# Extrapolated Random Tree for Regression

**Yuchao Cai** [*] [1] **Yuheng Ma** [*] [1] **Yiwei Dong** [1] **Hanfang Yang** [1] [2]

## Abstract

In this paper, we propose a novel tree-based algorithm named *Extrapolated Random Tree for Regression* (ERTR) that adapts to arbitrary smoothness of the regression function while maintaining the interpretability of the tree. We first put forward the *homothetic random tree for regression* (HRTR) that converges to the target function as the homothetic ratio approaches zero. Then ERTR uses a linear regression model to extrapolate HRTR estimations with different ratios to the ratio zero. From the theoretical perspective, we for the first time establish the optimal convergence rates for ERTR when the target function resides in the general Hölder space $C^{k,\alpha}$ for $k \in \mathbb{N}$, whereas the lower bound of the convergence rate of the random tree for regression (RTR) is strictly slower than ERTR in the space $C^{k,\alpha}$ for $k \geq 1$. This shows that ERTR outperforms RTR for the target function with high-order smoothness due to the extrapolation. In the experiments, we compare ERTR with state-of-the-art tree algorithms on real datasets to show the superior performance of our model. Moreover, promising improvements are brought by using the extrapolated trees as base learners in the extension of ERTR to ensemble methods.

## 1. Introduction

Ensemble of trees methods (Breiman, 2001; Friedman, 2002; Chen & Guestrin, 2016) are powerful and well-known methods that are successfully adopted for regression problems in many applications and machine learning competitions (Yu et al., 2021; Basak et al., 2022; Künzel et al., 2022; Amini et al., 2022). In fact, the success of ensemble methods stems from the advantages inherited from the tree methods. That is, the wide applicability of tree methods due to weak distributional assumption on data, the computational efficiency, as well as the interpretability after we build the model.

Despite their empirical success, the merit of regression trees is also addressed from theoretical perspectives. For instance, Gao & Zhou (2020) considered the convergence rate for random trees which attains a mini-max optimal rate for Lipschitz functions. Mourtada et al. (2020) derived similar results for Mondrian trees. As another line of work, bayesian additive regression trees (Chipman et al., 2010) impose prior distribution on the parameters of the partitioning process and are shown to achieve optimal convergence rate for Lipschitz functions (Ročková & Saha, 2019; Ročková & van der Pas, 2020). These works imply that standard regression trees, which are based on piece-wise constant functions, can only adapt to functions that are at most Lipschitz smooth. Therefore, they may be difficult to adapt to the high-order smoothness of the target function.

To overcome this issue, researchers seek to improve regression trees. Most works focus on employing smooth estimations, which are achieved by attributing data by soft assignment instead of hard assignment. Several closely related works, including Suárez & Lutsko (1999); Da Rosa et al. (2008); Irsoy et al. (2012); Frosst & Hinton (2017); Alkhoury et al. (2020), assign each data point to all leaves with a certain class membership, and the final prediction is a smooth combination of the prediction at each node. Despite their contribution from the methodology perspective, none of the works above managed to explain how their smooth/flexible tree structures facilitate adaption to high-order smoothness theoretically. A recent work (Linero & Yang, 2018) on Bayesian additive regression tree utilized the soft tree ensembles and derived mini-max rate for arbitrary smoothness. However, its Bayesian nature yields a heavy computational burden. Besides, the choice of priors is strongly restricted to achieve theoretical optimality and computation efficiency.

Under such background, borrowing the extrapolation techniques from Brezinski & Zaglia (2013); Okuno & Shimodaira (2020), we propose a novel tree-based algorithm called *Extrapolated Random Tree for Regression* (ERTR) that is adaptive to high order smoothness of the target func-

[*]Equal contribution [1]School of Statistics, Renmin University of China [2]Center for Applied Statistics, School of Statistics, Renmin University of China. Correspondence to: Hanfang Yang <hyang@ruc.edu.cn>.

*Proceedings of the 40th International Conference on Machine Learning*, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

tion while maintaining the interpretability. First, we introduce the random tree partition that divides the input space recursively by cutting the longest edge of the cell. Then we apply the homothetic transformation to shrink the cell by a scale factor that is the same in all directions, called homothetic ratio, according to a center point. Based on the transformations, we put forward the *homothetic random tree for regression* (HRTR) that represents the average of the response variable on these homothetic cells. Then, by using Taylor's expansion concerning the homothetic ratio, we show that HRTR converges to the target function as the ratio approaches zero. Finally, we propose *extrapolated random tree for regression* (ERTR) that extrapolates HRTR estimations with a series of pre-specified ratios to the ratio zero. More specifically, we train a linear regression model regarding the ratios as covariates and the corresponding HRTR estimations as responses to estimate the value at the ratio zero. Two advantages of ERTR are listed as follows. First, the extrapolation procedure eradicates the dominant terms in approximation error, which leads to the adaptivity to the target function with high-order smoothness. Moreover, ERTR inherits the interpretability from the tree model and linear regression model, which can tell us how important different input data are in prediction. Our contributions are summarized as follows.

(i) We propose a tree-based algorithm called *extrapolated random tree for regression* (ERTR) that is adaptive to the high-order smoothness of the target function while preserving the interpretability of the tree.

(ii) From the learning theory perspective, we establish optimal convergence rate $\mathcal{O}(n^{-\frac{2(k+\alpha)}{2(k+\alpha)+d}})$ for ERTR when the target function resides in the general Hölder space $C^{k,\alpha}, k \in \mathbb{N}$, whereas the lower bound of the convergence rates of the random tree for regression (RTR) is mere of the order $O(n^{-\frac{2}{2+d}})$ in the space $C^{k,\alpha}$, $k \geq 1$. This shows that ERTR outperforms RTR for the target function with high-order smoothness due to the extrapolation.

(iii) From the experimental perspective, we first conduct synthetic experiments to illustrate the power of extrapolation, which coincides with the established theoretical results. Moreover, we demonstrate that ERTR outperforms other state-of-the-art tree methods on almost real-world data sets. Furthermore, we extend ERTR to ensemble methods including random forest and gradient boosting. Promising improvements are brought by using the extrapolated trees as base learners.

## 2. Methodology

We dedicate this section to the methodology of ERTR. In Section 2.1, we first present preliminaries related to regression problems. Then we introduce the standard random tree for regression in Section 2.2. Next, in Section 2.3, we propose the homothetic random tree for regression. Finally, we apply the extrapolation method to these estimations to obtain the ERTR algorithm in Section 2.4.

### 2.1. Preliminaries

Regression is to predict the value of an unobserved output variable $Y$ based on the observed input variable $X$, based on a dataset $D := \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ consisting of i.i.d. observations drawn from an unknown probability measure $\mathrm{P}$ on $\mathcal{X} \times \mathcal{Y}$. Throughout this paper, we assume that $\mathcal{X} = [0,1]^d \subset \mathbb{R}^d$ and that $\mathcal{Y} \subset [-M, M]$ are compact and non-empty.

Recall that for $1 \leq p < \infty$, the $L_p$-norm of $x = (x_1, \dots, x_d)$ is defined by $\|x\|_p := (|x_1|^p + \cdots + |x_d|^p)^{1/p}$, and the $L_\infty$-norm is defined by $\|x\|_\infty := \max_{i=1,\dots,d} |x_i|$. Throughout this paper, we use the notation $a_n \lesssim b_n$ and $a_n \gtrsim b_n$ to denote that there exist positive constant $c$ and $c'$ such that $a_n \leq c b_n$ and $a_n \geq c' b_n$, for all $n \in \mathbb{N}$. In addition, we denote $a_n \asymp b_n$ if there hold $a_n \lesssim b_n$ and $b_n \lesssim a_n$. Moreover, for any $x \in \mathbb{R}$, let $\lfloor x \rfloor$ denote the largest integer less than or equal to $x$. In the sequel, the following multi-index notations are used frequently. For any vector $x = (x_i)_{i=1}^d \in \mathbb{R}^d$, we write $\lfloor x \rfloor := (\lfloor x_i \rfloor)_{i=1}^d$, $x^{-1} := (x_i^{-1})_{i=1}^d$, $\log(x) := (\log x_i)_{i=1}^d$, $\overline{x} = \max_{i=1,\dots,d} x_i$, and $\underline{x} = \min_{i=1,\dots,d} x_i$. In addition, for any matrix $R$, let $R^\top$ denote the transpose of the matrix $R$. Besides, for any set $A \subset \mathbb{R}^d$, the dimameter of $A$ is defined by $\mathrm{diam}(A) := \sup_{x,x' \in A} \|x - x'\|_2$.

It is legitimate to consider the least square loss $L : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$ defined by $L(x, y, f(x)) := (y - f(x))^2$ for our target of regression. Then, for a measurable decision function $f : \mathcal{X} \to \mathcal{Y}$, the risk is defined by $\mathcal{R}_{L,\mathrm{P}}(f) := \int_{\mathcal{X} \times \mathcal{Y}} L(x, y, f(x)) \, d\mathrm{P}(x, y)$ and the empirical risk is defined by $\mathcal{R}_{L,\mathrm{D}}(f) := \frac{1}{n} \sum_{i=1}^n L(X_i, Y_i, f(X_i))$. The Bayes risk, which is the smallest possible risk with respect to $\mathrm{P}$ and $L$, is given by $\mathcal{R}^*_{L,\mathrm{P}} := \inf\{\mathcal{R}_{L,\mathrm{P}}(f) | f : \mathcal{X} \to \mathcal{Y} \text{ measurable}\}$. The function that achieves the Bayes risk is called *Bayes function*, namely, $f^*_{L,\mathrm{P}}(x) := \mathbb{E}(Y | X = x)$.

### 2.2. Random Tree for Regression

Breiman's original algorithm (Li et al., 1984) is difficult to analyze as a result of the complexity of the partitioning procedure. Nevertheless, grown independently of the samples, random trees are amenable to theoretical analysis. In this paper, we investigate the random tree partitions with the mid-point splitting rule on the max edges suggested by Gao & Zhou (2020). To be specific, let $A_0^1 := [0,1]^d$ be the initial rectangular cell and $\pi_0 := \{A_0^1\}$ be the initialized cell partition. In addition, let $p \in \mathbb{N}$ represent the depth of the tree, which is fixed beforehand by the user, and possibly depending on $n$.

In the first step, we choose one of the coordinates $X = (X_1, \ldots, X_d)$ with the $\ell$-th feature $X_\ell$ having a probability $1/d$ of being selected and then split $A_0^1$ into two rectangular cells along the midpoint of the chosen side. In this way, we get a partition with two rectangular cells denoted as $\pi_1 := \{A_1^1, A_1^2\}$. Suppose after $i-1$ steps of the recursion, $1 \leq i \leq p$, we have obtained a partition $\pi_{i-1}$ of $\mathcal{X}$ with $2^{i-1}$ rectangular cells. In the $i$-th step, further partitioning of the region is defined as follows:

(i) For each rectangular cell $A_{i-1}^j$, $1 \leq j \leq 2^{i-1}$. Let $e_{i-1,j}^\ell$ denote the length of edge of $A_{i-1}^j$ along the $\ell$-th coordinate for $1 \leq l \leq d$ and $\overline{e}_{i-1,j} = \max_{1\leq \ell \leq d} e_{i-1,j}^\ell$. Then a coordinate of $X = (X_1, \ldots, X_d)$, namely $Z_{i,j}$ is uniformly selected among the coordinates with maximal side length,

$$\mathrm{P}(Z_{i,j} = \ell) = \frac{\mathbf{1}(e_{i-1,j}^\ell = \overline{e}_{i-1,j})}{\sum_{\ell=1}^d \mathbf{1}(e_{i-1,j}^\ell = \overline{e}_{i-1,j})}. \qquad (1)$$

(ii) For each rectangular cell $A_{i-1}^j$, $1 \leq j \leq 2^{i-1}$, once the coordinate is selected, the split is at the midpoint of the chosen side. Then, each cell $A_{i-1}^j$ is divided into two new ones, namely $A_i^{2j-1}$ and $A_i^{2j}$. We denote the set of all these cells $\{A_i^j, 1 \leq j \leq 2^i\}$ by $\pi_i$.

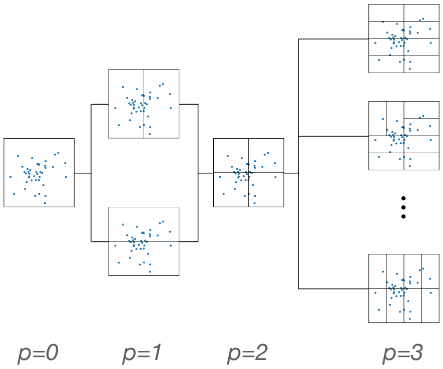


*Figure 1.* Illustration of random tree partition.

After $p$ recursive steps, we obtain the partition of $[0,1]^d$, i.e.

$$\pi_p := \{A_p^j\}_{j\in\mathcal{I}_p} := \{A_p^j, 1 \leq j \leq 2^p\}. \qquad (2)$$

We call it a *random tree partition* with the maximal edge (max-edge) splitting rule of depth $p$. The complete process is presented in Algorithm 1 with an illustration in Figure 1.

For any $x \in [0,1]^d$, there exists $j \in \mathcal{I}_p$ such that $x \in A_p^j$. Then we denote the cell containing $x$ as $A(x) := A_p^j$. With these preparations, we introduce the *random tree for regression* (RTR) based on the random tree partition $\pi_p$.

**Algorithm 1** Random Tree Partition

```
Input: Depth of the random tree $p$;
       Initial partition $\pi_0 = \{A_0^1 := [0,1]^d\}$.
for $i = 1$ to $p$ do
  for $j = 1$ to $2^{i-1}$ do
    For rectangular cell $A_{i-1}^j$, randomly choose one coordinate $Z_{i,j}$ among the longest edges as in (1);
    Divide the cell $A_{i-1}^j$ into two subregions, that is, $A_{i-1}^j = A_i^{2j-1} \cup A_i^{2j}$, along the midpoint of the dimension $Z_{i,j}$;
  end for
  Get $\pi_i = \{A_i^j, 1 \leq j \leq 2^i\}$.
end for
Output: Partition $\pi_p$.
```

**Definition 2.1.** Let Q be a probability measure and $\pi_p := \{A_p^j\}_{j\in\mathcal{I}_p}$ be a random tree partition with depth $p$ as in (2). Then, the *random tree for regression* (RTR), namely, $f_\mathrm{Q} : [0,1]^d \to \mathbb{R}$ is defined by

$$f_\mathrm{Q}(x) := \frac{\int_{A(x)\times\mathcal{Y}} Y \, d\mathrm{Q}}{\int_{A(x)\times\mathcal{Y}} d\mathrm{Q}}. \qquad (3)$$

Taking $Q := \mathrm{P}$, we get the population RTR as

$$f_\mathrm{P}(x) := \frac{\int_{A(x)} f_{L,\mathrm{P}}^*(x') \, d\mathrm{P}_X(x')}{\int_{A(x)} d\mathrm{P}_X(x')}. \qquad (4)$$

From the above definition, $f_\mathrm{P}(x)$ represents the average of the Bayes function $f_{L,\mathrm{P}}^*(x')$ on $A(x)$. Since P is inaccessible, we need to estimate it from the data. When Q is taken as the empirical measure $\mathrm{D} := \frac{1}{n}\sum_{i=1}^n \delta_{(x_i,y_i)}$ given the data set $D = \{(X_1,Y_1), \ldots, (X_n,Y_n)\}$, the empirical RTR is expressed as

$$f_\mathrm{D}(x) := \frac{\sum_{i=1}^n \mathbf{1}\{X_i \in A(x)\} Y_i}{\sum_{i=1}^n \mathbf{1}\{X_i \in A(x)\}}. \qquad (5)$$

The denominator on the right-hand side of (5) counts the number of observations falling in $A(x)$ and thus $f_\mathrm{D}(x)$ is the average of the responses in the cell $A(x)$.

### 2.3. Homothetic Random Tree for Regression

In this section, we put forward the homothetic random tree for regression and present Taylor's expansion concerning the homothetic ratio.

In mathematics, a *homothetic transformation* of an affine space is determined by a point $x$ called its center and a nonzero number $r$ called its *ratio*, which sends point $z$ to a point $z'$ by the rule $z' = x + r(z - x)$. Recall that for $x \in [0,1]^d$, we use $A(x)$ to represent the cell of the partition $\pi_p$ containing $x$. Then for any $0 \leq r \leq 1$, we define the homothetic transformation $T_{x,r} : A(x) \to \mathbb{R}^d$ with the center $x$ and the ratio $0 \leq r \leq 1$ as

$$T_{x,r}(z) := x + r(z - x), \quad z \in A(x). \tag{6}$$

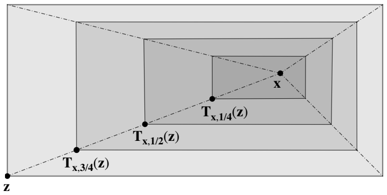


*Figure 2.* Illustration of homothetic transformations. The back dots between $x$ and $z$ denote the images of homothetic transformation $T_{x,1/4}(z)$, $T_{x,1/2}(z)$, and $T_{x,3/4}(z)$.

