more than one forest (Pratola et al. 2020), we could develop flexible gamma regression models with $Y_i \sim \text{Gam}\{\alpha(X_i), \alpha(X_i)/\mu(X_i)\}$ or beta regression models with $Y_i \sim \text{Beta}\{\mu(X_i)\,\phi(X_i), \phi(X_i) - \phi(X_i)\,\mu(X_i)\}$. This could be done using either separate forests, in which case our methodology extends directly, or using the shared forests approach of Linero, Sinha, and Lipsitz (2020). The shared forests approach is likely more difficult to implement due to the need for a multivariate Metropolis-Hastings proposal for the reversible jump move.

An alternative to RJMCMC is to use an approximation of the tree's marginal likelihood for Bayesian backfitting, as discussed in DiCiccio et al. (1997) and Llorente et al. (2023). In particular, the Laplace approximation we've used could also approximate the marginal likelihood and full conditional of the leaf node parameters; this approximation is likely accurate given our use of informative priors, even for leaf nodes with little data. This approach sacrifices exactness for simplicity, but computational

costs remain largely unchanged: after computing the Laplace approximation, we see little reason for using an inexact update when an exact update can be performed instead. More advanced methods, such as those described by Llorente et al. (2023), likely improve upon the Laplace approximation, but this should be balanced against the fact that the marginal likelihood must be evaluated many times over the course of the chain.

An additional application of our RJMCMC algorithm is that it can be extended to the *soft* BART models of Linero and Yang (2018). These models—which have better theoretical and empirical properties than standard BART models when the underlying function $r_0(x)$ is smooth—can only use conjugate updates for the model (1) to the best of our knowledge.

For the generalized gamma model, we crudely avoided computing the score and Fisher information by using numerical differentiation; this approximates the likelihood, score, and Fisher information using a total of three likelihood evaluations, and so is relatively efficient. We note that it is, in principle, possible to eliminate the need for the user to explicitly compute the derivatives of the likelihood by using software that performs automatic differentiation such as `TensorFlow`.

A lingering advantage of gradient boosting over BART is that gradient boosting is much faster and scales better to large datasets. Recently, He, Yalov, and Hahn (2019) and He and Hahn (2021) substantially closed this gap with their XBART algorithm; however, this approach also requires the same sort of conditional conjugacy as the generalized BART model. It is worth exploring whether our RJMCMC algorithm might be combined with XBART, either to be used after a "warm-start" with XBART or to be used to construct a replacement for the XBART splitting criterion.

Supplementary Materials

The online supplement contains additional illustrations, a heuristic justification of the algorithm, and proofs of all results.

Acknowledgments

The author thanks the reviewers for their helpful comments.

Disclosure Statement

The author reports no competing interest to declare.

Funding

This work is partially supported by NSF award DMS-2144933.

References

Albert, J. H., and Chib, S. (1993), "Bayesian Analysis of Binary and Polychotomous Response Data," *Journal of the American Statistical Association*, 88, 669–679. [2,4]

Bhattacharya, A., Pati, D., and Yang, Y. (2019), "Bayesian Fractional Posteriors," *The Annals of Statistics*, 47, 39–66. [7]

Breiman, L. (2001), "Random Forests," *Machine Learning*, 45, 5–32. [1]

Chipman, H. A., George, E. I., and McCulloch, R. E. (1998), "Bayesian CART Model Search," *Journal of the American Statistical Association*, 93, 935–948. [4,5]

——— (2010), "BART: Bayesian Additive Regression Trees," *The Annals of Applied Statistics*, 4, 266–298. [1,2,3,4,6,7]

Chipman, H. A., George, E. I., McCulloch, R. E., and Shively, T. S. (2021), "mBART: Multidimensional Monotone BART," *Bayesian Analysis*, 17, 515–544. [4]

DiCiccio, T. J., Kass, R. E., Raftery, A., and Wasserman, L. (1997), "Computing Bayes Factors by Combining Simulation and Asymptotic Approximations," *Journal of the American Statistical Association*, 92, 903–915. [12]

Dorie, V., Hill, J., Shalit, U., Scott, M., and Cervone, D. (2019), "Automated versus Do-It-Yourself Methods for Causal Inference: Lessons Learned from A Data Analysis Competition," *Statistical Science*, 34, 43–68. [1]

Freund, Y., Schapire, R., and Abe, N. (1999), "A Short Introduction to Boosting," *Journal-Japanese Society For Artificial Intelligence*, 4, 771–780. [1]

Friedman, J. H. (2001), "Greedy Function Approximation: A Gradient Boosting Machine," *The Annals of Statistics*, 29, 1189–1232. [1]

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013), *Bayesian Data Analysis*, Boca Raton, FL: CRC Press. [5]

George, E., Laud, P., Logan, B., McCulloch, R., and Sparapani, R. (2019), "Fully Nonparametric Bayesian Additive Regression Trees," in *Topics in Identification, Limited Dependent Variables, Partial Observability, Experimentation, and Flexible Modeling: Part B*, eds. I. Jeliazkov and J. L. Tobias, pp. 89–110, Bingley: Emerald Publishing Limited. [3]

Ghosal, S., and van der Vaart, A. (2017), *Fundamentals of Nonparametric Bayesian Inference* (Vol. 44), Cambridge: Cambridge University Press. [7]

Green, P. J. (1995), "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination," *Biometrika*, 82, 711–732. [2,5]

