Computational Statistics and Data Analysis 167 (2022) 107377

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

www.elsevier.com/locate/csda

Uncertainty quantification for honest regression trees

Suofei Wu [a], Jan Hannig [b,*], Thomas C.M. Lee [a]

[a] Department of Statistics, University of California at Davis, One Shields Ave, Davis, 95616 CA, United States of America
[b] Department of Statistics & Operations Research, University of North Carolina at Chapel Hill, 318 Hanes Hall, CB #3260, Chapel Hill, 27599 NC, United States of America

ARTICLE INFO

Article history:
Received 2 June 2021
Received in revised form 14 October 2021
Accepted 17 October 2021
Available online 21 October 2021

Keywords:
Bootstrap
Confidence intervals
Generalized fiducial inference
Jackknife
Prediction intervals

ABSTRACT

A new method is developed for quantifying the uncertainties of the estimates and predictions produced by honest random forests. This new method is based on the generalized fiducial methodology, and provides a fiducial density function that measures how likely each single honest tree is the true model. With such a density function, estimates and predictions, as well as their confidence/prediction intervals, can be obtained. The promising empirical properties of the proposed method are demonstrated by numerical comparisons with several state-of-the-art methods, and by applications to a few real data sets. Lastly, the proposed method is theoretically backed up by an asymptotic guarantee.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

Due to its robustness and accuracy, ensemble learning is a popular method in regression and classification (e.g., Mendes-Moreira et al., 2012). It is commonly used to make predictions for future observations. Denote the observed sample as $\{Y_i, \boldsymbol{X}_i\}$, $i = 1, \ldots, n$, where $Y_i \in \mathbb{R}$ are scalar responses and $X_i \in \mathbb{R}^p$ are vector predictors. The general regression model is

$$Y_i = f(\boldsymbol{X}_i) + \epsilon_i, \tag{1}$$

where the iid noise ϵ_i's follows $N(0, \sigma^2)$. An ensemble learning method approximates the model $f(\cdot)$ by a weighted sum of weak learners $T_i(\cdot)$'s with weights w_i's:

$$f(\boldsymbol{X}_i) = \sum_{i=1}^{a} w_i T_i(\boldsymbol{X}_i). \tag{2}$$

With their excellent interpretability, decision trees are often chosen as the weak learners. Notable earlier examples include bagging (Breiman, 1996) and random forests (Breiman, 2001). Recently, Athey et al. (2019) proposed generalized random forests that can be naturally extended to other statistical tasks such as quantile regression and heterogeneous treatment effect estimation. All of these three methods use the basic ensemble method (Perrone and Cooper, 1992) in regression, which takes all w_i's in (2) equally as $1/a$.

Wang et al. (2003) proposed a weighted ensemble approach for classification, where the classifiers are weighted by their accuracies in classifying their own training data. Their work can be straightforwardly extended to the regression case.

* Corresponding author at: 330 Hanes Hall, CB3260, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA.
E-mail addresses: swu@ucdavis.edu (S. Wu), jan.hannig@unc.edu (J. Hannig), tcmlee@ucdavis.edu (T.C.M. Lee).

https://doi.org/10.1016/j.csda.2021.107377
0167-9473/© 2021 Elsevier B.V. All rights reserved.

Bayesian ensemble learning (Chipman et al., 2007; Wu et al., 2007) is another approach that takes a weighted average of the trees. The posterior probabilities are used as weights in this scenario.

Despite the above efforts, the study of uncertainty quantification for ensemble learning is somewhat limited. One exception is Wager et al. (2014), where the authors proposed a method that produces standard error estimates $\hat{\sigma}$ for random forests predictions. The method is based on jackknife and infinite jackknife (Efron, 2014) and can be used for constructing Gaussian confidence intervals.

Another related work is Mentch and Hooker (2016), where the authors showed that under some strong assumptions, random forests based on subsampling are asymptotically normal, allowing for confidence intervals to accompany predictions. In addition, Chipman et al. (2010) developed a Bayesian Additive Regression Trees model (BART) that produces both point and interval estimates via posterior inference. And most recently Rockova (2020) provided some theoretical results on using BART for inference.

In this paper, we apply generalized fiducial inference (Hannig et al., 2016) to construct a probability density function on the set of honest trees in an honest random forests model. We shall show that such a new ensemble method of honest trees provides more precise confidence intervals as well as point estimates.

The rest of this paper is organized as follows. First, a brief introduction of generalized fiducial inference is provided in Section 2. Then the main methodology is presented in Section 3, and the theoretical properties of the method are studied in Section 4. Section 5 illustrates the practical performances of the proposed method. Lastly, concluding remarks are offered in Section 6 while technical details are delayed to the appendix.

2. Generalized fiducial inference

Fiducial inference was first introduced by Fisher (1930). It aims to construct a statistical distribution for the parameter space when no prior information is available. Under such condition, the usage of the classical Bayesian framework receives criticism because it requires a prior distribution of the parameter space. Alternatively, Fisher considered a switching mechanism between the parameters and the observations, which is quite similar to how parameters are estimated by the maximum likelihood method. Despite Fisher's continuous effort on the theory of fiducial inference, this framework was overlooked by the majority of the statistics community for several decades. Hannig et al. (2016) have a detailed introduction on the history of the original fiducial inference.

In recent years, there has been a renewed interest in extending Fisher's idea. The modified versions include Dempster-Shafer theory (Dempster, 2008; Martin et al., 2010), inferential models (Martin and Liu, 2015a,b), confidence distributions (Xie and Singh, 2013; Xie et al., 2011) and generalized inference (Weerahandi, 1995, 2013). In this paper we focus on the successful extension known as generalized fiducial inference (GFI) (Hannig et al., 2016). It has been successfully applied to a variety of problems, including wavelet regression (Hannig and Lee, 2009), ultrahigh-dimensional regression (Lai et al., 2015), nonparametric additive models (Gao et al., 2020) and principal component regression (Wu et al., 2021).

Under the GFI framework, the relationship between the data $\boldsymbol{y}$ and the parameter $\boldsymbol{\theta}$ is expressed by a data generating algorithm G:

$$\boldsymbol{y} = G(\boldsymbol{u}, \boldsymbol{\theta}),$$

where $\boldsymbol{u}$ is a random component whose distribution is completely *known*. Selection of the data generating algorithm plays a similar role to selecting a model in a more traditional statistical analysis. In this paper we will use data generating algorithm (1), where the function $f(X_i)$ is chosen as a tree with $l(T)$ leaves, i.e., $T(X_i|\mu_1, \ldots, \mu_{l(T)}) = \mu_k$ if X_i is in the kth leaf, and $\epsilon_i = \sigma U_i$, where U_i are i.i.d. N(0,1). Thus the random component of the data generating algorithm is $\boldsymbol{u} = (U_1, \ldots, U_n)$ and the parameters are $\boldsymbol{\theta} = (\mu_1, \ldots, \mu_{l(T)}, \sigma)$.

