Stochastic Tree Ensembles for Estimating Heterogeneous Effects

Nikolay Krantsevich
Arizona State University

Jingyu He
City University of Hong Kong

P. Richard Hahn
Arizona State University

Abstract

Determining subgroups that respond especially well (or poorly) to specific interventions (medical or policy) requires new supervised learning methods tailored specifically for causal inference. Bayesian Causal Forest (BCF) is a recent method that has been documented to perform well on data generating processes with strong confounding of the sort that is plausible in many applications. This paper develops a novel algorithm for fitting the BCF model, which is more efficient than the previous Gibbs sampler. The new algorithm can be used to initialize independent chains of the existing Gibbs sampler leading to better posterior exploration and coverage of the associated interval estimates in simulation studies. The new algorithm is compared to related approaches via simulation studies as well as empirical analysis.

1 INTRODUCTION

Causal effect estimation of binary interventions on continuous outcomes has been a problem of great interest in disciplines of applied research including social sciences, education, public health, and policy research. In recent years, the focus of many applied projects has been switching from average treatment effects (ATE) to conditional average treatment effects (CATE). While ATE findings are instructive for a general understanding of intervention effectiveness, CATE estimation enables the discovery of subpopulations whose effects deviate from average (which could be of critical importance for groups that experience an effect opposite in sign to the ATE).

There are two state-of-the-art methods in CATE estimation. The first is Bayesian Causal Forest (Hahn et al., 2020), which allows for separate regularization on the prognostic and treatment functions and relies on Bayesian Additive Regression Trees (Chipman et al., 2010). The second is Causal Random Forests (Wager and Athey, 2018), which extends Random Forests (Breiman, 2001) to causal inference.

The Bayesian Causal Forest model was documented to perform well in a number of separate, rigorous simulation studies (McConnell and Lindner, 2019; Hahn et al., 2019; Dorie et al., 2019; Wendling et al., 2018). It was used to estimate CATEs in the high-profile Growth Mindset intervention (Yeager et al., 2019), as well as other applied work (Ghosh et al., 2020; King et al., 2019; Bail et al., 2020; Bryan et al., 2019).

The contribution of this paper is two-fold. First, we apply the computational strategies of Accelerated BART (He and Hahn, 2021) to fit a BCF model, which we call the *Accelerated Bayesian Causal Forest (XBCF)* model. Second, we propose a procedure *warm-start BCF,* that utilizes trees obtained from a fitted XBCF model to initialize the BCF fitting procedure, allowing for more efficient posterior space exploration. Both methods provide similar or better coverage and are many orders of magnitude (15x and 100x) faster compared to the BCF model, making them powerful tools for working with data sets that contain hundreds of thousands of observations. In general, warm-start BCF provides superior results compared to XBCF, but requires additional runtime and computational memory space, which makes XBCF particularly valuable for very large data sets.

2 BACKGROUND

2.1 Notation

Let Y_i represent the scalar response variable, Z_i denote a binary treatment variable, and $\mathbf{x}_i$ represent a length d row vector of observed control variables for observation i. Let Y and Z be length n column vectors comprising variables Y_i and Z_i, respectively; let $\mathbf{X}$ denote the $n \times d$ matrix of control variables. We will use lowercase Roman letters, such as y and z, to denote the values assumed by variables. Our data will consist of n independent observations $(Y_i, Z_i, \mathbf{x}_i)$.

Following the potential outcomes framework (Imbens and Rubin, 2015), let $Y_i(1)$ and $Y_i(0)$ represent the outcomes under treatment and control, respectively; each observed response may be expressed as $Y_i = Z_i Y_i(1) + (1 - Z_i) Y_i(0)$.

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

Throughout, we assume the following standard conditions licensing regression estimates of treatment effects:

1. **SUTVA (Stable Unit Treatment Value Assumption):** No treatment assignment to a particular individual should affect the observed outcomes on other individuals and that there is no variation in treatment.

2. **Strong ignorability assumption:** First, there are no unmeasured confounders:

$$Y_i(0), Y_i(1) \perp\!\!\!\perp Z_i \mid \mathbf{X}_i.$$

Second, every individual has a non-zero probability of being assigned to either treatment arm:

$$0 < \Pr(Z_i = 1 \mid \mathbf{x}_i) < 1.$$

Under these assumptions, the CATE of units with covariates $\mathbf{x}$ may be estimated as the difference between two identified conditional expectations:

$$\tau(\mathbf{x}) := \mathbf{E}(Y \mid \mathbf{x}, Z = 1) - \mathbf{E}(Y \mid \mathbf{x}, Z = 0).$$

Further assuming a mean-zero additive error,

$$Y_i = f(\mathbf{x}_i, Z_i) + \epsilon_i, \quad \epsilon_i \sim \mathbf{N}(0, \sigma^2),$$

it follows that $\mathbf{E}(Y_i \mid \mathbf{x}_i, Z_i = z_i) = f(\mathbf{x}_i, z_i)$ and

$$\tau(\mathbf{x}_i) := f(\mathbf{x}_i, 1) - f(\mathbf{x}_i, 0).$$

In the XBCF method, these conditional expectations are estimated using the aforementioned Bayesian tree ensemble method Accelerated BART (XBART) of He et al. (2019), which is related to a well-known method called Bayesian Additive Regression Trees, or BART (Chipman et al., 2010).

2.2 BART

BART represents the outcome of interest as a sum of an unknown function $f(\cdot)$ and an error term,

$$Y_i = f(\mathbf{x}_i) + \epsilon_i, \quad \epsilon_i \sim \mathbf{N}(0, \sigma^2) \tag{1}$$

The mean function $f(\mathbf{x})$ is represented as a sum of many piecewise constant binary regression trees

$$f(\mathbf{x}) = \sum_{l=1}^{L} g_l(\mathbf{x}; T_l, \mathbf{m}_l) \tag{2}$$

where T_l denotes a regression tree, which represents a partition of the covariate space (say $\mathcal{A}_1, \ldots, \mathcal{A}_{B(l)}$) and consists of a set of internal decision nodes and a set of terminal nodes (or leaves) which correspond to each element of the partition. Each element of the partition $\mathcal{A}_b$ is assigned a leaf parameter value, m_{lb}, and $\mathbf{m}_l = (m_{l1}, \cdots, m_{lB(l)})$ denotes the vector corresponding to all leaf parameters of the l-th tree, T_l. The

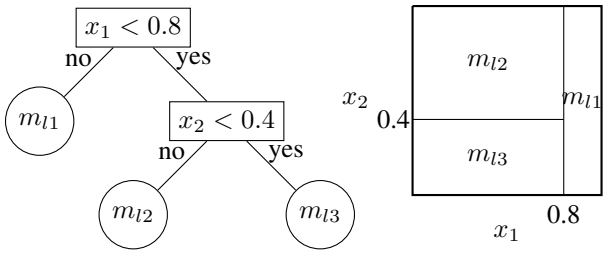

Figure 1: An example binary tree, with the corresponding partition of the sample space and the step function.

piecewise constant function comprising the partition and the leaf parameters is defined as $g_l(\mathbf{x}) = m_{lb}$ if $\mathbf{x} \in \mathcal{A}_b$; see Figure 1 for a demonstration.

Within each leaf, the mean parameters are given independent normal priors, $m_{lb} \sim \mathbf{N}(0, \nu)$. The prior over trees $p(T_l)$ is specified by the probability of a node having children at depth d as $\alpha(1+d)^{-\beta}, \alpha \in (0,1), \beta \in [0, \infty)$ to induce regularization of size of the tree (Chipman et al., 1998).

BART explores the posterior of the trees by a random walk Metropolis-Hastings Markov chain Monte Carlo (MCMC) algorithm. It can be slow to converge and limits the broader adoption of BART, especially for large datasets.

