Complex & Intelligent Systems (2024) 10:6805–6823
https://doi.org/10.1007/s40747-024-01516-x

ORIGINAL ARTICLE

An adaptive trimming approach to Bayesian additive regression trees

Taoyun Cao[1,2] · Jinran Wu[3] · You-Gan Wang[4]

Received: 6 November 2023 / Accepted: 30 May 2024 / Published online: 22 June 2024

Abstract
A machine learning technique merging Bayesian method called Bayesian Additive Regression Trees (BART) provides a nonparametric Bayesian approach that further needs improved forecasting accuracy in the presence of outliers, especially when dealing with potential nonlinear relationships and complex interactions among the response and explanatory variables, which poses a major challenge in forecasting. This study proposes an adaptive trimmed regression method using BART, dubbed BART(Atr) to improve forecasting accuracy by identifying suspected outliers effectively and removing these outliers in the analysis. Through extensive simulations across various scenarios, the effectiveness of BART(Atr) is evaluated against three alternative methods: default BART, robust linear modeling with Huber's loss function, and data-driven robust regression with Huber's loss function. The simulation results consistently show BART(Atr) outperforming the other three methods. To demonstrate its practical application, BART(Atr) is applied to the well-known Boston Housing Price dataset, a standard regression analysis example. Furthermore, random attack templates are introduced on the dataset to assess BART(Atr)'s performance under such conditions.

Keywords Robust regression · Outliers · Adaptive trimmed regression · Bayesian additive regression trees · Forecasting

Introduction

Bayesian Additive Regression Trees (BART), introduced by Chipman et al. [1], has gained significant attention due to its ability to effectively capture nonlinear relationships even in complex scenarios [2–6]. It combines the precision of likelihood-based inference with the flexibility of machine learning algorithms. BART is a non-parametric Bayesian regression approach that models the response variable as a sum of many small trees, each contributing a small portion to the overall prediction. This ensemble of trees allows for capturing intricate patterns and interactions in the data, while the Bayesian framework provides a principled way to quantify uncertainty and prevent overfitting. Theoretical investigations explore the concentration of posterior distributions, providing empirical evidence of BART's effectiveness [2]. Furthermore, BART's Bayesian nature allows for the incorporation of prior information and the quantification of uncertainty in the predictions, making it a valuable tool for decision-making under uncertainty. BART models have been successfully applied to a wide range of problems, including predicting solar radiation [7], forecasting sales [8], and identifying accident hot spot [9].

Despite the advancements made by the BART, it still faces several fundamental challenges when handling outliers. Firstly, BART employs a sum of regression trees as its base learners. Outliers in the responses can lead to splits that overfit those outlier points. Secondly, unlike other robust regression techniques like M-estimators, BART lacks an explicit mechanism to downweight or remove the influence of outliers during model fitting. It relies solely on the tree ensemble structure and regularization to provide robustness.

✉ Jinran Wu
ryan.wu@acu.edu.au

Taoyun Cao
cty@gdufe.edu.cn

You-Gan Wang
ygwanguq2012@gmail.com

1 School of Statistics and Mathematics, Guangdong University of Finance and Economics, Guangzhou 510320, People's Republic of China

2 Big Data and Educational Statistics Application Laboratory, Guangdong University of Finance and Economics, Guangzhou 510320, People's Republic of China

3 Institute for Positive Psychology and Education, Australian Catholic University, Banyo 4014, Australia

4 School of Mathematics and Physics, The University of Queensland, St Lucia 4067, Australia

Thirdly, BART models the mean response as a sum of trees, and outliers can increase the residual error, which is then modeled by a separate residual error term. If there are many outliers, say, more than 25%, modeling this complex residual error becomes more challenging [3]. Lastly, BART uses Bayesian posterior averaging over many trees. While this provides robustness against overfitting, it may also cause the influence of outliers to persist across multiple trees in the ensemble [10]. Hence, it is apparent that achieving improved forecasting accuracy in the presence of outliers using BART remains a challenging task. Therefore, in this work, we aim to enhance the reliability and trustworthiness of BART-based forecasting models, enabling their deployment in real-world applications where data is contaminated with outliers.

In statistics, traditional regression analysis relies on least squares procedures, which are potent tools. However, the presence of outliers in the data poses a significant challenge for accurate analysis within these traditional regression models. Hence, there is a pressing need to devise more suitable methods capable of handling outliers effectively. The pioneering work in robust statistics, including Tukey, Huber, and Hampel, are recognized for their seminal contributions [11–13]. Their work laid the foundation for robust statistical techniques. Recently, a review of robust statistics has been elaborated upon [14], where the M-estimation method, the most important class for the field of data rectification, is the main focus of these papers [14–17]. In robust linear regression, emphasis is placed on considering only residuals that do not deviate excessively when estimating regression parameters. This approach is encapsulated by the least trimmed squares estimator. Additionally, an adaptive least trimmed squares method has been devised to robustly estimate both the proportion of affected data and the coefficients in a multiple linear regression model [18–20], where the adaptive least trimmed squares method is a data-driven method which adaptively estimates the proportion of unaffected data p according to the data. Recently, such techniques have been successfully applied to address challenges in electricity demand forecasting against cyberattacks, leveraging an adaptive trimmed regression method [18].

As mentioned before, the presence of outliers in data poses a significant challenge for accurate analysis using the traditional BART computational algorithm. Therefore, our motivation is to construct a robust version of BART to enhance prediction performance by introducing the trimmed regression approach. Outliers can lead to splits that overfit the affected branches. We propose a trimmed approach so that the suspected outliers are trimmed, and their bad effects can therefore be alleviated. A key and challenging issue is what proportion we should trim—over-trimming will lose useful data necessarily while under-trimming will cause some outliers to be treated as "normal" data and hence have more detrimental effects on prediction.

Following our motivation, we introduce an adaptive trimmed regression approach based on BART, BART(Atr), which extends the robust approach by incorporating data-driven tuning parameters, as described in VandenHeuvel et al. [18], where it was originally applied to linear regression. More precisely, we extend the adaptive trimmed method to BART, which is a nonparametric regression model involving nonlinear regression, employing an inclusive framework that incorporates the estimation of the scale parameter, the proportion, and the weight function. Specifically, a suitable weight function for each data point, only residuals that do not exceed a certain threshold are utilized to estimate the true regression function. We explore the performance of our proposed method through simulation studies featuring varying attack rates and diverse distributions, as well as real data analysis involving random attacks, under the guidance of Lemma and Theorem. To demonstrate the effectiveness of BART(Atr), we perform a comparative analysis with three alternative methods: BART(Def), RLM, and RLMD. BART(Def), as mentioned in Chipman et al. [1], is a simpler and more efficient version of BART. RLM utilizes robust regression with Huber's loss function [12], while RLMD applies data-driven robust regression with the same loss function [21]. It is worth noting that Huber's loss function is widely used in robust regression, particularly in the M-estimation method [14]. Both RLM and RLMD are recognized as powerful tools for robust regression. So, we compare BART(Atr) against these three methods: BART(Def), RLM, and RLMD.

Therefore, the main contributions in this paper can be summarized as follows.

a. We present an extended robust regression technique based on BART, incorporating data-driven tuning parameters. This approach builds upon the original application of the method in linear regression, as described in VandenHeuvel et al. [18].
b. We propose an iterative procedure that robustly trains a comprehensive framework, encompassing the estimation of the scale parameter σ, the proportion of outliers to be trimmed, $1 - p$, and the weight function $\psi(\cdot)$ under the guidance of Lemma and Theorem. The weight function $\psi(\cdot)$ is carefully selected for each data point.
c. To establish the efficacy of our training procedure, we conduct simulation studies and evaluate its performance. Furthermore, we analyze a real-world scenario involving random attacks. The experimental results unequivocally demonstrate the superior performance and forecasting capabilities of our proposed method compared to three alternative methods.

The paper is organized as follows: In "An adaptive trimmed BART" Section outlines the proposed method,

followed by "Simulation studies" Section, which presents simulation studies evaluating its performance. In "Real data analysis" Section illustrates the application of the proposed method to real data subjected to random attacks. Finally, "Conclusions" Section provides conclusions and outlines avenues for future research.

An adaptive trimmed BART

BART

BART consists of three primary components: a sum-of-trees model, a regularization prior to the parameters of this model, and a Bayesian backfitting MCMC algorithm. Each of these components will be briefly described below.

Sum-of-trees model

Suppose that observations $D = \{(\boldsymbol{x_i}, y_i), x_i \in R^c, y_i \in R, i = 1, 2, \ldots, n\}$ are independent identical distribution $(i.i.d.)$ from a population, where $\boldsymbol{x_i}$ and y_i represent c dimensional vector and variable, respectively. A BART model by Chipman et al. [1] assumes the observations D follow:

$$y_i = f(\boldsymbol{x_i}) + \epsilon_i, \quad i = 1, \ldots, n, \tag{1}$$

where $f(\boldsymbol{x_i})$ is represented as the regression function by the following expression:

$$f(\boldsymbol{x_i}) = g(\boldsymbol{x_i}, \mathcal{T}_1, \mathcal{M}_1) + \ldots + g(\boldsymbol{x_i}, \mathcal{T}_m, \mathcal{M}_m). \tag{2}$$

The error terms $\epsilon_1, \ldots, \epsilon_n$ in Eq. (1) are i.i.d. from a normal distribution, denoted as $\epsilon_i \sim N(0, \sigma^2)$. In Eq. (2), $\mathcal{T}_j$ represents the jth regression tree, where j ranges from 1 to m. Each of the trees is a binary regression tree consisting of a set of interior node splitting rules and a set of terminal nodes. $\mathcal{M}_j = \{\mu_{j1}, \ldots, \mu_{jb_j}\}$ represents the collection of parameters for the b_j leaves of the jth tree. Given the jth tree $\mathcal{T}_j$, the function $g(\boldsymbol{x_i}, \mathcal{T}_j, \mathcal{M}_j)$ assigns a value in $\mathcal{M}_j$ to $\boldsymbol{x_i}$ based solely on which terminal node $\boldsymbol{x_i}$ belongs to.

Regularization prior

The entities $\mathcal{T}_1, \ldots, \mathcal{T}_m$, $\mathcal{M}_1, \ldots, \mathcal{M}_m$, and σ correspond to the structures of the trees, the parameters of the terminal nodes, and the standard deviation of the model described in Eq. (1), respectively. To account for a regularization prior applied to these parameters, the following prior distribution is considered:

$$P((\mathcal{T}_1, \mathcal{M}_1), \ldots, (\mathcal{T}_m, \mathcal{M}_m), \sigma) = \left\{ \prod_{j=1}^{m} \left(\prod_{k=1}^{b_j} P\left(\mu_{jk} \mid \mathcal{T}_j\right) \right) P\left(\mathcal{T}_j\right) \right\} P(\sigma). \tag{3}$$

This prior distribution is determined based on a set of independence and symmetry assumptions. As a result, the priors for $\mathcal{T}_j, \mu_{jk} \mid \mathcal{T}_j$, and σ as depicted in Eq. (3) can be expressed by introducing hyperparameters such as α, β, k, m, ν, and q, along with the application of conjugate prior distributions. The default settings for hyperparameters are recommended in this study (as described in "Simulation studies" Section). For more comprehensive details, refer to Chipman et al. [1].

