Supplementary Material: Online Binary Space Partitioning Forests

Xuhui Fan
School of Mathematics and Statistics
University of New South Wales
xuhui.fan@unsw.edu.au

Bin Li
School of Computer Science
Fudan University
libin@fudan.edu.cn

Scott A. Sisson
School of Mathematics and Statistics
University of New South Wales
scott.sisson@unsw.edu.au

1 Algorithm of Generating A Cut in $\Diamond'$ and Not Crossing into $\Diamond$

Algorithm 1 GenCutNorCross($\Diamond', \Diamond$)

1: $(d_1^*, d_2^*) \sim \text{Cat}(\tilde{L}_{(1,2)}(\Diamond') - \tilde{L}_{(1,2)}(\Diamond), \ldots, \tilde{L}_{(d-1,d)}(\Diamond') - \tilde{L}_{(d-1,d)}(\Diamond))$
2: $\theta \sim p(\theta) \propto |\boldsymbol{l}_{\Pi_{(d_1^*,d_2^*)}(\Diamond')}(\theta)| - |\boldsymbol{l}_{\Pi_{(d_1^*,d_2^*)}(\Diamond)}(\theta)|, \theta \in (0, \pi]$
3: Sample $\boldsymbol{u}$ uniformly on $|\boldsymbol{l}_{\Pi_{(d_1^*,d_2^*)}(\Diamond')}(\theta)| - |\boldsymbol{l}_{\Pi_{(d_1^*,d_2^*)}(\Diamond)}(\theta)|$
4: Form cutting hyperplane based on $(d_1^*, d_2^*), \theta, \boldsymbol{u}$

2 Some visualisations

Left panel of Figure 1 visualizes the difference between the convex hull representation of tree node in the BSP-Tree partition and Mondrian tree partition. Convex hulls are formed recursively, and larger hulls contain smaller ones. The BSP-Tree generates smaller convex hulls than the Mondrian-Tree, which means the BSP-Tree is a "tight" representation of the space. Right panel of Figure 1 visualizes oblique line slice of the BSP-Tree Process.

3 Proof of Lemma 1

Lemma 1. *(Oblique line slice) For any oblique line that crosses into the domain of a BSP-Tree process with budget τ, its intersection points with the partition forms a homogeneous Poisson process with intensity 2τ.*

Proof. The self-consistent property of the BSP-Tree process guarantees that this 1-dimensional slice follows the same way of directly generating a BSP-Tree partition on the line.

To define the BSP-Tree partition on the line segment, we first consider the BSP-Tree partition in an obtuse triangle. Two vertices of the triangle form a line segment with the length L. Another vertex lies between these two vertices and has an ϵ distance to the line segment. Based on the generative process of the BSP-Tree process, the cost of cut in this triangle follows an Exponential distribution with rate parameter being the perimeter of the triangle, which is $PE = (L_1 + L_2) + \sqrt{L_1^2 + \epsilon^2} + \sqrt{L_2^2 + \epsilon^2}$. As $PE \to 2(L_1 + L_2)$ when $\epsilon \to 0$, the cost has an exponential distribution with rate parameter $2L$ accordingly. As a result, the number of cuts follows a Poisson distribution with parameter $\tau \cdot 2L$. The cut position is Uniformly distributed in the line segment. (For each projection in the direction of θ, the crossing point between the cuts and line segment is Uniformly distributed.) This can verify the independent increments of the partition points in the line.

As the two condition of Poisson process is satisfied, according to Theorem 1.10 in [2], we can get the conclusion. □

4 Proof of Theorem 2

It is noted the main idea of the following proof largely follows the work of [6]. We make modifications to make the proof suitable to online BSP-Forest case.