To explain how fiducial inference accounts for uncertainty, suppose for the moment that the inverse function G^{-1} exists for any $\boldsymbol{u}$; i.e., one can always calculate

$$\boldsymbol{\theta} = G^{-1}(\boldsymbol{u}, \boldsymbol{y})$$

for any $\boldsymbol{u}$. Then a random sample $\{\tilde{\boldsymbol{\theta}}_1, \tilde{\boldsymbol{\theta}}_2, \ldots\}$ of $\boldsymbol{\theta}$ can be obtained by first generating a random sample $\{\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, \ldots\}$ of $\boldsymbol{u}$ and then calculate

$$\tilde{\boldsymbol{\theta}}_1 = G^{-1}(\tilde{\boldsymbol{u}}_1, \boldsymbol{y}), \quad \tilde{\boldsymbol{\theta}}_2 = G^{-1}(\tilde{\boldsymbol{u}}_2, \boldsymbol{y}), \quad \ldots$$

Recall that as the distribution of $\boldsymbol{u}$ is known, one can always generate $\{\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, \ldots\}$ Notice that the roles of $\boldsymbol{\theta}$ and $\boldsymbol{y}$ are "switched" in the above, as in the maximum likelihood method of Fisher. When the inverse G^{-1} does not exist for some $\boldsymbol{u}$, the sample $\{\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, \ldots\}$ are generated from a distribution of $\boldsymbol{u}$ truncated to the set $\{\tilde{\boldsymbol{u}} : \tilde{\boldsymbol{\theta}} = G^{-1}(\tilde{\boldsymbol{u}}, \boldsymbol{y}) \text{ exists}\}$. We call the above random sample $\{\tilde{\boldsymbol{\theta}}_1, \tilde{\boldsymbol{\theta}}_2, \ldots\}$ a *generalized fiducial sample* of $\boldsymbol{\theta}$, which can be used to form point estimates and confidence intervals for $\boldsymbol{\theta}$, as with a posterior sample in the Bayesian context. We also call the corresponding density $r(\boldsymbol{\theta})$ the *generalized fiducial density* of $\boldsymbol{\theta}$.

When G is not invertible, Hannig et al. (2016) provide a general definition and a user friendly formula for the generalized fiducial density:

$$r(\boldsymbol{\theta}) = \frac{h(\boldsymbol{y},\boldsymbol{\theta})\,J(\boldsymbol{y},\boldsymbol{\theta})}{\int_{\Theta} h(\boldsymbol{y},\boldsymbol{\theta}')\,J(\boldsymbol{y},\boldsymbol{\theta}')d\boldsymbol{\theta}'}, \tag{3}$$

where $h(\boldsymbol{y},\boldsymbol{\theta})$ is the likelihood and

$$J(\boldsymbol{y},\boldsymbol{\theta}) = D\left\{\nabla_{\boldsymbol{\theta}}\boldsymbol{G}(\boldsymbol{u},\boldsymbol{\theta})|_{\boldsymbol{u}=\boldsymbol{G}^{-1}(\boldsymbol{y},\boldsymbol{\theta})}\right\}$$

with $\nabla_{\boldsymbol{\theta}}\boldsymbol{G}(\boldsymbol{u},\boldsymbol{\theta})$ being a gradient of the data generating algorithm with respect to $\boldsymbol{\theta}$ and $D(\boldsymbol{A}) = \{\det(\boldsymbol{A}^T\boldsymbol{A})\}^{\frac{1}{2}}$. Notice that (3) resembles Bayesian posterior with the role of a prior being played by the volume of the parallelepiped determined by the gradient matrix of the data generating equation $\nabla_{\boldsymbol{\theta}}\boldsymbol{G}(\boldsymbol{u},\boldsymbol{\theta})$.

Formula (3) is applicable to a wide selection of data generating algorithms differentiable with respect to the parameters but it assumes that the model dimension is known. When model selection is involved, the generating function of a certain model T becomes:

$$\boldsymbol{y} = G(\boldsymbol{u}, T, \boldsymbol{\theta}_T).$$

The formula (3) is not directly applicable because T is a discrete parameter. However under identifiability assumptions Hannig et al. (2016) derived the marginal fiducial probability of a model T as

$$r(T) = \frac{n^{-\frac{l(T)}{2}} \int h_T(\boldsymbol{y},\boldsymbol{\theta}_T)\,J_T(\boldsymbol{y},\boldsymbol{\theta}_T)\,d\boldsymbol{\theta}_T}{\sum_{T'\in\mathcal{T}} n^{-\frac{l(T')}{2}} \int h_{T'}(\boldsymbol{y},\boldsymbol{\theta}_{T'})\,J_{T'}(\boldsymbol{y},\boldsymbol{\theta}_{T'})\,d\boldsymbol{\theta}_{T'}}, \tag{4}$$

where $\mathcal{T}$ is the set of all possible models and $l(T)$ is the number of parameters in model T. We note that closed form expressions for (3) and (4) do not always exist so one may need to resort to MCMC techniques.

3. Methodology

3.1. Regression trees and honest regression trees

A decision tree models the function $f(\cdot)$ in (1) by recursively partitioning the feature space, i.e., the space of all $\boldsymbol{X}$'s, into different subsets. These subsets are called *leaves*. Let $\boldsymbol{X}_0$ be any point in the feature space and $L(\boldsymbol{X}_0)$ be the leaf that contains $\boldsymbol{X}_0$. The decision tree estimate $\hat{f}(\boldsymbol{X}_0)$ for $f(\boldsymbol{X}_0)$ is the average of those responses that are in the same leaf as $\boldsymbol{X}_0$:

$$\hat{f}(\boldsymbol{X}) = \frac{1}{|\{i : \boldsymbol{X}_i \in L(\boldsymbol{X})\}|} \sum_{i:\boldsymbol{X}_i\in L(\boldsymbol{X})} Y_i.$$

Naturally one may like a partition that minimizes the loss function: $\sum_{i=1}^{n}\{y_i - \hat{f}(\boldsymbol{X}_i)\}^2$. However, very often in practice a serious drawback is that the number of potential partitions is huge which makes it infeasible to obtain the partition that minimizes the above loss. Therefore, a greedy search algorithm is usually considered.

