Hand-Waving Derivation on Generalized Random Forest

June 21, 2021

Reference: Athey, Tibshirani, and Wager (2018) Generalized Random Forest.

I uses the proposition 1,2, and their proofs in the appendix.

1 General Framework

The authors propose a general framework for Random Forest in two steps: relabeling and splitting.

In the relabeling step, we estimate a target function that minimizes the scoring function in a parent node level. In the splitting step, we loop over all columns of X to find the best splitting point that minimize the mean square error between in sample and test sample estimation.

More specifically, suppose we have an input data X, output O, a scoring function φ, weights α, parameter in focus θ, and optimal nuisance parameter ν. We want to find $\hat{\theta}$ and $\hat{\nu}$ that locally optimize the equation:

$$E[\varphi_{\theta(x),\nu(x)}(O_i)|X_i = x] = 0 \tag{1}$$

with the solution

$$(\hat{\theta}(x), \hat{\nu}(x)) \in \underset{\theta,\nu}{argmin}\Big\{\| \sum_{i}^{n} \alpha_i \varphi_{\theta,\nu}(O_i)\|_2\Big\} \tag{2}$$

In regression trees, output O is Y, but in instrument regression or causal inference, O is tuple (Y, W) where W is treatment assignment.

Let P be a parent node, and J be a sample of data, X be a test set, C_1 and C_2 be children nodes, n_{C_j} be number of observations in node j, X_p be center of mass in parent node , we want to minimize the equation:

$$err(C_1, C_2) = \sum_{j=1,2} P(X \in C_j | X \in P) E[(\hat{\theta}(J) - \theta(X))^2] \tag{3}$$

It's computational expensive to evaluate test sets during evaluating all possible splitting points. Instead, we use "honest" estimation (Athey and Imbens (2016)). We split data into two sets. One set is used to build trees, and the other set repopulate node values in the same trees. We then remove tree nodes if they are empty after repopulation (This step is more clear in the source code (function TreeTrainer::train)).

We proceed to derive splitting criteria. To use the same notations, $X = J_{C_1} + J_{C_2}$. We expand the square and use the $\text{Var}(X) = E(X^2) - [E[X]]^2$:

$$\begin{aligned} err(C_1, C_2) &= \sum_{j=1,2} \frac{n_{c_j}}{n_p} [E(\hat{\theta}_{c_j}(J))^2 + E(\theta(X)^2) - 2E(\hat{\theta}(J)\theta(X))] \\ &= \sum_{j=1,2} \frac{n_{c_j}}{n_p} [\text{Var}_{x \in C_j}(\theta(X)) + Var_{x \in C_j}(\hat{\theta}_{C_j}(X_p; J)) + \\ &\qquad [E(\hat{\theta}(X_p; J)) - E_{X \in C_j}(\theta(X))]^2] \end{aligned} \tag{4}$$

It's computational expensive to re-evaluate $\hat{\theta}$ for all possible splitting points, so we approximate $\hat{\theta}$ by $\tilde{\theta}$ so that $\tilde{\theta}$ is calculated at the parent node for one time.

By Taylor expansion and proof of proposition 2:

$$E(\varphi(X_p)) = E_{x \in C}(\varphi(x_C)) + (\hat{\theta}_P - \hat{\theta}_C) * E_{x \in P}(\frac{\partial \varphi_{\hat{\theta}, \hat{\nu}}}{\partial \hat{\theta}}) + O_P(r^2, 1/n_C) \tag{5}$$

The error term is bounded and ignored as n_C is large. Then, we have

$$\begin{aligned} \hat{\theta}_C \approx \tilde{\theta}_C &= \hat{\theta}_P - \frac{1}{n_p} A_P^{-1} \sum_{i \in C} \xi^T \varphi_{\hat{\theta}, \hat{\nu}}(O_i) \\ &= \hat{\theta}_P - \frac{1}{n_P} \sum_{i \in C} \rho_i \end{aligned} \tag{6}$$

where ξ is a column vector that selects θ, and A_P is $\frac{1}{n_P} \sum\limits_{i \in C} \frac{\partial \varphi}{\partial \hat{\theta}}(O_i)$. In the source code, ξ equivalent to looping over all columns of X.

Now we go back to equation (4) and replace $\hat{\theta}$ by $\tilde{\theta}$. The last term is bounded by $O(r^4)$ and removed.

$$err(C_1, C_2) = \sum_{j=1,2} \frac{n_{c_j}}{n_p} [\text{Var}_{x \in C_j}(\theta(X)) + \text{Var}_{x \in C_j}(\tilde{\theta}_{C_j}(X_p; J))] + O_P(\frac{1}{n_{c_1}}, \frac{1}{n_{n_2}}) \tag{7}$$