Then we define $A_r(x)$ as the image of $T_{x,r}$, that is,

$$A_r(x) := \{T_{x,r}(z) : z \in A(x)\}. \tag{7}$$

$A_r(x)$ defines a collection of rectangles that is parametrized by $r$. For $r = 1$, $A_1(x)$ remains $A(x)$. As $r$ approaches 0, $A_r(x)$ gradually shrinks and degenerates to $x$ in the end. With these preparations, we can define a broader class of estimations as below.

**Definition 2.2.** Let Q be a probability measure and $\pi_p := \{A_p^j\}_{j \in \mathcal{I}_p}$ be a random tree partition with depth $p$ as in (2). Moreover, let $A_r(x)$ be defined by (7), then the *homothetic random tree for regression* (HRTR) with the ratio $r$, namely, $f_{\mathrm{Q},r} : [0,1]^d \to [0,\infty)$ is defined by

$$f_{\mathrm{Q},r}(x) := \frac{\int_{A_r(x)\times\mathcal{Y}} Y \, dQ}{\int_{A_r(x)\times\mathcal{Y}} \, dQ}. \tag{8}$$

Taking $Q := \mathrm{P}$, we get the population HRTR as

$$f_{\mathrm{P},r}(x) := \frac{\int_{A_r(x)} f^*_{L,\mathrm{P}}(x') \, d\mathrm{P}_X(x')}{\int_{A_r(x)} \, d\mathrm{P}_X(x')}. \tag{9}$$

Clearly, $f_{\mathrm{P},r}(x)$ is the average of $f^*_{L,\mathrm{P}}(x')$ on $A_r(x)$. Similar to (5), we can estimate HRTR from the data by

$$f_{\mathrm{D},r}(x) := \frac{\sum_{i=1}^n \mathbf{1}\{X_i \in A_r(x)\} Y_i}{\sum_{i=1}^n \mathbf{1}\{X_i \in A_r(x)\}}. \tag{10}$$

To make HRTR converge to $f(x)$ as $r$ approaches 0, we need the following definition of Hölder continuity space.

**Definition 2.3.** Let $k \in \mathbb{N}$, $\alpha \in (0,1]$. We say that a function $f : \mathcal{X} \to \mathbb{R}$ is $(k,\alpha)$-Hölder continuous, if there exists a finite constant $c_L > 0$ such that $\|\nabla^\ell f\| \leq c_L$ for all $\ell \in \{1, \ldots, k\}$ and $\|\nabla^k f(x) - \nabla^k f(x')\| \leq c_L \|x - x'\|^\alpha$ for all $x, x' \in \mathcal{X}$. The set of such functions is denoted by $C^{k,\alpha}(\mathcal{X})$.

We remark that $k$ decides the order of smoothness for $f \in C^{k,\alpha}$, and larger $k$ indicates that $f$ enjoys a higher order of smoothness. For the special case $k = 0$, the function space $C^{0,\alpha}$ coincides with the commonly used $\alpha$-Hölder continuous function space $C^\alpha$.

With these preparations, the next proposition shows that when the Bayes function $f^*_{L,\mathrm{P}} \in C^{k,\alpha}$. HRTR can be approximated by the Taylor series concerning the homothetic ratio whose degree depends on the smoothness of the Bayes function.

**Proposition 2.4.** *Suppose that* $\mathrm{P}_X$ *has upper and lower bounded function over* $[0,1]^d$ *and the Bayes function* $f^*_{L,\mathrm{P}} \in C^{k,\alpha}$ *Moreover, let* $f_{\mathrm{P},r}$ *be defined by* (9)*. Then we have*

$$f_{\mathrm{P},r}(x) - f^*_{L,\mathrm{P}}(x) = \sum_{j=1}^k b_j r^j + \delta_{r,A}, \tag{11}$$

*where the remainder* $\delta_{r,A} \lesssim \mathrm{diam}(A(x))^{k+\alpha}$ *and* $b_1, \cdots b_k < B$ *are constants that depends on* $x$*.*

(11) tells us that HRTR converges to the target function as the ratio converges to zero. The theorem above directly indicates that

$$f_{\mathrm{D},r}(x) - f^*_{L,\mathrm{P}}(x) = \sum_{j=1}^k b_j r^j + \varepsilon_r, \tag{12}$$

where $\varepsilon_r = \delta_{r,A} + f_{\mathrm{D},r}(x) - f_{\mathrm{P},r}(x)$. This implies that the $f_{\mathrm{D},r}(x)$ behaves similar to $f_{\mathrm{P},r}(x)$ with additional error $f_{\mathrm{D},r}(x) - f_{\mathrm{P},r}(x)$ due to the empirical measure D.

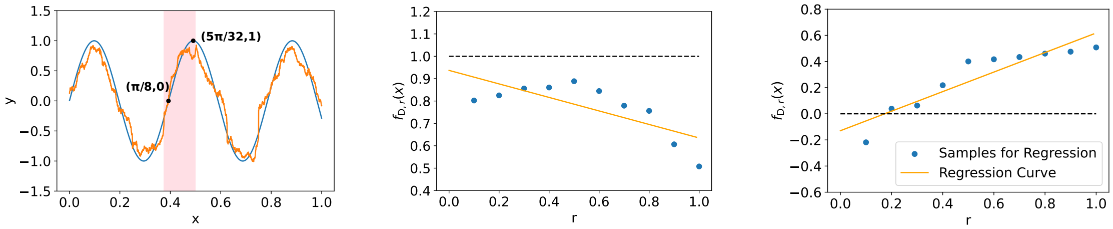
## 2.4. Extrapolated Random Tree for Regression

We dedicate this section to the extrapolation procedure. To be specific, we first fix a ratio sequence $\{r_i\}_{i=1}^V$ with $r_i = i/V$ and a pre-specified order parameter $L \leq V - 1$. Then, we compute the HRTR estimation for $r = r_i$, $1 \leq i \leq V$. Motivated by (12), we consider the following linear regression model to extrapolate these estimations to $r = 0$,

$$f_{\mathrm{D},r_i}(x) = b_0 + \sum_{j=1}^L b_j r_i^j + \epsilon_i, \quad 1 \leq i \leq V, \tag{13}$$

where $\epsilon_i := \delta_{r_i,A} + f_{\mathrm{D},r_i}(x) - f_{\mathrm{P},r_i}(x)$, and $b = (b_0, \ldots, b_L)^\top$ is the regression coefficient vector to be estimated. For $1 \leq i \leq k$, $b_i$ is specified in Proposition 2.4 and $b_i = 0$ for $k+1 \leq i \leq L$. Note that the regression function is a polynomial of $r_i$. Therefore, it can be solved efficiently through the standard least square method.

$$\hat{b} := \operatorname*{arg\,min}_{b \in \mathbb{R}^{L+1}} \sum_{i=1}^V \left( f_{\mathrm{D},r_i}(x) - b_0 - \sum_{j=1}^L b_j r_i^j \right)^2. \tag{14}$$



(a) ERTR estimation (b) Extrapolation at $(5\pi/32, 1)$ (c) Extrapolation at $(\pi/8, 0)$

*Figure 3.* Illustration of extrapolation. In (a), the blue line stands for the target function and the orange line stands for ERTR estimation. In (b) and (c), the blue points are $(r_i, f_{\mathrm{D},r_i}(x))$ with $V = 10$ for extrapolation. The green line represents the linear regression of the extrapolation. The Bayes function value at two points is represented by the horizontal dashed line.

The estimation can be explicitly expressed as

$$\hat{b} = (R^\top R)^{-1} R^\top (f_{\mathrm{D},r_1}(x), \cdots, f_{\mathrm{D},r_V}(x))^\top \tag{15}$$

where $R = (R_{ij})$ is a $V \times (L+1)$ matrix with $R_{ij} = r_i^{j-1}$. Then we propose the *extrapolated random tree for regression* (ERTR) as

$$f_{\mathrm{D,E}}(x) = \hat{b}_0. \tag{16}$$

The complete algorithm is presented in Algorithm 2.

**Algorithm 2** Extrapolated Random Tree for Regression

```
Input: Depth of the random tree $p$, order $L$, and number of estimations $V$.
Generate random tree partition $\pi_p$ by Algorithm 1;
for $i = 1$ to $V$ do
    Compute $A_{r_i}(x)$ with $r_i = i/V$ by (7);
    Compute $f_{\mathrm{D},r_i}(x)$ by (10);
end for
Compute the coefficient estimation $\hat{b}$ by (14);
Output: Extrapolated random tree for regression $f_{\mathrm{D,E}}(x)$ by (16).
```

ERTR first randomly splits the input space by the max-edge rule introduced in Algorithm 1. Then, for the test instance $x$, ERTR computes several HRTRs with a sequence of homothetic ratios $r_i$, $1 \le i \le V$. Finally, we obtain the estimation by using the linear regression model to extrapolate these estimations to $r = 0$.

Since ERTR eradicates the dominant terms in Taylor's expansion, i.e. $r^j$, $1 \le j \le k$, ERTR adapts to the high-order smoothness of the target function. Moreover, $f_{\mathrm{D,E}}(x)$ is actually a weighted average of $f_{\mathrm{D},r_i}(x)$, $1 \le i \le V$ from (15) and (16). This implies that ERTR is a well-interpretable model, which can not only tell us which node of the tree is taken for the prediction but also quantify the contributions of these data points in the prediction.

An explanation of ERTR is presented in Figure 3 for the following synthetic model employed in Cai et al. (2020),

$$Y = \sin(16X) + \varepsilon, \tag{17}$$

where $X \sim \mathrm{Unif}[0,1]$ and $\varepsilon \sim \mathcal{N}(0,1)$. We generate 2000 samples from this model for training and set $p = 3$, $L = 1$, and $V = 10$ for our algorithm. Figure 3(a) displays the target function and the estimation. To visualize the regression model in (14), we plot the curve of $(r, f_{\mathrm{D},r}(x))$ in blue for two fixed points $x = 5\pi/32$ and $\pi/8$ which reside in the same cell $(0.375, 0.5)$, respectively. The samples selected for the regression model (13) are colored in orange. It is clear to see that RTR predicts both points by an identical value of around $0.5$. On the contrary, from Figure 3(b) and 3(c), we see that the curve of HRTR can be approximated by a polynomial function. As a result, ERTR can make a more precise prediction thanks to the extrapolation method.

## 3. Theoretical Results

In this section, we present the theoretical results and related comments. To demonstrate the benefits of extrapolation, we establish optimal convergence rates of ERTR and the lower bound of the convergence rates for RTR in section 3.1 and 3.2, respectively. Finally, in Section 3.3, we conduct complexity analysis for ERTR.

### 3.1. Convergence Rates for ERTR in $C^{k,\alpha}$

**Theorem 3.1.** *Suppose that* $\mathrm{P}_X$ *has upper and lower bounded density over* $[0,1]^d$ *and the Bayes function* $f^*_{L,\mathrm{P}} \in C^{k,\alpha}$*. Let* $f_{\mathrm{D,E}}(x)$ *be the random tree extrapolation for regression defined by* (16)*. Moreover, let* $p_n$*,* $V$*, and* $L$ *be chosen as* $p_n \asymp \log(n/\log n)$ *and* $V - 1 \ge L \ge k$*. Then for all sufficiently large* $n$*, with probability* $\mathrm{P}^n$ *at least* $1 - 2/n^2$*, we have*

$$\mathcal{R}_{L,\mathrm{D}}(f_{\mathrm{D,E}}) - \mathcal{R}^*_{L,\mathrm{P}} \lesssim (\log n/n)^{\frac{2(k+\alpha)}{2(k+\alpha)+d}}. \tag{18}$$

Theorem 3.1 shows that when the Bayes function lies in the function space $C^{k,\alpha}$, our ERTR achieves optimal convergence rates with properly chosen depth $p$ and a sufficiently large $V$. Although Linero & Yang (2018) established similar convergence rates, their conclusion is derived from a bayesian perspective. To the best of our knowledge, we for the first time establish the optimal convergence rates for the smooth regression tree from the learning theory perspective.

### 3.2. Convergence Rates for RTR in $C^{k,\alpha}$

The following theorem presents the lower bound of the excess risk of the random tree for regression with some mild conditions on the gradients.

**Theorem 3.2.** *Let the regression model be defined by $Y := f(X) + \varepsilon$, where $\mathrm{P}_X$ is the uniform distribution over $[0,1]^d$ , $\mathrm{E}(\varepsilon|X = x) = 0$, and $\mathrm{Var}(\varepsilon|X = x) = \sigma^2 < \infty$ for $x \in [0,1]^d$. Moreover, assume that $f \in C^{k,\alpha}, k \geq 1$ and there exists a constant $\underline{c}_f \in (0,\infty)$ such that $\sqrt{(12d-7)/(12d-9)} \cdot \|\nabla f(x)\|_2 \geq \|\nabla f(x)\|_1 \geq \underline{c}_f$ for all $x \in [0,1]^d$. Then for all $n > N_1$, there holds*

$$\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{D}}) - \mathcal{R}^*_{L,\mathrm{P}} \gtrsim n^{-2/(2+d)} \tag{19}$$

*in expectation with respect to $\mathrm{P}^n$, where the constant $N_1$ is specified in the proof.*

The theorem above shows that in the space $C^{k,\alpha}$ with $k \geq 1$ and $0 < \alpha \leq 1$, the lower bound of the convergence rates RTR is merely of the order $O(n^{-2/(2+d)})$. This together with 3.1 implies that for any $k \geq 1$ and $\alpha \in (0,1]$, the upper bound of the convergence rate (18) for ERTR will be strictly smaller than the lower bound (19) for RTR, which explains the benefits of the extrapolation in our method for the target function with high-order smoothness. In fact, as pointed out in Theorem A.3 in Appendix A.1.2, RTR only achieves the optimal convergence rates in the space $C^{0,\alpha}$.

Since the edges of each cell are parallel to the axes, it is more convenient to use the conditions on the derivatives to derive the lower bound of RTR. The condition on the gradients in Theorem 3.2 can be satisfied in the general space $\mathbb{R}^d$ with $d \geq 1$. Let us consider the following example, where $f(x) = \sum_{i=1}^d a_i x_i + 1$ with $a_i = (12d-8)^{i-1}$ for $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$. It is clear to verify that $\sqrt{(12d-7)/(12d-9)}\|\nabla f(x)\|_2 \geq \|\nabla f(x)\|_1$. In fact, this condition is satisfied for a large class of functions whose derivatives along different directions vary widely.

### 3.3. Complexity Analysis

We consider the computation complexity of ERTR including both the training and testing stages. The training stage consists of only the Algorithm 1, whose average cost is $O(n \log n)$. Then for each test sample, computation of $f_{\mathrm{D},r_i}, 1 \leq i \leq V$ takes $\mathcal{O}\big(dn^{\frac{2(k+\alpha)}{2(k+\alpha)+d}}\big)$ time since the number of samples in each cell is around $\mathcal{O}(n/2^p) = \mathcal{O}\big(n^{\frac{2(k+\alpha)}{2(k+\alpha)+d}}\big)$, where we use $2^p \asymp \big((n/\log n)^{\frac{d}{2(k+\alpha)+d}}\big)$ by Theorem 3.1. Therefore, the test complexity of ERTR is $\mathcal{O}\big(dn^{\frac{2(k+\alpha)}{2(k+\alpha)+d}}\big)$. Thus, ERTR is an efficient method with sub-linear complexity.

In comparison, we discuss the complexities of several other tree models. For the standard decision tree, the training complexity is $O(n \log n)$ and the testing complexity is $O(\log n)$. For PR tree (Alkhoury et al., 2020), the training stage involves solving the weight matrix for all training samples which yields computation cost of $\mathcal{O}(nK^2 d)$ where $K$ is the number of cells. To ensure consistency, $K$ needs to grow with some order of $n$, which yields the super-linear cost of the PR tree. Soft Bayesian additive trees (Linero & Yang, 2018) require MCMC sampling which brings a heavy computation burden. For soft trees (Irsoy et al., 2012), its optimization relies on first-order methods, and thus their complexities are not comparable to ours. In short, though ERTR requires additional computation compared to the decision tree, it is still an efficient method compared to other regression trees. Therefore, an extension of ERTR to ensemble methods appears to be computationally feasible.