Backfitting MCMC algorithm

A Bayesian backfitting MCMC algorithm can be used to sample, and Bayesian setup induces a posterior distribution $P((\mathcal{T}_1, \mathcal{M}_1), \ldots, (\mathcal{T}_m, \mathcal{M}_m), \sigma \mid D)$ given observations D. Specifically, the sample $\mathcal{T}_j$ can be obtained using the Metropolis-Hastings algorithm of CGM98 in Chipman et al. [22]; the draw of $\mathcal{M}_j$ conditionally on $\mathcal{T}_j$, $\mathcal{M}_j = \{\mu_{j1}, \ldots, \mu_{jb_j}\}$, is implemented by independently draws from a normal distribution; to sample σ^2, conditional on the updated tree structures $\{\mathcal{T}_1, \ldots, \mathcal{T}_m\}$ and the terminal node parameters $\{\mathcal{M}_1, \ldots, \mathcal{M}_m\}$, is implemented from an inverse χ^2 distribution. Thus, generating a sequence of draws of $(\mathcal{T}_1, \mathcal{M}_1), \ldots, (\mathcal{T}_m, \mathcal{M}_m), \sigma$, it is converging (in distribution) to the posterior $P((\mathcal{T}_1, \mathcal{M}_1), \ldots, (\mathcal{T}_m, \mathcal{M}_m), \sigma \mid D)$. $\hat{f}(\boldsymbol{x_i}) = \frac{1}{\kappa} \sum_{\iota=1}^{\kappa} f_\iota^*(\boldsymbol{x_i})$ is a natural choice [1], where $1, \ldots, \kappa$ denotes the index of sequence after burn-in period, and $f_\iota^*(\boldsymbol{x_i}) = \sum_{j=1}^{m} g(\boldsymbol{x_i}, \mathcal{T}_j, \mathcal{M}_j)$. It may be of interest to note that $\hat{f}(\boldsymbol{x_i})$ approximates the posterior mean. Thereby, $\hat{\epsilon}_i = y_i - \hat{f}(\boldsymbol{x_i}), i = 1, \ldots, n$.

For clarity, Algorithm 1 presents the outline of BART as referenced in Wang et al. [23].

An adaptive trimmed BART

In this subsection, we will examine the statistical properties of the residuals, $\hat{\epsilon}_i = y_i - \hat{f}(\boldsymbol{x_i})$, which can be approximated with a mean of $E(\hat{\epsilon}_i) = 0$ and a variance of $Var(\hat{\epsilon}_i) = \sigma^2$. This approximation is justified based on the following reasoning.

Firstly, let's consider the mean:

$$E(\hat{\epsilon}_i) = E(y_i) - E(\hat{f}(\boldsymbol{x_i})) = f(\boldsymbol{x_i}) - E\left(\frac{1}{\kappa} \sum_{\iota=1}^{\kappa} f_\iota^*(\boldsymbol{x_i})\right) \approx f(\boldsymbol{x_i}) - f(\boldsymbol{x_i}) = 0,$$

Algorithm 1 BART Algorithm

Input:
1: $(\boldsymbol{x}, \boldsymbol{y})$ with $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_n})^{\mathrm{T}}$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)^{\mathrm{T}}$;
2: α, β, k, ν, q: hyper-parameters in priors;
3: x_0: a test point to predict;
4: κ: number of posterior samples drawn to make predictions;
5: m: number of trees used in BART;
Output:
6: $\hat{f}(x_0)$: the predicted value by using predictor x_0;
7:
8: Initialize σ^0 by drawing a sample from $p(\sigma)$;
9: **for** $s = 1, 2, \ldots, \kappa$ **do**
10: **for** $j = 1, 2, \ldots, m$ **do**
11: **if** $s = 1$ and $j = 1$ **then**
12: Initialize residual vector $R_{-1} = \boldsymbol{y}$ and $\mathcal{T}_1^0 =$ a decision stump;
13: **else**
14: Update $R_{-j} = \boldsymbol{y} - \sum_{\iota \neq j}^{\kappa} g(\boldsymbol{x}; \mathcal{T}_\iota^{(s-1)}, \mathcal{M}_\iota^{(s-1)})$;
15: **end if**
16: Sample a $\mathcal{T}_j^{(s)}$ from $p(\mathcal{T}_j \mid R_{-j}, \sigma^{(s-1)})$ using Gibbs sampling;
17: Sample a $\mathcal{M}_j^{(s)}$ from $p(\mathcal{M}_j \mid \mathcal{T}_j^{(s)}, R_{-j}, \sigma^{(s-1)})$ using Gibbs sampling;
18: **end for**
19: Sample a $\sigma^{(s)}$ from $p(\sigma \mid (\mathcal{T}_1^{(s)}, \mathcal{M}_1^{(s)}), \ldots, (\mathcal{T}_m^{(s)}, \mathcal{M}_m^{(s)}), \boldsymbol{y})$ based on an inverse gamma distribution;
20: **end for**
21: Draw $f_1^*(x_0), \ldots, f_\kappa^*(x_0)$ from posterior distribution and compute $\hat{f}(x_0) = (1/\kappa)\sum_{s=1}^{\kappa} f_s^*(x_0)$ with $f_s^*(x_0) = \sum_{j=1}^{m} g(x_0, \mathcal{T}_j^{*(s)}, \mathcal{M}_j^{*(s)}), s = 1, 2, \ldots, \kappa$.

where f_ι^* refers to the function induced by the ιth posterior draw, and converges to the posterior distribution on the true function f, as mentioned in Chipman et al. [1].

Secondly, let's consider the variance:

$$\begin{aligned} Var(\hat{\epsilon}_i) &= Var(y_i - \hat{f}(\boldsymbol{x_i})) = Var(f(\boldsymbol{x_i}) + \epsilon_i - \hat{f}(\boldsymbol{x_i})) \\ &= Var(\epsilon_i - \hat{f}(\boldsymbol{x_i})) \approx \sigma^2; \end{aligned}$$

this approximation relies on the delta method and a condition. More precisely, it is Taylor's formula to the first order, $g(x) \approx g(x_0) + g^{'}(x_0)(x - x_0)$, and next, calculate the expected value conditionally on $E(\epsilon_i \hat{f}(\boldsymbol{x_i})) = E((\hat{f}(\boldsymbol{x_i}) - f(\boldsymbol{x_i}))\hat{f}(\boldsymbol{x_i}))$.

Considering y_i follows a normal distribution from the assumption of Eq. (1), as well as the same distribution for $\hat{f}(\boldsymbol{x_i})$ according to central limit theorem [24], we assume $\hat{\epsilon}_i = y_i - \hat{f}(\boldsymbol{x_i})$ follows a normal distribution with mean 0 and variance σ^2, and these $\hat{\epsilon}_i$'s are i.i.d.. With this assumption, the distribution of the absolute residuals $|\hat{\epsilon}_1|, \ldots, |\hat{\epsilon}_n|$ follows the folded normal distribution, denoted as $FN(0, \sigma^2)$, with location 0 and scale σ^2.

For the sake of simplification in notation, let's use φ, Φ, Φ^{-1} to represent the density function, cumulative distribution function, and quantile function of the standard normal distribution, respectively. Furthermore, we'll employ $f(\delta) = 2\varphi(\delta/\sigma)/\sigma$, $F(\delta) = 2\Phi(\delta/\sigma) - 1$, $F^{-1}(\delta) = \sigma\Phi^{-1}((1+\delta)/2)$ to characterize the density function, cumulative distribution function, and quantile function of the folded normal distribution, $FN(0, \sigma^2)$. Now, considering $|\hat{\epsilon}|_{(i)}$ as the i-th smallest value in the set of absolute residuals $|\hat{\epsilon}_1|, \ldots, |\hat{\epsilon}_n|$, ordered in ascending order, i.e., $|\hat{\epsilon}|_{(1)} \leq |\hat{\epsilon}|_{(2)} \leq \cdots \leq |\hat{\epsilon}|_{(n)}$, we can express the following relationships:

$$\tilde{p}_j = \frac{j}{n+1}, \quad \xi_j = F^{-1}(\tilde{p}_j), \quad j = 1, \ldots, n. \tag{4}$$

Define s_i^2 as follows

$$s_i^2 = \sum_{j=1}^{i} |\hat{\epsilon}|_{(j)}^2 / i. \tag{5}$$

To find the distribution of s_i^2, the following theorem is given as follows.

Lemma *Suppose* $i \in \{1, 2, \ldots, n\}$ *and consider* $|\hat{\epsilon}|_{(1)}, |\hat{\epsilon}|_{(2)}, \ldots, |\hat{\epsilon}|_{(n)}$. *Then, the asymptotic joint distribution of* $\sqrt{n}(|\hat{\epsilon}|_{(1)} - \xi_1), \ldots, \sqrt{n}(|\hat{\epsilon}|_{(i)} - \xi_i)$ *is* i*-dimensional normal with zero mean vector and covariance matrix* $\sum_{(i)}$, *defined such that its* (j, k) *entry* $\sigma_{(i),jk}$ *as* $\sigma_{(i),jk} = \frac{\tilde{p}_j(1-\tilde{p}_k)}{f(\xi_j)f(\xi_k)}$, $1 \leq j \leq k \leq i$.

Proof This lemma could be regarded as a modification of the more general result introduced by David and Nagaraja ([25], Theorem 10.3) [25], which is the asymptotics of a joint distribution of order statistics. We need to verify the hypotheses of the theorem. More precisely,

1. The first hypothesis is that $0 < \tilde{p}_1 < \tilde{p}_2 < \cdots < \tilde{p}_i < 1$, it holds, due to $\tilde{p}_j$ in (4), naturally, $0 < \frac{1}{n+1} < \frac{2}{n+1} < \cdots < \frac{i}{n+1} < 1$.
2. The second hypothesis is that $r_j/n - \tilde{p}_j = o(n^{-\frac{1}{2}})$, here $r_j = j$. Considering $\sqrt{n}[r_j/n - \tilde{p}_j] = j/[\sqrt{n}(n+1)]$, $j = 1, \ldots, i$, $\sqrt{n}[r_j/n - \tilde{p}_j]$ is infinitesimal of higher order for $n^{-\frac{1}{2}}$. This condition is true.
3. The third hypothesis is that $0 < f(\xi_j) < \infty, j = 1, \ldots, i$. Considering the density f is defined in terms of φ, according to the property of φ, $0 < f(\xi_j) < \infty$, $j = 1, \ldots, i$, it holds.
4. The final hypothesis is that F for these absolute residuals is differentiable at each ξ_j. Considering $F(\delta) = 2\Phi(\delta/\sigma) - 1$ and Φ is differentiable, so too is F.