Note, $\text{Var}_{x \in C_j}(\tilde{\theta}_{C_j}(X_p; J))$ is the sampling noise and can be ignored when n_P is large, and $O_P(\frac{1}{n_{c_1}}, \frac{1}{n_{n_2}})$ is bounded and small because we assume $n_P >> r^{-2}$, where r is some positive value away from node P (appendix, proof of proposition 1). We keep expanding the first term into parent node level:

$$\begin{aligned} err(C1, C2) &\approx \sum_{j=1,2} \frac{n_{c_j}}{n_P} [E_{x \in C_j}[\theta(X)^2] - [E_{x \in C_j}(\theta(X))]^2] \\ &= \text{Var}_{x \in P}(\theta(X)) + [E_{x \in P}(\theta(X))]^2 - \sum_{j=1,2} \frac{n_{c_j}}{n_P} [E(\theta(X))]^2 \\ &= \text{Var}_{x \in P}(\theta(X)) + [\frac{(\sum_{x \in C_1} \theta + \sum_{x \in C_2} \theta)^2}{n_P^2}] - \sum_{j=1,2} \frac{n_{c_j}}{n_P} [E(\theta(X))]^2 \\ &= \text{Var}_{x \in P}(\theta(X)) - \frac{n_{c_1} n_{c_2}}{n_P^2} [E_{x \in C_2}(\theta) - E_{x \in C_1}(\theta)]^2 \\ &= \text{Var}_{x \in P}(\theta(X)) - \frac{n_{c_1} n_{c_2}}{n_P^2} E[(\hat{\theta}_{C_2} - \hat{\theta}_{C_1})^2] \end{aligned} \tag{8}$$

Now the first term is calculated once in the parent node and becomes constant, so we can ignore it. We reassemble the proposition 1:

$$\Delta(C_1, C_2) := \frac{n_{c_1} n_{c_2}}{n_P^2} [\hat{\theta}_{C_2} - \hat{\theta}_{C_1}]^2 \tag{9}$$

We use the proof of Proposition 2 and have

$$\hat{\theta}_{C_j} - \tilde{\theta}_{C_j} = O_P(r, 1/\sqrt{n_{C_j}}) \tag{10}$$

We plug in equation (6) and approximate $\Delta(C_1, C_2)$ by $\tilde{\Delta}(C_1, C_2)$:

$$\Delta(C_1, C_2) = \tilde{\Delta}(C_1, C_2) = \sum_{j=1,2} \frac{1}{n_j} (\sum_{x \in C_j} \rho_i)^2 + O_P(r^2, 1/n_C 1, 1/n_c 2) \tag{11}$$

The error term can be ignored as n_C is large.

In words, the equation (11) is the splitting rule that maximize children nodes heterogeneity, and $err(C_1, C_2)$ maximize the difference between parent node and children node variance (I try to rewrite equation (4), but I am not able to show the error bounds). In addition, for a given node P, we only need to compute $\sum_{i \in C} \xi^T \varphi_{\hat{\theta}, \hat{\nu}}(O_i)$ per splitting point because A_P is a scaling constant.

We then use the optimal splitting rules to split the other subset of data. Node values are then estimated in the new subset.

2 Causal Forest

Section 6.1 in the paper.

We have the functional form for causal regression

$$Y = W\beta(X) + C(X) \tag{12}$$

where $\beta(X)$ is the parameter in focus, and $C(X)$ is interception term.

We have the scoring function:

$$\varphi = [Y - W\beta(X)]W^T \tag{13}$$

where $\theta = \xi\beta(X)$.

By setting the mean square error as the loss function, we get the solution $\hat{\theta}$:

$$\hat{\theta} = \xi^T (WW^T)^{-1} WY \tag{14}$$

and the splitting rule is:

$$\rho = \xi^T A_P^{-1} \varphi \tag{15}$$

$$A_P = \frac{1}{n_P} WW^T \tag{16}$$

Note, however, we demean the treatment assignment W and outcome Y so that the estimators can be robust to counfounding effect (section 6.1).

On the other hand, the splitting point is independent on X.

In the source code, the splitting rule is first calculating the local β from sample data