## 4. Experiments

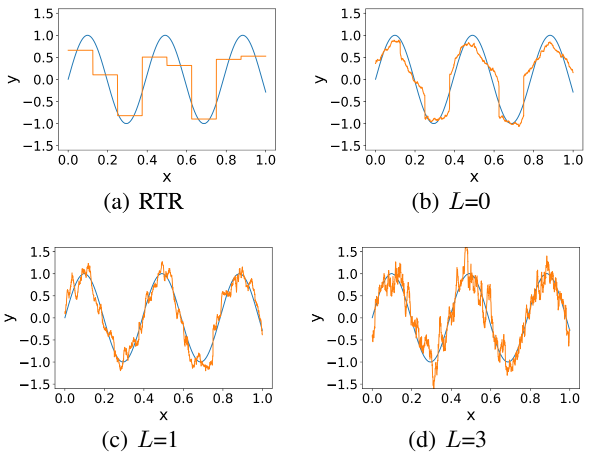
In experiments, we first conduct experiments on synthetic datasets in Section 4.1 to illustrate theoretical findings in Section 3. Then, in Section 4.2, we compare ERTR, as well as its ensemble extensions, with other tree-based regression models to show its superior performance.

### 4.1. Synthetic Experiments

In synthetic experiments, we first investigate the power of extrapolation in Section 4.1.1 and demonstrate the interpretability of ERTR in Section 4.1.2. Then, we conduct parameter analysis in Section 4.1.3.

#### 4.1.1. Power of Extrapolation

We first conduct experiments to show that the extrapolation method helps to improve the accuracy and the smoothness. To this end, we investigate the choice of the extrapolation order $L$. We fit RTR and ERTR with $L = 0, 1, 3$ on 2000 training samples from the synthetic model in (17). The depth of partition $p$ for both methods is set to 3. The regression curves as well as the ground truth are plotted in Figure 4. As in Figure 4(a), RTR using piece-wise constant functions fails to capture the variation of smooth functions and has poor performance. By contrast, ERTR is adaptive to the smoothness of the target function. From Figure 4(b), 4(c), and 4(d), we see that the choice of order $L$ affects the smoothness of the regressor. ERTR is under-fitting with a small $L$ due to poor approximation ability. In this case, ERTR can not benefit from high-order smoothness as in Figure 4(b). On the other hand, when the order $L$ is chosen too large, there are only a few samples in the cell. As a result, ERTR tends to overfit the model as shown in Figure 4(d). A proper extrapolation order can lead to a stable regressor with adequate approximation ability, as is shown in 4(c) for



(a) RTR (b) $L$=0 (c) $L$=1 (d) $L$=3

*Figure 4.* The estimated regression curve of RTR and ERTR with $L = 0, 1, 2$. The blue curves and the orange curves stand for ground truth and estimated target function respectively.

$L = 1$. These observations are compatible with Theorem A.3 that extrapolation brings a faster convergence rate with a suitably chosen order $L$ for a smoother target function.

### 4.1.2. Interpretability

We illustrate the interpretability of ERTR in prediction as mentioned in Section 2.4. (15) and (16) tell us that ERTR is the weighted average of the responses in the cell of the random tree partition. Therefore, to demonstrate the interpretability, we visualize the weights versus $r$ for $L = 0, 1, 2$ and RTR under the same simulation setting in Section 4. In contrast to RTR, which assigns equal weight to each sample, ERTR assigns larger weights to samples close to $x$ and smaller weights to samples far from $x$. Moreover, as $L$ increases, ERTR becomes overfitting as it puts too much weight on the nearby points. This implies that ERTR is interpretable in the sense that, we can not only recognize the points in the cell that influence the estimation but also quantify the influence of each data point.

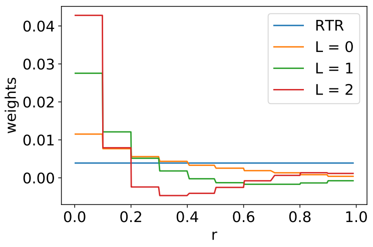


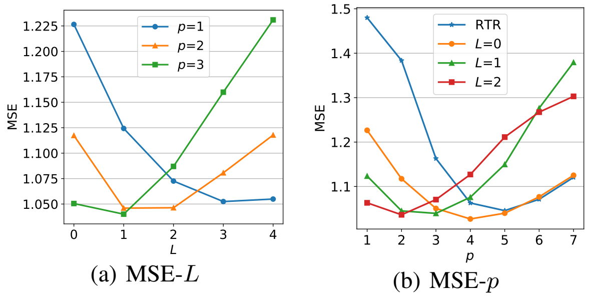
*Figure 5.* The weights of samples versus $r$ for RTR and ERTR with $L = 0, 1, 2$ at $x = 5\pi/32$.

We mention that the parameters $V$ and $L$ should be carefully chosen in terms of balancing the convergence rate and interpretability. $(i)$ According to the parameter choices in Theorem 3.1, a smaller $V$ yields a smaller $L$. In particular, if $V \leq k$, we can show that the convergence rate of ERTR becomes $(n/\log n)^{-2L/(2L+d)}$. In this case, ERTR can not fully use the target function's smoothness to achieve the optimal convergence rate. $(ii)$ Both the choices of $V$ and $L$ affect the interpretability of the model. More specifically, $L$ controls the pattern of the weights and $V$ affects the values of these weights. As shown in Figure 5, ERTR with larger $L$ assigns larger weights to samples close to $x$ and smaller weights to samples far from $x$, while ERTR with smaller $L$ assigns close weights to the samples in the cell $A(x)$. This implies that small $V$ and $L$ help to enhance the interpretability of the model. Therefore, we need to choose appropriate $V$ and $L$ to balance the convergence rate and interpretability.

### 4.1.3. Parameter Analysis

In this section, we conduct experiments to investigate the selections of $p$ and $L$ in terms of MSE. We pick $p \in \{1, 2, 3\}$ and plot MSE versus $L$ for $L \in \{0, 1, 2, 3, 4\}$ on (17). For each pair of $(p, L)$, we set $\lambda = 10^{-4}$ as the regularized parameter for ridge regression and choose $V \in \{15, 20, 25\}$ by cross-validation. We take 10 times averaged MSE with 1000 training samples in each repetition. The result is displayed in Figure 6(a). Apparently, for each $p$, as $L$ increases, MSE first decreases until $L$ reaches a certain value. Then MSE begins to increase as $L$ grows. This further confirms the trade-off observed in Section 4.1.1. Moreover, Figure 6(a) shows that the order $L$ at which the test error achieves the minimum decreases as $p$ increases. Intuitively, both the increasing of $p$ and $L$ enhance the approximation ability of the model. Therefore, a small $L$ is demanded to achieve the best performance for large $p$.

Next, we investigate the relation between depth $p$ and MSE under different $L$ under analogous settings. We also plot the MSE-$p$ curve for the RTR estimation. The result is displayed in Figure 6(b). The relation between MSE and $p$ is U-shaped under each $L$. This illustrates that a properly chosen $p$ is needed as explained in Theorem 3.1.



(a) MSE-$L$ (b) MSE-$p$

*Figure 6.* 6(a) MSE versus $L$ with different $p$. 6(b) MSE versus $p$ under different $L$.

*Table 1.* Average MSE over real data sets for tree methods. The best results are **bolded** and the second best results are underlined. The best results with significance are marked with $*$. Running of ST is corrupted on three data sets that are marked with -.

| | Random Partition | | | | Variance Reduction Partition | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | ERTR | RTR | STRT | PRT | ERTR | DT | ST | STRT | PRT |
| ABA | **5.60*** | 8.01 | 7.33 | 6.11 | 5.20 | 5.75 | **4.59*** | 5.53 | 5.89 |
| AIR | **2.79e+1*** | 3.92e+1 | 3.31e+1 | 3.56e+1 | **1.13e+1*** | 1.43e+1 | 1.22e+1 | 2.02e+1 | 3.49e+1 |
| ALG | **9.27e-2*** | 1.87e-1 | 1.51e-1 | 9.66e-2 | **3.03e-2*** | 3.44e-2 | 2.77e-1 | 3.21e-2 | 3.30e-2 |
| BIAS | 3.87 | 4.38 | 4.55 | **2.45*** | **1.60*** | 1.75 | - | 2.01 | 2.09 |
| CBM | **1.52e-1*** | 1.53 | 1.70e-1 | 3.61e-1 | 2.31e-10 | **4.24e-27*** | 3.70 | 9.08e-4 | 1.89e-1 |
| CCP | 3.95e+1 | 6.33e+1 | **3.60e+1** | 4.93e+1 | 3.92e+1 | **1.82e+1** | 2.93e+2 | 2.07e+1 | 2.56e+1 |
| CPU | **5.01e+1*** | 3.18e+2 | 3.23e+2 | 2.77e+2 | **1.36e+1*** | 1.54e+1 | 3.38e+2 | 1.63e+1 | 3.00e+1 |
| IST | 1.20e-4 | 1.58e-4 | 6.55e-5 | **2.71e-5*** | **3.69e-5** | 4.43e-5 | - | 5.57e-5 | 3.98e-5 |
| FOR | 4.56e+3 | **4.15e+3** | 4.70e+3 | 4.39e+3 | 4.25e+3 | 5.14e+3 | **4.08e+3** | 4.35e+3 | 4.11e+3 |
| MG | 1.90e-2 | 2.09e-2 | 2.75e-2 | **1.58e-2*** | **1.63e-2*** | 1.83e-2 | 2.15e-2 | 1.85e-2 | 2.58e-2 |
| WR | 5.72e-1 | 6.02e-1 | 5.85e-1 | **4.46e-1*** | 4.63e-1 | 4.64e-1 | 5.12e-1 | **4.29e-1*** | 4.63e-1 |
| SPA | **1.89e-2*** | 2.83e-2 | 2.25e-2 | 2.18e-2 | **1.72e-2** | 1.84e-2 | 1.72e-2 | 1.88e-2 | 2.46e-2 |
| WW | **5.95e-1*** | 7.59e-1 | 7.36e-1 | 7.62e-1 | **5.46e-1*** | 5.57e-1 | - | 5.50e-1 | 5.82e-1 |
| rank sum | **21** | 45 | 37 | 27 | **22** | 37 | 50 | 38 | 48 |

## 4.2. Real Data Performance

In this section, we conduct experiments on real-world data sets to show the promising performance of ERTR compared with other regression trees in Section 4.2.1. In Section 4.2.2, we further investigate the extension of extrapolated trees to ensemble methods.

**Splitting Rule** Note that most popular tree methods design their partition rules utilizing the information gained from the data. For a fair comparison, we also introduce the variance reduction scheme (Breiman, 2001) to the tree construction. To be more specific, we replace the random partition rule in Algorithm 2 with the variance reduction splitting rule. Each current region will be decomposed into two sub-regions with respect to a coordinate and a splitting point that minimizes the weighted sum of variances in the resulting sub-regions.

**Comparison Methods** Under the variance reduction splitting rule, the comparison methods include DT, ST (Irsoy et al., 2012), STRT (Da Rosa et al., 2008), and PRT (Alkhoury et al., 2020). Under the random splitting rule in Section 2.2, the comparison methods include RTR, PRT, and STRT. For forest methods, we compare the extension of ERTR, called ERF, with RF, PRRF, and SBART (Linero & Yang, 2018). For boosting methods, we compare the extension of ERTR, called GBERTR, with GBRT, GBPRT, and GBSTRT. The variance reduction splitting rule is used when ERTR is extened to the ensemble methods. All implementation details are presented in Appendix C.1.

**Experiment Setup** We conduct experiments on 12 real data sets. To ensure significance, we adopt the Wilcoxon signed-rank test (Wilcoxon, 1992) to check if the best result is significant. For ERTR, regression in (14) is ridge regularized with shrinkage parameter $\lambda$. We summarize the data sets and pre-processing details in Appendix C.1.

### 4.2.1. Comparison of Tree Methods

Results on the comparison of tree methods are presented in Table 1. For models with the random splitting rule, ERTR shows superior performance over the other models by achieving 7 best results and rank sum 21. Especially, ERTR outperforms RTR on most of the data sets. For models with the variance reduction splitting rule, ERTR outperforms DT on 11 out of 13 data sets. Moreover, ERTR performs promisingly compared to the other popular tree methods by achieving 8 best results and rank sum 22. In both settings, extrapolated methods greatly improve the regression performance compared to their ordinary counterparts.

*Table 2.* Average MSE over real data sets for forest methods. The best results are **bolded** and the second best results are underlined. The best results with significance are marked with $*$. Running of SBART is corrupted on one data set which is marked with -.

| | ERF | RF | PRRF | SBART |
|---|---|---|---|---|
| ABA | **4.64*** | 4.71 | 4.87 | 4.91 |
| AIR | 6.35 | 6.11 | 2.00e+1 | **3.90*** |
| ALG | **1.76e-2*** | 2.13e-2 | 4.83e-2 | 2.59e-2 |
| BIAS | 1.12 | 1.19 | 1.90 | **8.67e-1*** |
| CBM | 5.88e-9 | **1.21e-27*** | 3.11e-4 | - |
| CCP | **1.27e+1*** | 1.41e+1 | 1.96e+1 | 1.55e+1 |
| CPU | 9.79 | 9.87 | 2.02e+1 | **7.67*** |
| IST | 3.45e-5 | 3.40e-5 | 3.03e-5 | **2.93e-5** |
| FOR | **3.24e+3** | 3.52e+3 | 5.14e+3 | 3.81e+3 |
| MG | **1.39e-2** | 1.42e-2 | 1.88e-2 | 1.43e-2 |
| WR | 3.89e-1 | **3.69e-1** | 3.97e-1 | 3.90e-1 |
| SPA | 1.36e-2 | 1.37e-2 | 1.94e-2 | **1.07e-2*** |
| WW | **4.55e-1*** | 4.62e-1 | 5.40e-1 | 4.93e-1 |
| rank sum | **23** | 28 | 50 | 30 |

### 4.2.2. Comparison of Ensemble Methods

Extending our estimator to a forest method that adopts ERTR as base learners is straightforward. We compare the arising forest method, which we refer to as ERF, with other forest approaches using different base learners. The results for forest methods are presented in Table 2. ERF not only achieves much smaller MSE than single tree methods in Table 1 but also outperforms other competitors. ERF has the best performance on 6 data sets and the lowest rank sum 22. The promising performance of ERF comes from the complementary property of random forest and extrapolation. Random forests aim at reducing the variance with a small cost of increasing the bias, whereas extrapolation intends to reduce the bias. Therefore, base learners with low bias are averaged to have low variance, which leads to a method that significantly outperforms RF. Similarly, ERTR is extended to gradient boosting with results displayed in Table 4, Appendix C.3, where GBERTR attains the best ranking sum 25. These results show that the application of extrapolated trees enhances the performance of ensemble methods.

## 5. Conclusion

In this paper, we propose a novel tree-based algorithm called *Extrapolated Random Tree for Regression* (ERTR) that adapts to the high-order smoothness of the target function while maintaining the interpretability of the tree. On the theoretical side, we for the first time establish optimal convergence rates for ERTR when the target function resides in the general Hölder space and the lower bound of the convergence rates of the RTR, which shows that ERTR outperforms RTR for the target function with high-order smoothness by taking advantage of extrapolation. In experiments, we empirically demonstrate the power of the extrapolation method. Moreover, we show the experimental preponderance of ERTR compared to state-of-the-art regression trees. Furthermore, we extend ERTR to ensemble methods including random forest and gradient boosting. Promising improvements are brought by using the extrapolated trees as base learners. The application of the extrapolation method to enhance the performance of learning algorithms can be served as future works.

## Acknowledgements

The authors would like to thank the reviewers for their constructive comments, which led to a significant improvement in this work. This work was supported in part by the National Key R&D Program of China (2022YFB2702401). This research was supported by Public Computing Cloud, Renmin University of China.

## References

Abid, F. and Izeboudjen, N. Predicting forest fire in algeria using data mining techniques: Case study of the decision tree algorithm. In *International Conference on Advanced Intelligent Systems for Sustainable Development*, pp. 363–370. Springer, 2020.

Akbilgic, O., Bozdogan, H., and Balaban, M. E. A novel hybrid rbf neural networks model as a forecaster. *Statistics and Computing*, 24(3):365–375, 2014.

Alkhoury, S., Devijver, E., Clausel, M., Tami, M., Gaussier, É., et al. Smooth and consistent probabilistic regression trees. *Advances in Neural Information Processing Systems*, 33:11345–11355, 2020.

Altosole, M., Benvenuto, G., Figari, M., and Campora, U. Real-time simulation of a cogag naval ship propulsion system. *Proceedings of the institution of mechanical engineers, part M: journal of engineering for the maritime environment*, 223(1):47–62, 2009.