Thus, all hypotheses given by David and Nagaraja (2003, Theorem 10.3) [25] are satisfied, and the lemma is proved. □

Theorem *Suppose the residuals* $\hat{\epsilon}_1, \hat{\epsilon}_2, \ldots, \hat{\epsilon}_n$ *be i.i.d., each following the distribution* $N(0, \sigma^2)$, $\sigma > 0$. *Then, provided that* $|\hat{\epsilon}|_{(1)} > 0$,

$$\sqrt{n}\left\{s_i^2 - \frac{1}{i}\sum_{j=1}^{i}\xi_j^2\right\} \Big/ \left(\frac{2}{i}\sqrt{\zeta_i^T \Sigma_{(i)}\zeta_i}\right) \xrightarrow{d} N(0,1), i = 1, \ldots, n,$$

as $n \to \infty$, *where* $\xrightarrow{d}$ *denotes convergence in distribution.*

Proof Define $|\hat{\varepsilon}|_{(i)} = (|\hat{\epsilon}|_{(1)}, \ldots, |\hat{\epsilon}|_{(i)})^T, \zeta_i = (\xi_1, \ldots, \xi_i)^T$. Lemma shows that $\sqrt{n}(|\hat{\varepsilon}|_{(i)} - \zeta_i) \xrightarrow{d} N_i(\mathbf{0}_{(i)}, \Sigma_{(i)})$, where $\mathbf{0}_{(i)}$ is an i-dimensional zero vector. Furthermore, note that $s_i^2 = g(|\hat{\varepsilon}|_{(i)}) = \sum_{j=1}^{i}|\hat{\epsilon}|_{(j)}^2/i$, according to the multivariate delta method (Wasserman, 2004, Theorem 5.15) [26] we considered the following

$$\sqrt{n}\left\{s_i^2 - \frac{1}{i}\sum_{j=1}^{i}\xi_j^2\right\} \Big/ \left(\frac{2}{i}\sqrt{\zeta_i^T \Sigma_{(i)}\zeta_i}\right) \xrightarrow{d} N(0,1)$$

conditional on $|\hat{\epsilon}|_{(1)} > 0$ and the gradient of $g(|\hat{\varepsilon}|_{(i)}) > 0$. For continuous y_i, $|\hat{\epsilon}|_{(1)} = 0$ is a probability zero event, the gradient function of $g(|\hat{\varepsilon}|_{(i)})$ is equal to $\frac{2}{i}|\hat{\varepsilon}|_{(i)}$.

Thus, all conditions given by Wasserman ([26], Theorem 5.15) [26] satisfied, the theorem is proved. □

From the theorem, we have the following modified MAD estimator for σ^2

$$\hat{\sigma}^2 = \sum_{i=1}^{\lfloor np\rfloor}|\hat{\epsilon}|_{(i)}^2 \Big/ \sum_{i=1}^{\lfloor np\rfloor}\left[\Phi^{-1}\left(\frac{1+\tilde{p}_i}{2}\right)\right]^2, \tag{6}$$

where $\lfloor np\rfloor$ denotes the value of np rounded down. The upper index in the sums in (6) are set to $\lfloor n/4\rfloor$, which ensures that only 25% clean data are used for estimating σ^2.

Thus, the asymptotic mean of s_i^2 is given by $E(s_i^2) = \frac{1}{i}\sum_{j=1}^{i}\xi_j^2$. Define the set $S = \{i \in \{1, \ldots, n\} : s_i^2/E(s_i^2) \le Q\}$ and set

$$\hat{p}^{(\text{Iter})} = |S|/n, \text{Iter} = 1, 2, \ldots, \text{Iter}_{\max}, \tag{7}$$

where Iter, $|S|$, $\text{Iter}_{\max}$ and Q denote the iteration number, the number of indices in S, the maximum number of iterations and the tuning parameter, respectively.

Next, the weight function is defined by

$$\psi(\hat{\epsilon}_i) = \begin{cases} 0, & |\hat{\epsilon}_i| > |\hat{\epsilon}|_{(\lfloor np\rfloor)}, \\ 1, & |\hat{\epsilon}_i| \le |\hat{\epsilon}|_{(\lfloor np\rfloor)}, \end{cases} \tag{8}$$

where $\psi(\hat{\epsilon}_i)$ assigns a weight of 1 to nonoutliers and 0 to outliers.

Run BART model using the sample with $\psi(\hat{\epsilon}_i) = 1$. Afterwards, update $\hat{\epsilon}_i$ with the new model and calculate the revised estimations for σ^2, p, and the weights $\psi(\hat{\epsilon}_1), \ldots, \psi(\hat{\epsilon}_n)$. Continue this process iteratively until the absolute difference between $\hat{p}^{(\text{Iter})}$ and $\hat{p}^{(\text{Iter}-1)}$ divided by $\hat{p}^{(\text{Iter})}$ is less than a specified tolerance τ, where τ is a predefined value, or a maximum number of iterations, $\text{Iter}_{\max}$, is met. It's important to note that this iterative process converges within a finite number of steps [19], and in this context, τ is set to be 10^{-4} as referenced in VandenHeuvel et al. [18].

We now outline the procedure of our BART(Atr) as follows:

- Step 1: Start by obtaining the initial estimate of ϵ_i using the BART model for the observations $(x_{i1}, x_{i2}, \ldots, x_{ic}, y_i), i = 1, \ldots, n$, denoted as $\hat{\epsilon}_i^{(\text{Iter})}$.
- Step 2: Next, compute the estimate of σ^2 using (6) for the order statistics of $\hat{\epsilon}_i^{(\text{Iter})}$, and use (4) to obtain values for $\tilde{p}_i$ as well as set $\lfloor np\rfloor = \lfloor n/4\rfloor$.
- Step 3: Calculate the estimate of p using (7) and Theorem for s_i^2 and $E(s_i^2)$, then update the value of $\hat{p}^{(\text{Iter})}$; Noted that with the initial value $\hat{p}^{(0)} = 0.5$.
- Step 4: Compute the weights $w_i = \psi(\hat{\epsilon}_i)$ using (8). Fit the BART model to the samples with weights w_i and update $\hat{\epsilon}_i^{(\text{Iter}+1)}$.
- Step 5: Repeat Steps 2 to 4 iteratively until the condition $|\hat{p}^{(\text{Iter})} - \hat{p}^{(\text{Iter}-1)}|/\hat{p}^{(\text{Iter})} < \tau$ or a maximum number of iterations, $\text{Iter}_{\max}$, is met, where $\tau = 10^{-4}$, $\text{Iter}_{\max} = 20$.

In addition, the training flowchart for the proposed adaptive trimmed Bayesian additive regression trees is presented in Fig. 1.

Remark The BART(Atr) procedure begins by obtaining an initial estimate of the residuals ϵ_i using the BART model on the observations. Steps 2 to 4 then calculate estimates for the residual variance σ^2, the trimming proportion p, and the weights $w_i = \psi(\hat{\epsilon}_i)$, guided by the provided Lemma and Theorem. The residual estimate ϵ_i is then updated. Step 5 iterates Steps 2 to 4 until one of the specified conditions is met. Notably, the estimation of σ^2 and p in Steps 2 and 3 follows the guidance of the Lemma and Theorem. It is important to highlight that the primary objective is to improve forecasting accuracy in the presence of outliers by employing BART(Atr).

Simulation studies

In this section, we delve into a nonlinear model found in the BART literature [1, 24], originally introduced by Friedman [27]. We aim to employ this model to show the capabilities of BART(Atr) and compare it with the three methods (BART(Def), RLM, and RLMD), as stated in "Introduction" Section, where BART(Def) is vastly easier and

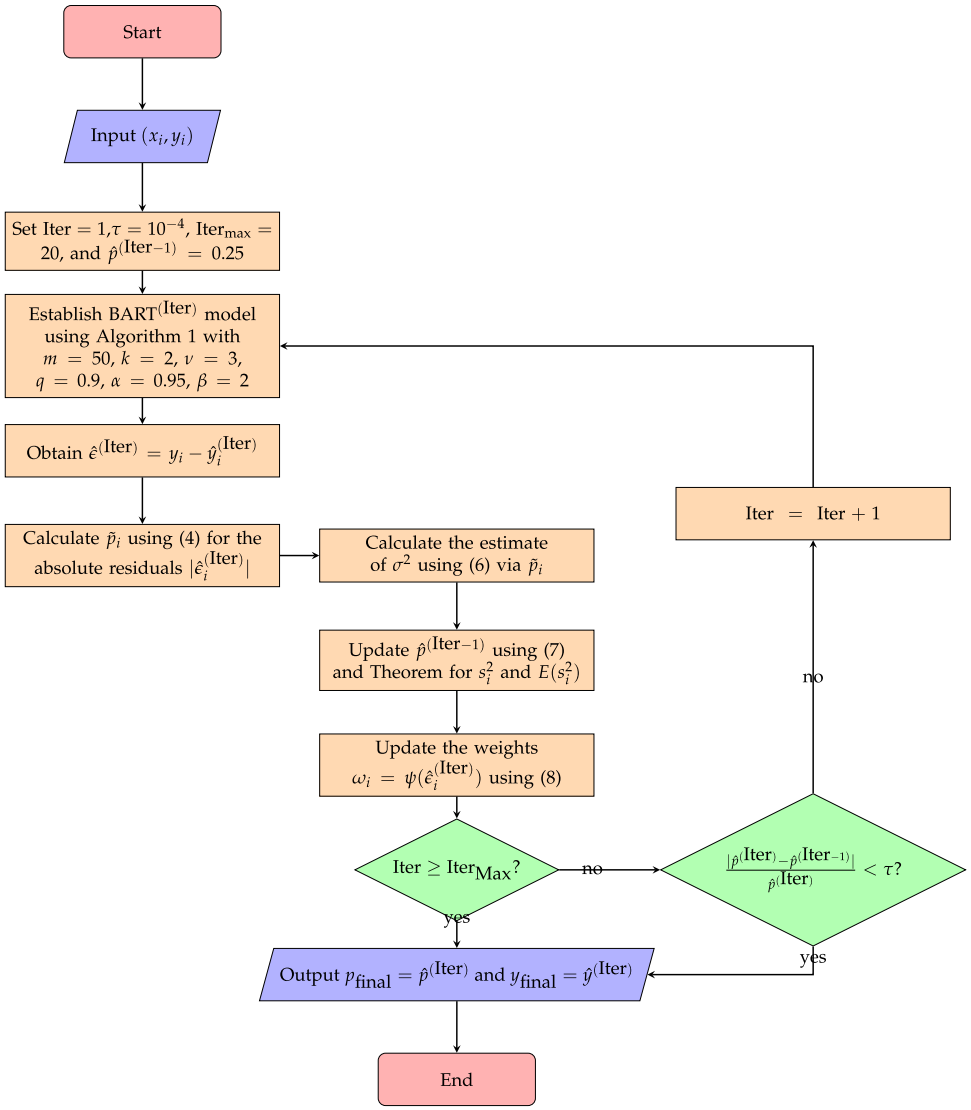

Fig. 1 The training procedure for the proposed adaptive trimmed Bayesian additive regression trees

faster to use in Chipman et al. [1], RLM is the linear model by robust regression with the Huber's loss function, and RLMD is the data-driven robust regression with the Huber's loss function, respectively. All experiments are conducted using R's 'bartMachine' [28], 'rlmDataDriven' [29], and 'MASS' [30] packages. To facilitate the easy implementation of the four methods, the default setting considered in this paper is shown in Table 1.