Lemma 2. *(Block diameter) Let $\boldsymbol{x} \in [0,1]^d$, and let $D(\boldsymbol{x})$ be the L^2-diameter of the block containing $\boldsymbol{x}$ in the BSP-Tree partition with budget $\tau/2$. If $\tau \to \infty$, then $D_\tau(\boldsymbol{x}) \to 0$ in probability. More precisely, for every $\delta, \tau > 0$, we have*

$$\mathbb{P}(D_\tau(\boldsymbol{x}) \geq \delta) \leq d(1+\frac{\tau\delta}{\sqrt{d}})\exp(-\frac{\tau\delta}{\sqrt{d}}), \quad \mathbb{E}[D_\tau(\boldsymbol{x})^2] \leq \frac{4d}{\tau^2}.$$

Proof. Let $\square_\tau(\boldsymbol{x})$ denotes the block of a Binary Space Partitioning-Tree partition containing $\boldsymbol{x} \in [0,1]^d$. In the space of $[0,1]^d$, we can build up d orthogonal basises to describe the block $\square_\tau(\boldsymbol{x})$, with one basis is in the direction of largest diameter in $\square_\tau(\boldsymbol{x})$. While it is obvious that rotations of the bock will not affect the diameter, w.l.o.g., we rotate the block and treat the direction with largest diameter as dimension 1. By definition, the L^∞-norm diameter $D_\tau(\boldsymbol{x})$ of $\square_\tau(\boldsymbol{x})$ is $\max\{\square_\tau^{(d')}(\boldsymbol{x})\}_{d'}$. While recording these smallest and largest interceptions in these rotated dimensions as $\{L_\tau^{(d')}, R_\tau^{(d')}\}_{d'}$, all of the random variables $R_\tau(\boldsymbol{x}) - L_\tau(\boldsymbol{x})$ have the same distribution, it suffices to consider $D_\tau^{(1)}(\boldsymbol{x}) = R_\tau^{(1)}(\boldsymbol{x}) - L_\tau^{(1)}(\boldsymbol{x})$.

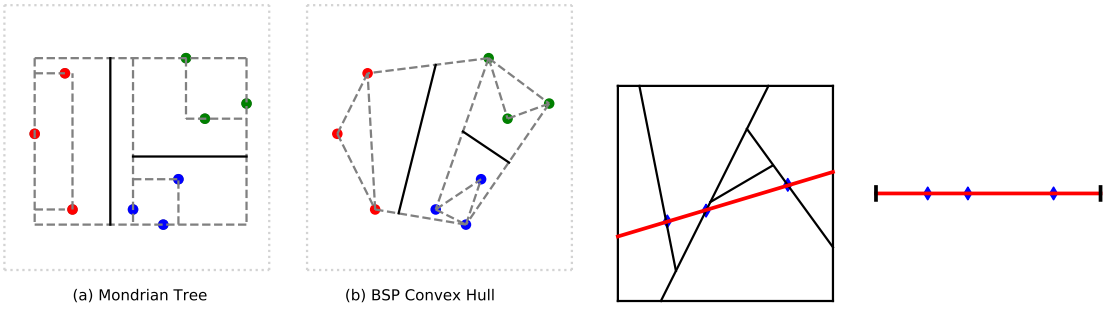

Figure 1: Left: example visualization comparing the Mondrian-Tree and BSP-Tree convex hulls. Point colors identify different data labels, and the dotted, dashed and solid lines denote the whole space, the convex hulls and the cuts, respectively. Right: $2d$ visualization of oblique line slice of the BSP-Tree Process partition. Red solid line denotes the oblique line and blue dots represents the intersection points.

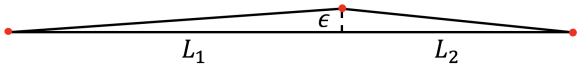

Figure 2: Visualization for the 1-dimansional space case.