Amini, S., Saber, M., Rabiei-Dastjerdi, H., and Homayouni, S. Urban land use and land cover change analysis using random forest classification of landsat time series. *Remote Sensing*, 14(11):2654, 2022.

Basak, P., Linero, A., Sinha, D., and Lipsitz, S. Semiparametric analysis of clustered interval-censored survival data using soft bayesian additive regression trees (sbart). *Biometrics*, 78(3):880–893, 2022.

Bernstein, S. N. *The Theory of Probabilities*. Gastehizdat Publishing House, Moscow, 1946.

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

Brezinski, C. and Zaglia, M. R. *Extrapolation methods: theory and practice*. Elsevier, 2013.

Brooks, T. F., Pope, D. S., and Marcolini, M. A. Airfoil self-noise and prediction. Technical report, 1989.

Cai, Y., Hang, H., Yang, H., and Lin, Z. Boosted histogram transform for regression. In *International Conference on Machine Learning*, pp. 1251–1261. PMLR, 2020.

Chang, C.-C. and Lin, C.-J. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

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

Cho, D., Yoo, C., Im, J., and Cha, D.-H. Comparative assessment of various machine learning-based bias correction methods for numerical weather prediction model forecasts of extreme air temperatures in urban areas. *Earth and Space Science*, 7(4):e2019EA000740, 2020.

Cortez, P. and Morais, A. d. J. R. A data mining approach to predict forest fires using meteorological data. 2007.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4): 547–553, 2009.

Cucker, F. and Zhou, D.-X. *Learning Theory: An Approximation Theory Viewpoint*. Cambridge University Press, 2007.

Da Rosa, J. C., Veiga, A., and Medeiros, M. C. Tree-structured smooth transition regression models. *Computational Statistics & Data Analysis*, 52(5):2469–2488, 2008.

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

Flake, G. W. and Lawrence, S. Efficient svm regression training with smo. *Machine Learning*, 46(1):271–290, 2002.

Friedman, J. H. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

Frosst, N. and Hinton, G. Distilling a neural network into a soft decision tree. *arXiv preprint arXiv:1711.09784*, 2017.

Gao, W. and Zhou, Z.-H. Towards convergence rate analysis of random forests for classification. *Advances in neural information processing systems*, 33:9300–9311, 2020.

Giné, E. and Nickl, R. *Mathematical Foundations of Infinite-Dimensional Statistical Models*. Cambridge University Press, 2021.

Horn, R. A. and Johnson, C. R. *Matrix analysis*. Cambridge university press, 2012.

Irsoy, O., Yıldız, O. T., and Alpaydın, E. Soft decision trees. In *Proceedings of the 21st international conference on pattern recognition (ICPR2012)*, pp. 1819–1822. IEEE, 2012.

Kosorok, M. R. *Introduction to Empirical Processes and Semiparametric Inference*. Springer Series in Statistics. Springer, New York, 2008.

Künzel, S. R., Saarinen, T. F., Liu, E. W., and Sekhon, J. S. Linear aggregation in tree-based estimators. *Journal of Computational and Graphical Statistics*, pp. 1–18, 2022.

Li, B., Friedman, J., Olshen, R., and Stone, C. Classification and regression trees (cart). *Biometrics*, 40(3):358–361, 1984.

Linero, A. R. and Yang, Y. Bayesian regression tree ensembles that adapt to smoothness and sparsity. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):1087–1110, 2018.

Massart, P. *Concentration Inequalities and Model Selection*, volume 1896 of *Lecture Notes in Mathematics*. Springer, Berlin, 2007.

Mourtada, J., Gaïffas, S., and Scornet, E. Minimax optimal rates for mondrian trees and forests. *The Annals of Statistics*, 48(4):2253–2276, 2020.

Nash, W. J., Sellers, T. L., Talbot, S. R., Cawthorn, A. J., and Ford, W. B. The population biology of abalone (haliotis species) in tasmania. i. blacklip abalone (h. rubra) from the north coast and islands of bass strait. *Sea Fisheries Division, Technical Report*, 48:p411, 1994.

Okuno, A. and Shimodaira, H. Extrapolation towards imaginary 0-nearest neighbour and its improved convergence rate. *Advances in Neural Information Processing Systems*, 33:21889–21899, 2020.

Pace, R. K. and Barry, R. Quick computation of spatial autoregressive estimators. *Geographical analysis*, 29(3): 232–247, 1997.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Ročková, V. and Saha, E. On theory for bart. In *The 22nd international conference on artificial intelligence and statistics*, pp. 2839–2848. PMLR, 2019.

Ročková, V. and van der Pas, S. Posterior concentration for bayesian regression trees and forests. *The Annals of Statistics*, 48(4):2108–2131, 2020.

Steinwart, I. and Christmann, A. *Support Vector Machines*. Springer Science & Business Media, 2008.

Suárez, A. and Lutsko, J. F. Globally optimal fuzzy decision trees for classification and regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12): 1297–1311, 1999.

Tüfekci, P. Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140, 2014.

van der Vaart, A. W. and Wellner, J. A. *Weak Convergence and Empirical Processes*. Springer Series in Statistics. Springer-Verlag, New York, 1996.

Wilcoxon, F. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pp. 196–202. Springer, 1992.

Yu, X., Rao, Y., Zhao, W., Lu, J., and Zhou, J. Group-aware contrastive regression for action quality assessment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7919–7928, 2021.

The appendix consists of supplementary for both theoretical analysis and experiments. In Appendix A, we present the error analysis for both RTR and ERTR. The proofs for theoretical results in the main content and Appendix A are presented in Appendix B. In Appendix C, we show the supplementary for numerical experiments, including implementation details, data set details and additional real data results.

## A. Error Analysis

In this section, we present the error analysis of RTR and ERTR in Section A.1 and A.2, respectively.

### A.1. Error Analysis of RTR

In this section, we present the error decompositions for the lower bound and upper bound of the convergence for RTR in Section A.1.1 and A.1.2, respectively.

### A.1.1. Lower bound for the convergence rates of RTR

For the base regressor RTR, we are concerned with the lower bound for $f_{\mathrm{D}}$. We make the error decomposition

$$\begin{aligned}\mathbb{E}_{\nu_n}\big(\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{D}})-\mathcal{R}^*_{L,\mathrm{P}}\big)&=\mathbb{E}_{\nu_n}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f^*_{L,\mathrm{P}}(x)\big)^2\\&=\mathbb{E}_{\nu_n}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X)\big)^2+\mathbb{E}_{\nu_n}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{P}}(X)-f^*_{L,\mathrm{P}}(x)\big)^2.\end{aligned}\tag{20}$$

It is important to note that the two terms on the right-hand side of (20) are data- and partition-independent due to the expectation with respect to D and $Z$. Loosely speaking, the first error term corresponds to the expected estimation error of the estimator $f_{\mathrm{D}}$, while the second one demonstrates the expected approximation error.

The following two propositions present the lower bound of approximation error and sample error of RTR respectively.

**Proposition A.1.** *Let the random tree for regression $f_{\mathrm{P}}$ be defined as in (4) and the regression model be defined by*

$$Y:=f(X)+\varepsilon,\tag{21}$$

*where $\mathrm{P}_X$ is the uniform distribution over $[0,1]^d$, $\mathrm{E}(\varepsilon|X)=0$ and $\mathrm{Var}(\varepsilon|X)=\sigma^2<\infty$. Moreover, assume that $f\in C^{k,\alpha}$, $k\geq 1$ and there exists a constant $\underline{c}_f\in(0,\infty)$ such that $\|\nabla f(x)\|_2\geq c\|\nabla f(x)\|_1\geq \underline{c}_f$ for all $x\in[0,1]^d$ with the constant $c:=\sqrt{(12d-9)/(12d-7)}$. Then for all $n\geq 1$, there holds*

$$\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{P}})-\mathcal{R}^*_{L,\mathrm{P}}\geq d\cdot c_f^2(384d-288)^{-1}\cdot 2^{-2p/d}.$$

*in expectation with respect to $\mathrm{P}_Z$.*

**Proposition A.2.** *Let the random tree for regression $f_{\mathrm{P}}$ and its empirical estimate $f_{\mathrm{D}}$ be defined by (4) and (5), respectively. Moreover, let the regression model be defined as in (21) with $f\in C^{k,\alpha}$, $k\geq 1$. Moreover, assume that $\mathrm{P}_X$ is the uniform distribution over $[0,1]^d$ , $\mathrm{E}(\varepsilon|X=x)=0$, and $\mathrm{Var}(\varepsilon|X=x)=\sigma^2<\infty$ for $x\in[0,1]^d$. Then there holds*

$$\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{D}})-\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{P}})\geq\sigma^2\cdot 2^p(1-2e^{-1})\cdot n^{-1}$$

*in expectation with respect to $\mathrm{P}^n$.*

### A.1.2. Upper bound for the convergence rates of RTR

In addition to the lower bound of RTR, we also provide the following upper bound.

**Theorem A.3.** *Suppose that $\mathrm{P}_X$ has upper and lower bounded density over $[0,1]^d$ and the Bayes function $f^*_{L,\mathrm{P}}(x)\in C^{k,\alpha}$. Let $f_{\mathrm{D}}(x)$ be the random tree for regression defined by (5) and $p_n\asymp\log(n/\log n)$. Then for all sufficiently large $n$, with probability $\mathrm{P}^n$ at least $1-2/n^2$, we have*

$$\mathcal{R}_{L,\mathrm{D}}(f_{\mathrm{D}})-\mathcal{R}^*_{L,\mathrm{P}}\lesssim(n/\log n)^{-\frac{2((\alpha+k)\wedge 1)}{2((\alpha+k)\wedge 1)+d}}.\tag{22}$$

The theorem above implies that when the Bayes function lies in the function space $C^{k,\alpha}$, under mild assumptions, RTR has the convergence rates of the order $\mathcal{O}(n^{-\frac{2\alpha}{2\alpha+d}})$. This rate is optimal only when the target function lies in the space $C^{0,\alpha}$. In other words, RTR can not take full advantage of the smoothness of the Bayes function and thus have slower convergence rates.

The proof relies on the following error decomposition.

$$\|f_{\mathrm{D}} - f^*_{L,\mathrm{P}}\|_\infty \leq \|f_{\mathrm{D}} - f_{\mathrm{P}}\|_\infty + \|f^*_{L,\mathrm{P}} - f_{\mathrm{P}}\|_\infty.$$

The two terms are bounded by the following two propositions.

**Proposition A.4.** *Let $\pi_p$ be a random tree partition of $[0,1]^d$ as in (2). Let $f_{\mathrm{P}}(x)$ be defined by (4) and $f^*_{L,\mathrm{P}}(x)$ be the Bayes function. Then we have*

$$\|f_{\mathrm{P}} - f^*_{L,\mathrm{P}}\|_\infty \leq c_L (2\sqrt{d})^{(k+\alpha)\wedge 1} 2^{-\big(p(k+\alpha)\wedge 1\big)/d}.$$

**Proposition A.5.** *Let $\pi_p$ be a random tree partition of $[0,1]^d$ as in (2). Let $f_{\mathrm{P}}(x)$ and $f_{\mathrm{D}}(x)$ be defined by (4) and (5), respectively. Assume that $\mathrm{P}_X$ has uppper and lower bounded density over $[0,1]^d$ and $\mathcal{Y} \subset [-M, M]$. Then for all $n \geq 1$, with probability $\mathrm{P}^n \otimes \mathrm{P}_Z$ at least $1 - 2/n^2$, there holds*

$$\|f_{\mathrm{D}} - f_{\mathrm{P}}\|_\infty \lesssim 2M\sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} \cdot M(4d+5)\log n}{3n} + \frac{2^{p+3} \cdot M}{n}.$$

### A.2. Error Analysis of ERTR

Let $e_1$ denote the vector $(1, 0, \cdots, 0)^\top$. The definition of ERTR in (16) can be reformalized as

$$f_{\mathrm{D,E}}(x) = e_1^\top (R^\top R)^{-1} R^\top (f_{\mathrm{D},r_1}(x), \cdots, f_{\mathrm{D},r_V}(x))^\top.$$

We also define

$$f_{\mathrm{P,E}}(x) = e_1^\top (R^\top R)^{-1} R^\top (f_{\mathrm{P},r_1}(x), \cdots, f_{\mathrm{P},r_V}(x))^\top \tag{23}$$

which stands for the population version of the extrapolated estimator. Then we have the following error decomposition

$$\|f_{\mathrm{D,E}} - f^*_{L,\mathrm{P}}\|_\infty \leq \|f_{\mathrm{D,E}} - f_{\mathrm{P,E}}\|_\infty + \|f^*_{L,\mathrm{P}} - f_{\mathrm{P,E}}\|_\infty.$$

**Proposition A.6.** *Let $f_{\mathrm{P,E}}(x)$ be defined by (23) and $f^*_{L,\mathrm{P}}(x)$ be the Bayes function. If we choose $V - 1 \geq L \geq k$, there holds*

$$\|f_{\mathrm{P,E}} - f^*_{L,\mathrm{P}}\|_\infty \leq c_{V,L} (2\sqrt{d})^{k+\alpha} 2^{-p(k+\alpha)/d}$$

*for constant $c_{V,L}$ depending only on $V$ and $L$.*

**Proposition A.7.** *Let $f_{\mathrm{P,E}}(x)$ and $f_{\mathrm{D,E}}(x)$ be defined by (23) and (16), respectively. Suppose that $\mathrm{P}_X$ has upper and lower bounded density over $[0,1]^d$ and $\mathcal{Y} \subset [-M, M]$. Then for all $n \geq 1$, with probability $\mathrm{P}^n \otimes \mathrm{P}_Z$ at least $1 - 2/n^2$, there holds*

$$\|f_{\mathrm{D,E}} - f_{\mathrm{P,E}}\|_\infty \lesssim 2c_{V,L}V^d M\sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} \cdot c_{V,L}V^d M(4d+5)\log n}{3n} + \frac{2^{p+3} \cdot c_{V,L}V^d M}{n}$$

*for constant $c_{V,L}$ depending only on $V$ and $L$.*

## B. Proofs

### B.1. Fundamental Results on the Properties of the Random Tree

**Lemma B.1.** *Let $\pi_p := \{A_p^j, j \in \mathcal{I}_p\}$ be the random tree partition as in (2). Then for any $A_p^j := \times_{i=1}^d [a_i, b_i]$, $j \in \mathcal{I}_p$, we have $\sqrt{d}/2 \cdot 2^{-p/d} \leq \mathrm{diam}(A_p^j) \leq 2\sqrt{d} \cdot 2^{-p/d}$. Moreover, for any $1 \leq i \leq d$, there holds*

$$\mathrm{diam}(A_p^j)^2 \leq (4d-3)(b_i - a_i)^2. \tag{24}$$

*Proof of Lemma B.1.* According to the random tree partition rule, when the depth of the tree $p$ is a multiple of dimension $d$, each cell of the random tree partition is a high-dimensional cube with side length of $2^{-p/d}$. On the other hand, when the depth of the tree $p$ is not a multiple of dimension $d$, we consider the random tree partition with depth $\lfloor p/d \rfloor$ and $\lceil p/d \rceil$, whose corresponding side length of the higher dimensional cube is $2^{-\lfloor p/d \rfloor}$ and $2^{-\lceil p/d \rceil}$. Note that under the splitting criterion of random tree, the side length of each sub-rectangle decreases monotonically with the increase of $p$, so the side length of a random tree partition cell is between $2^{-\lceil p/d \rceil}$ and $2^{-\lfloor p/d \rfloor}$. This implies that

$$\sqrt{d} \cdot 2^{-\lceil p/d \rceil} \leq \mathrm{diam}(A_p^j) \leq \sqrt{d} \cdot 2^{-\lfloor p/d \rfloor}$$

Since $p/d - 1 \leq \lfloor p/d \rfloor \leq \lceil p/d \rceil \leq p/d + 1$, we immediately get $\sqrt{d}/2 \cdot 2^{-p/d} \leq \mathrm{diam}(A_p^j) \leq 2\sqrt{d} \cdot 2^{-p/d}$. This shows the first assertion.