Experimental settings

We examine the nonlinear model given by:

$$y_i = 10\sin(\pi x_{i1}x_{i2}) + 20(x_{i3} - 0.5)^2 + 10x_{i4} + 5x_{i5} + \epsilon_i, \quad i = 1, \ldots, n. \tag{9}$$

In this model, the variables $x_{i1}, x_{i2}, \ldots, x_{i5}$ are i.i.d. from Uniform(0,1). The error term ϵ_i follows a mixture distribution, represented as $pN(0, 0.1^2) + (1-p)F$. Here, p, $1-p$, and n denote the proportion of unattacked data, attacked data, and the sample size, respectively. We consider three different distributions for F as follows:

- **Case 1.** Normal distribution, where the errors are attacked by a normal distribution, $N(0, 1.5^2)$, with the proportion of attacked $1-p$. Different $1-p$ are considered, that is, $1-p = 0\%, 5\%, 10\%, 15\%, 20\%, 25\%, 30\%, 35\%, 40\%, 45\%$, and 50%.
- **Case 2.** Student's t distribution, where the errors are attacked by $t(3)$ with three degrees of freedom, with the proportion of attacked $1-p$. Different $1-p$ are considered, that is, $1-p = 0\%, 5\%, 10\%, 15\%, 20\%, 25\%, 30\%, 35\%, 40\%, 45\%$, and 50%.
- **Case 3.** Exponential distribution, where the errors are attacked by an exponential distribution, $Exp(0.5) - 2$, with the proportion of attacked $1-p$. Different $1-p$

Table 1 Methods, packages, and default settings

Method	Package	Default setting
BART (Atr)	bartMachine	$m = 50, k = 2, \nu = 3, q = 0.9, \alpha = 0.95, \beta = 2$
BART (Def)	bartMachine	$m = 50, k = 2, \nu = 3, q = 0.9, \alpha = 0.95, \beta = 2$
RLM	MASS	Psi = "psi.huber", $k = 1.345$
RLMD	rlmDataDriven	Method ="huber"

are considered, that is, $1 - p = 0\%, 5\%, 10\%, 15\%, 20\%, 25\%, 30\%, 35\%, 40\%, 45\%$, and 50%.

Cases 1–3 are the normal errors, attacked via three distributions: $N(0, 1.5^2)$, $t(3)$, and $Exp(0.5) - 2$ with the proportion of attacked $1 - p$. We simulate i.i.d. datasets with $n = 1000$ and fit the model using the proposed method, BART(Atr), for $p \in \{0.5, 0.55, \ldots, 0.95, 1.0\}$. Considering $Q \in \{1.0, 1.07, \ldots, 1.49\}$, at the cost of increased computation time, $Q = 1.28$ is selected in this work based on fitting the model (9), according to the majority of the minimal MAE and MAPE values in Case 1. Thus, we use $Q = 1.28$ for all cases in BART(Atr). All are implemented 100 times in all cases for the four methods, BART(Atr), BART(Def), RLM, and RLMD.

Evaluation criterion

We assess the performance of the four methods using three key metrics: the estimates $\hat{p}$ for p, mean absolute error (MAE), and mean absolute percentage error (MAPE). In the case of $\hat{p}$ estimates, we collect 100 simulation results and then calculate the average and median for each p. The last two metrics are defined as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\mu_i - \hat{y}_i|,$$

and

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\frac{|\mu_i - \hat{y}_i|}{|\mu_i|},$$

where $\hat{y}_i$ and μ_i represent the predicted value for y_i and the expected value in the model (9), respectively. Notably, the expected value μ_i corresponds to the observed value without the additional noise.

Results and discussion

The simulation results are presented in Tables 2, 3, and 4. Table 2 provides the results in Case 1 for the estimates of p, $\hat{p}$, and the MAE and MAPE values. Our method, BART(Atr), performs better than the other three methods in terms of all the metrics. For instance, at $p = 0.75$, firstly, the estimate $\hat{p}$ for p lies within $(0.701, 0.754)$, where the interval is given by the mean and median from BART(Atr), while it is not provided in BART(Def), RLM and RLMD. Moreover, the MAE of 0.309 (mean) and 0.306 (median) are given by BART(Atr), whereas 0.383 (mean) and 0.381 (median) are given by BART(Def); 1.770 and 1.763 are given by RLM and RLMD, respectively. Here 1.770 and 1.763 largely outweigh 0.309 and 0.383, so, the following focus on the two methods, BART(Atr) and BART(Def). An obvious decline from 0.383 (or 0.381) to 0.309 (or 0.306), a difference of about 24% times. Similarly, we have a MAPE of 2.6% with BART(Atr), whereas 3.2%, 15%, and 14.9% are provided in BART(Def), RLM, and RLMD, respectively. In the same way, BART(Atr) and BART(Def) are focused on, for BART(Atr), a difference of about 23%. At the same time, it should be noted that there is no difference between the two methods, BART(Atr) and BART(Def) when $p = 1$. To summarise, BART(Atr) provides better estimates of $\hat{p}$ in Case 1 and provides greater forecasting accuracy in the four methods.

Table 3 presents the findings for Case 2, including estimates of p represented as $\hat{p}$, as well as the accompanying MAE and MAPE values. Just as in the previous case, BART(Atr) outperforms the other three methods across all metrics. For example, at $p = 0.85$, the estimated value $\hat{p}$ falls within the interval of $(0.848, 0.898)$, as defined by the mean and median values from BART(Atr). However, BART(Def), RLM, and RLMD do not provide such an interval. Furthermore, BART(Atr) reports an MAE of 0.267 (mean) and 0.261 (median), while BART(Def) yields an MAE of 0.342 (mean) and 0.340 (median), 1.769 and 1.763 are given in RLM and RLMD, respectively. Here 1.769, and 1.763 largely outweigh 0.267 and 0.342, as in the previous case, the following focus on the two methods, BART(Atr) and BART(Def). This results in a significant reduction, approximately 28%, when transitioning from BART(Def) to BART(Atr). Similarly, in terms of MAPE, BART(Atr) records a value of 2.2%, while BART(Def) provides a MAPE of 2.8%, leading to a difference of about 27%. It's important to note that there is no difference between the two methods, BART(Atr) and BART(Def) when $p = 1$. Therefore, BART(Atr) delivers improved estimates of $\hat{p}$ that consistently overestimate the true p values, particularly when $p = 0.65, 0.6$, and it signif-

Table 2 Simulation study results in Case 1

Method	p	$\hat{p}$		MAE		MAPE (%)	
		Mean	Median	Mean	Median	Mean	Median
BART(Atr)	1	0.972	1	**0.222**	**0.220**	**1.8**	**1.8**
BART(Def)		–	–	**0.220**	**0.219**	**1.8**	**1.8**
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.95	0.893	0.956	**0.243**	**0.240**	**2.0**	**2.0**
BART(Def)		–	–	0.272	0.272	2.2	2.2
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.762	14.9	14.9
BART(Atr)	0.9	0.864	0.922	**0.254**	**0.252**	**2.1**	**2.1**
BART(Def)		–	–	0.308	0.306	2.5	2.5
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.85	0.813	0.867	**0.275**	**0.269**	**2.3**	**2.2**
BART(Def)		–	–	0.336	0.335	2.8	2.8
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.8	0.741	0.792	**0.295**	**0.288**	**2.4**	**2.4**
BART(Def)		–	–	0.361	0.361	3.0	3.0
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.75	0.701	0.754	**0.309**	**0.306**	**2.6**	**2.6**
BART(Def)		–	–	0.383	0.381	3.2	3.2
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.7	0.663	0.700	**0.328**	**0.324**	**2.7**	**2.7**
BART(Def)		–	–	0.399	0.402	3.3	3.3
RLM		–	–	1.771	1.771	15.0	15.0
RLMD		–	–	1.764	1.764	14.9	14.9
BART(Atr)	0.65	0.617	0.611	**0.350**	**0.348**	**2.9**	**2.9**
BART(Def)		–	–	0.418	0.417	3.4	3.5
RLM		–	–	1.771	1.771	15.0	15.0
RLMD		–	–	1.765	1.765	14.9	14.9
BART(Atr)	0.6	0.600	0.570	**0.367**	**0.363**	**3.1**	**3.0**
BART(Def)		–	–	0.430	0.429	3.5	3.5
RLM		–	–	1.772	1.772	15.0	15.0
RLMD		–	–	1.767	1.765	14.9	14.9
BART(Atr)	0.55	0.548	0.500	**0.399**	**0.399**	**3.4**	**3.3**
BART(Def)		–	–	0.446	0.445	3.7	3.7
RLM		–	–	1.772	1.772	15.0	15.0
RLMD		–	–	1.767	1.765	14.9	15.0
BART(Atr)	0.5	0.552	0.500	**0.415**	**0.416**	**3.5**	**3.5**
BART(Def)		–	–	0.457	0.457	3.8	3.8
RLM		–	–	1.773	1.773	15.0	15.0
RLMD		–	–	1.769	1.767	14.9	14.9

– Implies null
Values in bold indicate the best predictive performance

Table 3 Simulation study results in Case 2

Method	p	$\hat{p}$		MAE		MAPE (%)	
		Mean	Median	Mean	Median	Mean	Median
BART(Atr)	1	0.937	1	**0.227**	**0.222**	**1.9**	**1.8**
BART(Def)		–	–	**0.220**	**0.220**	**1.8**	**1.8**
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.95	0.892	0.973	**0.245**	**0.238**	**2.0**	**2.0**
BART(Def)		–	–	0.269	0.268	2.2	2.2
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.762	14.9	14.9
BART(Atr)	0.9	0.882	0.941	**0.251**	**0.248**	**2.1**	**2.0**
BART(Def)		–	–	0.312	0.310	2.6	2.6
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.85	0.848	0.898	**0.267**	**0.261**	**2.2**	**2.2**
BART(Def)		–	–	0.342	0.340	2.8	2.8
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.8	0.800	0.853	**0.284**	**0.281**	**2.4**	**2.3**
BART(Def)		–	–	0.374	0.369	3.1	3.1
RLM		–	–	1.769	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.75	0.750	0.809	**0.299**	**0.293**	**2.5**	**2.4**
BART(Def)		–	–	0.392	0.389	3.2	3.2
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.7	0.705	0.768	**0.314**	**0.311**	**2.6**	**2.6**
BART(Def)		–	–	0.411	0.407	3.4	3.4
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.764	1.763	14.9	14.9
BART(Atr)	0.65	0.679	0.721	**0.329**	**0.326**	**2.7**	**2.7**
BART(Def)		–	–	0.435	0.431	3.6	3.6
RLM		–	–	1.771	1.771	15.0	15.0
RLMD		–	–	1.765	1.764	14.9	14.9
BART(Atr)	0.6	0.642	0.629	**0.355**	**0.339**	**3.0**	**2.9**
BART(Def)		–	–	0.448	0.448	3.7	3.7
RLM		–	–	1.772	1.772	15.0	15.0
RLMD		–	–	1.765	1.765	14.9	14.9
BART(Atr)	0.55	0.599	0.5	**0.370**	**0.366**	**3.1**	**3.1**
BART(Def)		–	–	0.465	0.463	3.8	3.8
RLM		–	–	1.772	1.771	15.0	15.0
RLMD		–	–	1.765	1.764	14.9	14.9
BART(Atr)	0.5	0.580	0.5	**0.387**	**0.386**	**3.2**	**3.2**
BART(Def)		–	–	0.481	0.477	4.0	3.9
RLM		–	–	1.772	1.773	15.0	15.0
RLMD		–	–	1.766	1.765	14.9	14.9