One criticism of the above decision tree is that the same data are used to grow the tree and make predictions. To ensure good statistical behaviors and as a response to this criticism, *honest decision trees* were proposed (Biau, 2012; Denil et al., 2014). An honest tree is grown using one subsample of the training data and uses a different subsample for making predictions at its leaves. If there are no observations falling to a specific leaf, its prediction will be made by one of its parents. A corresponding honest random forest can be generated by using the same mechanism to generate random forests from decision trees. Wager and Athey (2018) proved that under some regularity conditions, the leaves of an honest tree become small in all dimensions of the feature space when n becomes large. Hence, if the true generating function is Lipschitz continuous, honest trees are unbiased and so are honest random forests. In our context, we need honest trees to get a valid uncertainty quantification.

3.2. Ensemble of honest trees using generalized fiducial inference

The goal is to solve the regression problem (1) using an ensemble of honest binary trees $\{T_j\}_{j=1}^{q}$ and apply GFI to conduct statistical inference.

Suppose there exists a binary tree structured function T_0 such that $f(\boldsymbol{X}) = T_0(\boldsymbol{X})$ for any $\boldsymbol{X}\in\mathbb{R}^P$; we will call any such tree a *true model*. An example of a binary tree structured function is the "AND" function mentioned in Wager et al. (2014): $T(\boldsymbol{X}) = 10\cdot\text{AND}(X_1 > 0.3; X_2 > 0.3; X_3 > 0.3; X_4 > 0.3)$. Notice that a true model is not necessarily unique. For example, if a true model is nested within another binary tree structured function then the larger binary tree would also act as a true model albeit unnecessarily complex.

We assign a generalized fiducial probability to each tree T using (4). To this end, suppose T has $l(T)$ leaves $L_1,\ldots,L_{l(T)}$. Denote the number of observations in the j-th leaf is n_j and hence $n = n_1 + \cdots + n_{l(T)}$. Also denote the response value of the j-th leaf as μ_j. Its least square estimator is the average of all the Y_i's that belong to this leaf: $\hat{\mu}_j = \frac{1}{|L_j|}\sum_{i:\boldsymbol{X}_i\in L_j} Y_i$.

The likelihood for this tree is

$$h_T(\boldsymbol{y}, \boldsymbol{\mu}, \sigma) = (2\pi\sigma^2)^{-\frac{n}{2}} e^{\frac{-\sum_{i=1}^{n}(Y_i - \hat{\mu}_{i(X_i)})^2}{2\sigma^2}},$$

where $i(X_i)$ is the index of the leaf that $\boldsymbol{X}_i$ belongs to; i.e., $\boldsymbol{X}_i \in L_{i(X_i)}$. The Jacobian term is $J(\boldsymbol{\mu}, \sigma^2) = \sigma^{-1}\sqrt{\mathrm{SSE}\prod_{i=1}^{l(T)} n_i}$ with $\mathrm{SSE} = \sum_{i=1}^{n}(Y_i - \hat{\mu}_{i(X_i)})^2$. Integrating out the parameters in (4) we see that the marginal fiducial probability of the tree T is proportional to

$$r(T) \propto R(T) := \frac{\Gamma(\frac{n-l(T)-1}{2}) n^{-\frac{l(T)}{2}}}{\mathrm{SSE}^{\frac{n-l(T)}{2}-1}\pi^{\frac{n-l(T)}{2}}}. \tag{5}$$

3.3. A practical method for generating fiducial samples

This subsection presents a practical method for generating a fiducial sample of honest trees.

Even when n is only of moderate size, the set $\mathcal{T}$ of all possible trees is huge. Therefore we only consider a subset of trees $\mathcal{T}^* \subset \mathcal{T}$. More precisely, $\mathcal{T}^*$ is an honest random forest with an adequate number of trees such that one can assume it contains at least one good model T_0; i.e., a model that approximates the actual data generating process well. We follow the usual practice and generate the trees in $\mathcal{T}^*$ randomly. In particular, each tree samples $\lfloor n/4 \rfloor$ observations without replacement to grow, and uses a different group of $\lfloor n/4 \rfloor$ observations to calculate the averages $\hat{\mu}_j$'s (i.e., make predictions) at the leaves; see Wager and Athey (2018, Procedure 1).

There are three steps involved in generating a fiducial sample of trees directly from the generalized fiducial distribution. The first step is to generate the structure of the tree. The second step is, given a tree, to generate the noise variance. The last step is, given the sampled tree and variance, to generate the leaf values μ_j's. We describe details of each step in turn.

For each tree $T \in \mathcal{T}^*$, we calculate $r(T)$ with

$$r(T) = \frac{R(T)}{\sum_{T' \in \mathcal{T}^*} R(T')}. \tag{6}$$

After a particular tree T is sampled from (6), $\tilde{\sigma}^2$ is sampled using

$$\tilde{\sigma}^2 = \mathrm{SSE}/\chi^2_{\lfloor n/2 \rfloor - l(T)}, \tag{7}$$

where $l(T)$ denotes the number of leaves in T, SSE is computed using the training data of size $\lfloor n/2 \rfloor$ and $\chi^2_{\lfloor n/2 \rfloor - l(T)}$ denotes a sample from chi-square distribution with $\lfloor n/2 \rfloor - l(T)$ degrees of freedom. In the third step we draw without replacement $\lfloor n/4 \rfloor$ observations from the part of the data that was not used to grow T. Denote these drawn observations as $\{\tilde{\boldsymbol{X}}_i, \tilde{Y}_i\}_{i=1}^{\lfloor n/4 \rfloor}$. Then a generalized fiducial tree sample $\tilde{T}$ can be obtained by updating the leaf values of T using

$$\tilde{\mu}_j = \frac{1}{\tilde{n}_j} \sum_{i:\tilde{\boldsymbol{X}}_i \in L_j} \tilde{Y}_i + \frac{\tilde{\sigma}}{\sqrt{\tilde{n}_j}} \tilde{U}_j, \quad j = 1, \ldots, l(T), \tag{8}$$

where $\tilde{U}_l \overset{\text{iid}}{\sim} N(0,1)$, L_j are the leaves and $\tilde{n}_j$ is the number $\tilde{\boldsymbol{X}}_i$ in leaf L_j.