Since the random tree partition rule divides the rectangles along the midpoint of the coordinate with maximal length, for every $1 \leq i \leq d$, there holds

$$|b_j - a_j| \leq 2|b_i - a_i|, \quad 1 \leq j \leq d, \ j \neq i.$$

This implies that

$$\mathrm{diam}(A_p^j)^2 \leq \sum_{\ell=1}^{d} (b_\ell - a_\ell)^2 \leq 4(d-1)(b_i - a_i)^2 + (b_i - a_i)^2 = (4d-3)(b_i - a_i)^2.$$

This completes the proof. □

### B.2. Proofs of the Results for RTR

#### B.2.1. Proofs Related to Section A.1.1

*Proof of Proposition A.1.* Recall that the regression model is defined as $Y = f(X) + \varepsilon$ with $X$ following the uniform distribution. Let $\pi_p = \{A_p^j, \, j \in \mathcal{I}_p\}$ be the random tree partition in (2). Then we have

$$\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{P}}) - \mathcal{R}_{L,\mathrm{P}}^* = \mathbb{E}_{\mathrm{P}_X} \big( f_{\mathrm{P}}(X) - f_{L,\mathrm{P}}^*(x) \big)^2 = \sum_{j \in \mathcal{I}_p} \int_{A_p^j} \big( f_{\mathrm{P}}(x) - f_{L,\mathrm{P}}^*(x) \big)^2 \, dx. \tag{25}$$

Since $\mathrm{P}_X$ is the uniform distribution over $[0,1]^d$, for any $x \in A_p^j$, we have

$$f_{\mathrm{P}}(x) = \frac{1}{\mathrm{P}_X(A_p^j)} \int_{A_p^j} f_{L,\mathrm{P}}^*(x') \, dx' = \frac{1}{\mu(A_p^j)} \int_{A_p^j} f_{L,\mathrm{P}}^*(x') \, dx'.$$

where $\mu$ is the Lesbugue measure. Since the Bayes function $f_{L,\mathrm{P}}^*(x)$ is continuous on $A_p^j$, the mean value theorem implies that there exists an $\xi \in A_p^j$ such that $f_{\mathrm{P}}(x) = f_{L,\mathrm{P}}^*(\xi)$. Consequently, we get

$$\int_{A_p^j} \big( f_{\mathrm{P}}(x) - f_{L,\mathrm{P}}^*(x) \big)^2 \, dx = \int_{A_p^j} \big( f_{L,\mathrm{P}}^*(\xi) - f_{L,\mathrm{P}}^*(x) \big)^2 \, dx. \tag{26}$$

For every fixed $x \in A_p^j$, we define $h(t) := f_{L,\mathrm{P}}^*((1-t)x + t\xi)$ for $0 \leq t \leq 1$. It is clear to see that $h(0) = f_{L,\mathrm{P}}^*(x)$ and $h(1) = f_{L,\mathrm{P}}^*(\xi)$. Then by Lagrange's mean value theorem, there exists $0 \leq t_{x,\xi} \leq 1$ such that

$$f_{L,\mathrm{P}}^*(x) - f_{L,\mathrm{P}}^*(\xi) = h(1) - h(0) = h'(t_{x,\xi}) = \nabla f_{L,\mathrm{P}}^*((1 - t_{x,\xi})x + t_{x,\xi}\xi)^\top (\xi - x).$$

For the sake of notation simplicity, we write $\eta := \nabla f_{L,\mathrm{P}}^*((1 - t_{x,\xi})x + t_{x,\xi}\xi)$. Then, we obtain

$$\begin{aligned} \int_{A_p^j} \big( f_{L,\mathrm{P}}^*(\xi) - f_{L,\mathrm{P}}^*(x) \big)^2 \, dx &= \int_{A_p^j} \big( \eta^\top (\xi - x) \big)^2 \, dx = \int_{A_p^j} \bigg( \sum_{i=1}^{d} \eta_i (\xi_i - x_i) \bigg)^2 \, dx \\ &= \int_{A_p^j} \sum_{i=1}^{d} \big( \eta_i (\xi_i - x_i) \big)^2 \, dx + 2 \int_{A_p^j} \sum_{1 \leq i < l \leq d} \eta_i \eta_l (\xi_i - x_i)(\xi_l - x_l) \, dx := (I) + (II). \end{aligned} \tag{27}$$

Therefore, to derive the lower bound of (26), it suffices to calculate the upper bound and the lower bound of $(I)$ and $|(II)|$ respectively. Let us first consider the lower bound of $(I)$. Assume that $A_p^j := \times_{i=1}^d [a_i, b_i]$. Then, a simple calculation yields that

$$(I) = \sum_{i=1}^d \int_{A_p^j} \bigl(\eta_i(\xi_i - x_i)\bigr)^2 \, dx = \sum_{i=1}^d \eta_i^2 \int_{A_p^j} (\xi_i - x_i)^2 \, dx = \sum_{i=1}^d \eta_i^2 \int_{a_i}^{b_i} (\xi_i - x_i)^2 \, dx_i \cdot \prod_{l \neq i} (b_l - a_l). \tag{28}$$

Minimizing the following quadratic function with respect to $\xi_i$, we obtain

$$\int_{a_i}^{b_i} (\xi_i - x_i)^2 \, dx_i = \xi_i^2 \cdot \int_{a_i}^{b_i} dx_i - 2\xi_i \cdot \int_{a_i}^{b_i} x_i \, dx_i + \int_{a_i}^{b_i} x_i^2 \, dx_i \geq \frac{(b_i - a_i)^3}{12}.$$

This together with (28) and Lermma B.1 implies that

$$(I) \geq \frac{\mu\bigl(A_p^j\bigr)}{12} \sum_{i=1}^d \eta_i^2 (b_i - a_i)^2 \geq \frac{\mu\bigl(A_p^j\bigr) \cdot \|\eta\|_2^2}{12(4d-3)} \cdot \operatorname{diam}\bigl(A_p^j\bigr)^2. \tag{29}$$

Next, let us consider the upper bound of $(II)$. Assume that $A_p^j := \times_{i=1}^d [a_i, b_i]$. Then, a simple calculation yields that

$$\begin{aligned}
(II) &= 2 \sum_{1 \leq i < l \leq d} \int_{A_p^j} \eta_i \eta_l \cdot (\xi_i - x_i)(\xi_l - x_l) \, dx \\
&= 2 \sum_{1 \leq i < l \leq d} \eta_i \eta_l \cdot \int_{a_i}^{b_i} (\xi_i - x_i) \, dx_i \cdot \int_{a_l}^{b_l} (\xi_l - x_l) \, dx_l \cdot \prod_{k \neq i, l} (b_k - a_k) \\
&\leq 2 \sum_{1 \leq i < l \leq d} \eta_i \eta_l \cdot \int_{a_i}^{b_i} |\xi_i - x_i| \, dx_i \cdot \int_{a_l}^{b_l} |\xi_l - x_l| \, dx_l \cdot \prod_{k \neq i, l} (b_k - a_k) \\
&\leq \frac{\mu\bigl(A_p^j\bigr)}{8} \sum_{1 \leq i < l \leq d} \eta_i \eta_l (b_i - a_i)(b_l - a_l) \leq \frac{\mu\bigl(A_p^j\bigr) \cdot \operatorname{diam}\bigl(A_p^j\bigr)^2 \cdot \bigl(\|\eta\|_1^2 - \|\eta\|_2^2\bigr)}{16}.
\end{aligned} \tag{30}$$

Recall that $\eta = \nabla f_{L,\mathrm{P}}^*((1 - t_{x,\xi})x + t_{x,\xi}\xi)$. Therefore, the condition $\sqrt{(12d-7)/(12d-9)} \cdot \|\nabla f(x)\|_2 \geq \|\nabla f(x)\|_1$ implies that

$$\|\eta\|_1^2 - \|\eta\|_2^2/16 \leq \|\eta\|_2^2/(16d - 12).$$

This together with (29) and (30) yields that

$$(I) + (II) \geq (I) - |(II)| \geq \mu\bigl(A_p^j\bigr) \cdot \|\eta\|_2^2 \cdot \operatorname{diam}\bigl(A_p^j\bigr)^2 / \bigl(24(4d-3)\bigr).$$

On combining this with (27) and the condition $\|\eta\|_2^2 \geq \underline{c}_f^2$, we have

$$\int_{A_p^j} \bigl(f_{L,\mathrm{P}}^*(\xi) - f_{L,\mathrm{P}}^*(x)\bigr)^2 \, dx \geq \frac{\mu\bigl(A_p^j\bigr) \cdot \|\eta\|_2^2}{24(4d-3)} \cdot \operatorname{diam}\bigl(A_p^j\bigr)^2 \geq \frac{c_f^2 \cdot \mu\bigl(A_p^j\bigr)}{24(4d-3)} \cdot \operatorname{diam}\bigl(A_p^j\bigr)^2 \geq \frac{d \cdot c_f^2 \cdot \mu\bigl(A_p^j\bigr)}{96(4d-3) \cdot 2^{2p/d}},$$

where the last inequality follows from Lemma B.1. This together with (25) implies that

$$\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{P}}) - \mathcal{R}_{L,\mathrm{P}}^* \geq \sum_{j \in \mathcal{I}_p} \frac{d \cdot c_f^2 \cdot \mu\bigl(A_p^j\bigr)}{96(4d-3) \cdot 2^{2p/d}} = \frac{d \cdot c_f^2}{96(4d-3) \cdot 2^{2p/d}}.$$

This completes the proof. □

*Proof of Proposition A.2.* Let $\pi_p = \{A_p^j, 1 \leq j \leq 2^p\}$ be the random tree partition as in (2). Then for any fixed $1 \leq j \leq 2^p$, we define the random variable $Z_j$ by

$$Z_j := \sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i).$$

Since the random variables $\{\mathbf{1}_{A_p^j}(X_i)\}_{i=1}^n$ are i.i.d. Bernoulli distributed with parameter $\mathrm{P}(X \in A_p^j)$, elementary probability theory implies that the random variable $Z_j$ is Binomial distributed with parameters $n$ and $\mathrm{P}(X \in A_p^j)$. Therefore, for any $j \in \mathcal{I}_p$, we have

$$\mathbb{E}(Z_j) = n \cdot \mathrm{P}(X \in A_p^j).$$

Moreover, the RTR regressor $f_{\mathrm{D}}$ can be defined by

$$f_{\mathrm{D}}(x) = \begin{cases} \dfrac{\sum_{i=1}^n Y_i \mathbf{1}_{A_p^j}(X_i)}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)} \cdot \mathbf{1}_{A_p^j}(x) & \text{if } Z_j > 0, \\ 0 & \text{if } Z_j = 0. \end{cases}$$

By the law of total probability, we get

$$\begin{aligned} \mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X) - f_{\mathrm{P}}(X)\big)^2 &= \sum_{j \in \mathcal{I}_p} \mathbb{E}_{\mathrm{P}_X}\big(\big(f_{\mathrm{D}}(X) - f_{\mathrm{P}}(X)\big)^2 \big| X \in A_p^j\big) \cdot \mathrm{P}(X \in A_p^j) \\ &= \sum_{j \in \mathcal{I}_p} \mathbb{E}_{\mathrm{P}_X}\big(\big(f_{\mathrm{D}}(X) - f_{\mathrm{P}}(X)\big)^2 \big| X \in A_p^j, Z_j > 0\big) \cdot \mathrm{P}(Z_j > 0) \cdot \mathrm{P}(X \in A_p^j) \qquad (31) \\ &\quad + \sum_{j \in \mathcal{I}_p} \mathbb{E}_{\mathrm{P}_X}\big(\big(f_{\mathrm{D}}(X) - f_{\mathrm{P}}(X)\big)^2 \big| X \in A_p^j, Z_j = 0\big) \cdot \mathrm{P}(Z_j = 0) \cdot \mathrm{P}(X \in A_p^j). \qquad (32) \end{aligned}$$

For the term (31), we have

$$\begin{aligned} &\sum_{j \in \mathcal{I}_p} \mathbb{E}_{\mathrm{P}_X}\big((f_{\mathrm{D}}(X) - f_{\mathrm{P}}(X))^2 \big| X \in A_p^j, Z_j > 0\big) \cdot \mathrm{P}(Z_j > 0)\mathrm{P}(X \in A_p^j) \\ &= \sum_{j \in \mathcal{I}_p} \bigg( \frac{\sum_{i=1}^n Y_i \mathbf{1}_{A_p^j}(X_i)}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)} - \mathbb{E}(f_{L,\mathrm{P}}^*(X) | X \in A_p^j) \bigg)^2 \cdot \mathrm{P}(Z_j > 0)\mathrm{P}(X \in A_p^j) \\ &= \sum_{j \in \mathcal{I}_p} \bigg( \sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\big(Y_i - \mathbb{E}(f_{L,\mathrm{P}}^*(X) | X \in A_p^j)\big) \bigg)^2 \cdot \frac{\mathrm{P}(X \in A_p^j)}{(\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i))^2} \cdot \mathrm{P}(Z_j > 0), \end{aligned}$$

which yields that for a fixed $j \in \mathcal{I}_p$, there holds

$$\begin{aligned} &\mathbb{E}\bigg( \sum_{j \in \mathcal{I}_p} \bigg( \sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\big(Y_i - \mathbb{E}(f_{L,\mathrm{P}}^*(X) | X \in A_p^j)\big) \bigg)^2 \cdot \frac{\mathrm{P}(X \in A_p^j)}{(\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i))^2} \bigg| X_i \in A_p^j \bigg) \\ &= \sum_{j \in \mathcal{I}_p} \sum_{i=1}^n \mathbf{1}_{A_p^j}^2(X_i) \mathbb{E}\big(\big(Y - f_{\mathrm{P}}(X)\big)^2 \big| X \in A_p^j\big) \cdot \frac{\mathrm{P}(X \in A_p^j)}{(\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i))^2} \\ &= \sum_{j \in \mathcal{I}_p} \frac{\mathrm{P}(X \in A_p^j)}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)} \cdot \mathbb{E}\big((Y - f_{\mathrm{P}}(X))^2 \big| X \in A_p^j\big). \qquad (33) \end{aligned}$$

Obviously, for any fixed $j \in \mathcal{I}_p$, there holds

$$\mathbb{E}(f_{\mathrm{P}}(X) | X \in A_p^j) = \mathbb{E}(f_{L,\mathrm{P}}^*(X) | X \in A_p^j)$$

and consequently we obtain

$$\begin{aligned} \mathbb{E}\big((Y - f_{\mathrm{P}}(X))^2 \big| X \in A_p^j\big) &= \mathbb{E}\big((Y - f_{L,\mathrm{P}}^*(X))^2 \big| X \in A_p^j\big) + \mathbb{E}\big((f_{L,\mathrm{P}}^*(X) - f_{\mathrm{P}}(X))^2 \big| X \in A_p^j\big) \\ &= \sigma^2 + \mathbb{E}\big((f_{L,\mathrm{P}}^*(X) - f_{\mathrm{P}}(X))^2 \big| X \in A_p^j\big). \end{aligned}$$

Taking expectation over both sides of (33) with respect to $\mathrm{P}^n$, we get

$$\begin{aligned}
\mathbb{E}_{D\sim \mathrm{P}^n}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X)\big)^2 &= \mathbb{E}_{D\sim \mathrm{P}^n}\big(\mathbb{E}\big(\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X)\big)^2\big|X_i\in A_p^j\big)\big)\\
&= \sum_{j\in\mathcal{I}_p}\Bigg(\mathrm{P}(X\in A_p^j)\mathbb{E}_{D\sim\mathrm{P}^n}\Bigg(\Bigg(\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\Bigg)^{-1}\Bigg|Z_j>0\Bigg)\Bigg)\\
&\qquad\cdot\big(\sigma^2+\mathbb{E}(f^*_{L,\mathrm{P}}(X)-f_{\mathrm{P}}(X))^2\big)\cdot\mathrm{P}(Z_j>0)\\
&= \big(\sigma^2+\mathbb{E}(f^*_{L,\mathrm{P}}(X)-f_{\mathrm{P}}(X))^2\big)\cdot\sum_{j\in\mathcal{I}_p}\big(\mathrm{P}(X\in A_p^j)\mathbb{E}_{D\sim\mathrm{P}^n}(Z_j^{-1}|Z_j>0)\big)\mathrm{P}(Z_j>0)\\
&= n^{-1}\big(\sigma^2+\mathbb{E}(f^*_{L,\mathrm{P}}(X)-f_{\mathrm{P}}(X))^2\big)\cdot\sum_{j\in\mathcal{I}_p}\big(\mathbb{E}(Z_j)\cdot\mathbb{E}(Z_j^{-1}|Z_j>0)\big)\mathrm{P}(Z_j>0).
\end{aligned}$$

Clearly, $x^{-1}$ is convex for $x>0$. Therefore, by Jensen's inequality, we get

$$\begin{aligned}
\mathbb{E}(Z_j)\cdot\mathbb{E}(Z_j^{-1}|Z>0)\mathrm{P}(Z_j>0) &\geq \mathbb{E}(Z_j)\cdot\mathbb{E}(Z_j|Z_j>0)^{-1}\mathrm{P}(Z_j>0) = \mathbb{E}(Z)\cdot\mathbb{E}(Z\mathbf{1}_{\{Z>0\}})^{-1}\mathrm{P}(Z>0)\mathrm{P}(Z>0)\\
&= \mathrm{P}(Z>0)^2 = (1-\mathrm{P}(Z=0))^2 = \big(1-(1-\mathrm{P}(X\in A_p^j))^n\big)^2 \geq 1-2e^{-n\mathrm{P}(X\in A_p^j)},
\end{aligned}$$

where the last inequality follows from $(1-x)^n\leq e^{-nx}$, $x\in(0,1)$.