– Implies null
Values in bold indicate the best predictive performance

Table 4 Simulation study results in Case 3

Method	p	$\hat{p}$		MAE		MAPE (%)	
		Mean	Median	Mean	Median	Mean	Median
BART(Atr)	1	0.956	1	**0.223**	**0.220**	**1.8**	**1.8**
BART(Def)		-	–	**0.219**	**0.218**	**1.8**	**1.8**
RLM		–	–	1.769	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.95	0.884	0.956	**0.247**	**0.241**	**2.0**	**2.0**
BART(Def)		–	–	0.288	0.286	2.4	2.3
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.9	0.841	0.894	**0.256**	**0.253**	**2.1**	**2.1**
BART(Def)		–	–	0.338	0.338	2.8	2.8
RLM		–	–	1.770	1.769	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.85	0.782	0.845	**0.276**	**0.270**	**2.3**	**2.2**
BART(Def)		–	–	0.375	0.371	3.1	3.0
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.8	0.720	0.773	**0.298**	**0.291**	**2.5**	**2.4**
BART(Def)		–	–	0.412	0.410	3.4	3.4
RLM		–	–	1.770	1.770	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.75	0.663	0.723	**0.319**	**0.309**	**2.6**	**2.5**
BART(Def)		–	–	0.439	0.439	3.6	3.6
RLM		–	–	1.771	1.772	15.0	15.0
RLMD		–	–	1.763	1.763	14.9	14.9
BART(Atr)	0.7	0.645	0.662	**0.337**	**0.334**	**2.8**	**2.8**
BART(Def)		–	–	0.459	0.459	3.8	3.8
RLM		–	–	1.773	1.772	15.0	15.0
RLMD		–	–	1.765	1.765	14.9	14.9
BART(Atr)	0.65	0.602	0.603	**0.359**	**0.356**	**3.0**	**3.0**
BART(Def)		–	–	0.480	0.476	4.0	3.9
RLM		–	–	1.773	1.773	15.0	15.0
RLMD		–	–	1.766	1.766	14.9	14.9
BART(Atr)	0.6	0.574	0.5	**0.388**	**0.388**	**3.2**	**3.2**
BART(Def)		–	–	0.499	0.496	4.1	4.1
RLM		–	–	1.774	1.774	14.9	14.9
RLMD		–	–	1.767	1.766	14.8	14.8
BART(Atr)	0.55	0.552	0.5	**0.416**	**0.415**	**3.4**	**3.4**
BART(Def)		–	–	0.520	0.522	4.3	4.3
RLM		–	–	1.775	1.775	14.9	14.9
RLMD		–	–	1.768	1.768	14.8	14.8
BART(Atr)	0.5	0.533	0.5	**0.451**	**0.450**	**3.7**	**3.7**
BART(Def)		–	–	0.534	0.533	4.4	4.4
RLM		–	–	1.776	1.776	14.9	14.9
RLMD		–	–	1.770	1.769	14.8	14.8

– Implies null
Values in bold indicate the best predictive performance

icantly outperforms BART(Def) as well as RLM and RLMD in the context of Case 2.

Table 4 displays the outcomes for Case 3, encompassing the estimates of p denoted as $\hat{p}$, and the associated MAE and MAPE values. Much like in the previous cases, BART(Atr) outperforms the other three methods across all metrics. For example, when $p = 0.95$, the estimate $\hat{p}$ falls within the range of $(0.884, 0.956)$, as defined by the mean and median values from BART(Atr). However, BART(Def), RLM, and RLMD do not provide such an interval. Furthermore, BART(Atr) produces an MAE of 0.247 (mean) and 0.241 (median), whereas BART(Def) yields an MAE of 0.288 (mean) and 0.286 (median), 1.770 and 1.763 are given in RLM and RLMD. Here 1.770, and 1.763 largely outweigh 0.247 and 0.288, as in the previous cases, the following focus on the two methods, BART(Atr) and BART(Def). This indicates a noticeable reduction, approximately 17% when transitioning from BART(Def) to BART(Atr). Similarly, in terms of MAPE, BART(Atr) reports a value of 2%, while BART(Def) provides a MAPE of 2.4%, resulting in a difference of about 20%. It's important to note that there is no difference between the two methods, BART(Atr) and BART(Def) when $p = 1$. In summary, BART(Atr) offers estimates of $\hat{p}$ that consistently underestimate the true p values, particularly when $p = 0.8, 0.75, 0.7, 0.65, 0.6$, and it significantly outperforms the other three methods in the context of Case 3.

Tables 2, 3, and 4 demonstrate that BART(Atr) consistently outperforms the other three methods across all cases, as viewed from three different metrics, the estimates of p, $\hat{p}$, MAE and MAPE values. It's important to highlight that both the mean and median values of MAE and MAPE increase as the parameter p decreases from 1 to 0.5 in both BART(Atr) and BART(Def). This trend appears reasonable given that the proportion of attacked data, represented by $1 - p$, increases from 0 to 0.5. Furthermore, it's worth noting that BART(Atr) delivers the best performance among all cases, especially when it comes to estimating the parameter p, while the other three methods cannot provide estimates for the parameter p.

As depicted in Fig. 2, it is evident that for Case 1, the true values of p are almost consistently found within the intervals represented by Esti (Mean) and Esti (Median). In other words, these intervals tend to encompass the true values of p. Similarly, for Case 2 and Case 3, the true values of p are also well-contained within such intervals. The novel method, BART(Atr), performs exceptionally well and consistently yields similar results for Cases 1, 2, and 3. It's worth noting that in Case 2, two intervals overestimate p. For instance, when the true value is $p = 0.6$, the estimated value falls within the range of $\hat{p} \in [0.629, 0.642]$. On the other hand, in Case 3, five intervals underestimate p. For instance, when the true value is $p = 0.7$, the estimated value falls within the range of $\hat{p} \in [0.645, 0.662]$. As previously described in VandenHeuvel et al. [18], it is preferable to have a method that underestimates p, which means potentially removing clean data points during the regression function estimation. BART(Atr) contributes to greater forecasting accuracy, in contrast to the other three methods, especially for BART(Def).

Real data analysis

In this section, we demonstrate the application of the BART(Atr) model using the Boston Housing Price dataset, a well-known example commonly used for regression analysis [31, 32]. This dataset is accessible via the R package MASS and comprises 506 entries with 14 variables. The response variable is "medv" (median value of owner-occupied homes), and the dataset includes the following covariates: "crim" (per capita crime rate by town), "zn" (proportion of residential land zoned for lots over 25,000 square feet), "indus" (proportion of non-retail business acres per town), "chas" (Charles River, equal to one if the tract borders the river, zero otherwise), "nox" (nitrogen oxides concentration), "rm" (average number of rooms per dwelling), "age" (proportion of owner-occupied units built before 1940), "dis" (weighted mean of distances to five Boston employment centers), "rad" (index of accessibility to radial highways), "tax" (full-value property-tax rate), "ptratio" (pupil-teacher ratio by town), "black" $(1000(B - 0.63)^2$, where B represents the proportion of black residents by town), and "lstat" (lower status of the population). Descriptive statistics for the Boston Housing Price dataset can be found in Table 5.

Table 5 provides statistics for the variables under consideration, which include minimum and maximum values, as well as mean and median values. It's worth noting that, due to the discrete nature of the "chas" variable (taking values 0 or 1), mean statistics are not computed for it. Notably, the "black" and "zn" variables display a substantial difference between their maximum and minimum values, indicating higher volatility compared to the other variables. Additionally, the response variable "medv" exhibits nearly identical mean and median values, suggesting a symmetric distribution for "medv". In the subsequent analysis, we use $\mu_0 = 23$, which is consistent with the mean of "medv".

Random attacks

We consider random attacks, defined by

$$y_{i,a} = \left(1 + \frac{s}{100}\right) y_i,$$

where y_i is the unattacked data, $y_{i,a}$ is the attacked data, and $s \sim N(\mu, \sigma^2)$. The $1 + s\%$ factor denotes randomly scales

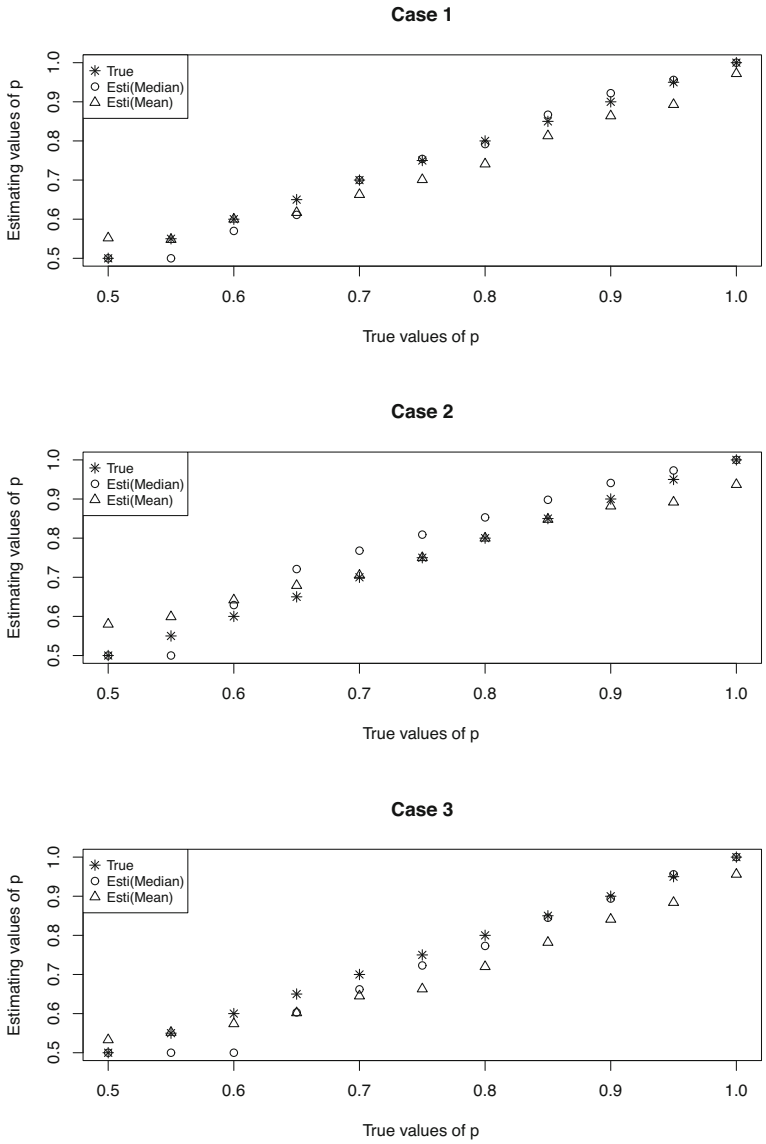

Fig. 2 Estimates $\hat{p}$ for p in all Cases. Here $*$,$\circ$, and $\triangle$ refer to true values of p, estimates of p using the new method by the median and mean for 100 simulations, respectively. Namely, $*$,$\circ$ and $\triangle$ denote True, Esti (median), Esti (mean)