Repeating the above procedure multiple times provides fiducial sample $\{\tilde{\sigma}, \tilde{T}\}$. Statistical inference can then be conducted in a similar fashion as with a posterior sample in the Bayesian context. For any design point $\boldsymbol{X}$, averaging over all the $\tilde{T}(X)$'s will deliver a point estimate for $f(\boldsymbol{X})$. The $\alpha/2$ and $1-\alpha/2$ percentiles of $\tilde{T}(\boldsymbol{X})$ will give a $100(1-\alpha)\%$ confidence interval for $f(\boldsymbol{X})$, while the $\alpha/2$ and $1-\alpha/2$ percentiles of $\tilde{T}(\boldsymbol{X}) + \tilde{\sigma}u$ will provide a prediction interval for the corresponding Y.

We summarize the above procedure in Algorithm 1.

Algorithm 1 A generalized fiducial method for generating honest tree ensemble.

1: Choose $|\mathcal{T}^*|$ and M.
2: Train an honest random forest with $|\mathcal{T}^*|$ honest trees $\{T_j\}_{j=1}^{|\mathcal{T}^*|}$.
3: For each tree T_j, calculate the generalized fiducial probability $r(T_j)$ using (6).
4: **for** $i = 1, \ldots, M$ **do**
5: Draw a $T \in \{T_j\}_{j=1}^{|\mathcal{T}^*|}$ using (6).
6: Draw a $\tilde{\sigma}^2$ from (7).
7: Draw without replacement $\lfloor n/4 \rfloor$ observations from the part of the data that was not used to grow T. Denote these drawn observations as $\{\boldsymbol{X}_i^*, Y_i^*\}_{i=1}^{\lfloor n/4 \rfloor}$.
8: Obtain $\tilde{T}$ by updating its leaf values using (8).
9: **end for**
10: Output the M copies of generalized fiducial sample $\{\tilde{\sigma}^2, \tilde{T}\}$ obtained from above for further inference.

We conducted a set of numerical experiments to help determine a practical choice for the number of trees $|\mathcal{T}^*|$, and found that $|\mathcal{T}^*| = 1,000$ provides very satisfactory results and offers a good compromise between computational and statistical efficiencies. Therefore, in all our numerical work, we set $|\mathcal{T}^*| = 1,000$.

4. Asymptotic properties

If our ensemble contains a tree that is a good approximation to the true function, we are actually dealing with a strong learner like in the case of a Bayesian CART (Denison et al., 1998). Under this situation we show that the fiducial distribution recognizes this and assigns high probability to this tree. The theoretical properties of the proposed method are established under the following conditions.

A1) The generating function $f(x)$ has a binary tree structure. Denote the training data set of T as $\mathcal{D}_T = \{Y_i, \boldsymbol{X}_i\}_{i=1}^{\lfloor n/4 \rfloor}$. We say this binary tree T is a true model if, for any $\boldsymbol{X}$ in the training set $\mathcal{D}_T$, $\mathbb{E}(T(\boldsymbol{X})) = f(\boldsymbol{X})$. Notice that such a binary tree is not unique. We denote the collections of true models as $\mathcal{T}_0$. This assumption defines precisely what true models $\mathcal{T}_0$ are.

A2) Let $\mathcal{T}^*$ be the collection of honest trees in a trained random forests model. We assume that with a high probability $\mathcal{T}^*$ has at least one tree that belongs to $\mathcal{T}_0$: $P(\mathcal{T}^* \cap \mathcal{T}_0 = \emptyset) \to 0$. In other words, we assume that with high probability the collection of honest trees $\mathcal{T}^*$ contains at least one good model.

A3) Meanwhile, we assume that the size of $\mathcal{T}^*$ is not too large: $|\mathcal{T}^*| = o(\sqrt{\frac{\log(n)}{\log\log n}})$. That is, we do not need to consider a large set of honest trees.

A4) Let $\boldsymbol{H}_T = \{h_{ij}\}_{i,j=1}^n$ be the projection matrix of T with

$$h_{ij} = \begin{cases} \frac{1}{n_j} \text{ if } \boldsymbol{X_i} \in L(\boldsymbol{X_j}) \text{ in } T, \\ 0 \text{ otherwise,} \end{cases}$$

and let $\Delta_T = ||\boldsymbol{\mu} - \boldsymbol{H}_T\boldsymbol{\mu}||^2$, where $\boldsymbol{\mu} = E(\boldsymbol{y})$. Assume

$$\lim_{n\to\infty} \min_{T\in\mathcal{T}^*\setminus\mathcal{T}_0} \left\{ \frac{\Delta_T}{l(T)\log n} \right\} = \infty.$$

Heuristically speaking, this assumption means that $\mathcal{T}^*$ contains only trees that are different enough from each other. The assumption ensures that the true models are identifiable.

Theorem 4.1. *Let $\mathcal{T}_{l_0}$ be the trees in $\mathcal{T}^* \cap \mathcal{T}_0$ with number of leaves equal to $l_0 = \min\{l(T), T \in \mathcal{T}^* \cap \mathcal{T}_0\}$. Then under the above assumptions,*

$$\sum_{T\in\mathcal{T}_{l_0}} r(T) \to_p 1.$$

The heuristic interpretation of this theorem is that if $\mathcal{T}^*$ is not too dense and contains a tree close to the true data generating model, the GFD will concentrate on that tree. The proof proceeds by controlling the rate at which $R(T')/R(T) \to_p 0$, for $T' \in \mathcal{T}^* \setminus \mathcal{T}_{l_0}$ and $T \in \mathcal{T}_{l_0}$. This is achieved by using tail estimates on the chi-square distribution separately for the case $T' \notin \mathcal{T}_0$ and the case $T' \in \mathcal{T}_0$ but $l(T') > l_0$. The details can be found in the appendix.

We note that the assumption stated above, such as $f(x)$ following a binary tree structure, may not be realistic in many practical situations. However, we remark that the assumptions are needed for the strong learner result obtained by Theorem 4.1. We believe this theorem is a useful first step in understanding how fiducial inference works on trees. Numerical results reported below suggest that our method works well in practice even when this assumption is violated.

5. Empirical properties

This section illustrates the practical performance of the above proposed method via a sequence of simulation experiments and real data applications. We shall call the proposed method *FiT*, short for Fiducial Trees.

5.1. Simulation experiments

In our simulation experiments three test functions were used:

- Cosine: $3 \cdot \cos(\pi \cdot (X_1 + X_2))$,
- XOR: $5 \cdot \text{XOR}(X_1 > 0.6; X_2 > 0.6) + \text{XOR}(X_3 > 0.6; X_4 > 0.6)$,
- AND: $10 \cdot \text{AND}(X_1 > 0.3; X_2 > 0.3; X_3 > 0.3; X_4 > 0.3)$.