Hahn, P. R., Murray, J. S., and Carvalho, C. M. (2020), "Bayesian Regression Tree Models for Causal Inference: Regularization, Confounding, and Heterogeneous Effects," (with Discussion), *Bayesian Analysis*, 15, 965–1056. [3]

He, J., and Hahn, P. R. (2021), "Stochastic Tree Ensembles for Regularized Nonlinear Regression," *Journal of the American Statistical Association*, advance online publication. [13]

He, J., Yalov, S., and Hahn, P. R. (2019), "Accelerated Bayesian Additive Regression Trees," in *22nd Proceedings of the International Conference on Artificial Intelligence in Statistics (AISTATS)*. [13]

Henderson, N. C., Louis, T. A., Rosner, G. L., and Varadhan, R. (2020), "Individualized Treatment Effects with Censored Data via Fully Nonparametric Bayesian Accelerated Failure Time Models," *Biostatistics*, 21, 50–68. [3]

Hill, J., Linero, A., and Murray, J. (2020), "Bayesian Additive Regression Trees: A Review and Look Forward," *Annual Review of Statistics and Its Application*, 7, 251–278. [2]

Hill, J. L. (2011), "Bayesian Nonparametric Modeling for Causal Inference," *Journal of Computational and Graphical Statistics*, 20, 217–240. [3]

Holmes, C. C., and Held, L. (2006), "Bayesian Auxiliary Variable Models for Binary and Multinomial Regression," *Bayesian Analysis*, 1, 145–168. [7,8]

Johndrow, J. E., Smith, A., Pillai, N., and Dunson, D. B. (2019), "MCMC for Imbalanced Categorical Data," *Journal of the American Statistical Association*, 114, 1394–1403. [8]

Kapelner, A., and Bleich, J. (2016), "bartMachine: Machine Learning with Bayesian Additive Regression Trees," *Journal of Statistical Software*, 70, 1–40. [4]

Kindo, B. P., Wang, H., Hanson, T., and Pena, E. A. (2016), "Bayesian Quantile Additive Regression Trees," arXiv preprint arXiv:1607.02676. [3,7]

Li, Y., Linero, A. R., and Murray, J. (2023), "Adaptive Conditional Distribution Estimation with Bayesian Decision Tree Ensembles," *Journal of the American Statistical Association*, 118, 2129–2142. [3,6]

Linero, A. R. (2018), "Bayesian Regression Trees for High-Dimensional Prediction and Variable Selection," *Journal of the American Statistical Association*, 113, 626–636. [6,7]

Linero, A. R., Basak, P., Li, Y., and Sinha, D. (2022), "Bayesian Survival Tree Ensembles with Submodel Shrinkage," *Bayesian Analysis*, 17, 997–1020. [1,3,4,6,10]

Linero, A. R., Sinha, D., and Lipsitz, S. R. (2020), "Semiparametric Mixed-Scale Models Using Shared Bayesian Forests," *Biometrics*, 76, 131–144. [1,3,4,11,12]

Linero, A. R., and Yang, Y. (2018), "Bayesian Regression Tree Ensembles that Adapt to Smoothness and Sparsity," *Journal of the Royal Statistical Society*, Series B, 80, 1087–1110. [1,3,6,7,13]

Llorente, F., Martino, L., Delgado, D., and López-Santiago, J. (2023), "Marginal Likelihood Computation for Model Selection and Hypothesis Testing: An Extensive Review," *SIAM Review*, 65, 3–58. [12,13]

Murray, J. S. (2021), "Log-Linear Bayesian Additive Regression Trees for Multinomial Logistic and Count Regression Models," *Journal of the American Statistical Association*, 116, 756–769. [1,3,4,8]

Orlandi, V., Murray, J., Linero, A., and Volfovsky, A. (2021), "Density Regression with Bayesian Additive Regression Trees," arXiv preprint arXiv:2112.12259. [6]

Polson, N. G., Scott, J. G., and Windle, J. (2013), "Bayesian Inference for Logistic Models Using Pólya–Gamma Latent Variables," *Journal of the American statistical Association*, 108, 1339–1349. [8]

Pratola, M. (2016), "Efficient Metropolis-Hastings Proposal Mechanisms for Bayesian Regression Tree Models," *Bayesian Analysis*, 11, 885–911. [4]

Pratola, M. T., Chipman, H. A., George, E. I., and McCulloch, R. E. (2020), "Heteroscedastic BART via Multiplicative Regression Trees," *Journal of Computational and Graphical Statistics*, 29, 405–417. [4,9,12]

Ročková, V., and van der Pas, S. (2020), "Posterior Concentration for Bayesian Regression Trees and Forests," *The Annals of Statistics*, 48, 2108–2131. [1,3,6,7]

Saha, E. (2023), "Theory of Posterior Concentration for Generalized Bayesian Additive Regression Trees," arXiv preprint arXiv:2304.12505. [3,6,7]

Sparapani, R., Spanbauer, C., and McCulloch, R. (2021), "Nonparametric Machine Learning and Efficient Computation with Bayesian Additive Regression Trees: The BART R Package," *Journal of Statistical Software*, 97, 1–66. [7]

Sparapani, R. A., Logan, B. R., McCulloch, R. E., and Laud, P. W. (2016), "Nonparametric Survival Analysis Using Bayesian Additive Regression Trees (BART)," *Statistics in Medicine*, 35, 2741–2753. [1,3]

Wei, L. J. (1992), "The Accelerated Failure Time Model: A Useful Alternative to the Cox Regression Model in Survival Analysis," *Statistics in Medicine*, 11, 1871–1879. [9]