Table 5 Descriptive statistics in the Boston housing price dataset

Variables	Min	Max	Mean	Median	Variables	Min	Max	Mean	Median
Crim	0.006	88.976	3.614	0.257	Dis	1.130	12.127	3.795	3.207
zn	0	100	11.364	0	Rad	1	24	9.549	5
Indu	0.46	27.74	11.137	9.690	Tax	187	711	408.237	330
Chas	0	1	–	0	Ptra	12.60	22.00	18.456	19.050
Nox	0.385	0.871	0.555	0.538	Black	0.32	396.90	356.674	391.440
rm	3.561	8.780	6.285	6.209	Lsta	1.73	37.97	12.653	11.360
Age	2.9	100	68.575	77.5	Medv	5	50	22.533	21.2

The discrete values (0,1) in chas, the statistics of the mean is not computed

the data in the random attack. Here considering rand-attack 1 and 2:

1) **rand-attack 1:** $\mu = \mu_0, \sigma = 50, 80, 150$; and
2) **rand-attack 2:** $\mu = 2\mu_0, \sigma = 50, 80, 150$.

Generate a training dataset by randomly choosing 70% of the entries, while reserving the remaining 30% for the test dataset. Ensure a fair comparison by subjecting the training set to a random attack 100 times, and subsequently assess the outcomes, which encompass MAE and MAPE values, on the test set.

Results

Tables 6 and 7 display the results for "rand-attack 1" and "rand-attack 2". In "rand-attack 1", Table 6 presents the MAE and MAPE values, as well as the running time, where MAE and MAPE are derived from 100 independent realizations, and the running time is the mean value of time in 100 realizations. Notably, when comparing BART(Atr) to the other three methods, BART(Atr) consistently yields significantly lower MAE and MAPE values. For instance, at a given parameter setting of $1 - p = 0.1$ and $\sigma = 50$, BART(Atr) achieves a mean MAE of 2.634, while 2.888, 3.319 and 3.292 are provided in BART(Def), RLM and RLMD, respectively. This represents a substantial decrease, approximately 10%, 26%, 25%, in MAE from BART(Def), RLM, and RLMD to BART(Atr). At the same time, BART(Atr) achieves a median MAE of 2.616, while 2.850, 3.326, and 3.302 are provided in BART(Def), RLM, and RLMD, respectively. This represents a substantial decrease, approximately 9%, 27%, 26%, in MAE from BART(Def), RLM, and RLMD to BART(Atr). Similarly, the mean MAPE for BART(Atr) is 12.8%, whereas 14.2%, 16.5%, and 16.2% are provided in BART(Def), RLM, and RLMD, respectively. For BART(Def), RLM, and RLMD, they resulted in a roughly 11%, 29%, and 27% improvement with BART(Atr). At the same time, the median MAPE for BART(Atr) is 13%, whereas 14%, 16.6%, and 16.2% are provided in BART(Def), RLM, and RLMD, respectively. For BART(Def), RLM, and RLMD, they resulted in a roughly 8%, 28%, and 25% improvement with BART(Atr). Therefore, BART(Atr) exhibits superior performance to the three methods across MAE and MAPE metrics, offering significantly higher accuracy in the context of "rand-attack 1". However, it should be noted that BART(Atr) needs more computational costs. More computational costs are required to find a proper value of p, where p signifies the proportion of unaffected data as stated in "Introduction" Section.

Table 7 displays the results for "rand-attack 2", presenting the MAE and MAPE values, as well as the running time. In the context of "rand-attack 2", BART(Atr) almost consistently outperforms BART(Def) by yielding notably lower MAE and MAPE values. For instance, under the specified parameter settings of $1 - p = 0.1$ and $\sigma = 50$, BART(Atr) achieves a mean MAE of 2.725, in contrast, BART(Def), RLM, and RLMD produce mean MAE of 2.930, 3.341 and 3.275. These result in a significant reduction, approximately 8%, 23%, and 20% in mean MAE when transitioning from BART(Def), RLM, and RLMD to BART(Atr). At the same time BART(Atr) achieves a median MAE of 2.694, in contrast, BART(Def), RLM, and RLMD produce mean MAE of 2.900, 3.342, and 3.265. These result in a significant reduction, approximately 8%, 24%, and 21% in median MAE when transitioning from BART(Def), RLM, and RLMD to BART(Atr). Similarly, the mean MAPE for BART(Atr) is 13.7%, whereas BART(Def), RLM, and RLMD yield the MAPE of 15%, 17.3%, and 16.7% indicating a substantial improvement of approximately 9.5%, 26.3% and 21.9% with BART(Atr). At the same time, the median MAPE for BART(Atr) is 13.6%, whereas BART(Def), RLM, and RLMD yield the MAPE of 14.9%, 17.1%, and 16.7% indicating a substantial improvement of approximately 9.6%, 25.7% and 22.8% with BART(Atr). In summary, BART(Atr) almost consistently demonstrates superior performance to the three methods across the MAE and MAPE values metrics within the context of "rand-attack 2", showcasing a considerable increase in accuracy.

It's important to note that BART(Atr) isn't the best in the specified parameter settings of $1-p = 0.4, 0.5$ and $\sigma = 150$, RLMD is the best in four methods. It shows that RLMD is a good choice under "rand-attack 2" with $1 - p = 0.4, 0.5$ and $\sigma = 150$. Just as in the previous "rand-attack 1" BART(Atr) gets the maximum value in four methods from a running time standpoint. It is noted that more computational costs in BART(Atr) are required to find a proper value of p, as stated in "Introduction" Section, where p signifies the proportion of unaffected data.

Figure 3 indicates boxplots in "rand-attack 1" with $\mu = \mu_0, \sigma = 50$. According to the reviewer's opinion, considering six boxplots as Fig. 3 for "rand-attack 1" and "rand-attack 2" with different $1 - p = 0.1, 0.2, 0.3, 0.4, 0.5$ and different $\sigma = 50, 80, 150$, here only Fig. 3 as an example. Each boxplot in Fig. 3 represents 100 MAE and MAPE values in the test set based on 100 independent realizations. BART(Atr) is the best, which coincides with Table 6. Note also that Fig. 3 shows the MAE and MAPE values increase with a growth of $1 - p$ in the methods.

To demonstrate the significance of our proposed BART(Atr) method in forecasting, we conducted a Wilcoxon signed-rank test using MAE and MAPE indexes. These indexes were obtained from 100 repeated experiments on the test set, specifically under rand-attack 1, as mentioned in Wu and Wang [33]. The results of the statistical tests are documented in Table 8. Based on the tests, it was determined

Table 6 Forecasting results of the Boston housing price data under rand-attack 1

Method	Parameters			MAE		MAPE (%)		Time (sec)
	$1-p$	μ	σ	Mean	Median	Mean	Median	Mean
BART(Atr)	0.1	μ_0	50	**2.634**	**2.616**	**12.8**	**13.0**	4.712
BART(Def)				2.888	2.850	14.2	14.0	2.644
RLM				3.319	3.326	16.5	16.6	0.016
RLMD				3.292	3.302	16.2	16.2	0.056
BART(Atr)	0.2	μ_0	50	**2.717**	**2.720**	**13.3**	**13.3**	11.487
BART(Def)				2.975	2.974	14.8	14.8	6.800
RLM				3.395	3.384	16.8	16.9	0.035
RLMD				3.323	3.327	16.2	16.1	0.186
BART(Atr)	0.3	μ_0	50	**2.814**	**2.830**	**13.8**	**13.8**	10.746
BART(Def)				3.604	3.562	17.5	17.3	7.082
RLM				3.499	3.480	17.6	17.4	0.035
RLMD				3.359	3.324	16.7	16.6	0.247
BART(Atr)	0.4	μ_0	50	**3.069**	**3.046**	**15.1**	**15.0**	10.754
BART(Def)				3.507	3.457	17.1	16.9	6.830
RLM				3.569	3.539	17.9	17.9	0.036
RLMD				3.404	3.392	16.9	17.0	0.227
BART(Atr)	0.5	μ_0	50	**3.698**	**3.681**	**17.9**	**17.9**	8.374
BART(Def)				4.405	4.349	21.3	21.2	6.839
RLM				4.234	4.200	21.9	21.9	0.036
RLMD				3.896	3.882	20.0	20.1	0.256
BART(Atr)	0.1	μ_0	80	**2.698**	**2.696**	**13.2**	**13.1**	15.668
BART(Def)				3.368	3.313	16.8	16.6	8.326
RLM				3.289	3.302	16.5	16.5	0.040
RLMD				3.247	3.238	16.0	16.1	0.335
BART(Atr)	0.2	μ_0	80	**2.736**	**2.716**	**13.5**	**13.5**	13.225
BART(Def)				3.619	3.596	17.9	18.0	8.047
RLM				3.348	3.339	16.4	16.4	0.048
RLMD				3.300	3.278	15.8	15.8	0.439
BART(Atr)	0.3	μ_0	80	**2.811**	**2.792**	**14.7**	**14.6**	13.666
BART(Def)				4.041	4.041	20.6	20.3	9.012
RLM				3.478	3.452	18.3	18.1	0.043
RLMD				3.329	3.322	17.2	17.1	0.472
BART(Atr)	0.4	μ_0	80	**3.154**	**3.150**	**15.5**	**15.6**	10.412
BART(Def)				4.393	4.372	21.0	20.7	9.520
RLM				3.734	3.747	19.0	19.0	0.042
RLMD				3.516	3.518	17.6	17.7	0.479
BART(Atr)	0.5	μ_0	80	**3.373**	**3.338**	**16.6**	**16.3**	10.900
BART(Def)				4.464	4.471	22.1	22.0	8.943
RLM				3.896	3.899	19.6	19.6	0.038
RLMD				3.665	3.681	18.3	18.0	0.529
BART(Atr)	0.1	μ_0	150	**2.727**	**2.653**	**13.2**	**13.0**	16.168
BART(Def)				5.248	5.284	24.1	23.9	8.378
RLM				3.304	3.324	16.6	16.5	0.035
RLMD				3.277	3.298	16.3	16.3	0.362