2.3 XBART

XBART was introduced to improve the fitting time of BART-like models. XBART blends regularization and stochastic search strategies from Bayesian modeling with computationally efficient techniques ranging from recursive partitioning approaches to tree-fitting. XBART fits the same sum-of-trees ensemble model as BART, but regrows each tree *recursively* at each iteration according to a stochastic process inspired by Bayesian updating.

We review the stochastic tree-growing approach of XBART (Algorithm 1). Let $\mathcal{C}$ denote a matrix of cutpoint candidates with elements c_{jk}, where $j = 1, \ldots, p$ indexes a variable and k indexes a candidate cutpoint. Assume the leaf parameter m has prior $N(0, \nu)$. At each node, the probability of splitting at cutpoint c_{jk} is proportional to

$$\begin{aligned} L(c_{jk}) \propto \exp\Bigg\{ \frac{1}{2} \Bigg[& \log\left(\frac{\sigma^2}{\sigma^2 + \nu n^l_{jk}}\right) + \\ & + \frac{\nu}{\sigma^2(\sigma^2 + \nu n^l_{jk})} \left(s^l_{jk}\right)^2 + \log\left(\frac{\sigma^2}{\sigma^2 + \nu n^r_{jk}}\right) \\ & + \frac{\nu}{\sigma^2(\sigma^2 + \nu n^r_{jk})} \left(s^r_{jk}\right)^2 \Bigg] \Bigg\}, \end{aligned} \tag{3}$$

where σ^2 is the residual variance as in equation (1), n^l_{jk} and n^r_{jk} are the number of observations in the left or right child node if split at the splitting rule c_{jk}, and s^l_{jk} and s^r_{jk} are the

corresponding sufficient statistics for the children nodes:

$$s^l_{jk} = \sum_{x_i \in \mathcal{A}^{\text{left}}_{jk}} y_i, \quad s^r_{jk} = \sum_{x_i \in \mathcal{A}^{\text{right}}_{jk}} y_i$$
$$s_{\text{all}} = s^l_{jk} + s^r_{jk} = \sum_{i=1}^{n} y_i, \tag{4}$$

where $n = n^l_{jk} + n^r_{jk}$ is number of observations in the current node. The probability of not splitting anywhere (no-split option) is proportional to

$$L(\emptyset) \propto |\mathcal{C}| \left(\frac{(1+d)^\beta}{\alpha} - 1 \right) \times \exp\left\{ \frac{1}{2} \left[\log\left(\frac{\sigma^2}{\sigma^2 + \nu n} \right) + \frac{\nu}{\sigma^2(\sigma^2 + \nu n)} s^2_{\text{all}} \right] \right\}, \tag{5}$$

where $|\mathcal{C}|$[1] is the total number of candidate splitting rules, and d is the depth of the current node in the tree. The tree is fitted recursively where at each node, a cutpoint (or the no-splitting option) is randomly drawn from a multinomial distribution using probabilities (3) and (5). If no-splitting is sampled, or other pre-set stopping conditions are satisfied, the current node becomes a terminal (leaf) node, and its associated leaf parameter m is updated by conjugate Gaussian sampling. To form an ensemble of trees, XBART uses a similar strategy as Bayesian backfitting, residualizing the data with respect to the partial fit corresponding to the forest. Specifically, the h-th tree is grown to fit the partial residual of all other trees: $\mathrm{y} - \sum_{l \neq h} g_l(\mathbf{x}; T_l, \mathbf{m}_l)$.

He and Hahn (2021) details how these strategies contribute to the improved efficiency of XBART over BART as well as improved posterior coverage of interval estimates obtained by initializing multiple Markov chains at XBART estimates. Here, we adapt the XBART approach to the BCF model and demonstrate comparable performance gains in the heterogeneous treatment effect setting.

2.4 Bayesian Causal Forest

Hahn et al. (2020) demonstrate the inability of BART to handle confounding for certain simple data generative processes (DGPs) via simulation studies. They refine the BART model to overcome this limitation with several modifications. First, rather than representing $f(\mathbf{x}, z)$ as a single BART model, as in Hill (2011), they propose using the representation

$$f(\mathbf{x}_i, z_i) = \mu(\mathbf{x}_i) + \tau(\mathbf{x}_i) z_i, \tag{6}$$

where μ and τ are prognostic and treatment functions, respectively; both are given independent BART priors, permitting control and treatment effects to be regularized independently. Second, they propose including an estimate of the propensity score $\widehat{\pi}_i = P(Z_i = 1 \mid \mathbf{x}_i)$ to combat unintended bias of treatment effects due to the regularization of μ. See Hahn et al. (2020) for more details on this phenomenon, which the authors refer to as *regularization induced confounding* (RIC). We also refer readers to Hahn and Herren (2022) and Herren and Hahn (2020) for a discussion on finite sample properties of non-parametric propensity score methods.

[1]For heuristics about categorical and continuous variables, see Appendix A.

Algorithm 1 Grow From Root (GFR)

Input: GFR$(\mathrm{y}, \mathbf{X}, \sigma, d, T, \texttt{node})$.
Calculate full sufficient statistics s_{all} by (4).
for $c_{jk} \in \mathcal{C}$, partition data to left and right sides **do**
    Calculate s^l_{jk} and s^r_{jk} by equation (4).
    Calculate $L(c_{jk})$ by equation (3).
end for
Calculate probability of no-split $L(\emptyset)$ by equation (5).
Draw a cutpoint or no-split using probability $L(c_{jk})$ and $L(\emptyset)$.
if no-split is chosen or stop conditions are met **then**
    Update leaf parameter $m_{\texttt{node}}$.
    return.
else
    Create two new nodes as children of `node`, denoted `left_node` and `right_node`.
    Sift the data into `left_node` and `right_node`.
    GFR$(\mathrm{y}_{\text{left}}, \mathbf{X}_{\text{left}}, \sigma, d+1, T, \texttt{left_node})$
    GFR$(\mathrm{y}_{\text{right}}, \mathbf{X}_{\text{right}}, \sigma, d+1, T, \texttt{right_node})$
end if
Output: The grown tree T, including the vector of sampled leaf parameters, $\mathbf{m}$.

Finally, adding scaling factors b_0 and b_1 in the model makes the priors invariant with respect to which group is designated as the treated group. An additional scaling factor, a, enhances the learning of the prognostic term.

Putting these modifications together, the complete Bayesian causal forest (BCF) model is

$$y_i = a\tilde{\mu}(\mathbf{x}_i, \widehat{\pi}_i) + b_{z_i}\tilde{\tau}(\mathbf{x}_i) + \epsilon_i, \quad \epsilon_i \sim \mathbf{N}(0, \sigma^2)$$
$$a \sim \mathbf{N}(0,1), \quad b_0, b_1 \sim \mathbf{N}(0, 1/2) \tag{7}$$

According to the parametrization above, treatment effects are given by $\tau(\mathbf{x}_i) = (b_1 - b_0)\tilde{\tau}(\mathbf{x}_i)$.

Computationally, BCF is built upon the same random walk Metropolis-Hastings algorithm that underpins BART. As such, it suffers from the same slow fitting time on large data sets and the same slow posterior exploration.