In this rotated block, consider the segment $I^{(1)}(\boldsymbol{x}) = [0, \sqrt{d}] \times \boldsymbol{x}^{-i}$ containing $\boldsymbol{x}$, and denote $\phi^1_\tau(\boldsymbol{x}) \subset [0, \sqrt{d}]$ the restriction of the partition to $I^{(1)}(x)$. Note that $R^{(1)}_\tau(\boldsymbol{x})$ ($L^{(1)}_\tau(\boldsymbol{x})$) is the lowest (highest) element of $\phi^{(1)}_\tau(\boldsymbol{x})$ that is larger (smaller) than x_1, and is equal to $\sqrt{d}$ (0) if $\phi^{(1)}_\tau(\boldsymbol{x}) \cap [x_1, \sqrt{d}]$ ($\phi^{(1)}_\tau(\boldsymbol{x}) \cap [0, x_1]$) is empty. By Theorem 1 and Lemma 1, $\phi_\tau(\boldsymbol{x})$ is a Poisson process with intensity τ.

This implies the distribution of $(L^{(1)}_\tau(\boldsymbol{x}), R^{(1)}_\tau(\boldsymbol{x}))$ is the same as that of $(\tilde{L}^{(1)}_\tau(\boldsymbol{x}) \vee 0, \tilde{R}^{(1)}_\tau(\boldsymbol{x}) \wedge \sqrt{d})$, where $\tilde{\phi}^1_\tau(\boldsymbol{x})$ is a Poisson process on $\mathbb{R}$ with intensity τ, and $\tilde{L}^{(1)}_\tau(\boldsymbol{x}) = \sup(\tilde{\phi}^{(1)}_\tau(x_1) \cup (-\infty, x_1))$, $\tilde{R}^{(1)}_\tau(\boldsymbol{x}) = \inf(\tilde{\phi}^{(1)}_\tau(x_1) \cap (x_1, \infty))$. By the property of the Poisson point process, this implies that $x_1 - L^{(1)}_\tau(\boldsymbol{x}), R^{(1)}_\tau(\boldsymbol{x}) - x_1 \overset{d}{=} (E_1, E_2)$, where E_1, E_2 are independent exponential random variables with parameter τ. $D^{(1)}_\tau(\boldsymbol{x}) = R^{(1)}_\tau(\boldsymbol{x}) - x_1 + x_1 - L^{(1)}_\tau(\boldsymbol{x})$ is upper bounded by $E_1 + E_2 \sim \text{Gamma}(2, \tau)$. Thus, we have $\forall \delta > 0, \mathbb{P}(D^1_\tau(\boldsymbol{x})) \geq \delta) \leq (1 + \tau\delta)e^{-\tau\delta}$ and $\mathbb{E}[D^1_\tau(\boldsymbol{x})^2] \leq \mathbb{E}(E^2_1) + \mathbb{E}(E^2_2) = \frac{4}{\tau^2}$. The bound of $D_\tau(\boldsymbol{x})$ can be obtained by $D_\tau(\boldsymbol{x}) = \sqrt{\sum_{d'} D^{d'}_\tau(\boldsymbol{x})}$. □

Lemma 3. *If K_τ denotes the number of cuts in the BSP-Tree process, we have $\mathbb{E}[K_\tau] \leq (1 + \tau)^d e^{d(d-1)}$.*

Proof. Let $\square \subset [0, 1]^d$ be an arbitrary block, and let $K^\square_\tau$ denotes the number of splits performed in the BSP-Tree process with budget value $\tau/2$. As shown in [4][5], the waiting time of a cut occurs in a leaf node ϕ of the BSP-Tree process follows an exponential distribution of rate $L(\square_\phi) \leq L(\square)$, where $L(\square)$ denotes the perimeter of the block $\square$. The number of leaves $K_t + 1 \geq K_t$ at time t is dominated by the number of individuals in a Yule process with rate $L(\square)$ [7]. Thus, we have: $\mathbb{E}(K^\square_\tau) \leq e^{\tau L(\square)}$.