Table 6 continued

Method	Parameters			MAE		MAPE (%)		Time (sec)
	$1-p$	μ	σ	Mean	Median	Mean	Median	Mean
BART(Atr)	0.2	μ_0	150	**2.962**	**2.950**	**14.0**	**13.9**	14.903
BART(Def)				5.172	5.029	24.6	24.2	8.841
RLM				3.343	3.309	16.5	16.4	0.037
RLMD				3.263	3.228	16.0	15.9	0.353
BART(Atr)	0.3	μ_0	150	**3.232**	**3.196**	**15.6**	**15.3**	14.272
BART(Def)				6.570	6.538	31.5	31.2	9.656
RLM				3.499	3.478	17.6	17.6	0.051
RLMD				3.299	3.275	16.3	16.3	0.580
BART(Atr)	0.4	μ_0	150	**3.410**	**3.330**	**15.9**	**15.5**	11.779
BART(Def)				8.053	7.888	35.1	34.6	9.335
RLM				3.575	3.598	18.1	17.9	0.046
RLMD				3.371	3.339	16.7	16.6	0.464
BART(Atr)	0.5	μ_0	150	**3.672**	**3.595**	**17.5**	**17.3**	11.230
BART(Def)				7.120	7.090	32.6	32.3	10.518
RLM				4.069	4.040	21.0	20.9	0.053
RLMD				3.567	3.554	18.3	18.1	0.671

Values in bold indicate the best predictive performance
The unit of execution time: second

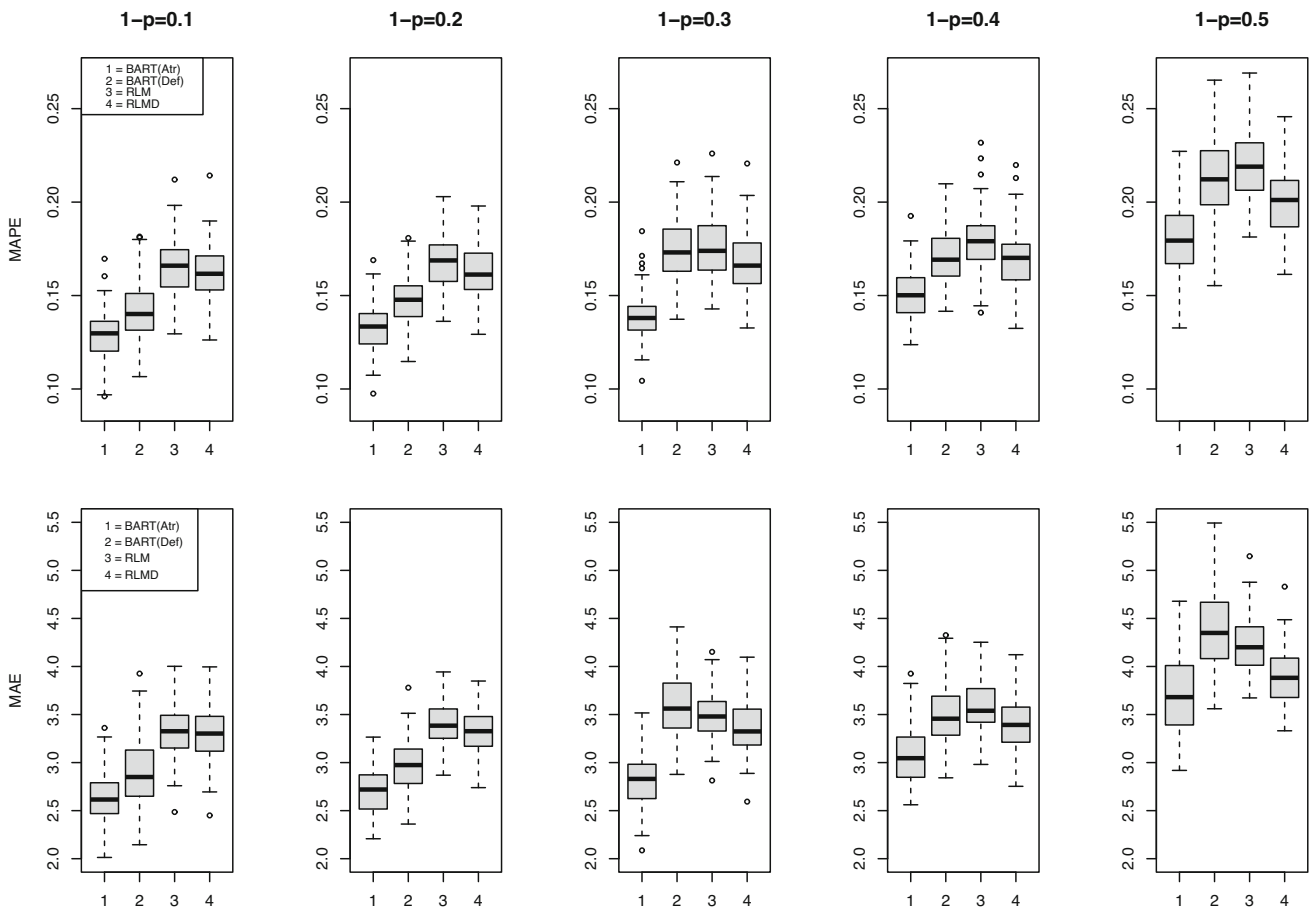

Fig. 3 Boxplots of results under rand-attack 1 with $\mu = \mu_0, \sigma = 50$

Table 7 Forecasting results of the Boston housing price data under rand-attack 2

Method	Parameters			MAE		MAPE (%)		Time (sec)
	$1-p$	μ	σ	Mean	Median	Mean	Median	Mean
BART(Atr)	0.1	$2\mu_0$	50	**2.725**	**2.694**	**13.7**	**13.6**	11.744
BART(Def)				2.930	2.900	15.0	14.9	8.862
RLM				3.341	3.342	17.3	17.1	0.125
RLMD				3.275	3.265	16.7	16.7	0.955
BART(Atr)	0.2	$2\mu_0$	50	**2.867**	**2.846**	**14.6**	**14.7**	11.320
BART(Def)				3.535	3.563	18.1	17.9	7.983
RLM				3.604	3.608	18.5	18.4	0.124
RLMD				3.425	3.422	17.3	17.3	0.866
BART(Atr)	0.3	$2\mu_0$	50	**3.182**	**3.177**	**16.0**	**15.7**	11.719
BART(Def)				4.528	4.493	22.1	21.8	9.209
RLM				4.005	3.983	20.6	20.6	0.045
RLMD				3.690	3.653	18.8	18.7	1.114
BART(Atr)	0.4	$2\mu_0$	50	**3.612**	**3.630**	**18.1**	**18.0**	10.099
BART(Def)				4.847	4.782	24.3	24.3	9.012
RLM				4.407	4.386	22.9	22.9	0.074
RLMD				4.003	4.008	20.6	20.3	0.753
BART(Atr)	0.5	$2\mu_0$	50	**4.396**	**4.388**	**22.3**	**22.3**	3.791
BART(Def)				5.771	5.712	29.1	28.9	2.777
RLM				5.085	5.052	26.5	26.5	0.014
RLMD				4.577	4.506	23.7	23.6	0.097
BART(Atr)	0.1	$2\mu_0$	80	**2.701**	**2.676**	**13.4**	**13.4**	10.789
BART(Def)				3.350	3.309	16.5	16.8	5.335
RLM				3.362	3.364	17.1	17.0	0.028
RLMD				3.315	3.322	16.6	16.6	0.143
BART(Atr)	0.2	$2\mu_0$	80	**2.798**	**2.797**	**13.8**	**13.7**	11.784
BART(Def)				4.434	4.325	21.1	21.0	6.646
RLM				3.331	3.296	16.7	16.6	0.038
RLMD				3.300	3.278	15.8	15.8	0.687
BART(Atr)	0.3	$2\mu_0$	80	**3.259**	**2.230**	**16.7**	**16.5**	10.071
BART(Def)				4.740	4.746	24.2	24.3	6.972
RLM				3.801	3.793	19.8	19.8	0.085
RLMD				3.468	3.429	17.9	17.7	0.610
BART(Atr)	0.4	$2\mu_0$	80	**3.525**	**3.421**	**17.3**	**17.1**	9.216
BART(Def)				5.821	5.790	27.5	27.2	7.417
RLM				4.126	4.070	21.8	21.5	0.033
RLMD				3.593	3.546	18.9	18.7	0.386
BART(Atr)	0.5	$2\mu_0$	80	**2.701**	**2.676**	**13.4**	**13.4**	10.789
BART(Def)				3.350	3.309	16.5	16.8	5.335
RLM				3.362	3.364	17.1	17.0	0.028
RLMD				3.315	3.322	16.6	16.6	0.142
BART(Atr)	0.1	$2\mu_0$	150	**2.794**	**2.777**	**13.5**	**13.5**	13.196
BART(Def)				4.913	4.809	23.2	22.6	6.480
RLM				3.315	3.333	16.5	16.4	0.036
RLMD				3.246	3.266	15.9	15.9	0.266

Table 7 continued

Method	Parameters			MAE		MAPE (%)		Time (sec)
	$1-p$	μ	σ	Mean	Median	Mean	Median	Mean
BART(Atr)	0.2	$2\mu_0$	150	**2.962**	**2.952**	**14.6**	**14.4**	12.250
BART(Def)				6.194	6.081	29.0	28.5	6.789
RLM				3.534	3.489	18.1	17.8	0.032
RLMD				3.353	3.302	17.0	16.8	0.272
BART(Atr)	0.3	$2\mu_0$	150	**3.330**	**3.240**	**16.3**	**15.8**	9.662
BART(Def)				6.831	6.824	32.4	31.8	8.324
RLM				3.665	3.660	18.6	18.4	0.036
RLMD				3.371	3.351	16.7	16.6	0.384
BART(Atr)	0.4	$2\mu_0$	150	3.522	3.433	17.1	16.8	10.428
BART(Def)				7.190	7.195	33.8	34.0	8.078
RLM				3.839	3.793	19.2	19.0	0.032
RLMD				**3.420**	**3.409**	**16.6**	**16.5**	0.295
BART(Atr)	0.5	$2\mu_0$	150	4.090	4.037	20.2	20.3	7.986
BART(Def)				7.534	7.435	36.3	35.5	7.395
RLM				4.116	4.066	20.6	20.5	0.034
RLMD				**3.573**	**3.547**	**17.8**	**17.7**	0.380

Values in bold indicate the best predictive performance
The unit of execution time: second

that BART(Atr) outperformed three other methods in terms of prediction accuracy. However, it is important to note that BART(Atr) incurred higher computational costs, as indicated in Tables 6 and 7. Regarding MAE with $1-p=0.4, 0.5$ and $\sigma=150$, neither BART(Atr) nor the RLMD methods exhibited statistical significance when considering $p \leq 0.05$.