3 ACCELERATED BCF

3.1 The model

We now describe our first contributed method, which we call Accelerated BCF, or XBCF. The XBCF model differs in one substantive respect from the model presented in Hahn et al. (2020): the error standard deviations σ_0 and σ_1 are allowed to differ between the control and treatment groups,

respectively, whereas the original BCF model (7) has a common shared residual standard deviation. Thus, the XBCF model is

$$\begin{aligned} y_i = a\tilde{\mu}(\mathbf{x}_i, \widehat{\pi}_i) + b_{z_i}\tilde{\tau}(\mathbf{x}_i) + \epsilon_i, \;\; \epsilon_i \sim \mathbf{N}(0, \sigma^2_{z_i}), \\ a \sim \mathbf{N}(0,1), \;\; b_0, b_1 \sim \mathbf{N}(0, 1/2), \end{aligned} \tag{8}$$

and $\tilde{\mu}$ and $\tilde{\tau}$ are represented as sums of trees:

$$\begin{aligned} \tilde{\mu}(\mathbf{x}_i, \widehat{\pi}_i) &= \sum_{l=1}^{L} u_l(\mathbf{x}_i, \widehat{\pi}_i; T_l, \mathbf{m}_l^T), \\ \tilde{\tau}(\mathbf{x}_i) &= \sum_{k=1}^{K} v_k(\mathbf{x}_i; S_k, \mathbf{m}_k^S), \end{aligned}$$

where L, K represent the number of trees, T_l, S_k represent individual trees, and $\mathbf{m}_l^T, \mathbf{m}_k^S$ denote vectors of scalar means associated with the leafs nodes of T_l and S_k, respectively. We will reference the forests of trees as $T = \{T_l, \mathbf{m}_l^T\}_{l=1}^L$ and $S = \{S_k, \mathbf{m}_k^S\}_{k=1}^K$ for prognostic and treatment terms, respectively. Following BCF, we include a column vector of (estimated) propensity scores $\widehat{\pi}$ as an additional covariate for the prognostic term.

3.2 Model fitting procedure

The XBCF fitting algorithm uses a similar "backfitting" strategy as BART and XBART, iterating tree-by-tree through two forests (corresponding to the prognostic and treatment terms) rather than just one. The tree and parameter updates at each iteration are based on the following "residuals":

$$\begin{aligned} \text{Prognostic residual: } \mathrm{v} &\equiv \mathrm{y} - a\tilde{\mu}(\mathbf{X}, \widehat{\pi}), \\ \text{Treatment residual: } \mathrm{t} &\equiv \mathrm{y} - \mathrm{b} \cdot \tilde{\tau}(\mathbf{X}), \\ \text{Total residual: } \mathrm{r} &\equiv \mathrm{y} - a\tilde{\mu}(\mathbf{X}, \widehat{\pi}) - \mathrm{b} \cdot \tilde{\tau}(\mathbf{X}), \end{aligned} \tag{9}$$

where b is a length n vector with i-th component equal to b_{z_i}, and '$\cdot$' denotes element-wise multiplication. The update steps for trees T_l or S_k, depend on the vectors of *partial* residuals, which are obtained by subtracting the partial fit (corresponding to the forests *without* the current tree) from the observed response variable:

$$\begin{aligned} \mathrm{r}^T_{-l} &\equiv \mathrm{r} + a u_l(\mathbf{X}, \hat{\pi}; T_l, \mathbf{m}_l^T), \quad l = 1, \ldots, L, \\ \mathrm{r}^S_{-k} &\equiv \mathrm{r} + \mathrm{b} \cdot v_k(\mathbf{X}; S_k, \mathbf{m}_k^S), \quad k = 1, \ldots, K. \end{aligned} \tag{10}$$

With these terms defined, the sequence of stochastic updates is as follows:

1. **Stage 1: update prognostic forest.** We first grow L trees comprising the forest for the prognostic term $\mu(\mathbf{x}_i, \widehat{\pi}_i)$. For each of the trees ($l = 1, \ldots, L$) the sequence of updates is the following:
   (a) $T_l, \mathbf{m}_l^T \mid \mathrm{r}^T_{-l}, \sigma_0^2, \sigma_1^2, a, b_0, b_1$, which is done compositionally as
      i. $T_l \mid \mathrm{r}^T_{-l}, \sigma_0^2, \sigma_1^2$
      ii. $\mathbf{m}_l^T \mid T_l, \sigma_0^2, \sigma_1^2, a, b_0, b_1$
   (b) $a \mid \mathrm{t}, T_l$
   (c) $b_0, b_1 \mid \mathrm{v}, T_l$
   (d) $\sigma_0^2, \sigma_1^2 \mid \mathrm{r}$.
2. **Stage 2: update treatment forest.** We then grow K trees comprising the forest for the treatment term $\tau(\mathbf{x}_i)$. The sequence of updates for each tree S_k ($k = 1, \ldots, K$) is similar:
   (a) $S_k, \mathbf{m}_k^S \mid \mathrm{r}^S_{-k}, \sigma_0^2, \sigma_1^2, a, b_0, b_1$, which is done compositionally as
      i. $S_k \mid \mathrm{r}^S_{-k}, \sigma_0^2, \sigma_1^2$
      ii. $\mathbf{m}_k^S \mid S_k, \sigma_0^2, \sigma_1^2, a, b_0, b_1$
   (b) $a \mid \mathrm{t}, S_k$
   (c) $b_0, b_1 \mid \mathrm{v}, S_k$
   (d) $\sigma_0^2, \sigma_1^2 \mid \mathrm{r}$,

These two stages are repeated I times, which we refer to as "sweeps". Pseudocode is given in Algorithm 2. Although we use conditioning notation, note that these stochastic updates are *not* full conditional distributions in the usual Gibbs sampling sense. The tree-growing updates (Stage 1(a) and Stage 2(a)) are given in Algorithm 1, applied to the partial residuals defined in expression 10. Parameter updates are detailed in the next subsection.

After I sweeps, the CATE estimate for individuals with features $\mathbf{x}$ is calculated as an average of the $(b_1 - b_0)\tilde{\tau}(\mathbf{x})$ samples, as if one were taking a traditional posterior mean.

3.2.1 Parameter updates

If the no-split option is selected, or other pre-set stopping conditions are satisfied, the current node becomes a leaf node, and the associated leaf parameter is updated as follows (lines 8 and 16 in Algorithm 2). This update is a conditionally conjugate Gaussian mean update; we incorporate the control group and treatment group data sequentially to accommodate their differing variances (σ_0^2 and σ_1^2):

$$\nu_{n_0} = \left(\frac{1}{\nu} + \frac{n_0}{d_0^2}\right)^{-1}, \quad \beta_{n_0} = \frac{\bar{y}_0}{d_0^2}\nu_{n_0},$$

$$\nu_n = \left(\frac{1}{\nu_{n_0}} + \frac{n_1}{d_1^2}\right)^{-1}, \quad \beta_n = \left(\frac{\beta_{n_0}}{\nu_{n_0}} + \frac{\bar{y}_1}{d_1^2}\right)\nu_n,$$

where ν is the prior variance over the mean, $d_0 = \frac{\sigma_0}{b_0}, d_1 = \frac{\sigma_1}{b_1}$; n_0, n_1 are the number of individuals in the control and treatment groups respectively for this leaf node, and $\bar{y}_0, \bar{y}_1$ are the corresponding partial residual means of these two groups in this leaf node. The leaf mean parameter is then sampled according to $\mathbf{m} \sim \mathbf{N}(\beta_n, \nu_n^2)$.

Model parameters $a, b_0, b_1, \sigma_0, \sigma_1$ are sampled after each tree update for a total of $L + K$ times per sweep. After

Algorithm 2 Accelerated Bayesian Causal Forest (XBCF)