$$\hat{\beta} = \frac{(W - \bar{W})[Y - \bar{(Y)}]}{(W - \bar{W})^2} \tag{17}$$

then calculate $response_i$ per splitting point and column of feature

$$response_i = (W - \bar{W})[Y - \bar{Y} - \hat{\beta}(W - \bar{W})] \tag{18}$$

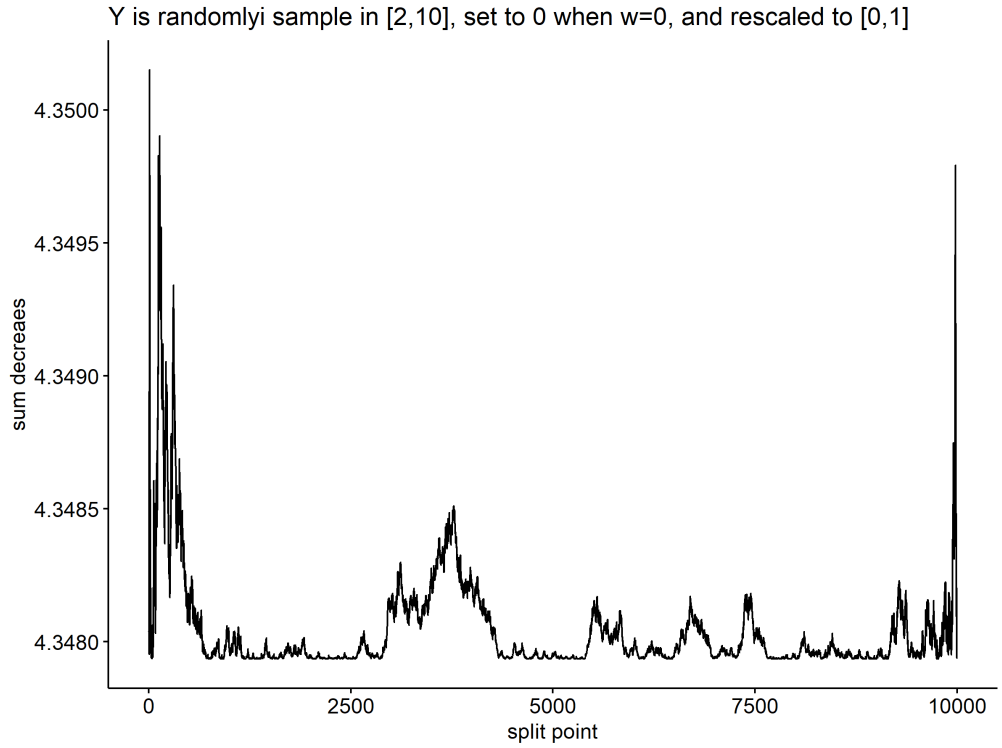
then we find the splitting point that give the max *decrease*

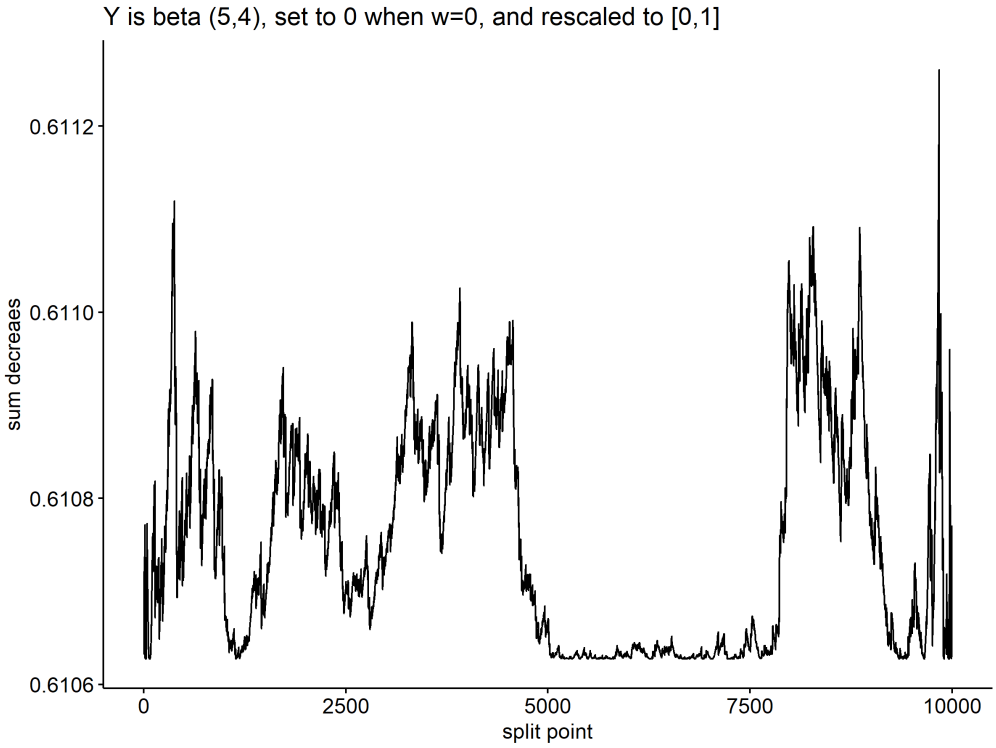
$$decrease = \frac{[\sum_{i \in left} response_i]^2}{n_{left}} + \frac{[\sum_{i \in right} response_i]^2}{n_{right}} \tag{19}$$

3 Distribution on Splitting Point

Suppose we have a protective profiles for users, such as race, gender, etc. We don't want to use them to estimate treatment effect, but we want to make sure targets that have similar protective profiles share similar target probability.

When splitting the data, we purpose to add a penalty term. However, the splitting point range is not bounded. The plots below are simulations for the decrease value in different splitting point.




Y is beta (5,4), set to 0 when w=0, and rescaled to [0,1]
sum decreaes
0.6112
0.6110
0.6108
0.6106
0
2500
5000
7500
10000
split point