Conclusions

In this study, we have introduced an extension to the robust regression method originally developed by Vandenheuvel et al. [18]. Our extension involves adaptive robust regression based on BART, which is tailored for nonlinear regression models. This novel approach referred to as BART(Atr), was employed to conduct analyses akin to those in the model presented in Chipman et al. [1], Cao and Zhang [24]. Specifically, we focused on forecasting in the presence of outliers. We further compared the performance of BART(Atr), as well as RLM, RLMD and BART(Def) in terms of forecasting metrics, including estimates of p, $\hat{p}$, and MAE and MAPE values. Our new method expands upon the concept of an asymptotic distribution for the order statistics of the residuals, utilizing BART to handle outliers in a regression context. We conducted a real data analysis involving random attacks, similar to the approach used in VandenHeuvel et al. [18], to assess the performance of our new method. The evaluation was based on metrics such as MAE and MAPE, and we measured these values across 100 independent realizations. Our findings consistently showed that the forecasts generated by the new method outperformed those of BART(Def), RLM, and RLMD in both simulation studies and real data analysis.

The findings presented in this study come with several significant limitations. Firstly, we restricted our analysis to random attack templates. For a more comprehensive understanding, especially in the context of forecasting electricity loads, it would be valuable to investigate the performance of BART(Atr) with different attack templates, such as ramp attacks. Future work should explore how the method performs under varying attack scenarios. Secondly, this study was primarily focused on the nonlinear regression model with i.i.d. errors. It might be beneficial to extend this framework to include models with non-i.i.d. errors, as contemplated in Sela and Simonof [34]. Furthermore, we specifically concentrated on the weight function $\psi(\cdot)$ mentioned in (8), following the recommendation in VandenHeuvel et al. [18]. We acknowledge that there exist various alternative weight functions, as extensively discussed in papers such as Wang et al. [21, 32], Wu and Wang [33], and Pratola et al. [35]. Among these alternatives, the two most commonly utilized functions are Huber's weight function and the bisquare weight function shown in Wang et al. [21], Jiao et al. [36]. Future research endeavors should delve into exploring additional weight functions, expanding the scope of investigation in this area. Lastly, we limited our investigation to the default settings in the BART model, including parameters like $\alpha=0.95$ and $\beta=2$, as recommended in Chipman et al. [1]. We can consider α as an example; it represents the probability asso-

Table 8 Results of the Boston housing price data under rand-attack 1 with four different methods

$1-p$	μ	σ	Index	BART(Atr)	BART(Def)	RLM	RLMD
0.1	μ_0	50	MAE	2.616	2.850*	3.326*	3.302*
	μ_0	50	MAPE	13.0	14.0*	16.6*	16.2*
0.2	μ_0	50	MAE	2.720	2.974*	3.384*	3.327*
	μ_0	50	MAPE	13.3	14.8*	16.9*	16.1*
0.3	μ_0	50	MAE	2.830	3.562*	3.480*	3.324*
	μ_0	50	MAPE	13.8	17.3*	17.4*	16.6*
0.4	μ_0	50	MAE	3.046	3.457*	3.539*	3.392*
	μ_0	50	MAPE	15.0	16.9*	17.9*	17.0*
0.5	μ_0	50	MAE	3.681	4.349*	4.200*	3.882*
	μ_0	50	MAPE	17.9	21.2*	21.9*	20.1*
0.1	μ_0	80	MAE	2.696	3.313*	3.302*	3.238*
	μ_0	80	MAPE	13.1	16.6*	16.5*	16.1*
0.2	μ_0	80	MAE	2.716	3.596*	3.339*	3.278*
	μ_0	80	MAPE	13.5	18.0*	16.4*	15.8*
0.3	μ_0	80	MAE	2.792	4.041*	3.452*	3.322*
	μ_0	80	MAPE	14.6	20.3*	18.1*	17.1*
0.4	μ_0	80	MAE	3.150	4.372*	3.747*	3.518*
	μ_0	80	MAPE	15.6	20.7*	19.0*	17.7*
0.5	μ_0	80	MAE	3.338	4.471*	3.899*	3.681*
	μ_0	80	MAPE	16.3	22.0*	19.6*	18.0*
0.1	μ_0	150	MAE	2.653	5.284*	3.324*	3.298*
	μ_0	150	MAPE	13.0	23.9*	16.5*	16.3*
0.2	μ_0	150	MAE	2.950	5.029*	3.309*	3.228*
	μ_0	150	MAPE	13.9	24.2*	16.4*	15.9*
0.3	μ_0	150	MAE	3.196	6.538*	3.478*	3.275*
	μ_0	150	MAPE	15.3	31.2*	17.6*	16.3*
0.4	μ_0	150	MAE	3.330	7.888*	3.598*	3.339
	μ_0	150	MAPE	15.5	34.6*	17.9*	16.6*
0.5	μ_0	150	MAE	3.595	7.090*	4.040*	3.554
	μ_0	150	MAPE	17.3	32.3*	20.9*	18.1*

'$*$' Represents the forecasting results are significant ($p \leq 0.05$) to our BART(Atr) method by the Wilcoxon signed-rank test

ciated with a splitting node given $\beta = 0$. In future research, it would be advantageous to explore data-driven approaches and examine how they compare to fixed-tuning parameter settings, as they tend to yield superior performance.

Acknowledgements The authors are grateful to the editor and all reviewers for their valuable feedback. The work is supported by the Australian Research Council project (Grant No. DP160104292) and "Chunhui Program" Collaborative Scientific Research Project (202202004).

Author Contributions Taoyun Cao: Writing-original draft, Methodology, Supervision, Software & Project administration. JinranWu: Methodology, Writing-review, Software & Visualization. You-Gan Wang: Methodology, Writing-review, Project administration & editing.

Data availability https://www.stats.ox.ac.uk/pub/MASS4.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical approval Not applicable.

References

1. Chipman HA, George EI, McCulloch RE (2010) BART: Bayesian additive regression trees. Ann Appl Stat 6(1):266–298
2. Rocková V, Van der Pas S et al (2020) Posterior concentration for Bayesian regression trees and forests. Ann Stat 48(4):2108–2131
3. Linero AR (2018) Bayesian regression trees for high-dimensional prediction and variable selection. J Am Stat Assoc 113(522):626–636
4. Murray JS (2021) Log-linear Bayesian additive regression trees for multinomial logistic and count regression models. J Am Stat Assoc 116(534):756–769
5. Hill J, Linero A, Murray J (2020) Bayesian additive regression trees: a review and look forward. Annu Rev Stat Appl 7:251–278
6. Pratola MT, Chipman HA, George EI, McCulloch RE (2020) Heteroscedastic BART via multiplicative regression trees. J Comput Graph Stat 29(2):405–417
7. Wu W, Tang X, Lv J, Yang C, Liu H (2021) Potential of Bayesian additive regression trees for predicting daily global and diffuse solar radiation in arid and humid areas. Renew Energy 177:148–163
8. Haselbeck F, Killinger J, Menrad K, Hannus T, Grimm DG (2022) Machine learning outperforms classical forecasting on horticultural sales predictions. Mach Learn Appl 7:100239
9. Krueger R, Bansal P, Buddhavarapu P (2020) A new spatial count data model with Bayesian additive regression trees for accident hot spot identification. Accident Anal Prevent 144:105623
10. Tan YV, Roy J (2019) Bayesian additive regression trees and the general BART model. Stat Med 38(25):5048–5069
11. Tukey JW (1960) A survey of sampling from contaminated distributions. In: Contributions to Probability and Statistics, pp 448–485
12. Huber PJ (1964) Robust estimation of a location parameter. Ann Math Stat 35(1):73–101
13. Hampel FR (1968) Contributions to the theory of robust estimation. PhD thesis, University of California, Berkeley
14. De Menezes D, Prata DM, Secchi AR, Pinto JC (2021) A review on robust M-estimators for regression analysis. Comput Chem Eng 147:107254
15. Fu L, Wang Y-G, Cai F (2020) A working likelihood approach for robust regression. Stat Methods Med Res 29(12):3641–3652
16. Wu J, Wang Y-G (2022) Iterative learning in support vector regression with heterogeneous variances. IEEE Trans Emerg Top Comput Intell 7(2):513–522
17. Song Y, Wu J, Fu L, Wang Y-G (2024) Robust augmented estimation for hourly $PM_{2.5}$ using heteroscedastic spatiotemporal models. Stoch Env Res Risk Assess 38(4):1423–1451
18. VandenHeuvel D, Wu J, Wang Y-G (2023) Robust regression for electricity demand forecasting against cyberattacks. Int J Forecast 39(4):1573–1592
19. Bacher R, Chatelain F, Michel O (2016) An adaptive robust regression method: application to galaxy spectrum baseline estimation. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, pp 4423–4427
20. Zhao S, Wu Q, Zhang Y, Wu J, Li X-A (2022) An asymmetric bisquare regression for mixed cyberattack-resilient load forecasting. Expert Syst Appl 210:118467
21. Wang Y-G, Lin X, Zhu M, Bai Z (2007) Robust estimation using the Huber function with a data-dependent tuning constant. J Comput Graph Stat 16(2):468–481
22. Chipman HA, George EI, McCulloch RE (1998) Bayesian CART model search. J Am Stat Assoc 93(443):935–948
23. Wang G, Zhang C, Yin Q (2019) RS-BART: a novel technique to boost the prediction ability of Bayesian additive regression trees. Chin J Eng Math 36(4):461–477
24. Cao T, Zhang R (2022) Research and application of Bayesian additive regression trees model for asymmetric error distribution. J Syst Sci Math Sci 42(11):15
25. David HA, Nagaraja HN (2003) Order statistics. John Wiley & Sons, Hoboken, New Jersey
26. Wasserman L (2004) All of statistics: a concise course in statistical inference. Springer, New York
27. Friedman JH (1991) Multivariate adaptive regression splines. Ann Stat 19(1):1–67
28. Kapelner A, Bleich J (2016) bartMachine: Machine learning with Bayesian additive regression trees. J Stat Softw 70:1–40
29. Wang Y-G, Liquet B, Callens A, Wang N (2019) rlmDataDriven: Robust regression with data driven tuning parameter. https://cran.r-project.org/web/packages/rlmDataDriven/rlmDataDriven.pdf
30. Ripley B, Venables B, Bates DM, Hornik K, Gebhardt A, Firth D, Ripley MB (2013) Package mass. Cran R 538:113–120
31. Breiman L, Friedman JH (1985) Estimating optimal transformations for multiple regression and correlation. J Am Stat Assoc 80(391):580–598
32. Wang X, Jiang Y, Huang M, Zhang H (2013) Robust variable selection with exponential squared loss. J Am Stat Assoc 108(502):632–643
33. Wu J, Wang Y-G (2023) A working likelihood approach to support vector regression with a data-driven insensitivity parameter. Int J Mach Learn Cybern 14(3):929–945
34. Sela RJ, Simonoff JS (2012) RE-EM trees: a data mining approach for longitudinal and clustered data. Mach Learn 86:169–207
35. Pratola MT, George EI, McCulloch RE (2024) Influential observations in Bayesian regression tree models. J Comput Graph Stat 33(1):47–63
36. Jiao J, Tang Z, Zhang P, Yue M, Yan J (2022) Cyberattack-resilient load forecasting with adaptive robust regression. Int J Forecast 38(3):910–919