1: **Input:** $\mathrm{y}, \mathbf{X}, L, K, I$
2: Initialize $\mathrm{r}, \mathrm{v}, \mathrm{t}$, partial residuals $\mathrm{r}^T_{-l}, \mathrm{r}^S_{-k}$ and scale parameters $a, b_0, b_1, \sigma_0, \sigma_1$.
3: **for** iter in 1 to I **do**
4: **for** l in 1 to L **do**
5: Compute partial residual r^T_{-l} by equation (10).
6: Create `new_node` to initialize tree T_l^{iter} with root node.
7: GFR($\mathrm{r}^T_{-l}, \mathbf{X}, \sigma_0^2, \sigma_1^2, d = 0, T_l^{\text{iter}}$, `new_node`).
8: Update leaf parameter $\mathbf{m}_h^{T,\text{iter}}$ for T_l^{iter}.
9: Update full residual r, v by equation (9).
10: Sample $a, b_0, b_1, \sigma_0, \sigma_1$ based on $\mathrm{r}, \mathrm{v}, \mathrm{t}$.
11: **end for**
12: **for** k in 1 to K **do**
13: Compute partial residual r^S_{-k} by equation (10).
14: Create `new_node` to initialize tree S_k^{iter} with root node.
15: GFR($\mathrm{r}^S_{-k}, \mathbf{X}, \sigma_0^2, \sigma_1^2, d = 0, S_k^{\text{iter}}$, `new_node`).
16: Update leaf parameter $\mathbf{m}_k^{S,\text{iter}}$ for S_k^{iter}.
17: Update full residual r, t by equation (9).
18: Sample $a, b_0, b_1, \sigma_0, \sigma_1$ based on $\mathrm{r}, \mathrm{v}, \mathrm{t}$.
19: **end for**
20: **end for**
21:
22: **output:** $\{\{T_l^{\text{iter}}, \mathbf{m}_l^{T,\text{iter}}\}_{l=1}^L, \{S_k^{\text{iter}}, \mathbf{m}_k^{S,\text{iter}}\}_{k=1}^K\}_{\text{iter}=1}^I$, I posterior draws of the prognostic and treatment forests, and $\{a^{\text{iter}}, b_0^{\text{iter}}, b_1^{\text{iter}}, \sigma_0^{\text{iter}}, \sigma_1^{\text{iter}}\}_{\text{iter}=1}^I$, I posterior draws of other model parameters.

updating trees, the model parameters are sampled based on the residual vectors in equation (9) – the prognostic residual v, the treatment residual t and the total residual r (lines 9 and 17 in Algorithm 2). Since the general update sequence is similar for the two stages above, we will provide an explicit update scheme for each step for Stage 2 only.

In order to update parameter a we first reshape (8) in a regression problem, where the treatment residual vector t, with each component divided by the corresponding σ_{z_i}, is the response variable:

$$\begin{bmatrix} \frac{y_1 - b_{z_1}\tau(x_1)}{\sigma_{z_1}} \\ \vdots \\ \frac{y_n - b_{z_n}\tau(x_n)}{\sigma_{z_n}} \end{bmatrix} = \begin{bmatrix} \frac{\mu(x_1)}{\sigma_{z_1}} \\ \vdots \\ \frac{\mu(x_n)}{\sigma_{z_n}} \end{bmatrix} a + \begin{bmatrix} \frac{\epsilon_1}{\sigma_{z_1}} \\ \vdots \\ \frac{\epsilon_n}{\sigma_{z_n}} \end{bmatrix}.$$

Then updating a is essentially implemented as a two-step regression update:

$$\nu_{n_0} = \left(1 + \frac{\mu_0^t \mu_0}{\sigma_0^2}\right)^{-1}, \quad \beta_{n_0} = \frac{\mathrm{t}_0^t \mu_0}{\sigma_0^2}\nu_{n_0};$$
$$\nu_n = \left(\frac{1}{\nu_{n_0}} + \frac{\mu_1^t \mu_1}{\sigma_1^2}\right)^{-1}, \quad \beta_n = \left(\frac{\beta_{n_0}}{\nu_{n_0}} + \frac{\mathrm{t}_1^t \mu_1}{\sigma_1^2}\right)\nu_n,$$

where μ_0 is a vector with elements corresponding to $\mu(\cdot)$ evaluated at rows of $\mathbf{X}$ for which $z_i = 0$, and similarly for μ_1; t_0 is the part of residual vector t corresponding to only individuals with $z_i = 0$, and similarly for t_1. The parameter a is then sampled according to $a \sim \mathbf{N}(\beta_n, \nu_n^2)$.

For the scaling factors b_0 and b_1, we rearrange (8) in the form of a linear regression problem, where the prognostic residual vector v, with each component divided by corresponding σ_{z_i}, is the response variable:

$$\begin{bmatrix} \frac{y_1 - a\mu(x_1)}{\sigma_{z_1}} \\ \vdots \\ \frac{y_n - a\mu(x_n)}{\sigma_{z_n}} \end{bmatrix} = \begin{bmatrix} \frac{\tau(x_1)z_1}{\sigma_{z_1}} & \frac{\tau(x_1)(1-z_1)}{\sigma_{z_1}} \\ \vdots & \vdots \\ \frac{\tau(x_n)z_n}{\sigma_{z_n}} & \frac{\tau(x_n)(1-z_n)}{\sigma_{z_n}} \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} + \begin{bmatrix} \frac{\epsilon_1}{\sigma_{z_1}} \\ \vdots \\ \frac{\epsilon_n}{\sigma_{z_n}} \end{bmatrix}.$$

We then update b_0, b_1 as the regression coefficients. We first update their sampling parameters as follows:

$$\nu_{n_0} = \left(\frac{1}{\frac{1}{2}} + \frac{\tau_0^t \tau_0}{\sigma_0^2}\right)^{-1}, \quad \beta_{n_0} = \frac{\mathrm{v}_0^t \tau_0}{\sigma_0^2}\nu_{n_0};$$
$$\nu_{n_1} = \left(\frac{1}{\frac{1}{2}} + \frac{\tau_1^t \tau_1}{\sigma_1^2}\right)^{-1}, \quad \beta_{n_1} = \frac{\mathrm{v}_1^t \tau_1}{\sigma_1^2}\nu_{n_1},$$

where τ_0 is a vector with elements corresponding to $\tau(\cdot)$ evaluated at rows of $\mathbf{X}$ for which $z_i = 0$, and similarly for τ_1; m_0 is the part of residual vector m corresponding to only individuals with $z_i = 0$, and similarly for m_1. Then b_0 and b_1 are sampled as $b_0 \sim \mathbf{N}(\beta_{n_0}, \nu_{n_0}^2), b_1 \sim \mathbf{N}(\beta_{n_1}, \nu_{n_1}^2)$.

Lastly, updating the residual variances σ_0^2 and σ_1^2 is a conditionally conjugate inverse-Gamma update:

$$\sigma_0^2 \sim \mathbf{IG}\left(\frac{n_0 + \kappa_0}{2}, \frac{2}{\mathrm{r}_0^t \mathrm{r}_0 + s_0}\right)$$
$$\sigma_1^2 \sim \mathbf{IG}\left(\frac{n_1 + \kappa_1}{2}, \frac{2}{\mathrm{r}_1^t \mathrm{r}_1 + s_1}\right),$$

where n_0, n_1 are the total number of individuals fit in the control and treatment groups respectively, $\mathrm{r}_0, \mathrm{r}_1$ are the total residuals for the same corresponding groups; $\kappa_0, \kappa_1, s_0, s_1$ are hyperparameters of the inverse-Gamma prior.

4 WARM-START BCF

Preliminary work on simulation studies revealed that coverage of both BCF and XBCF often does not reach the desired nominal rate. On the one hand, complex Bayesian models do not guarantee a nominal coverage rate of credible intervals. On the other hand, very poor coverage is obviously undesirable. One contributor to under-coverage is inadequate Monte Carlo exploration of the posterior distribution, resulting in artificially narrow reported intervals. Because XBCF provides a fast approximation to the BCF posterior, initializing the BCF MCMC at XBCF trees rather than roots is a promising strategy to improve posterior exploration.

Specifically, we propose the following: First, use XBCF (s sweeps, b burn-in) to obtain the tree draws for each of the $s-b$ sweeps after the burn-in period. Second, initialize $s-b$ BCF Markov chains at the forests obtained from XBCF. Initializing BCF on the trees obtained from XBCF

Table 1: Results of root mean squared error (RMSE), interval coverage (Coverage) and interval length (I.L.) for ATE and CATE estimators under various settings. Sample size is 500. Time is running time in seconds.