We now turn to estimate the term (32). By the definition of $f_{\mathrm{D}}$, we have

$$\begin{aligned}
&\sum_{j\in\mathcal{I}_p}\mathbb{E}_{\mathrm{P}_X}\big((f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X))^2\big|X\in A_p^j, Z_j=0\big)\cdot\mathrm{P}(Z_j=0)\cdot\mathrm{P}(X\in A_p^j)\\
&= \sum_{j\in\mathcal{I}_p}\mathbb{E}_{\mathrm{P}_X}\big(\big(f_{\mathrm{P}}(X)\big)^2\big|X\in A_p^j\big)\cdot\mathrm{P}(Z_j=0)\cdot\mathrm{P}(X\in A_p^j)\geq 0.
\end{aligned}$$

Then we obviously have $\mathrm{P}(X\in A_p^j)=\mu\big(A_p^j\big)\geq 2^{-p-d}$ for all $j\in\mathcal{I}_p$. Combing the above results, we obtain

$$\begin{aligned}
\mathbb{E}_{D\sim\mathrm{P}^n}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X)\big)^2 &= \sum_{j\in\mathcal{I}_p}\mathbb{E}_{\mathrm{P}_X}\big((f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X))^2|X\in A_p^j, Z_j>0\big)\cdot\mathrm{P}(Z_j>0)\cdot\mathrm{P}(X\in A_p^j)\\
&\quad+\sum_{j\in\mathcal{I}_p}\mathbb{E}_{\mathrm{P}_X}\big((f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X))^2\big|X\in A_p^j, Z_j=0\big)\cdot\mathrm{P}(Z_j=0)\cdot\mathrm{P}(X\in A_p^j)\\
&\geq \sum_{j\in\mathcal{I}_p}\mathbb{E}_{\mathrm{P}_X}\big((f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X))^2\big|X\in A_p^j, Z_j>0\big)\cdot\mathrm{P}(Z_j>0)\cdot\mathrm{P}(X\in A_p^j)\\
&\geq \frac{1}{n}\sum_{j\in\mathcal{I}_p}\big(1-2e^{-n\mathrm{P}(X\in A_p^j)}\big)\cdot\big(\mathbb{E}(f^*_{L,\mathrm{P}}(X)-f_{\mathrm{P}}(X))^2+\sigma^2\big)\\
&\geq \frac{\sigma^2}{n}\Bigg(|\mathcal{I}_p|-\sum_{j\in\mathcal{I}_p}2e^{-n\mathrm{P}(X\in A_p^j)}\Bigg).
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\mathbb{E}_{D\sim\mathrm{P}^n}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f_{\mathrm{P}}(X)\big)^2 &\geq \frac{\sigma^2}{n}\Bigg(|\mathcal{I}_p|-\sum_{j\in\mathcal{I}_p}2e^{-n\mathrm{P}(X\in A_p^j)}\Bigg) = \frac{\sigma^2}{n}\Big(|\mathcal{I}_p|-2|\mathcal{I}_p|\exp\big(-n2^{-p-d}\big)\Big)\\
&\geq \sigma^2\cdot 2^p(1-2e^{-1})\cdot n^{-1}. \qquad (34)
\end{aligned}$$

Taking expectation with respect to $\mathrm{P}_Z$, we obtain the desired assertion. □

### B.2.2. Proofs Related to Section 3.2

*Proof of Theorem 3.2.* By Proposition A.1 and A.2, the error decomposition in (20) tells us that

$$\begin{aligned}
\mathbb{E}_{\mathrm{P}_Z}\big(\mathcal{R}_{L,\mathrm{P}}(f_{\mathrm{D}})-\mathcal{R}^*_{L,\mathrm{P}}\big) &= \mathbb{E}_{\mathrm{P}_Z}\mathbb{E}_{\mathrm{P}_X}\big(f_{\mathrm{D}}(X)-f^*_{L,\mathrm{P}}(x)\big)^2\\
&\geq d\cdot c_f^2(384d-288)^{-1}\cdot 2^{-2p/d}+\sigma^2\cdot 2^p(1-2e^{-1})\cdot n^{-1}\gtrsim n^{-\frac{2}{2+d}},
\end{aligned}$$

where the last inequality holds if and only if $2^p\asymp n^{d/(2+d)}$. This yields the desired assertion. □

### B.2.3. Proofs Related to Section A.1.2

*proof of Proposition A.4.* By the definition of $f_{\mathrm{P}}$, we have

$$|f_{\mathrm{P}}(x) - f^*_{L,\mathrm{P}}(x)| = \left| \frac{\int_{A(x)} f^*_{L,\mathrm{P}}(x') d\mathrm{P}_X(x')}{\int_{A(x)} d\mathrm{P}_X(x')} - f^*(x) \right| \leq \frac{\int_{A(x)} |f^*_{L,\mathrm{P}}(x') - f^*_{L,\mathrm{P}}(x)|\, d\mathrm{P}_X(x')}{\int_{A(x)} d\mathrm{P}_X(x')}$$

Since $f^*_{L,\mathrm{P}}(x) \in C^{k,\alpha}$, we have $|f^*_{L,\mathrm{P}}(x') - f^*_{L,\mathrm{P}}(x)| \leq c_L \|x' - x\|^{(k+\alpha)\wedge 1}$. Then, by Lemma B.1, we get $|f^*_{L,\mathrm{P}}(x') - f^*_{L,\mathrm{P}}(x)| \leq c_L \mathrm{diam}(A(x))^{(k+\alpha)\wedge 1}$. Consequently, we have

$$|f_{\mathrm{P}}(x) - f^*_{L,\mathrm{P}}(x)| \leq \frac{c_L \mathrm{diam}(A(x))^{(k+\alpha)\wedge 1} \int_{A(x)} \mathrm{P}_X(x')}{\int_{A(x)} \mathrm{P}_X(x')} \leq c_L \mathrm{diam}(A(x))^{(k+\alpha)\wedge 1}.$$

This together with Lemma B.1 yields the desired assertion. □

To conduct our analysis, we first need to recall the definitions of *VC dimension* (*VC index*) and *covering number*, which are frequently used in capacity-involved arguments and measure the complexity of the underlying function class (van der Vaart & Wellner, 1996; Kosorok, 2008; Giné & Nickl, 2021).

**Definition B.2** (VC dimension)**.** Let $\mathcal{B}$ be a class of subsets of $\mathcal{X}$ and $A \subset \mathcal{X}$ be a finite set. The trace of $\mathcal{B}$ on $A$ is defined by $\{B \cap A : B \subset \mathcal{B}\}$. Its cardinality is denoted by $\Delta^{\mathcal{B}}(A)$. We say that $\mathcal{B}$ shatters $A$ if $\Delta^{\mathcal{B}}(A) = 2^{\#(A)}$, that is, if for every $A' \subset A$, there exists a $B \subset \mathcal{B}$ such that $A' = B \cap A$. For $n \in \mathrm{N}$, let

$$m^{\mathcal{B}}(n) := \sup_{A \subset \mathcal{X},\, \#(A) = n} \Delta^{\mathcal{B}}(A). \tag{35}$$

Then, the set $\mathcal{B}$ is a Vapnik-Chervonenkis class if there exists $n < \infty$ such that $m^{\mathcal{B}}(n) < 2^n$ and the minimal of such $n$ is called the VC dimension of $\mathcal{B}$, and abbreviate as $\mathrm{VC}(\mathcal{B})$.

Since an arbitrary set of $n$ points $\{x_1, \ldots, x_n\}$ possess $2^n$ subsets, we say that $\mathcal{B}$ *picks out* a certain subset from $\{x_1, \ldots, x_n\}$ if this can be formed as a set of the form $B \cap \{x_1, \ldots, x_n\}$ for a $B \in \mathcal{B}$. The collection $\mathcal{B}$ *shatters* $\{x_1, \ldots, x_n\}$ if each of its $2^n$ subsets can be picked out in this manner. From Definition B.2 we see that the VC dimension of the class $\mathcal{B}$ is the smallest $n$ for which no set of size $n$ is shattered by $\mathcal{B}$, that is,

$$\mathrm{VC}(\mathcal{B}) = \inf\Big\{ n : \max_{x_1, \ldots, x_n} \Delta^{\mathcal{B}}(\{x_1, \ldots, x_n\}) \leq 2^n \Big\},$$

where $\Delta^{\mathcal{B}}(\{x_1, \ldots, x_n\}) = \#\{B \cap \{x_1, \ldots, x_n\} : B \in \mathcal{B}\}$. Clearly, the more refined $\mathcal{B}$ is, the larger is its index.

To further bound the capacity of the function sets, we need to introduce the following fundamental descriptions of *covering number* which enables an approximation of an infinite set by finite subsets.

**Definition B.3** (Covering Number)**.** Let $(\mathcal{X}, d)$ be a metric space and $A \subset \mathcal{X}$. For $\varepsilon > 0$, the $\varepsilon$-covering number of $A$ is denoted as

$$\mathcal{N}(A, d, \varepsilon) := \min\Big\{ n \geq 1 : \exists x_1, \ldots, x_n \in \mathcal{X} \text{ such that } A \subset \bigcup_{i=1}^n B(x_i, \varepsilon) \Big\},$$

where $B(x, \varepsilon) := \{x' \in \mathcal{X} : d(x, x') \leq \varepsilon\}$.

To prove Lemma B.4, we need the following fundamental lemma concerning with the VC dimension of random partitions in Section 2.2, which follows the idea put forward by Gao & Zhou (2020) of the construction of random forest. To this end, let $p \in \mathbb{N}$ be fixed and $\pi_p$ be a partition of $\mathcal{X}$ with number of splits $p$ and $\pi_{(p)}$ denote the collection of all partitions $\pi_p$.

**Lemma B.4.** *Let $\tilde{A}$ be the collection of all cells $\times_{i=1}^d [a_i, b_i]$ in $\mathbb{R}^d$. The VC index of $\tilde{A}$ equals $2d + 1$. Moreover, for all $0 < \varepsilon < 1$, there exists a universal constant $C$ such that*

$$\mathcal{N}(\mathbf{1}_{\tilde{A}}, \|\cdot\|_{L_1(Q)}, \varepsilon) \leq C(2d+1)(4e)^{2d+1}(1/\varepsilon)^{2d}.$$

*Proof of Lemma B.4.* The first result of VC index follows from Example 2.6.1 in van der Vaart & Wellner (1996). The second result of covering number follows directly from Theorem 9.2 in Kosorok (2008). □

Before we proceed, we list the well-known Bernstein's inequality that will be used frequently in the proofs. Lemma B.5 was introduced in Bernstein (1946) and can be found in many statistical learning textbooks, see e.g., Massart (2007); Cucker & Zhou (2007); Steinwart & Christmann (2008).

**Lemma B.5** (Bernstein's inequality)**.** *Let $B > 0$ and $\sigma > 0$ be real numbers, and $n \geq 1$ be an integer. Furthermore, let $\xi_1, \dots, \xi_n$ be independent random variables satisfying $\mathbb{E}_P \xi_i = 0$, $\|\xi_i\|_\infty \leq B$, and $\mathbb{E}_P {\xi_i}^2 \leq \sigma^2$ for all $i = 1, \dots, n$. Then for all $\tau > 0$, we have*

$$P\biggl(\frac{1}{n}\sum_{i=1}^n \xi_i \geq \sqrt{\frac{2\sigma^2\tau}{n}} + \frac{2B\tau}{3n}\biggr) \leq e^{-\tau}.$$

*Proof of Proposition A.5.* Let $\pi_p = \{A_p^j, j \in \mathcal{I}_p\}$ be the random tree partition of $[0,1]^d$ as in (2). Then by the definition of $f_{\mathrm{P}}(x)$ and $f_{\mathrm{D}}(x)$ and the triangle inequality, we have

$$\begin{aligned}
\|f_{\mathrm{P}} - f_{\mathrm{D}}\|_\infty &= \sup_{j\in\mathcal{I}_p} \biggl| \frac{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)} - \frac{\int_{A_p^j} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')}{\int_{A_p^j} d\mathrm{P}_X(x')} \biggr| \\
&= \sup_{j\in\mathcal{I}_p} \frac{\bigl|\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i \cdot \int_{A_p^j} d\mathrm{P}_X(x') - \sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\cdot \int_{A_p^j} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\bigr|}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\cdot\int_{A_p^j} d\mathrm{P}_X(x')} \\
&\leq \sup_{j\in\mathcal{I}_p} \frac{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\cdot \bigl|\int_{A_p^j} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x') - \sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i\bigr|}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\cdot\int_{A_p^j} d\mathrm{P}_X(x')} \\
&\quad + \sup_{j\in\mathcal{I}_p} \frac{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i\cdot \bigl|\int_{A_p^j} d\mathrm{P}_X(x') - \sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\bigr|}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)\cdot\int_{A_p^j} d\mathrm{P}_X(x')}.
\end{aligned}$$

For the notation simplicity, we write

$$(I) := \sup_{j\in\mathcal{I}_p}\biggl|\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i) - \int_{A_p^j} d\mathrm{P}_X(x')\biggr|, \quad (II) := \sup_{j\in\mathcal{I}_p}\biggl|\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i - \int_{A_p^j} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\biggr|$$

Then the estimation error in $L_\infty$-norm is bounded by

$$\|f_{\mathrm{P}} - f_{\mathrm{D}}\|_\infty \lesssim (I)\cdot 2^p \cdot \sup_{j\in\mathcal{I}_p} \cdot \frac{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i}{\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)} + (II)\cdot 2^p.$$

Since $\mathcal{Y} \subset [-M, M]$, we have $|Y_i| \leq M$ for $1 \leq i \leq n$. Consequently, there holds

$$\|f_{\mathrm{P}} - f_{\mathrm{D}}\|_\infty \lesssim \bigl(M\cdot(I) + (II)\bigr)\cdot 2^p. \tag{36}$$

Therefore, it suffice to bound $(I)$ and $(II)$ respectively. Let us first consider the term $(I)$. Let $\tilde{\mathcal{A}}$ be the collection of all cells $\times_{i=1}^d [a_i, b_i]$ in $\mathbb{R}^d$. Applying Lemma B.4 with $Q := (\mathrm{D}_X + \mathrm{P}_X)/2$, there exists an $\varepsilon$-net $\{\tilde{A}_k\}_{k=1}^K \subset \tilde{\mathcal{A}}$ with

$$K \leq C(2d+1)(4e)^{2d+1}(1/\varepsilon)^{2d} \tag{37}$$

such that for any $j \in \mathcal{I}_p$, there exist some $k \in \{1, \dots, K\}$ such that

$$\|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1((\mathrm{D}_X+\mathrm{P}_X)/2)} \leq \varepsilon,$$

Since $\|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1((\mathrm{D}_X+\mathrm{P}_X)/2)} = 1/2\cdot\|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{D}_X)} + 1/2\cdot\|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{P}_X)}$, we get

$$\|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{D}_X)} \leq 2\varepsilon, \quad \|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{P}_X)} \leq 2\varepsilon. \tag{38}$$

Consequently, by the definition of the covering number and the triangle inequality, for any $j \in \mathcal{I}_p$, there holds

$$\begin{aligned}\biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i) - \mathrm{P}_X\bigl(A_p^j\bigr)\biggr| &\leq \biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)\biggr| + \|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{D}_X)} + \|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{P}_X)} \\ &\leq \biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)\biggr| + 4\varepsilon.\end{aligned}$$

Therefore, we get

$$(I) = \sup_{j \in \mathcal{I}_p}\biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i) - \mathrm{P}_X\bigl(A_p^j\bigr)\biggr| \leq \sup_{1\leq k\leq K}\biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)\biggr| + 4\varepsilon. \tag{39}$$

