Generalized Random Forests

Daegeon Lee

Applied Econometrics Reading Group

March 6, 2025

Recap on Random Forests

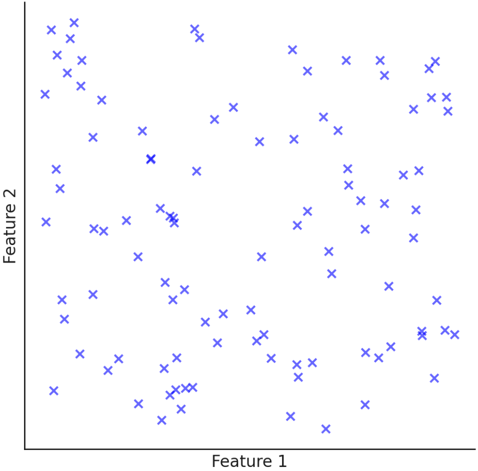


Figure: Training Sample

Recap on Random Forests

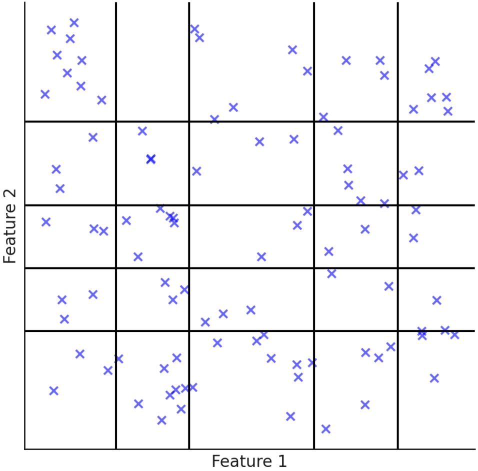


Figure: After Split

Recap on Random Forests

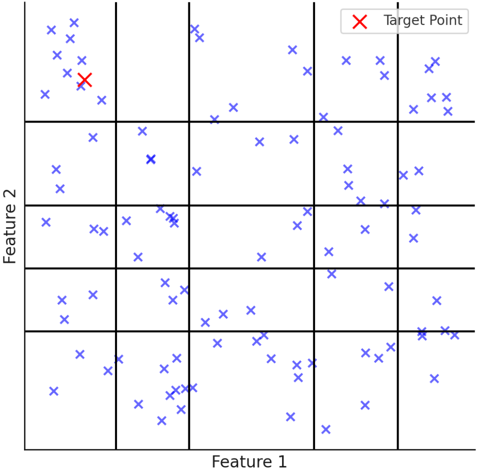


Figure: Predict Based on The Leaf

Recap on Random Forests

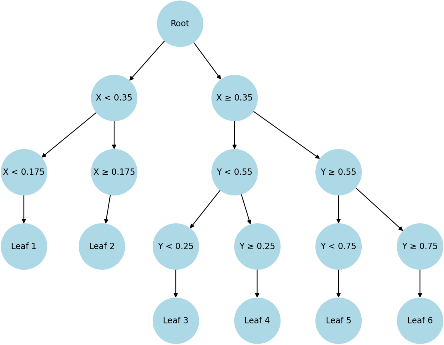


Figure: Tree

S. Wager and S. Athey (2018)

- Using **Subsampling** and **Honesty**, Wager and Athey developed the desired consistency and asymptotic theory
  - **Subsampling**: Draw a random subsample of size s from $\{1, ..., n\}$ **without replacement**
  - **Honesty**: A tree is honest if, for each training example i, it only uses the response Y_i to estimate the within-leaf treatment effect, or to decide where to place the splits but **not both**

Generalized Random Forests

Main goal of GRF is to estimate solutions to local estimation equations of the form

$$\mathbb{E}[\psi_{\theta(x),\nu(x)}(O_i)|X_i = x] = 0 \;\; for \;\; all \;\; x \in \mathcal{X}$$

- $\theta(x)$: a parameter we care about

  $\nu(x)$: an optional nuisance parameter
- Suppose we have n i.i.d. samples, for which we have access to observable O_i that encodes information relevant to $\theta(\cdot)$ along with auxiliary covariates X_i