Considering the covering $\mathcal{C}$ of $\square$ by a regular grid of $\lceil\tau\rceil^d$ boxes obtained by equally dividing each coordinate of $\square$ in $\lceil\tau\rceil$ parts. Each cut in $\square$ will induce a split in at least one box Cin $\mathcal{C}$ and $B^{\mathcal{C}}_\tau$ is also a BSP-Tree process in box C (due to the self-consistency of the BSP-Tree process), we have: $\mathbb{E}(K^\square_\tau) \leq \sum_{C \in \mathcal{C}} \mathbb{E}(K^C_\tau) \leq \lceil\tau\rceil^d e^{\tau \frac{L(\square)}{\lceil\tau\rceil}} \leq (\tau + 1)^d e^{L(\square)}$. □

Lemma 4. *Assume that the total number of splits K_τ performed by the BSP-Tree partition satisfies $\lim_{n\to\infty} \mathbb{E}(K_{\tau_n})/n \to 0$. For $N_n(\boldsymbol{x})$ being the number of datapoints in $\boldsymbol{x}_{1:N}$ fall in $A_{\tau_n}(\boldsymbol{x})$, we have $N_n(\boldsymbol{x}) \to \infty$ in probability.*

Proof. We fix $n \geq 1$, and conditionally on the BSP-Tree partition at the budget of τ_n, B_{τ_n} is independent of $\boldsymbol{x}$ by construction. Note that the number of leaves is $|\mathcal{L}(B_{\tau_n})| = K_{\tau_n} + 1$, and $(\square_\phi)_{\phi \in \mathcal{L}(B_{\tau_n})}$ is the corresponding blocks, where ϕ refers tot he leaf node. For ϕ, we define N_ϕ to be the number of points among $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ that fall in the cell $\square_\phi$. Since $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are i.i.d., so that the joint distribution of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ is invariance under the permutation of the $n + 1$ datapoints, conditionally on the set $S = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ the probability that $\boldsymbol{x}$ falls in the block $\square_\phi$. Therefore, for each $t > 0$, we have:

$$\begin{aligned}\mathbb{P}(N_n(\boldsymbol{x}) \leq t) =& \mathbb{E}\{\mathbb{P}(N_n(\boldsymbol{x}) \leq t | S, B_{\tau_n})\} \\ =& \mathbb{E}\left\{\sum_{\phi \in \mathcal{L}(B_{\tau_n}): N_\phi \leq t} \frac{N_\phi}{n+1}\right\} \\ \leq& \mathbb{E}\left\{\frac{t|\mathcal{L}(B_{\tau_n})|}{n+1}\right\} = \frac{t(\mathbb{E}(K_{\tau_n}) + 1)}{n+1}\end{aligned} \tag{1}$$

which tends to 0 as $n \to \infty$. □

Before proving Theorem 2, we first invoke a consistency theorem (Theorem 6.1 in [3] and we use □ to denote the block for notation consistency)

Theorem 4. *Consider a sequence of randomised tree classifiers $(\tilde{g}_n(\cdot, Z))$, grown independently of the labels $Y_1, \ldots, Y_n$. For $\boldsymbol{x} \in [0,1]^d$, denote $\square_n(\boldsymbol{x}) = \square_n(\boldsymbol{x}, Z)$ the block containing $\boldsymbol{x}$, $D_n(\boldsymbol{x})$ its diameter and $N_n(\boldsymbol{x}) = N_n(\boldsymbol{x}, Z)$ the number of input vectors among $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ that fall in $\square_n(\boldsymbol{x})$. Assume that, if $\boldsymbol{x}$ is drawn from the distribution with the following conditions:*

1. $\lim_{n\to\infty} D_n(\boldsymbol{x}) \to 0$ *in probability;*
2. $\lim_{n\to\infty} N_n(\boldsymbol{x}) \to \infty$ *in probability.*

Then the tree classifier $\tilde{g}_n$ is consistent.

The proof of Theorem 2 is:

Proof. We can show the two conditions in Theorem 4 are satisfied. First, Lemma 1 ensures that, if $\tau_n \to \infty$, $D_{\tau_n}(\boldsymbol{x}) \to 0$ in probability for every $\boldsymbol{x} \in [0,1]^d$. In particular, for every $\delta > 0$, we have $\mathbb{P}(\square_{\tau_n}(\boldsymbol{x}) \geq \delta) = \int_{[0,1]^d} \mathbb{P}(D_{\tau_n}(\boldsymbol{x}) > \delta)\mu(d\boldsymbol{x}) \to 0$ as $n \to \infty$ by the dominated convergence theorem.