		Homogeneous Treatment							Heterogeneous Treatment						
Prognostic Term	Method	RMSE		Coverage		I.L.		Time	RMSE		Coverage		I.L.		Time
		ATE	CATE	ATE	CATE	ATE	CATE		ATE	CATE	ATE	CATE	ATE	CATE	
Linear	ws-BCF	0.24	0.33	**0.92**	**0.97**	0.98	1.70	1.92	0.24	1.05	0.88	**0.93**	1.00	3.47	1.95
	XBCF	0.26	**0.31**	0.84	0.90	0.89	1.29	0.23	0.26	1.28	0.80	0.76	0.90	2.70	0.24
	BCF	0.24	0.34	0.90	0.96	0.96	1.65	4.69	0.23	1.07	0.90	0.84	0.96	2.98	4.98
	ps-BART	0.30	0.53	0.87	**0.98**	1.02	2.62	10.32	0.28	1.18	0.84	**0.93**	1.06	3.72	10.49
	CRF	0.39	0.57	0.80	0.83	1.23	1.62	0.21	0.45	1.41	0.76	0.74	1.38	2.59	0.21
	BART	0.39	0.61	0.67	0.96	1.00	2.57	10.35	0.38	1.21	0.72	0.92	1.03	3.66	10.57
	BART-$f_0 f_1$	0.59	1.03	0.36	0.95	1.03	4.14	12.83	0.54	1.39	0.56	0.93	1.07	5.06	12.84
	lm	0.19	**0.30**	**0.94**	**0.99**	0.90	1.74	1.54	0.23	0.40	0.92	**0.97**	0.96	1.93	1.44
Nonlinear	ws-BCF	0.33	**0.48**	**0.95**	**0.99**	1.65	2.78	1.92	0.33	**1.48**	**0.94**	**0.91**	1.61	4.63	1.93
	XBCF	0.35	**0.48**	0.88	0.92	1.49	2.09	0.23	0.36	1.62	0.88	0.78	1.45	3.59	0.24
	BCF	0.32	**0.46**	0.94	0.98	1.63	2.65	4.60	0.33	1.52	**0.93**	0.86	1.58	4.27	4.75
	ps-BART	0.41	0.88	0.90	**0.99**	1.72	4.70	10.42	0.40	1.57	0.92	**0.93**	1.7	5.53	10.46
	CRF	0.44	0.69	0.84	0.90	1.53	2.50	0.20	0.56	1.62	0.74	0.79	1.67	3.5	0.21
	BART	0.55	0.95	0.76	0.98	1.63	4.44	10.39	0.55	1.58	0.76	**0.93**	1.59	5.29	10.45
	BART-$f_0 f_1$	1.44	2.56	0.12	0.85	1.71	7.56	12.86	1.39	2.71	0.10	0.86	1.68	7.88	12.88
	lm	1.89	2.14	0.03	0.45	1.73	3.97	1.41	1.81	2.20	0.06	0.50	1.71	4.26	1.31

substantially reduces the necessary burn-in period for the BCF MCMC algorithm. We call this initialization strategy *warm-start BCF*, or ws-BCF.

We recommend retrieving 40 forests from XBCF and respectively initializing 40 independent BCF chains using those forests, with 10 burn-in iterations and 100 iterations post burn-in as a default setting. Running BCF for so few iterations is much faster compared to the regular BCF model, and it also grants additional time gains coming from a straightforward chain parallelization. For all experiments in this paper, we used eight cores which we believe is very common for modern machines.

The findings of Ronen et al. (2022) shed light on certain issues with mixing in BART-based models, which could be alleviated by seeding techniques like our warm-start method. Studying the simplified version of BART, the authors observed that the first split in the tree could create a bottleneck for the mixing of the chain, and overcoming this bottleneck is more difficult with larger amounts of data. With a warm-start approach described above, the BCF chains are initialized at sets of stochastically grown trees that are likely to differ in their overall structure and have a greater variety of splitting rules at the top the trees.

5 SIMULATION STUDIES

5.1 Simulation 1: Small data

First, we reproduce the simulation study of Hahn et al. (2020), focusing on estimating CATE on the basis of three metrics: average root mean square error, coverage, and average interval length. The data are generated according to four different processes: the conditional expectation can be linear or nonlinear, and the treatment effect can be homogeneous or heterogeneous. The covariate vector $\mathbf{x}$ contains five variables; three are continuous, standard normal random variables, one is dichotomous, and one is unordered categorical with three levels (denoted 1, 2, 3). Specifically, the treatment effect is either

$$\tau(\mathbf{x}) = \begin{cases} 3 & \text{homogeneous,} \\ 1 + 2x_2x_5 & \text{heterogeneous,} \end{cases}$$

and the prognostic function is defined as either

$$\mu(\mathbf{x}) = \begin{cases} 1 + g(x_4) + x_1x_3 & \text{linear,} \\ -6 + g(x_4) + 6|x_3 - 1| & \text{nonlinear,} \end{cases}$$

where $g(1) = 2$, $g(2) = -1$ and $g(3) = -4$. The propensity function is given by

$$\pi(\mathbf{x}_i) = 0.8\Phi(3\mu(\mathbf{x}_i)/s - 0.5x_1) + 0.05 + u_i/10,$$

where s is the standard deviation of $\mu(\mathbf{x})$ taken over the observed sample, with $u_i \sim \text{Uniform}(0, 1)$. Including μ in the treatment probability induces strong confounding.

The set of methods that we use to estimate treatment effects on this data includes our two new methods proposed in this paper, XBCF and warm-start BCF; the original BCF method; a naive version of BART with binary treatment assignment added as a non-distinguished covariate; ps-BART, which in addition to the treatment assignment also incorporates propensity score estimates as another covariate; BART-$f_0 f_1$, which fits two separate BART models for the treatment and control groups; Causal Random Forest (Wager and Athey, 2018), which also incorporates propensity score estimates; and a Bayesian linear model with a horseshoe prior (Carvalho et al., 2010) on the regression coefficients.

Table 2: Results of root mean squared error (RMSE), interval coverage (Coverage) and interval length (I.L.) for ATE and CATE estimators for the simulation study with 100,000 datapoints and 50 covariates. The number in parenthesis for BCF indicates the number of burn-in and follow-up iterations. The column Time is running time in seconds. The results are averaged over 50 independent replications.

Method	RMSE		Coverage		I.L.		Time
	ATE	CATE	ATE	CATE	ATE	CATE	
ws-BCF	0.006	**0.064**	**0.960**	**0.805**	0.023	0.159	**334**
XBCF	0.006	0.076	0.900	0.685	0.025	0.157	**43**
BCF(4)	0.015	0.098	0.640	0.513	0.022	0.109	2494
BCF(10)	0.011	0.085	0.78	0.55	0.022	0.107	6084

For each of the methods, we averaged the results on the three metrics over 200 independent replications. The results on a sample of $n = 500$ data points are presented in Table 1. For this simulation study, we used the default recommended settings for all of the methods. Two methods, warm-start BCF and Causal Random Forest, took advantage of parallelization on eight cores.

Broadly, we recapitulate the findings of Hahn et al. (2020). Their key takeaways are that one, the propensity score is an important feature for accurate estimation of treatment effects in problems with strong confounding, and two, separate regularization of μ and τ improves estimation accuracy. Here, we highlight the differences between BCF, XBCF, and warm-start BCF.

- warm-start BCF always performs better than regular BCF in CATE estimation, in both RMSE and coverage.
- XBCF provides the most narrow credible interval length, but often under covers compared to BCF and warm-start BCF.
- Overall, warm-start BCF provides the best coverage among all three methods, for both ATE and CATE.

5.2 Simulation 2: Large data

An additional simulation is designed to compare the performance and computational speed of XBCF, warm-start BCF, and the original BCF on large data. Here the time comparisons and CATE coverage are the main interest, as we expect these methods will concur on any data set given sufficient run time.

We generate $n = 100,000$ data points with $p = 50$ covariates (25 continuous and 25 binary) as the input matrix. Of those 50 variables, we choose two possibly overlapping sets of 10, sampled uniformly, to contribute to the treatment and prognostic functions, respectively. The data is unbalanced on average, with approximately $\frac{2}{3}$ data points in the control group, and treatment effects were stratified into 10 levels. Full details of the DGP are available in the supplement.