Forest-Based Local Estimation

Recall) If we estimate $\mu(x) = \mathbb{E}[Y_i | X_i = x]$ with RF, we have

$$\hat{\mu}_b(x) = \frac{1}{|\{i : X_i \in L_b\}|} \sum_{\{i : X_i \in L_b\}} Y_i, \;\; \hat{\mu}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{\mu}_b(x)$$

With GRF, using $\psi_{\mu(x)}(Y_i) = Y_i - \mu(x)$, we have

$$\sum_{i=1}^{n} \frac{1}{B} \sum_{b=1}^{B} \alpha_{bi}(x)(Y_i - \hat{\mu}(x)) = 0$$

$$\iff \hat{\mu}_b(x) = \frac{1}{|\{i : X_i \in L_b\}|} \sum_{\{i : X_i \in L_b\}} Y_i, \;\; \hat{\mu}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{\mu}_b(x)$$

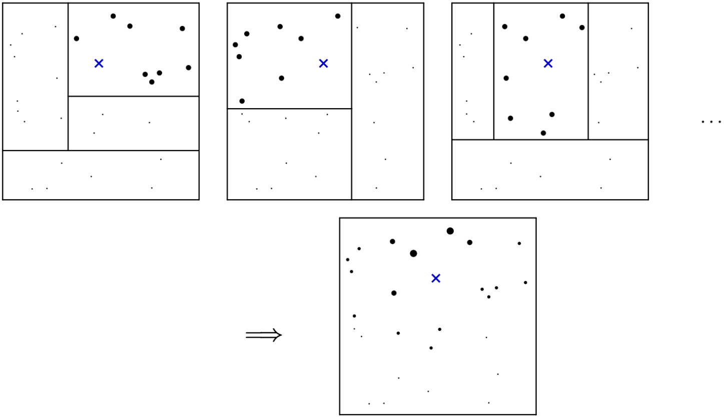
Forest-Based Local Estimation

Therefore, our problem of estimating $\theta(x)$ reduces to

$$(\hat{\theta}(x), \hat{\nu}(x)) \ \in \ argmin_{\theta,\nu} \left\{ \left\| \sum_{i=1}^{n} \alpha_i(x) \psi_{\theta,\nu}(O_i) \right\|_2 \right\}$$

When the above expression has a unique root, we can simply say that $(\hat{\theta}(x), \hat{\nu(x)})$ solves $\sum_{i=1}^{n} \alpha_i(x) \psi_{\hat{\theta}(x),\hat{\nu}(x)}(O_i) = 0$

Forest-Based Local Estimation



Forest-Based Local Estimation

As in Hothorn et al. (2004) and Meinshausen (2006), GRF obtains α_i by averaging the neighborhood

We first grow a set of B trees, and define $L_b(x)$. Then

$$\alpha_{bi}(x) = \frac{\mathbf{1}(\{X_i \in L_b(x)\})}{|L_b(x)|}, \;\; \alpha_i = \frac{1}{B}\sum_{b=1}^{B} \alpha_{bi}(x)$$

Note that

$\sum_{i=1}^{n} \alpha_i = \sum_i \frac{1}{B} \sum_b \alpha_{bi}(x) = \frac{1}{B} \sum_b \sum_i \alpha_{bi}(x) = \frac{1}{B} \sum_b 1 = 1$

Forest-Based Local Estimation

In summary...

- our goal: estimate solutions $(\theta(x), \nu(x))$ to the equations

  $$\mathbb{E}[\psi_{\theta(x),\nu(x)}(O_i)|X_i = x] = 0 \ \forall \ x \in \mathcal{X}$$

- we do so by solve

  $$(\hat{\theta}(x), \hat{\nu}(x)) \ \in \ argmin_{\theta,\nu} \left\{ \left\| \sum_{i=1}^{n} \alpha_i(x) \psi_{\theta,\nu}(O_i) \right\|_2 \right\}$$

Splitting to Maximize Heterogeneity

To assign weights to each training example, we need to establish the split rule.