Since Lemma 4 provides the proof for the second condition, the proof of Theorem 2 is concluded. □

5 Proof of Theorem 3

It is noted the main idea of the following proof largely follows the work of [6]. We make modifications to make the proof suitable to online BSP-Forest case.

Proof. By the convexity of the quadratic loss function and the fact that all the BSP-Tree has the same distribution, we have that $\mathbb{E}\left[(g(\boldsymbol{x}) - \widehat{g}_n(\boldsymbol{x}))^2\right] \leq \frac{1}{m}\sum_{k=1}^m \mathbb{E}\left[(g(\boldsymbol{x}) - \widehat{g}_{n,k}(\boldsymbol{x}))^2\right] = \mathbb{E}\left[(g(\boldsymbol{x}) - \widehat{g}_{n,1}(\boldsymbol{x}))^2\right]$. Thus, we can prove the result for a single treee algorithm to get the conclusion. □

We firs write use the bias-variance decomposition of the quadratic loss by:

$$R(\widehat{f}_n) = \mathbb{E}\left[(f(\boldsymbol{x}) - \bar{f}_n(\boldsymbol{x}))^2\right] + \mathbb{E}\left[(\widehat{f}_n(\boldsymbol{x}) - \bar{f}_n(\boldsymbol{x}))^2\right] \tag{2}$$

where $\bar{f}_n(\boldsymbol{x}) := \mathbb{E}\left[f(\boldsymbol{x}|\boldsymbol{x} \in A_n(\boldsymbol{x}))\right]$ denotes the groundtruth label value for the block containing $\boldsymbol{x}$. The first term is bias and it measures the closeness of $f(\boldsymbol{x})$ to the best approximator $\bar{f}_n(\boldsymbol{x})$ (of which the label value is constant on the block containing $\boldsymbol{x}$). The second term is variance and it measures the closeness of he best approximator $\bar{f}_n(\boldsymbol{x})$ to the empirical approximator $\widehat{f}_n(\boldsymbol{x})$.

For the bias term, we have:

$$\begin{aligned}|f(\boldsymbol{x}) - \bar{f}_n(\boldsymbol{x})| \leq & |\frac{1}{\mu(A_n(\boldsymbol{x}))}\int_{A_n(\boldsymbol{x})}(f(\boldsymbol{x}) - \bar{f}_n(\boldsymbol{z}))\mu(d\boldsymbol{z})| \\ \leq & \sup_{\boldsymbol{z}\in A_n(\boldsymbol{x})}|f(\boldsymbol{x}) - f(\boldsymbol{z})| \\ \leq & L \sup_{\boldsymbol{z}\in A_n(\boldsymbol{x})}\|\boldsymbol{x} - \boldsymbol{z}\|_2 = L \cdot D_n(\boldsymbol{x}) \end{aligned} \tag{3}$$

where $D_n(\boldsymbol{x})$ is the l^2-diameter of $A_n(\boldsymbol{x})$. According to the result of Lemma 1, we get:

$$\mathbb{E}\left[(f(\boldsymbol{x}) - \bar{f}_n(\boldsymbol{x}))^2\right] \leq L^2\mathbb{E}\left[D_n(\boldsymbol{x})^2\right] \leq \frac{4dL^2}{\tau_n^2} \tag{4}$$

For the variance term, based on the Proposition 2 of [1]: if U is a random tree partition of the unit space with $k+1$ blocks, we have:

$$\mathbb{E}\left[(\bar{f}_U(\boldsymbol{x}) - \widehat{f}_U(\boldsymbol{x}))^2\right] \leq \frac{k+1}{n}(2\sigma^2 + 9\|f\|_\infty) \tag{5}$$