Table 1
Empirical coverage rates for the 90% confidence intervals for $E(Y^*|\boldsymbol{X}^*)$ obtained by the various methods. The results that are closest to the target coverage rate are highlighted in **bold**. The numbers in parentheses are the average widths of the intervals.

Function	n	p	FiT	Bootstrap	Jackknife	BART
Cosine	50	2	**82.7 (4.29)**	34.6 (1.63)	23.6 (1.40)	57.6 (2.33)
Cosine	200	2	87.4 (3.11)	51.0 (2.12)	39.2 (1.05)	**91.1 (1.66)**
XOR	50	50	**73.2 (4.64)**	5.0 (1.72)	3.8 (1.48)	62.9 (4.53)
XOR	200	50	92.6 (2.61)	26.3 (2.96)	32.0 (1.02)	**89.1 (3.53)**
AND	50	500	60.3 (8.19)	3.4 (3.07)	0.7 (2.30)	**66.1 (6.87)**
AND	200	500	**87.2 (4.79)**	35.0 (5.09)	0.0 (1.94)	59.2 (6.10)

Table 2
Similar to Table 1 but for the 95% confidence intervals for $E(Y^*|\boldsymbol{X}^*)$.

Function	n	p	FiT	Bootstrap	Jackknife	BART
Cosine	50	2	**91.6 (5.27)**	41.9 (1.94)	27.9 (1.67)	66.3 (2.78)
Cosine	200	2	93.6 (3.80)	58.8 (2.52)	47.5 (1.26)	**95.6 (1.98)**
XOR	50	50	**83.9 (5.67)**	8.0 (2.05)	6.7 (1.77)	77.3 (5.39)
XOR	200	50	**96.1(3.22)**	38.0 (3.53)	37.4 (1.21)	95.4 (4.21)
AND	50	500	**71.5 (9.91)**	8.3 (3.65)	2.6 (2.74)	71.4 (8.18)
AND	200	500	**90.5 (5.86)**	50.3 (6.06)	0.4 (2.31)	67.9 (7.26)

Table 3
Empirical coverage rates for the 90% and 95% confidence intervals for σ obtained by FiT and BART. The results closest to the target rate are highlighted in **bold**. The numbers in parentheses are the average widths of the intervals.

Function	n	p	FiT 90%	BART 90%	FiT 95%	BART 95
Cosine	50	2	**93.4 (0.68)**	8.5 (0.67)	**96.4 (0.83)**	15.5 (0.80)
Cosine	200	2	96.8 (0.26)	**86.5 (0.20)**	99.1 (0.31)	**92.3 (0.24)**
XOR	50	50	**71.8 (0.67)**	5.4 (1.03)	**81.4 (0.82)**	9.6 (1.24)
XOR	200	50	**90.0 (0.24)**	93.6 (0.62)	**95.4 (0.29)**	96.1 (0.76)
AND	50	500	**24.3 (1.04)**	0.0 (1.59)	**26.1 (1.27)**	0.0 (1.90)
AND	200	500	**1.4 (0.45)**	0.0 (0.78)	**2.0 (0.53)**	0.0 (0.94)

The design points $\boldsymbol{X}_i$'s are iid random uniform $(0, 1)$ variables, and the error standard deviation $\sigma = 1$. We tested different combinations of n and p following experimental configurations used by previous authors (e.g., Chipman et al., 2010; Wager et al., 2014). The number of repetitions for each experimental configuration is 1000.

We applied FiT to the simulated data and calculated the mean coverages of various confidence intervals. We also applied the following three methods to obtain other confidence intervals:

- BART: Bayesian Additive Regression Trees of Chipman et al. (2010) (R package `BART`),
- Bootstrap: the bootstrap method of Mentch and Hooker (2016) (R package `surfin`), and
- Jackknife: the infinite jackknife method of Wager et al. (2014) (R package `grf`).

Tables 1 and 2 report the empirical coverage rates of the 90% and 95% confidence intervals, respectively, produced by these methods for $E(Y^*|\boldsymbol{X}^*)$, where $(\boldsymbol{X}^*, Y^*)$ is a random future data point, i.e., the proportion of times the confidence interval contains its target conditional expected value.

Overall FiT provided good and stable coverages. The performances of Bootstrap and Jackknife are somewhat disappointing. The possible reasons are that in Jackknife the uncertainty of the residual noise was not taken into account, and that Bootstrap is, in general, not asymptotically unbiased, as argued in Wager and Athey (2018). BART sometimes gave better results than FiT. However, for those cases where BART were better, results from FiT were not far behind, but for some other cases, BART's results could be substantially worse than FiT's. Therefore it seems that FiT is the preferred and safe method if one is targeting $E(Y^*|\boldsymbol{X}^*)$.

Next we examine the coverage rates for the noise standard deviation σ. Since Bootstrap and Jackknife do not produce convenient confidence intervals for σ, we only focus on FiT and BART. The results are summarized in Table 3. Overall one can see that FiT is the preferred method, although the performances of all the methods for the test function AND were rather disappointing.

5.2. Real data examples

This subsection reports the coverage rates of the FiT prediction intervals on five real data sets:

Table 4
Empirical coverage rates for the 95% FiT prediction intervals for various real data sets, i.e., the proportion of the validation set covered by its respective prediction intervals. The numbers in parentheses are the averaged widths of the intervals.

Data	Coverage
Air Foil	93.2% (14.2)
Auto Mpg	91.8% (11.5)
CCPP	95.1% (11.5)
Boston House	87.9% (12.4)
CCS	92.8% (30.4)

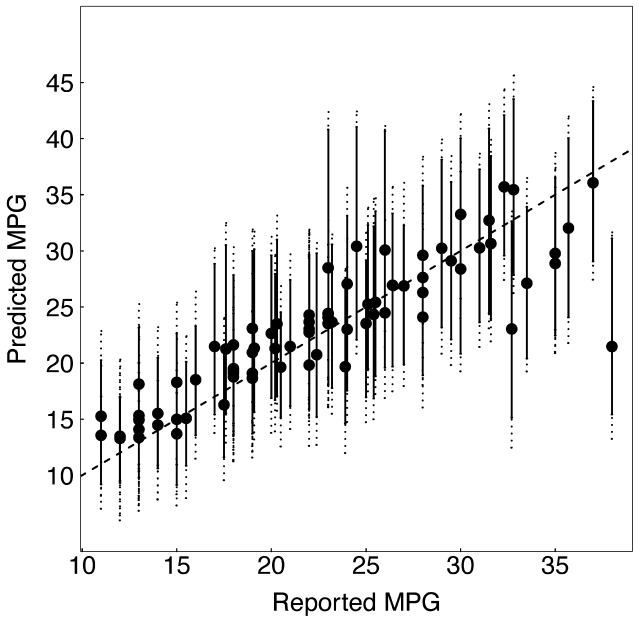

Fig. 1. Predictions (dots) and 95% intervals obtained by the proposed FiT for the *Auto MPG* data set. The solid lines are the 95% confidence intervals for $E(Y^*|\boldsymbol{X}^*)$ while the wider dotted lines show the 95% prediction intervals for Y^*. The FiT prediction intervals cover 98.7% of the validation set (diagonal line).