For any fixed $1 \leq k \leq K$, let the random variable $\xi_i$ be defined by $\xi_i := \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)$. Then we have $\mathbb{E}_{\mathrm{P}_X}\xi_i = 0$, $\|\xi\|_\infty \leq 1$, and $\mathbb{E}_{\mathrm{P}_X}\xi_i^2 \leq \mathrm{P}_X(\tilde{A}_k)$. Since $\mathrm{P}_X$ has upper and lower bounded density over $[0,1]^d$, there holds $\mathbb{E}_{\mathrm{P}_X}\xi_i^2 \lesssim 2^{-p}$. Applying Bernstein's inequality in Lemma B.5, we obtain

$$\biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)\biggr| \lesssim \sqrt{\frac{2^{1-p}\cdot\tau}{n}} + \frac{2\tau\log n}{3n}$$

with probability $\mathrm{P}^n$ at least $1 - 2e^{-\tau}$. Then the union bound together with the covering number estimate (37) implies that

$$\sup_{1\leq k\leq K}\biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)\biggr| \lesssim \sqrt{\frac{2^{1-p}(\tau + \log(2K))}{n}} + \frac{2(\tau + \log(2K))\log n}{3n}$$

with probability $\mathrm{P}^n$ at least $1 - e^{-\tau}$. Let $\tau = 2\log n$ and $\varepsilon = 1/n$. Then for any $n > N_1 := (2C) \wedge (2d+1) \wedge (4e)$, we have $\tau + \log(2K) = 2\log n + \log(2C) + \log(2d+1) + (2d+1)\log(4e) + 2d\log n \leq (4d+5)\log n$. Therefore, we have

$$\sup_{1\leq k\leq K}\biggl|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i) - \mathrm{P}_X(\tilde{A}_k)\biggr| \lesssim \sqrt{\frac{2^{1-p}(4d+5)\log n}{n}} + \frac{2(4d+5)\log n}{3n} \tag{40}$$

with probability $\mathrm{P}^n$ at least $1 - 1/n^2$. This together with (39) yields that

$$(I) \lesssim \sqrt{\frac{2^{1-p}(4d+5)\log n}{n}} + \frac{2(4d+5)\log n}{3n} + \frac{4}{n}. \tag{41}$$

Next, let us consider the term $(II)$. Let $\tilde{\mathcal{A}}$ be the collection of all cells $\times_{i=1}^d[a_i, b_i]$ in $\mathbb{R}^d$. Then there exists an $\varepsilon$-net $\{\tilde{A}_k\}_{k=1}^K \subset \tilde{\mathcal{A}}$ with $K$ bounded by (37) such that for any $j \in \mathcal{I}_p$, (38) holds for some $k \in \{1, \ldots, K\}$. Consequently, by the definition of the covering number and the triangle inequality, for any $j \in \mathcal{I}_p$, there holds

$$\begin{aligned}\biggl|\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i - \int_{A_p^j} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\biggr| &\leq \biggl|\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i)Y_i - \int_{\tilde{A}_k} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\biggr| \\ &\quad + \int_{\mathbb{R}^d}\bigl|\mathbf{1}_{A_p^j}(x') - \mathbf{1}_{\tilde{A}_k}(x')\bigr|\bigl|f_{L,\mathrm{P}}^*(x')\bigr|d\mathrm{P}_X(x') + \sum_{i=1}^n\bigl|\mathbf{1}_{\tilde{A}_k}(X_i) - \mathbf{1}_{A_p^j}(X_i)\bigr|\bigl|Y_i\bigr|.\end{aligned}$$

Consequently, we have

$$\begin{aligned}&\biggl|\sum_{i=1}^n \mathbf{1}_{A_p^j}(X_i)Y_i - \int_{A_p^j} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\biggr| \\ &\leq \biggl|\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i)Y_i - \int_{\tilde{A}_k} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\biggr| + \max_{1\leq i\leq n}|Y_i| \cdot \|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{D}_X)} + \|f_{L,\mathrm{P}}^*\|_\infty \cdot \|\mathbf{1}_{A_p^j} - \mathbf{1}_{\tilde{A}_k}\|_{L_1(\mathrm{P}_X)} \\ &\leq \biggl|\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i)Y_i - \int_{\tilde{A}_k} f_{L,\mathrm{P}}^*(x')d\mathrm{P}_X(x')\biggr| + 4M\varepsilon.\end{aligned} \tag{42}$$

where the last inequality follow from the condition $\mathcal{Y} \subset [-M, M]$.

For any fixed $1 \leq k \leq K$, let the random variable $\tilde{\xi}_i$ be defined by $\tilde{\xi}_i := \mathbf{1}_{\tilde{A}_k}(X_i)Y_i - \int_{\tilde{A}_k} f^*_{L,\mathrm{P}}(x')\, d\mathrm{P}_X(x')$. Then we have $\mathbb{E}_{\mathrm{P}}\tilde{\xi}_i = 0$, $\|\xi\|_\infty \leq 1$, and $\mathbb{E}_{\mathrm{P}}\tilde{\xi}_i^2 \leq M^2 \mathrm{P}_X(\tilde{A}_k)$. Since $\mathrm{P}_X$ has upper and lower bounded density over $[0,1]^d$, there holds $\mathbb{E}_{\mathrm{P}}\tilde{\xi}_i^2 \lesssim M^2 \cdot 2^{-p}$. Applying Bernstein's inequality in Lemma B.5, we obtain

$$\left|\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i)Y_i - \int_{\tilde{A}_k} f^*_{L,\mathrm{P}}(x')d\mathrm{P}_X(x')\right| \lesssim \sqrt{\frac{M^2 \cdot 2^{1-p} \cdot \tau}{n}} + \frac{2M\tau \log n}{3n}$$

with probability $\mathrm{P}^n$ at least $1 - 2e^{-\tau}$. Similar to (40), one can show that for any $n \geq N_1$, there holds

$$\sup_{1 \leq k \leq K}\left|\sum_{i=1}^n \mathbf{1}_{\tilde{A}_k}(X_i)Y_i - \int_{\tilde{A}_k} f^*_{L,\mathrm{P}}(x')d\mathrm{P}_X(x')\right| \lesssim M\sqrt{\frac{\cdot 2^{1-p} \cdot \tau}{n}} + \frac{2M\tau \log n}{3n}$$

with probability $\mathrm{P}^n$ at least $1 - 1/n^2$. This together with (42) yields that

$$(II) \lesssim M\sqrt{\frac{2^{1-p}(4d+5)\log n}{n}} + \frac{2M(4d+5)\log n}{3n} + \frac{4M}{n}. \tag{43}$$

On combining (41) and (43) with (36), we get

$$\|f_{\mathrm{P}} - f_{\mathrm{D}}\|_\infty \lesssim 2M\sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} \cdot M(4d+5)\log n}{3n} + \frac{2^{p+3} \cdot M}{n}.$$

This completes the proof of Proposition A.5. □

*Proof of Theorem A.3.* By Proposition A.4 and A.5, the triangle inequality tells us that

$$\begin{aligned}
&\|f_{\mathrm{D}} - f^*_{L,\mathrm{P}}\|_\infty \leq \|f_{\mathrm{D}} - f_{\mathrm{P}}\|_\infty + \|f_{\mathrm{P}} - f^*_{L,\mathrm{P}}\|_\infty \\
&\lesssim c_L(2\sqrt{d})^{(k+\alpha)\wedge 1}2^{-(p(k+\alpha)\wedge 1)/d} + 2M\sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} \cdot M(4d+5)\log n}{3n} + \frac{2^{p+3} \cdot M}{n}
\end{aligned}$$

holds with probability $\mathrm{P}^n \otimes \mathrm{P}_Z$ at least $1 - 2/n^2$. By choosing $2^p \asymp (n/\log n)^{d/(2((k+\alpha)\wedge 1)+d)}$, i.e. $p \asymp \log(n/\log n)$, we obtain the desired assertion. □

### B.3. Proofs of the Results for ERTR

#### B.3.1. Proofs Related to Section 2.4

**Lemma B.6.** *Suppose that $g(x) : [0,1] \to \mathbb{R} \in C^{k,\alpha}$ with the constant $c_L$. Let $g^{(i)}(x)$ be the i-th order derivative of $g(x)$. Then for $0 \leq r \leq 1$, we have*

$$\left|g(r) - \sum_{i=1}^k \frac{g^{(i)}(0)r^i}{i!}\right| \leq \frac{c_L}{k!}.$$

*Proof of Lemma B.6.* Let

$$h(x) = g(x) - \sum_{i=1}^{k-1} \frac{g^{(i)}(0)}{i!}x^i - \frac{x^k}{r^k}\left(g(r) - \sum_{i=1}^{k-1} \frac{g^{(i)}(0)}{i!}r^i\right).$$

A simple calculation yields that $h(0) = h(r) = 0$. Therefore, Rolle's theorem tells us that there exists an $0 \leq \xi_1 \leq r$ such that $h^{(1)}(\xi_1) = 0$. Note that $h^{(i)}(0) = 0$ for $1 \leq i \leq k-1$, applying Rolle's theorem $k-1$ times again implies that there exists an $\xi_k \in [0, r]$ such that $h^{(k)}(\xi_k) = 0$. This yields that

$$g(r) - \sum_{i=1}^{k-1} \frac{g^{(i)}(0)}{i!}r^i = \frac{r^k g^{(k)}(\xi_k)}{k!}.$$

Consequently, we have

$$\left|g(r)-\sum_{i=1}^{k}\frac{g^{(i)}(0)r^i}{i!}\right| = \left|g(r)-\sum_{i=1}^{k-1}\frac{g^{(i)}(0)}{i!}r^i+\sum_{i=1}^{k-1}\frac{g^{(i)}(0)}{i!}r^i-\sum_{i=1}^{k}\frac{g^{(i)}(0)r^i}{i!}\right|$$
$$= \frac{r^k|g^{(k)}(\xi_k)-g^{(k)}(0)|}{k!} \leq \frac{c_L r^k \xi_k^\alpha}{k!} \leq \frac{c_L}{k!}.$$

where the last inequality follows from $\|g^{(k)}(x)-g^{(k)}(x')\| \leq c_L\|x-x'\|^\alpha$ and $\xi_k \leq r \leq 1$. This completes the proof. □

*Proof of Proposition 2.4.* From the definition of $f_{\mathrm{P},r}(x)$ in (9), we get

$$f_{\mathrm{P},r}(x) = \frac{\int_{A_r(x)} f_{L,\mathrm{P}}^*(x')\,d\mathrm{P}_X(x')}{\int_{A_r(x)} d\mathrm{P}_X(x')}.$$

Since $\mathrm{P}_X$ has the upper and lower bounded density over $[0,1]^d$, we have

$$f_{\mathrm{P},r}(x) = \frac{\int_{A_r(x)} f_{L,\mathrm{P}}^*(x')f_X(x')\,dx'}{\int_{A_r(x)} f_X(x')\,dx'},$$

where $f_X(x)$ represents the density function of $\mathrm{P}_X$. From (6), we have $x' = x + r(z-x)$ with $z \in A(x)$. Therefore, by the substitution $x' = x + r(z-x)$, we get

$$f_{\mathrm{P},r}(x) = \frac{\int_{A(x)} f_{L,\mathrm{P}}^*(x+r(z-x))\cdot f_X(x+r(z-x))\,dz}{\int_{A(x)} f_X(x+r(z-x))\,dz}. \tag{44}$$

For fixed $x, z \in \mathbb{R}^d$, we define $g : [0,1] \to \mathbb{R}$ by $g(r) = f_{L,\mathrm{P}}^*(x+r(z-x))$. Then we have $g(r) \in C^{k,\alpha}$ since $f_{L,\mathrm{P}}^*(x) \in C^{k,\alpha}$. Moreover, for $0 \leq \ell \leq k$, the $l$-th order derivative of $g(r)$ at $r=0$ is

$$g^{(j)}(0) = \sum_{i_1+\cdots+i_d=j}\frac{j!\cdot\partial^j f_{L,\mathrm{P}}^*(x)}{\partial x_1^{i_1}\cdots\partial x_d^{i_d}}\prod_{\ell=1}^{d}\frac{(z_\ell-x_\ell)^{i_\ell}}{i_\ell!} \leq d_A^j(x)\cdot\sum_{i_1+\cdots+i_d=\ell}\frac{j!\cdot\partial^\ell f_{L,\mathrm{P}}^*(x)}{\partial x_1^{i_1}\cdots\partial x_d^{i_d}} \leq c_L j!\cdot(j+1)^d d_A^\ell(x) < \infty.$$

where the last inequality follows from $\|\nabla^\ell f\| \leq c_L$. Furthermore, for $0 \leq r \leq r' \leq 1$, we have

$$|g^{(k)}(r)-g^{(k)}(r')| \leq c_L(k+1)!\cdot(k+1)^d d_A^{k+\alpha}(x)|r-r'|^\alpha.$$

Consequently, by Lemma B.6, we get

$$\left|g(r)-\sum_{j=0}^{k}\frac{g^{(j)}(0)r^j}{j!}\right| \leq c_L(k+1)^{d+1}d_A^{k+\alpha}(x).$$

Therefore, we have

$$\left|f_{L,\mathrm{P}}^*(x+r(z-x))\cdot f_X(x+r(z-x))-\sum_{j=0}^{k}b_j' r^j\cdot f_X(x+r(z-x))\right| \leq c_L(k+1)^{d+1}d_A^{k+\alpha}(x)\cdot f_X(x+r(z-x)).$$

with $b_j'$ expressed as

$$b_j' = \sum_{i_1+\cdots+i_d=j}\frac{\partial^j f_{L,\mathrm{P}}^*(x)}{\partial x_1^{i_1}\cdots\partial x_d^{i_d}}\prod_{\ell=1}^{d}\frac{(z_\ell-x_\ell)^{i_\ell}}{i_\ell!}.$$

Consequently, we get

$$\begin{aligned}
&\left| \int_{A(x)} f_{L,\mathrm{P}}^{*}(x + r(z-x)) \cdot f_X(x + r(z-x))\, dz - \sum_{j=0}^{k} r^j \cdot \int_{A(x)} b_j' f_X(x + r(z-x))\, dz \right| \\
&\leq \int_{A(x)} c_L (k+1)^{d+1} d_A^{k+\alpha}(x) \cdot f_X(x + r(z-x))\, dz \\
&\leq c_L (k+1)^{d+1} d_A^{k+\alpha}(x) \cdot \int_{A(x)} f_X(x + r(z-x))\, dz
\end{aligned} \tag{45}$$

On combining (44) with (45), we get

$$\left| f_{\mathrm{P},r}(x) - \sum_{j=0}^{k} b_j r^j \right| \leq c_L (k+1)^{d+1} d_A^{k+\alpha}(x).$$

with

$$b_j := \frac{\int_{A(x)} b_j' f_X(x + r(z-x))\, dz}{\int_{A(x)} f_X(x + r(z-x))\, dz}.$$

This completes the proof. □

### B.3.2. Proofs Related to Section A.2

**Lemma B.7.** *Let $r_i = i/V$ for $i = 1, \cdots, V$ for $V > 0$. For $V \geq L+1$, let $R$ be a $V \times (L+1)$ matrix whose $i, j$-th entry is $r_i^{j-1}$. Then we have*

$$\|e_1^\top (R^\top R)^{-1} R^\top\|_1 \leq c_{V,L}$$

*for some constant $c_{V,L}$ depending only on $V$ and $L$.*

*Proof of Lemma B.7.* Note that $R$ is a Vandermonde matrix (Horn & Johnson, 2012). Then, for $V \geq L+1$ and $r_i \neq r_j$ for any $i \neq j$, $R$ has rank $L+1$ and its eigenvalues are strictly positive. Thus, the operator norm of $\|(R^\top R)^{-1}\|_2$ can be bounded by some constant $c_{V,L}'$ depending only on $V$ and $L$. Then, there holds

$$\|e_1^\top (R^\top R)^{-1} R^\top\|_1 \leq \sqrt{V} \|e_1^\top (R^\top R)^{-1} R^\top\|_2 = \sqrt{V} \sqrt{e_1^\top (R^\top R)^{-1} e_1} \leq \sqrt{V c_{V,L}'} := c_{V,L}.$$

This completes the proof. □

*Proof of Proposition A.6.* By Proposition 2.4, there exists $b_1, \cdots, b_k$ such that

$$\begin{pmatrix} f_{P,r_1} \\ f_{P,r_2} \\ \vdots \\ f_{P,r_V} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & r_1^1 & \cdots & r_1^L \\ 1 & r_2^1 & \cdots & r_2^L \\ \vdots & \vdots & \ddots & \vdots \\ 1 & , r_V^1 & \cdots & r_V^L \end{pmatrix}}_{R} \underbrace{\begin{pmatrix} f_{L,\mathrm{P}}^*(x) \\ b_1 \\ \vdots \\ b_V \end{pmatrix}}_{b} + \underbrace{\begin{pmatrix} \delta_{r_1,A} \\ \delta_{r_2,A} \\ \vdots \\ \delta_{r_V,A} \end{pmatrix}}_{\delta} \tag{46}$$

where $\delta_{r_i,A} \leq d_A^{k+\alpha}$ for $i = 1, \cdots, V$. For notation simplicity, let $b$ denote the vector $(f_{L,\mathrm{P}}^*(x), b_1, \cdots, b_V)^\top$ and $\delta$ denote the vector $(\delta_{r_1,A}, \cdots, \delta_{r_V,A})^\top$. Combining (46) and (23), we have

$$f_{\mathrm{P,E}}(x) - f^*(x) = e_1^\top (R^\top R)^{-1} R^\top \, (R\, b + \delta) - f_{L,\mathrm{P}}^*(x) \leq e_1^\top (R^\top R)^{-1} R^\top \delta.$$