Table 3: Results of root mean squared error (RMSE), interval coverage (Coverage) and interval length (I.L.) for ATE and CATE estimators for the simulation study with 500,000 datapoints and 250 covariates. The column Time is running time in seconds. The results are averaged over 50 independent replications.

Method	RMSE		Coverage		I.L.		Time
	ATE	CATE	ATE	CATE	ATE	CATE	
ws-BCF	0.002	**0.056**	**0.90**	**0.708**	0.095	0.128	3376
XBCF	0.003	0.063	0.86	0.626	0.097	0.119	**1121**

Table 2 shows that warm-start BCF with default parameters (100 iterations over 40 sweeps) performs better than the original BCF MCMC in all estimands of interest and especially improves in coverage. In general, MCMC methods need to run for long enough in order to converge, and when we run the original BCF for a significantly larger amount of iterations (10,000 after 10,000 iterations of burn-in), we still see that it does not match the performance of warm-start BCF, despite taking 15 times longer. While XBCF doesn't reach the levels of coverage that warm-start BCF does, the method still surpasses the standard BCF MCMC in all aspects while being 100 times faster.

5.3 Simulation 3: Largest data

Finally, we focus on estimating the performance of our proposed two methods in an even larger sample size. With a DGP similar to the simulation in Section 5.2, we generate samples of $n = 500,000$ data points with $p = 250$ covariates (125 continuous and 125 binary). We also increase the complexity of the problem by sampling two sets of 50 variables (instead of 10) to contribute to the prognostic and treatment functions, respectively. The results provided in Table 3 support the observed trend that warm-start BCF helps improve the coverage, with both methods finishing computations within an hour.

We did not include BCF in the latter simulation due to excessive run time demands. However, we would like to point out two potential issues of the method with regard to large data. First, as a BART-based model, BCF is expected to suffer from a greater degree of poor mixing as the number of observations increases (Ronen et al., 2022); this would lead to even longer run times till convergence than on smaller sample sizes. Second, as the method would need to run very long, even saving posterior draws every few hundred iterations, the output matrix may grow past RAM size. Warm-start BCF overcomes both of these issues.

5.4 Hardware specifications

The experiments underlying Tables 1 and 2 were performed on a Linux machine with an Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz processor and 64GB RAM; 8 cores were

Table 4: ATE estimates and respective lengths of the 95% credible/confidence intervals for the set of methods we considered.

Method	ATE	CI length	Time
ws-BCF	0.68	1.02	1.50
XBCF	0.62	0.91	0.52
BCF	0.67	1.02	8.02
ps-BART	0.67	0.98	11.07
BART	0.68	0.99	11.26
BART-$f_0 f_1$	0.73	1.04	13.26
CRF	0.64	1.18	0.45

used for warm-start BCF and CRF parallelization.

The experiment with $n = 500,000$ observations underlying Table 3 was performed on a more powerful Linux machine with an Intel Xeon Gold 5318Y CPU @ 2.1GHz processor and 314GB RAM; but 8 cores were used for warm-start BCF parallelization.

6 EMPIRICAL DEMONSTRATION

For empirical demonstration, we analyze data on student classroom performance in language arts classes collected from two public schools in Portugal during the 2005-2006 school year (Cortez and Silva, 2008). The data is available at the UCI Machine Learning Repository and was used in Cortez and Silva (2008) to predict students' final grades. The rich covariates in this data set make it possible to pose several questions regarding the causal impact of students' attributes on their final scores. Here, we focus on estimating the treatment effect of which school was attended, Gabriel Pereira (GP) or Mousinho da Silveira (MS). The course grade is an award on a 20-point scale.

From the original data set, which contained information on 649 students, we omit students whose final score is 0. We also restrict our analysis to those who state that they intend to pursue higher education, bringing the sample size to $n = 570$ students. We control for the following variables:

- age: age in years at the time of the survey (numeric)
- address: indicator whether student lives in a city or in a rural area (binary)
- famrel: quality of family relationship (5 levels)
- famsize: indicator whether student's family has more than 3 members or not (binary)
- famsup: family educational support (binary)
- Fedu: father's education level (5 levels)
- Fjob: father's job (5 categories)
- health: student's current health status (5 levels)
- internet: internet access at student's home (binary)
- Medu: mother's education level (5 levels)
- Mjob: mother's job (5 categories)
- nursery: indicator of attending nursery school (binary)
- Pstatus: parent's cohabitation status (binary)
- reason: reason to choose this school (4 categories)
- sex: student's sex assigned at birth (binary)

6.1 Treatment effect estimation

All methods considered in the simulation study are used here as well, except for the linear model. Table 4 reports point estimates and interval lengths (for 95% credible intervals for the Bayesian methods and for the 95% confidence interval for the Causal Random Forest method). All methods estimate the ATE to be in the range 0.6-0.8, with interval estimates lying above zero, suggesting a small positive average treatment effect.

Although the ATE estimates broadly concur, CATE estimates vary substantially across methods. Table 5 shows the correlation matrix of CATE estimates obtained from different methods. As desired, BCF and warm-start BCF are strongly positively correlated.

Table 5: The correlation matrix of CATE estimates obtained from different methods.

	CRF	BART	BART-$f_0 f_1$	ps-BART	BCF	XBCF
BART	0.65					
BART-$f_0 f_1$	0.63	0.88				
ps-BART	0.63	0.87	0.99			
BCF	0.73	0.62	0.73	0.71		
XBCF	0.63	0.63	0.64	0.61	0.49	
ws-BCF	0.76	0.73	0.83	0.82	0.98	0.57

6.2 Subgroup analysis

Posterior inference for subgroup average treatment effects can be obtained directly from the posterior draws sampled from warm-start BCF.

To discover subgroups of interest, we fit a regression tree to the posterior point estimates of the CATE, using the set of all covariates available from the original dataset; the resulting tree defines subgroups for which the CATE estimates differ. This should be considered a convenient form of posterior exploration and not a separate inference procedure. Posterior inferences are obtained simply as the sample average effects calculated according to each posterior draw. Of particular interest is the posterior difference between subgroup treatment effects: posterior credible intervals of this quantity allow us to determine if the difference between subgroups is statistically convincing.

The left panel in Figure 2 represents the fitted tree to posterior point estimates obtained from warm-start BCF. Subgroup 1, which benefited most from treatment, with subgroup ATE estimate of 1.3 points, consisted of 50 students

with the following characteristics: mother doesn't have a higher education degree (Medu < 4); family relationship is perceived by the student as average or lower (famrel < 4); there is educational support from the family (famsup ≥ 2).

At the other end of the spectrum we have Subgroup 2, which benefited the least from the treatment, with the subgroup ATE estimate of -0.46 points, consisting of 11 students with the following characteristics: mother has a higher education degree (Medu ≥ 4); father's job is teacher; there is no educational support from the family (famsup < 2).

The posterior difference in subgroup ATE is shown in the middle panel of Figure 2. The majority of differences are above 0; the 95% posterior credible interval is $(-0.2, 4.7)$.

Although it makes sense intuitively that students whose parents have less education may stand to benefit more from better in-school instruction, the fact that those students are receiving at-home support while the children of teachers are not defied expectation. We speculate that the reason a pupil whose father is a teacher would not receive at-home support is if the student is not in need of assistance. If this were the case, it would suggest that better in-school instruction benefits students who are not already excelling; this is consistent with the estimated subgroup average prognostic effects (see right panel in Figure 2) as well as with previous literature on educational interventions (Yeager et al., 2019).

7 SUMMARY

This paper introduces a novel algorithm for fitting the Bayesian causal forest model, a popular method for causal inference problems with heterogeneous treatment effects. Our warm-start BCF method produces BCF models capable of fitting substantially larger datasets than could be fit with the previous random walk Metropolis-Hastings algorithm, which can under-explore the vast space of regression tree ensembles. Furthermore, warm-start BCF produces superior results comapred to BCF in a much (15x) shorter runtime.