- *Air Foil*: This is a NASA data set, obtained from a series of aerodynamic and acoustic tests of two and three-dimensional air foil blade sections conducted in an anechoic wind tunnel (Dua and Graff, 2017). Five features were selected to predict the air foil noise. We used 1000 observations as the training data set and 503 observations as test data.
- *Auto Mpg*: This data set contains eight features to predict city-cycle fuel consumption in miles per gallon (Asuncion and Newman, 2007; Dua and Graff, 2017). After discarded samples with missing entries, we split the rest of the observations into a training set of size 314 and a test set of size 78.
- *CCPP*: This data set contains 9568 data points collected from a Combined Cycle Power Plant over six years (2006-2011), when the power plant was set to work with full load (Tüfekci, 2014; Kaya et al., 2012). There are four features aiming to predict the full load electrical power. We split the data into a training set of size 8000 and a test set of size 1568.
- *Boston House*: Originally published by (Harrison Jr and Rubinfeld, 1978), a collection of 506 observations associated with 14 features from the U.S. Census Service are used to predict the median value of owner-occupied homes. We split the data into a training set of size 400 and a test set of size 106.
- *CCS*: In civil engineering, concrete is the most important material (Yeh, 1998). This data set consists of eight features to predict the concrete compressive strength. We split it into a training set of size 750 and a test set of size 280.

For each of the above data sets, we applied FiT to the training data set to construct 95% prediction intervals for the observations in the test data set. We repeated this procedure 100 times by randomly splitting the whole data set into a training data set and a test data set. The empirical coverage rates of these prediction intervals, i.e., the proportion of times the validation value was covered, are reported in Table 4.

For visual inspection, we plotted the FiT intervals for the *Auto MPG* data set in Fig. 1. The FiT plot has two types of intervals: the solid lines are the 95% confidence intervals for $E(Y^*|\boldsymbol{X}^*)$ while the wider dotted lines show the 95% prediction

intervals for Y^*. One can see that the FiT prediction intervals are wide enough to cover 98.7% of the validation set, showing adequate uncertainty quantification.

6. Concluding remarks

In this paper, we applied generalized fiducial inference to ensembles of honest regression trees. In particular, we derived a fiducial probability for each honest tree in honest random forests, which shows how likely the tree contains the true model. A practical procedure was developed to generate fiducial samples of the tree models, variance of errors and predictions. These samples can further be used for point estimation, and constructing confidence intervals and prediction intervals that account for uncertainty in the estimation and prediction process. The proposed method compares favorably with other state-of-the-art methods in simulation experiments and real data analysis.

There are several potential extensions to this method. First, it would be interesting to see how fiducial inference can be used to quantify uncertainty for classification problems. The main challenge will be due to the fact that classification is inherently a discrete parameter problem.

Second, as an alternative to our implementation of FiT that is based on enumerating the fiducial probabilities for all members of a moderately sized $\mathcal{T}^*$, one could consider a very large $\mathcal{T}^*$ and implement an MCMC algorithm proposing a modification of the current tree and accepting it using Metropolis-Hastings ratio based on the unnormalized fiducial probability $R(t)$.

Third, the theoretical results could be extended. The current results give a strong model selection consistency and therefore require very strong assumptions; the data was generating using a tree. It would interesting to see if one can use the mathematical tools in Castillo and Rousseau (2015) to relax this assumption and still get Bernstein - von Mises theorem even if the true function is no longer a tree and the number of candidate trees $\mathcal{T}^*$ is large.

Acknowledgements

The authors are most grateful to the reviewers and the associate editor for their most constructive and useful comments. The work of Hannig was partially supported by the National Science Foundation under grants IIS-1633074, DMS-1916115 and DMS-2113404. The work of Lee was partially supported by the National Science Foundation under grants CCF-1934568, DMS-1811405, DMS-1811661, DMS-1916125 and DMS-2113605.

Appendix A. Technical details

This appendix provides the proof for Theorem 4.1.

WLOG, assume $\sigma^2 = 1$ and fix $T \in \mathcal{T}_{l_0}$. We first prove that

$$\max_{T' \notin \mathcal{T}_{l_0}, T' \in \mathcal{T}^*} R(T')/R(T) \to_p 0.$$

Rewrite

$$R(T')/R(T) = exp\{-D_1 - D_2\},$$

where

$$D_1 = \frac{n - l(T') - 1}{2} \log \frac{\mathrm{SSE}_{T'}}{\mathrm{SSE}_T}$$

and

$$D_2 = \log \frac{\Gamma(\frac{n-l_0}{2})}{\Gamma(\frac{n-l(T')}{2})} + \frac{l_0 - l(T')}{2} \log \pi + \frac{l_0 - l(T')}{2} \log \mathrm{SSE}_T + \frac{l(T') - l_0}{2} \log(n).$$

Case 1: $T' \notin \mathcal{T}_0$.

Now calculate

$$\mathrm{SSE}_{T'} - \mathrm{SSE}_T = \Delta_{T'} + 2\boldsymbol{\mu}'(\boldsymbol{I} - \boldsymbol{H}_{T'})\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}'(\boldsymbol{H}_{T'} - \boldsymbol{H}_T)\boldsymbol{\varepsilon}. \tag{A.1}$$