Then, by Lemma B.7, there holds

$$|f_{\mathrm{P,E}}(x) - f^*(x)| \leq \|e_1^\top (R^\top R)^{-1} R^\top\|_1 \|\delta\|_\infty \leq c_{V,L} d_A^{k+\alpha}$$

for all $x \in \mathcal{X}$. Applying Lemma B.1 completes the proof. □

*Proof of Proposition A.7.* Note that $f_{\mathrm{D},r}$ and $f_{\mathrm{P},r}$ consider the rectangles $A_{i/V}(x)$ for $i = 1, \cdots, V$. Since the collection of $A_{i/V}(x)$ is a subset of all cells in $\mathcal{X}$, as a direct corollary of Proposition A.5, we have

$$\|f_{\mathrm{P},r} - f_{\mathrm{D},r}\|_\infty \lesssim 2V^d M \sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} \cdot V^d M(4d+5)\log n}{3n} + \frac{2^{p+3} \cdot V^d M}{n}$$

with probability $\mathrm{P}^n \otimes \mathrm{P}_Z$ at least $1 - 2/n^2$. There is an additional constant $V^d$ since now $\int_{A_r(x)} d\mathrm{P}_X(x') \geq 2^{-p}V^{-d}$. Then, by Lemma B.7, there holds

$$\begin{aligned}|f_{\mathrm{P,E}}(x) - f_{\mathrm{D,E}}(x)| &\leq \|e_1^\top (R^\top R)^{-1} R^\top\|_1 \|f_{\mathrm{P},r} - f_{\mathrm{D},r}\|_\infty \\ &\lesssim c_{V,L}\left(2V^d M \sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} \cdot V^d M(4d+5)\log n}{3n} + \frac{2^{p+3} \cdot V^d M}{n}\right)\end{aligned}$$

for all $x \in \mathcal{X}$, which completes the proof. □

### B.3.3. PROOFS RELATED TO SECTION 3.1

*Proof of Theorem 3.1.* By Proposition A.6 and A.7, the triangle inequality tells us that

$$\begin{aligned}&\|f_{\mathrm{D,E}} - f^*_{L,\mathrm{P}}\|_\infty \leq \|f_{\mathrm{D,E}} - f_{\mathrm{P,E}}\|_\infty + \|f_{\mathrm{P,E}} - f^*_{L,\mathrm{P}}\|_\infty \\ &\lesssim c_{V,L}(2\sqrt{d})^{k+\alpha} 2^{-p(k+\alpha)/d} + 2c_{V,L}V^d M \sqrt{\frac{2^{p+1}(4d+5)\log n}{n}} + \frac{2^{p+2} c_{V,L} V^d M(4d+5)\log n}{3n} + \frac{2^{p+3} c_{V,L} V^d M}{n}\end{aligned}$$

holds with probability $\mathrm{P}^n \otimes \mathrm{P}_Z$ at least $1 - 2/n^2$. By choosing $2^p \asymp (n/\log n)^{d/(2(k+\alpha)+d)}$, i.e. $p \asymp \log(n/\log n)$, we obtain the desired assertion. □

## C. Experiments

### C.1. Implementation Details

All experiments are conducted on a machine with 72-core Intel Xeon 2.60GHz and 128GB main memory. All code is available on GitHub[1]. For ERTR, we use the parameter grids $p \in \{2, 3, 4, 5, 6, 7, 8\}$, $C \in \{0, 1\}$ and $\lambda \in \{0.001, 0.01, 0.1\}$. $V$ is fixed to be $\max(\lfloor n \cdot 2^{-(p+2)} \rfloor, 5)$. For each node, if the number of samples in the node is less than 5, then we stop the recursive partition process of the current node. The grids for each base learner in ERF and GBERTR are set similarly. For ERF, we set the number of trees to 200 and subsample $\{\lceil 0.5d \rceil, \lceil 0.75d \rceil, d\}$ features in each split procedure to look for the best cut. In addition, each base learner is trained on a $\{\lceil 0.8n \rceil, n, \lceil 1.2n \rceil\}$ samples bootstrapped with replacement from $D$. For GBERTR, we set the number of trees to 100 and the learning rate to 0.01. $\{\lceil 0.5d \rceil, \lceil 0.75d \rceil, d\}$ features are sub-sampled in each split procedure to look for the best cut. In addition, each base learner is trained on a $\{\lceil 0.8n \rceil, n, \lceil 1.2n \rceil\}$ samples bootstrapped with replacement from $D$.

The implementation details for deterministically partitioned tree methods are as follows.

- **Decision Tree (DT)**. For standard decision trees, we use the implementation by Scikit-Learn (Pedregosa et al., 2011). We select *max_depth* in $\{2, 4, 6, 8\}$ and other parameters by default.
- **Soft Tree (ST)** is proposed by Irsoy et al. (2012). We use the implementation in C++[2]. The implementation provides a self-contained validation procedure that requires an additional validation set. We take 30% of the training data as the validation set.
- **Smooth Transition Tree (STRT)** is proposed by Da Rosa et al. (2008). We use the implementation in R[3]. We set the depth *d_max* in $\{2, 4, 6, 8\}$ and choose the ratio of samples considered in each cell split $p$ to be 0.5.

[1] https://github.com/Karlmyh/ERTR
[2] https://github.com/oir/soft-tree
[3] https://github.com/gabrielrvsc/BooST

- **Probabilistic Regression Tree (PRT)** is proposed by Alkhoury et al. (2020). We use the implementation in Python [4]. We set $\sigma \in \{0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2\}$ and *min_leaf_percentage*=0.1. Their implementation is not compatible with cross-validation tools provided by Scikit-Learn. Thus, we take 30% of the training data as the validation set.

The implementation details for random partitioned tree methods are as follows.

- **Random Tree for Regression (RTR)** is defined in (5). We pick $p \in \{2, 3, 4, 5, 6, 7, 8\}$.
- **Smooth Transition Tree with max-edge partition (STRT)** replace the partition procedure in Da Rosa et al. (2008) with max-edge random partition. We modify the implementation on GitHub[5]. We set the depth *d_max* in $\{2, 4, 6, 8\}$.
- **Probabilistic Regression Tree with max-edge partition (PRT)** replace the partition procedure in Alkhoury et al. (2020) with max-edge random partition. We set $\sigma \in \{0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2\}$ and *min_leaf_percentage*=0.1.

The implementation details for other forest methods are as follows. All methods are ideally paralleled.

- **Random forest (RF)**. For the standard random forest, the Scikit-Learn package in python is applied with *n_estimators*$=$ 200 and *max_depth* $\in \{2, 4, 6, 8\}$.
- **Soft Bayesian Additive Regression Tree (SBART)** is proposed by Linero & Yang (2018) and is implemented in R[6]. To avoid unacceptable running time, we set *num_tree*$=$ 50 and let other parameters be default.
- **Probabilistic Regression Random Forest (PRRF)** is an direct extension of PRT in Alkhoury et al. (2020). We set *n_estimators* $= 200$ and *min_leaf_percentage*=0.1. $\sigma$ is fixed to 1.

The implementation details for other boosting methods are as follows.

- **Gradient Boosting Regression Tree (GBRT)**. For the standard gradient boosting proposed by Friedman (2002), the Scikit-Learn package in python is applied with *n_estimators*$=$ 100 and *max_depth* $\in \{2, 4, 6, 8\}$.
- **Gradient Boosted Smooth Transition (GBSTRT)** utilize smooth transition tree (Da Rosa et al., 2008) as base learner. We set the depth to 4 and the number of trees to 100.
- **Probabilistic Regression Boosting (GBPRT)** is an direct extension of PRT in Alkhoury et al. (2020). We set *n_estimators* $= 100$ and *min_leaf_percentage*=0.1.

### C.2. Details of Real Data Sets

We summarize the details of real data sets in Table 3, with the number of instances and features after pre-processing reported. Each feature is min-max scaled to the range $[0, 1]$ individually. We also present additional information of the data sets including the data source and the preprocessing details.

ABA: The *Abalone* dataset originally comes from biological research (Nash et al., 1994) and now it is accessible on UCI Machine Learning Repository (Dua & Graff, 2017). ABA contains 4177 observations of one target variable and 8 attributes related to the physical measurements of abalone.

AIR: The *Airfoil Self-Noise* dataset on UCI Machine Learning Repository records the result of a series of aerodynamic and acoustic tests of airfoil blade sections conducted in an anechoic wind tunnel (Brooks et al., 1989). It comprises 1503 instances of 6 attributes including wind tunnel speeds and angles of attack.

ALG: The *Algerian Forest Fires* dataset on UCI Machine Learning Repository contains 244 instances of 11 attributes and 1 output attribute. The task is to predict the condition of forest fires in Algeria (Abid & Izeboudjen, 2020). The attribute date is omitted when conducting regression in our experiments.

[4] https://gitlab.com/sami.kh/pr-tree

[5] https://github.com/gabrielrvsc/BooST

[6] https://github.com/theodds/SoftBART

*Table 3.* Description of real datasets

| DATASET | $n$ | $d$ | DATASET | $n$ | $d$ |
|---|---|---|---|---|---|
| ABA | 4177 | 8 | CPU | 8192 | 12 |
| AIR | 1503 | 6 | IST | 536 | 8 |
| ALG | 244 | 12 | FOR | 517 | 13 |
| BIAS | 7750 | 25 | MG | 1385 | 6 |
| CBM | 11934 | 16 | WR | 4898 | 12 |
| CCP | 9568 | 4 | SPA | 3107 | 6 |
| WW | 4898 | 12 | | | |

BIAS: The *Bias correction of numerical prediction model temperature forecast* dataset on UCI Machine Learning Repository is for the purpose of bias correction of next-day maximum and minimum air temperatures forecast of the model operated by Korea Meteorological Administration over Seoul (Cho et al., 2020). It contains 7750 instances of 23 input attributes and 2 output attributes. We chose the output attribute minimum air temperature as the target variable of our regression model.

CBM: The *Condition Based Maintenance of Naval Propulsion Plants* dataset (Altosole et al., 2009) on UCI Machine Learning Repository was generated from a sophisticated simulator of Gas Turbines. It contains $11934$ instances of $16$ features.

CCP: The *Combined Cycle Power Plant Data Set* dataset (Tüfekci, 2014) on UCI Machine Learning Repository contains 9568 data points. There are $4$ features that can be used to predict the net hourly electrical energy output of the power plant.

CPU: The *cpusmall* dataset is from LIBSVM(Chang & Lin, 2011). It contains $8192$ instances, each with $12$ attributes.

IST: The *Istanbul Stock Exchange* dataset (Akbilgic et al., 2014) on UCI Machine Learning Repository contains returns of the Istanbul Stock Exchange together with seven other international indexes. It has 536 instances and the time range is from January 5, 2009, to February 22, 2011. The date column in the dataset is dropped before constructing the regression models.

FOR: The *Forest Fires* dataset(Cortez & Morais, 2007) on UCI Machine Learning Repository comprises 517 instances of 13 attributes. The task is to predict the burned area of forest fires. In our experiments, two attributes, the month of the year and the day of the week were not used in the regression models.

MG: This dataset can be traced back to (Flake & Lawrence, 2002). It consists of $1385$ observations of dimension $6$.

WR: This dataset contains the information on red wine of the *Wine Quality* dataset (Cortez et al., 2009) on UCI Machine Learning Repository. There are $11$ input variables to predict the output variable wine quality. $4898$ instances are collected in the dataset.

SPA: The *Geographical Analysis Spatial* dataset is accessible in *Libstat* of CMU, originally uploaded by (Pace & Barry, 1997). It comprises 3107 observations of dimension 6.

WW: This dataset also originates from the *Wine Quality* dataset (Cortez et al., 2009) on UCI Machine Learning Repository. There are $11$ features related to white wine to predict the corresponding wine quality.

### C.3. Additional Experiment Results for GBERTR

We present the results for boosting methods in Table 4.

### C.4. Additional Experiment Results for Efficiency

In section 3.3, we justified that ERTR is efficient enough for adaption to ensemble methods. As evidence, we provide the computation time of forest methods. As we can see in Table 5, ERF achieves stably the second computation time on each data set and often outperforms PRRF and SBART by magnitudes. It is reasonable that ERF is always worse than RF. Also, we argue that ERF is implemented in pure python while RF is implemented by cython. This can also result in some performance gaps.

We argue that the test stage of ERTR is highly parallelizable since the computation of $f_{\mathrm{D},r}(x)$ only involves samples in $A(x)$. Thus, we can use the divide and conquer strategy to distribute the computation tasks on each cell to $t$ different

*Table 4.* Average MSE over real data sets for boosting methods.

| | GBERTR | GBRT | GBSTRT | GBPRT |
|---|---|---|---|---|
| ABA | 4.64 | 4.82 | **4.47** | 4.72 |
| AIR | **2.88*** | 3.53 | 8.01 | 1.34e+1 |
| ALG | **2.13e-2** | 2.18e-2 | 3.32e-2 | 3.30e-2 |
| BIAS | **7.28e-1*** | 7.40e-1 | 1.33 | 1.67 |
| CBM | 2.71e-8 | **4.85e-9*** | 9.40e-6 | 4.70e-5 |
| CCP | 1.76e+1 | **1.06e+1** | 1.65e+1 | 1.77e+1 |
| CPU | 8.37 | 7.79 | **7.52** | 1.29e+1 |
| DAK | 3.40e-5 | 3.27e-5 | 3.09e-5 | **3.01e-5** |
| FOR | **4.13e+3** | 6.84e+3 | 8.66e+3 | 4.89e+3 |
| MG | **1.43e-2*** | 1.51e-2 | 1.47e-2 | 1.62e-2 |
| WR | 3.81e-1 | **3.73e-1** | 4.03e-1 | 3.90e-1 |
| SPA | 1.15e-2 | 1.25e-2 | **1.07e-2** | 1.36e-2 |
| WW | 4.20e-1 | **4.05e-1*** | 4.74e-1 | 5.05e-1 |
| ranking sum | **25** | 28 | 33 | 44 |

*Table 5.* Average running time (s) over real data sets for forest methods.

| | ERF | RF | PRRF | SBART |
|---|---|---|---|---|
| ABA | 1.01e+1 | **1.15** | 2.31e+2 | 5.94e+2 |
| AIR | 2.12 | **3.05e-1** | 8.07e+1 | 3.02e+2 |
| ALG | 9.28e-1 | **2.36e-1** | 1.09e+1 | 4.06e+1 |
| BIAS | 2.76e+1 | **6.23** | 4.45e+2 | 2.71e+3 |
| CBM | 2.78e+2 | **4.33** | 9.66e+2 | - |
| CCP | 5.17e+2 | **2.13** | 5.94e+2 | 1.61e+3 |
| CPU | 2.93e+1 | **3.80** | 5.65e+2 | 1.71e+3 |
| DAK | 8.74e-1 | **4.00e-1** | 2.79e+1 | 9.89e+1 |
| FOR | 1.15 | **2.98e-1** | 3.41e+1 | 6.12e+1 |
| MG | 2.49 | **5.62e-1** | 7.35e+1 | 2.46e+2 |
| WR | 4.18 | **9.56e-1** | 9.75e+1 | 2.16e+2 |
| SPA | 5.82 | **1.09** | 1.66e+2 | 4.50e+2 |
| WW | 1.85e+1 | **1.90** | 2.87e+2 | 9.19e+2 |

† The best results are **bolded** and the second best results are underlined. The best results with significance are marked with $*$.

‡ The running of SBART is corrupted on three data sets that are marked with -.

machines. Since no additional storage is needed, the space complexity remains $\mathcal{O}(nd)$ and the prediction computation time is divided by $t$. In comparison, $k$ nearest neighbors-based methods, if paralleled, require $\mathcal{O}(ndt)$ memory to achieve the same computation time. Similarly, paralleled PR tree also has $\mathcal{O}(Kdt)$ storage. This means that, for large data sets, the overhead of parallel computing caused by memory bandwidth restrictions is significantly smaller for ERTR than for PR tree. Hence, ERTR is suitable for parallelism to promote computation speed.