The XBCF method also provides improved coverage compared to BCF (though inferior to warm-start BCF) on large datasets, while running 100x faster. XBCF can be used on datasets of size up to ≈ 1 million in our experiments. Additionally, even on smaller data sets, the warm-start BCF algorithm provides better interval estimates of conditional average treatment effects in simulations, a property that we believe to hold for empirical analyses as well, as the warm-start BCF intervals tend to be longer. We hope to apply our approach to large observational health databases.

In an extension of this work, Wang et al. (2022) augments Gaussian processes to XBCF to extrapolate non-overlapping region of treatment and control groups. Future directions include adapting the work for other causal inference methods that call for regularized regression, such as instrumental variables approaches or regression discontinuity designs.

SOFTWARE AVAILABILITY

Both warm-start BCF and XBCF are available in R; details on installation and use are available at `https://github.com/JingyuHe/bcf2`. Warm-start BCF is expected to be available on CRAN as part of the next `bcf` package version. XBCF is expected to become available on CRAN as part of the `XBART` package (`https://github.com/JingyuHe/XBART`), and it has a standalone Python implementation `xbcausalforest`.

Acknowledgements

The authors would like to thank the reviewers for the helpful feedback that allowed to improve the readability and organization of the paper. This project was supported by Facebook Statistics for Improving Insights and Decisions research award. Jingyu He gratefully acknowledges the support from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 21504921).

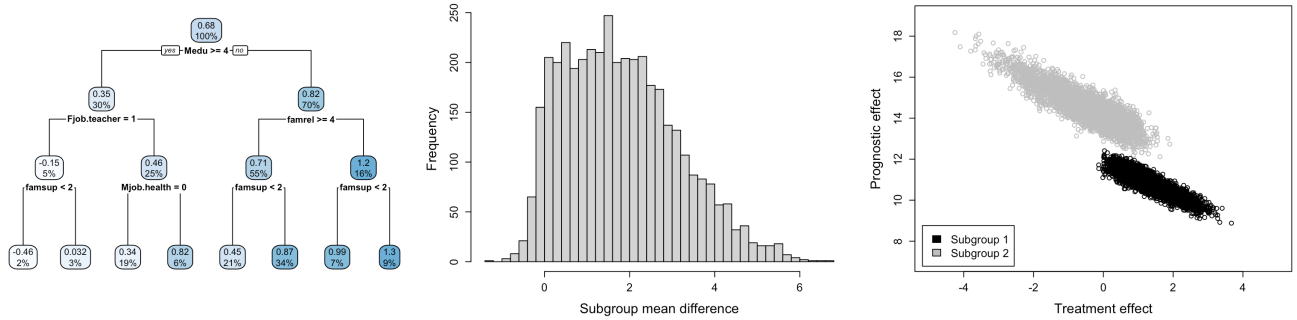

Figure 2: (Left) A single deterministic tree fit to the individual-level treatment estimates of warm-start BCF. The top number in each box is the average subgroup treatment effect, the lower number indicates the percentage of the total sample. (Middle) The histogram of differences in means of Subgroup 1 and Subgroup 2 over all posterior draws of warm-start BCF. (Right) Posterior draws of subgroup average treatment and prognostic effects for the two subgroups.

References

Bail, C. A., Guay, B., Maloney, E., Combs, A., Hillygus, D. S., Merhout, F., Freelon, D., and Volfovsky, A. (2020). Assessing the russian internet research agency's impact on the political attitudes and behaviors of american twitter users in late 2017. *Proceedings of the National Academy of Sciences*, 117(1):243–250.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Bryan, C. J., Yeager, D. S., and O'Brien, J. M. (2019). Replicator degrees of freedom allow publication of misleading failures to replicate. *Proceedings of the National Academy of Sciences*, 116(51):25535–25545.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2010). The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480.

Chipman, H. A., George, E. I., and McCulloch, R. E. (1998). Bayesian CART model search. *Journal of the American Statistical Association*, 93(443):935–948.

Chipman, H. A., George, E. I., McCulloch, R. E., et al. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298.

Cortez, P. and Silva, A. M. G. (2008). Using data mining to predict secondary school student performance. *Proceedings of 5th Annual Future Business Technology Conference*, pages 5–12.

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(1).

Ghosh, A., Orzol, S., Dale, S., Laird, J., Fu, N., Singh, P., Kim, M.-Y., Markovitz, A., Swankoski, K., Duda, N., et al. (2020). Independent evaluation of comprehensive primary care plus (cpc+) second annual report: Appendices to the supplemental volume. Technical report, Mathematica Policy Research.

Hahn, P. R., Dorie, V., and Murray, J. S. (2019). Atlantic causal inference conference (acic) data analysis challenge 2017. *arXiv preprint arXiv:1905.09515*.

Hahn, P. R. and Herren, A. (2022). Feature selection in stratification estimators of causal effects: lessons from potential outcomes, causal diagrams, and structural equations. *arXiv preprint arXiv:2209.11400*.

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(3).

He, J. and Hahn, P. R. (2021). Stochastic tree ensembles for regularized nonlinear regression. *Journal of the American Statistical Association*, pages 1–20.

He, J., Yalov, S., and Hahn, P. R. (2019). Xbart: Accelerated Bayesian additive regression trees. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 1130–1138. PMLR.

Herren, A. and Hahn, P. R. (2020). Semi-supervised learning and the question of true versus estimated propensity scores. *arXiv preprint arXiv:2009.06183*.

Hill, J. L. (2011). Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240.

Imbens, G. W. and Rubin, D. B. (2015). *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

King, C. R., Escallier, K. E., Ju, Y.-E. S., Lin, N., Palanca, B. J., McKinnon, S. L., and Avidan, M. S. (2019). Obstructive sleep apnoea, positive airway pressure treatment and postoperative delirium: protocol for a retrospective observational study. *BMJ Open*, 9(8):e026649.

McConnell, K. J. and Lindner, S. (2019). Estimating treatment effects with machine learning. *Health Services Research*, 54(6):1273–1282.

Ronen, O., Saarinen, T., Tan, Y. S., Duncan, J., and Yu, B. (2022). A mixing time lower bound for a simplified version of bart. *arXiv preprint arXiv:2210.09352*.

Wager, S. and Athey, S. (2018). Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242.

Wang, M., He, J., and Hahn, P. R. (2022). Local gaussian process extrapolation for bart models with applications to causal inference. *arXiv preprint arXiv:2204.10963*.

Wendling, T., Jung, K., Callahan, A., Schuler, A., Shah, N. H., and Gallego, B. (2018). Comparing methods for estimation of heterogeneous treatment effects using observational data from health care databases. *Statistics in Medicine*, 37(23):3309–3324.

Yeager, D. S., Hanselman, P., Walton, G. M., Murray, J. S., et al. (2019). A national experiment reveals where a growth mindset improves achievement. *Nature*, 573(7774):364–369.

A HEURISTICS FOR CATEGORICAL AND CONTINUOUS VARIABLES

Recall Equation (3), the split criterion for each cutpoint candidate. We implicitly assume that all cutpoint candidates are equally weighted. If all variables in $\mathbf{X}$ are continuous, and each variable has an equal number of candidates, then all variables are equally weighted as well. However, if $\mathbf{X}$ contains a mixture of continuous and categorical variables, they may have different weights, since the number of candidates for them can be different: users can, at most, choose split candidates equal to the number of classes for categorical variables. It is therefore possible to adjust Equation (3) for the j-th categorical variable by a $-\log(\text{num\_cutpoints}_{j,\text{categorical}}) + \log(\text{num\_cutpoints}_{\text{continuous}})$ term to induce a uniform prior on variables, but not cutpoint candidates. Neither option is inherently correct, but they may induce different results, especially when the number of candidates is quite different for continuous and categorical variables. In this paper, we use the fist strategy (equally weighted cutpoints) for the small sample simulation from Section 5.1, and the second strategy for the large sample simulations from Sections 5.2 and 5.3.