Let $c_{l(T)} = l(T) \log\log n$, consider the second term in equation (A.1) and denote

$$Z_{T'} = \boldsymbol{\mu}'(\boldsymbol{I} - \boldsymbol{H}_{T'})\boldsymbol{\varepsilon}/\sqrt{\Delta_{T'}},$$

then

$$\mu^T(\boldsymbol{I} - \boldsymbol{H}_{T'})\boldsymbol{\varepsilon} = \sqrt{\Delta_{T'}} Z_{T'}$$

and $Z_{T'} \sim N(0, I_n)$ since $var(Z_{T'}) = 1$. Furthermore,

$$\begin{aligned} P(\max_{T' \in \mathcal{T}^*} |Z_{T'}/\sqrt{c_{l(T')}}| > 1) &\le |\mathcal{T}| \max_{T' \in \mathcal{T}^*} P(Z_{T'}^2 > c_{l(T')}) \\ &= |\mathcal{T}^*| \max_{T' \in \mathcal{T}^*} P(\chi_1^2 > c_{l(T')}) \\ &\le |\mathcal{T}^*| \max_{T' \in \mathcal{T}^*} (c_{l(T')} e^{1-c_{l(T')}})^{1/2} \longrightarrow 0 \qquad \text{as} \quad n \longrightarrow \infty. \end{aligned}$$

Therefore,

$$P(|\boldsymbol{\mu}'(\boldsymbol{I} - \boldsymbol{H}_{T'})\boldsymbol{\varepsilon}| > \sqrt{\Delta_{T'} c_{l(T')}}) \longrightarrow 0 \qquad \text{as} \quad n \longrightarrow \infty.$$

Consider the third term in equation (A.1):
Notice that $\boldsymbol{\varepsilon}' \boldsymbol{H}_T \boldsymbol{\varepsilon} = \sum_{i=1}^{l(T)} n_i \bar{\epsilon}_i^2 \sim \chi_{l(T)}^2$. Thus,

$$\begin{aligned} P(\max_{T \in \mathcal{T}^*} \boldsymbol{\varepsilon}' \boldsymbol{H}_T \boldsymbol{\varepsilon}/c_{l(T)} > 1) &\le |\mathcal{T}^*| \max_{T \in \mathcal{T}^*} P(\boldsymbol{\varepsilon}^T \boldsymbol{H}_{l(T)} \boldsymbol{\varepsilon} > c_{l(T)}) \\ &= |\mathcal{T}^*| \max_{T \in \mathcal{T}^*} P(\chi_{l(T)}^2 > c_{l(T)}) \\ &\le |\mathcal{T}^*| \max_{T \in \mathcal{T}^*} (\frac{c_{l(T)}}{l(T)} e^{1-\frac{c_{l(T)}}{l(T)}})^{l(T)/2} \\ &= |\mathcal{T}^*| \max_{T \in \mathcal{T}^*} (\frac{e \log\log n}{\log n})^{l(T)/2} \longrightarrow 0 \qquad \text{as} \quad n \longrightarrow \infty. \end{aligned}$$

Therefore, $P(\boldsymbol{\varepsilon}^T \boldsymbol{H}_T \boldsymbol{\varepsilon} > c_{l(T)}) \longrightarrow 0$, and $P(\boldsymbol{\varepsilon}^T \boldsymbol{H}_{T'} \boldsymbol{\varepsilon} > c_{l(T')}) \longrightarrow 0$ as $n \longrightarrow \infty$. Thus, we have $P(\mathrm{SSE}_{T'} - \mathrm{SSE}_T < 0.5\Delta_{T'}) \longrightarrow 0$ as $n \longrightarrow \infty$.

In addition,

$$P(\chi_{n-L}^2 < \frac{n}{4}) \le P(\chi_{n-L}^2 < \frac{n-L}{2}) \le (\frac{\sqrt{e}}{2})^{\frac{n-L}{2}} \longrightarrow 0 \qquad \text{as} \quad n \longrightarrow \infty,$$

which means

$$P(\min_{T' \in \mathcal{T}^*} \chi_{n-l(T')}^2 < \frac{n}{4}) \longrightarrow 0 \qquad \text{as} \quad n \longrightarrow \infty.$$

Thus,

$$\begin{aligned} D_1 &= \frac{n - l(T') - 1}{2} \log(\frac{\mathrm{SSE}_{T'}}{\mathrm{SSE}_T}) \\ &= -\frac{n - l(T') - 1}{2} \log(\frac{\mathrm{SSE}_T}{\mathrm{SSE}_{T'}}) \\ &= -\frac{n - l(T') - 1}{2} \log(1 + \frac{\mathrm{SSE}_T - \mathrm{SSE}_{T'}}{\mathrm{SSE}_{T'}}) \\ &\ge \frac{n - l(T') - 1}{2} \frac{\mathrm{SSE}_{T'} - \mathrm{SSE}_T}{\mathrm{SSE}_{T'}} \\ &= \Omega_p(\Delta_{T'}). \end{aligned}$$

Moreover, $D_2 = \Omega_p(-l(T')\log(n))$. Therefore, $D_1 + D_2 = \Omega_p(\log n)$.

Case 2: $T' \in \mathcal{T}_0$ and $l(T') > l_0$.

Recall $T \in \mathcal{T}_{l_0}$ is fixed. First notice that $\mathrm{SSE}_T - \mathrm{SSE}_{T'} = \chi_{l(T')-l_0}^2(T')$, where $\chi_{l(T')-l_0}^2(T')$ is a chi-square random variable depending on T' with degrees of freedom $l(T') - l_0$.

$$\begin{aligned} P(\max_{T' \in \mathcal{T}_0, l(T') > l_0} \frac{\chi_{l(T')-l_0}^2(T')}{(l(T') - l_0)\log\log n} \ge 1) &\le |\mathcal{T}^*| \max_{T' \in \mathcal{T}_0, l(T') > l_0} (\log\log n e^{1-\log\log n})^{\frac{l(T')-l_0}{2}} \\ &= |\mathcal{T}^*| (\frac{e \log\log n}{\log n})^{\frac{1}{2}} \to 0. \end{aligned}$$

It implies that

$$\chi_{l(T')-l_0}^2 = O_p(c_{l(T')-l_0}).$$

Therefore,

$$\begin{aligned}\frac{n-l(T')-1}{2}\log\frac{\mathrm{SSE}_{T'}}{\mathrm{SSE}_T} &= -\frac{n-l(T')-1}{2}\log(1+\frac{\chi^2_{l(T')-l_0}(T')}{\chi^2_{n-l(T')}})\\ &\geq -\frac{n-l(T')-1}{2}(\frac{\chi^2_{l(T')-l_0}(T')}{\chi^2_{n-l(T')}})\\ &= \Omega_p(-c_{l(T')-l_0}),\end{aligned}$$

uniformly over $\{T' : T' \in \mathcal{T}_0,\ l(T') > l_0\}$. Thus, we show that

$$D_1 = \Omega_p(-\frac{l(T')}{2}\log\log n).$$

Meanwhile, the calculation of D_2 is similar to Case 1, $D_2 = \Omega_p((l(T')-l_0)\log(n))$, so we have $D_1 + D_2 = \Omega_p(\log n)$.