. Thus, we can have:

$$\begin{aligned}&\mathbb{E}\left[(\bar{f}_n(\boldsymbol{x}) - \widehat{f}_n(\boldsymbol{x}))^2\right] = \sum_{k=0}^\infty \mathbb{P}(k)\mathbb{E}\left[(\bar{f}_U(\boldsymbol{x}) - \widehat{f}_U(\boldsymbol{x}))^2|k\right] \\ \leq & \sum_{k=0}^\infty \mathbb{P}(k)\frac{k+1}{n}(2\sigma^2 + 9\|f\|_\infty) \\ = & \frac{\mathbb{E}(K_n)+1}{n}(2\sigma^2 + 9\|f\|_\infty)\end{aligned} \tag{6}$$

Based on the result of Lemma 2, we get

$$\begin{aligned}&\mathbb{E}\left[(\bar{f}_n(\boldsymbol{x}) - \widehat{f}_n(\boldsymbol{x}))^2\right] \\ \leq & \frac{(1+\tau_n)^d e^{d(d-1)}+1}{n}(2\sigma^2 + 9\|f\|_\infty)\end{aligned} \tag{7}$$

Combining the result of Eq. (4)(7), we get:

$$\begin{aligned}&R(\widehat{f}_n) \\ \leq & \frac{4dL^2}{\tau_n^2} + \frac{(1+\tau_n)^d e^{d(d-1)}+1}{n}(2\sigma^2 + 9\|f\|_\infty)\end{aligned} \tag{8}$$

Taking $\tau_n = n^{\frac{1}{d+2}}$ can make $R(\widehat{f}_n)$ scales to $\mathcal{O}(n^{-\frac{2}{d+2}})$.

6 Additional Experimental Results

Figure 3 illustrates the observed and predicted labels (and their difference) for the UK Apartment Price Data. The online BSP-Forest appears to be able to capture the price variation reasonably well, and provide an accurate prediction of the true test data. Spatially, the prediction error looks broadly pattern free (in colour distribution), indicating that the regression model is adequate for these data.

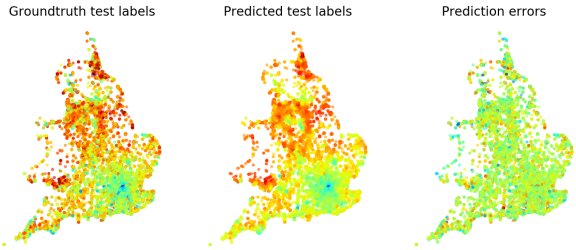

Figure 3: Visualisation of the online BSP-Forest's spatial predictions on the UK Apartment Price data. Plots show [L-R] actual test data, predictions, and prediction errors. Red–blue colour denotes low–high prices.

References

[1] Sylvain Arlot and Robin Genuer. Analysis of purely random forests bias. *arXiv preprint arXiv:1407.3939*, 2014.

[2] Matej Balog and Yee Whye Teh. The mondrian process for machine learning. *arXiv preprint arXiv:1507.05181*, 2015.

[3] Luc Devroye, László Györfi, and Gábor Lugosi. *A probabilistic theory of pattern recognition*, volume 31. Springer Science & Business Media, 2013.

[4] Xuhui Fan, Bin Li, and Scott A. Sisson. The binary space partitioning-tree process. In *AISTATS*, volume 84 of *Proceedings of Machine Learning Research*, pages 1859–1867, 2018.

[5] Xuhui Fan, Bin Li, and Scott A. Sisson. Binary space partitioning forests. In *AISTATS*, volume 89, pages 3022–3031, 2019.

[6] Jaouad Mourtada, Stéphane Gaïffas, and Erwan Scornet. Universal consistency and minimax rates for online mondrian forests. In *NIPS*, pages 3758–3767, 2017.

[7] George Udny Yule. A mathematical theory of evolution based on the conclusions of Dr. Jc Willis, FRS. *Philosophical Transactions of the Royal Society B*, 1925.