B DATA GENERATING PROCESSES FOR SIMULATION STUDIES 2 AND 3

Here we describe the Data Generating Process (DGP) for the simulation studies described in Sections 5.2 and 5.3. We will use the variable v to distinguish between the two simulations, with $v = 1$ for the DGP in Section 5.2 and $v = 5$ for the one in Section 5.3. We are interested in data that meet two criteria:

1. All input covariates are correlated;
2. Only some of those covariates are causally relevant.

First, we generate a feature matrix $\mathbf{X}$ meeting criteria (1), with $n = 100000v$ observations and $p = 50v$ features (half of them are continuous and the other half are binary).

From the set of covariates, we then choose at random two subsets of $10v$ variables, S_1 and S_2; we allow for these subsets to overlap. We then generate two vectors of length $50v$, α and β, component-wise for $k = 1, ..., 50$ as follows:

$$\alpha_k = \begin{cases} 0, & k \notin S_1 \\ a_k, & k \in S_1 \end{cases} \text{ and } \beta_k = \begin{cases} 0, & k \notin S_2 \\ b_k, & k \in S_2 \end{cases}$$

where $a_k, b_k \sim \mathbf{N}(0, 1)$. The vectors α and β are used for generation of the treatment and prognostic functions as described below.

To meet criteria (2), we only want some of the input features to be causally relevant, so we define the prognostic function in a way that only variables from S_2 impact the output of the function:

$$\mu(\mathbf{x}) = \left(\frac{\mathbf{x} \cdot \beta}{\sqrt{10v}} \right)^2.$$

Similarly, only variables from S_1 impact the output of the treatment function for this DGP, which is defined as a multilevel step-function. With the treatment function for this DGP, we intended to limit heterogeneity while having non-linearity of treatment.

Before we proceed to the treatment function, we will introduce two auxiliary functions which we utilized on the way to generating the treatment effects. First, we define function t as

$$t(\mathbf{x}) = \frac{\mathbf{x} \cdot \alpha}{\sqrt{10v}}.$$

We compute its minimum $m = \min_i t(\mathbf{x}_i)$ and range $r = \max_i t(\mathbf{x}_i) - \min_i t(\mathbf{x}_i)$ over rows $\mathbf{x}_i$ of the input matrix $\mathbf{X}$ ($i = 1, ..., 100000v$). Then we define an auxiliary step-function τ^* as

$$\tau^*(\mathbf{x}) = \begin{cases} 1, & t(\mathbf{x}) \in [m, m + \frac{1}{10}r) \\ 2, & t(\mathbf{x}) \in [m + \frac{1}{10}r, m + \frac{2}{10}r) \\ \vdots & \vdots \\ 10, & t(\mathbf{x}) \in [m + \frac{9}{10}r, m + r] \end{cases}$$

We also compute a scaling factor as the portion of the ratio between ranges of $\mu(\cdot)$ and $\tau(\cdot)$, evaluated at the rows $\mathbf{x}_i$ of input matrix $\mathbf{X}$ ($i = 1, ..., 100000v$):

$$h = 0.1 \times \frac{\max_i \mu(\mathbf{x}_i) - \min_i \mu(\mathbf{x}_i)}{\max_i \tau^*(\mathbf{x}_i) - \min_i \tau^*(\mathbf{x}_i)}$$

This scaling factor ensures that the treatment effects are not too large compared to the prognostic effects. We finally define the treatment function as follows:

$$\tau(\mathbf{x}) = h\tau^*(\mathbf{x}).$$

Similar to simulation study in Section 3, we include μ in defining the treatment probability to induce strong confounding:

$$\pi(\mathbf{x}) = 0.05 + 0.9(2\Phi(\mu(\mathbf{x})) - 1).$$

Lastly, we compute the binary vector of treatment assignments z as follows: $z_i \sim \mathbf{B}(\pi(\mathbf{x}_i))$.

Then for every individual observation i we compute the outcome variable in the following way:

$$y_i = \mu(\mathbf{x}_i) + \tau(x_i)z_i + 0.6\kappa\epsilon_i, \quad \epsilon_i \sim \mathbf{N}(0, 1),$$

where κ is the standard deviation of $\mu(x)$ taken over the observed sample.

C ADDITIONAL RESULTS FOR SIMULATION STUDY 1

We repeated the same simulation study shown in Table 1 of the paper, but with a smaller sample size of $n = 250$, to match the structure of the full simulation study in the original BCF paper. Table 6 presents the results of all considered methods on the estimands of interest obtained following the DGPs described in Section 5.1 of the paper, but on this smaller sample size of $n = 250$.

Table 6: Results of root mean squared error (RMSE), interval coverage (Coverage) and interval length (I.L.) for ATE and CATE estimators with different combinations of treatment term and prognostic term types. Sample size is 250. The column Time is running time in seconds.

		Homogeneous Treatment							Heterogeneous Treatment						
Prognostic Term	Method	RMSE		Coverage		I.L.		Time	RMSE		Coverage		I.L.		Time
		ATE	CATE	ATE	CATE	ATE	CATE		ATE	CATE	ATE	CATE	ATE	CATE	
Linear	ws-BCF	0.29	0.42	0.96	0.99	1.45	2.40	1.73	0.35	1.37	0.9	0.91	1.51	4.32	1.73
	XBCF	0.31	0.37	0.91	0.95	1.33	1.81	0.14	0.38	1.55	0.88	0.77	1.35	3.36	0.14
	BCF	0.30	0.45	0.96	0.98	1.43	2.39	2.53	0.35	1.39	0.88	0.87	1.47	4.07	2.63
	ps-BART	0.32	0.61	0.94	0.99	1.49	3.27	5.83	0.40	1.75	0.88	0.86	1.64	4.38	6.03
	CRF	0.43	0.56	0.83	0.88	1.57	1.86	0.10	0.55	1.76	0.82	0.72	1.96	2.89	0.10
	BART	0.45	0.73	0.80	0.96	1.44	3.22	5.95	0.53	1.78	0.76	0.84	1.57	4.32	6.03
	BART-f_0f_1	0.67	1.16	0.57	0.96	1.46	4.74	7.70	0.66	1.65	0.62	0.93	1.56	5.76	7.81
	lm	0.28	0.44	0.96	0.99	1.35	2.64	1.51	0.36	0.58	0.86	0.96	1.35	2.82	1.44
Nonlinear	ws-BCF	0.52	0.68	0.94	0.97	2.46	3.86	1.73	0.53	1.84	0.93	0.91	2.41	5.89	1.73
	XBCF	0.55	0.67	0.88	0.94	2.24	2.99	0.13	0.59	1.96	0.86	0.80	2.15	4.70	0.14
	BCF	0.52	0.69	0.93	0.97	2.43	3.79	2.51	0.50	1.89	0.93	0.89	2.38	5.64	2.56
	ps-BART	0.75	1.20	0.84	0.98	2.56	5.90	5.95	0.74	2.16	0.84	0.89	2.58	6.60	5.94
	CRF	0.87	1.01	0.65	0.82	2.15	3.06	0.09	0.93	2.12	0.64	0.72	2.45	4.10	0.09
	BART	1.11	1.47	0.56	0.94	2.39	5.65	5.94	1.11	2.30	0.56	0.86	2.40	6.36	6.02
	BART-f_0f_1	2.08	3.15	0.10	0.83	2.38	8.68	7.93	1.96	3.32	0.14	0.83	2.38	9.12	7.89
	lm	1.82	2.28	0.24	0.69	2.55	5.68	1.41	1.64	2.28	0.37	0.75	2.51	6.01	1.33

D OBSERVATIONAL DATA

The data set for the empirical data demonstration is publicly available at the UCI Machine Learning Repository (`https://archive.ics.uci.edu/ml/datasets/Student+Performance`).