Combining Case 1 and Case 2, we have:

$$\max_{T'\notin\mathcal{T}_{l_0}, T'\in\mathcal{T}^*} R(T')/R(T) = O_p(1/n).$$

Furthermore,

$$\sum_{T'\notin\mathcal{T}_{l_0}, T'\in\mathcal{T}^*} R(T')/R(T) \leq |\mathcal{T}^*| \max_{T'\notin\mathcal{T}_{l_0}, T'\in\mathcal{T}^*} R(T')/R(T) \leq \frac{|\mathcal{T}^*|}{n} \to_p 0.$$

Equivalently,

$$\sum_{T\in\mathcal{T}_{l_0}} r(T) \to_p 1.$$

References

Asuncion, A., Newman, D., 2007. UCI machine learning repository. http://www.ics.uci.edu/~mlearn/MLRepository.html.

Athey, S., Tibshirani, J., Wager, S., 2019. Generalized random forests. Ann. Stat. 47, 1148–1178.

Biau, G., 2012. Analysis of a random forests model. J. Mach. Learn. Res. 13, 1063–1095.

Breiman, L., 1996. Bagging predictors. Mach. Learn. 24, 123–140.

Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32.

Castillo, I., Rousseau, J., 2015. A Bernstein - von Mises theorem for smooth functionals in semiparametric models. Ann. Stat. 43, 2353–2383.

Chipman, H.A., George, E.I., McCulloch, R.E., 2007. Bayesian ensemble learning. In: Advances in Neural Information Processing Systems, pp. 265–272.

Chipman, H.A., George, E.I., McCulloch, R.E., 2010. BART: Bayesian additive regression trees. Ann. Appl. Stat. 4, 266–298.

Dempster, A.P., 2008. The Dempster–Shafer calculus for statisticians. Int. J. Approx. Reason. 48, 365–377.

Denil, M., Matheson, D., De Freitas, N., 2014. Narrowing the gap: random forests in theory and in practice. In: International Conference on Machine Learning, pp. 665–673.

Denison, D.G., Mallick, B.K., Smith, A.F., 1998. A Bayesian CART algorithm. Biometrika 85, 363–377.

Dua, D., Graff, C., 2017. UCI machine learning repository. http://archive.ics.uci.edu/ml.

Efron, B., 2014. Estimation and accuracy after model selection. J. Am. Stat. Assoc. 109, 991–1007.

Fisher, R.A., 1930. Inverse Probability. Mathematical Proceedings of the Cambridge Philosophical Society, vol. 26. Cambridge University Press, pp. 528–535.

Gao, Q., Lai, R.C.S., Lee, T.C.M., Li, Y., 2020. Uncertainty quantification for high-dimensional sparse nonparametric additive models. Technometrics 62, 513–524.

Hannig, J., Lee, T.C.M., 2009. Generalized fiducial inference for wavelet regression. Biometrika 96, 847–860.

Hannig, J., Iyer, H., Lai, R.C.S., Lee, T.C.M., 2016. Generalized fiducial inference: a review and new results. J. Am. Stat. Assoc. 111, 1346–1361.

Harrison Jr, D., Rubinfeld, D.L., 1978. Hedonic housing prices and the demand for clean air. J. Environ. Econ. Manag. 5, 81–102.

Kaya, H., Tüfekci, P., Gürgen, F.S., 2012. Local and global learning methods for predicting power of a combined gas & steam turbine. In: Proceedings of the International Conference on Emerging Trends in Computer and Electronics Engineering ICETCEE, pp. 13–18.

Lai, R.C., Hannig, J., Lee, T.C.M., 2015. Generalized fiducial inference for ultrahigh-dimensional regression. J. Am. Stat. Assoc. 110, 760–772.

Martin, R., Liu, C., 2015a. Conditional inferential models: combining information for prior-free probabilistic inference. J. R. Stat. Soc. Ser. B 77, 195–217.

Martin, R., Liu, C., 2015b. Marginal inferential models: prior-free probabilistic inference on interest parameters. J. Am. Stat. Assoc. 110, 1621–1631.

Martin, R., Zhang, J., Liu, C., 2010. Dempster–Shafer theory and statistical inference with weak beliefs. Stat. Sci. 25, 72–87.

Mendes-Moreira, J., Soares, C., Jorge, A.M., Sousa, J.F.D., 2012. Ensemble approaches for regression: a survey. ACM Comput. Surv. 45, 10.

Mentch, L., Hooker, G., 2016. Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. J. Mach. Learn. Res. 17, 841–881.

Perrone, M.P., Cooper, L.N., 1992. When networks disagree: Ensemble methods for hybrid neural networks. Technical Report. Brown Univ Providence Ri Inst for Brain and Neural Systems.

Rockova, V., 2020. On semi-parametric inference for BART. In: International Conference on Machine Learning, vol. 119, pp. 8137–8146.

Tüfekci, P., 2014. Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. Int. J. Electr. Power Energy Syst. 60, 126–140.

Wager, S., Athey, S., 2018. Estimation and inference of heterogeneous treatment effects using random forests. J. Am. Stat. Assoc. 113, 1228–1242.

Wager, S., Hastie, T., Efron, B., 2014. Confidence intervals for random forests: the jackknife and the infinitesimal jackknife. J. Mach. Learn. Res. 15, 1625–1651.

Wang, H., Fan, W., Yu, P.S., Han, J., 2003. Mining concept-drifting data streams using ensemble classifiers. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 226–235.
Weerahandi, S., 1995. Generalized confidence intervals. In: Exact Statistical Methods for Data Analysis. Springer, pp. 143–168.
Weerahandi, S., 2013. Exact Statistical Methods for Data Analysis. Springer Science & Business Media.
Wu, S., Hannig, J., Lee, T.C., 2021. Uncertainty quantification for principal component regression. Electron. J. Stat. 15, 2157–2178.
Wu, Y., Tjelmeland, H., West, M., 2007. Bayesian CART: prior specification and posterior simulation. J. Comput. Graph. Stat. 16, 44–66.
Xie, M.-g., Singh, K., 2013. Confidence distribution, the frequentist distribution estimator of a parameter: a review. Int. Stat. Rev. 81, 3–39.
Xie, M.-g., Singh, K., Strawderman, W.E., 2011. Confidence distributions and a unifying framework for meta-analysis. J. Am. Stat. Assoc. 106, 320–333.
Yeh, I.-C., 1998. Modeling of strength of high-performance concrete using artificial neural networks. Cem. Concr. Res. 28, 1797–1808.