- We denote a parent node by $P \subset \mathcal{X}$, and children node by $C_1, C_2 \subset \mathcal{X}$
- Then, we define $(\hat{\theta}_p, \hat{\nu}_p)(\mathcal{J})$ as follows:

$$(\hat{\theta}_P, \hat{\nu}_P)(\mathcal{J}) \in argmin_{\theta,\nu} \left\{ \left\| \sum_{\{i \in \mathcal{J} : X_i \in P\}} \psi_{\theta,\nu}(O_i) \right\|_2 \right\}$$

Splitting to Maximize Heterogeneity

Our split rule is to minimize

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

where

$$(\hat{\theta}_{C_j}, \hat{\nu}_{C_j})(\mathcal{J}) \in argmin_{\theta,\nu} \left\{ \left\| \sum_{\{i \in \mathcal{J} : X_i \in C_j\}} \psi_{\theta,\nu}(O_i) \right\|_2 \right\}$$

Splitting to Maximize Heterogeneity

Proposition 1

Suppose that the parent node P has a radius smaller than r for some value $r > 0$, and write $n_P = |\{i \in \mathcal{J} : X_i \in P\}|$ $n_{C_j} = |\{i \in \mathcal{J} : X_i \in C_j\}|$ $j = 1, 2$. Define

$$\Delta(C_1, C_2) \coloneqq n_{C_1} n_{C_2} / n_P^2 (\hat{\theta}_{C_1}(\mathcal{J}) - \hat{\theta}_{C_2}(\mathcal{J}))^2$$

Then, treating C_1, C_2, n_{C_1}, and n_{C_2} as fixed, and assuming that $n_{C_1}, n_{C2} \gg r^{-2}$, we have $err(C_1, C_2) = K(P) - \mathbb{E}[\Delta(C_1, C_2)] + o(r^2)$

$\Longrightarrow$Then, our split rule is equivalent to maximize $\Delta(C_1, C_2)$

Splitting to Maximize Heterogeneity

Problem with this approach?

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

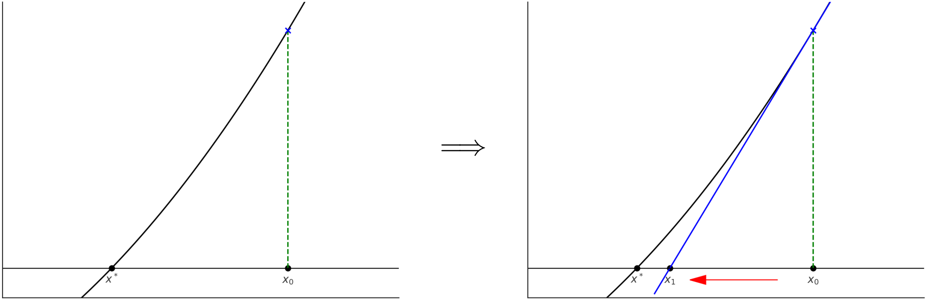
$$(\hat{\theta}_{C_j}, \hat{\nu}_{C_j})(\mathcal{J}) \in argmin_{\theta,\nu} \left\{ \left\| \sum_{\{i \in \mathcal{J} : X_i \in C_j\}} \psi_{\theta,\nu}(O_i) \right\|_2 \right\}$$

$\Longrightarrow$ Computationally too demanding!

The Gradient Tree Algorithm

Newton-Raphson Method

$$f'(x_0) = \frac{f(x_0) - f(x_1)}{x_0 - x_1} = \frac{f(x_0)}{x_0 - x_1} \implies x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$



The Gradient Tree Algorithm

We optimize an approximate criterion $\tilde{\Delta}(C_1, C_2)$ built using gradient-based approximations. For each child C, we use $\tilde{\theta}_C \approx \hat{\theta}_C$ as follows.

$$\tilde{\theta}_C = \hat{\theta}_P - \frac{1}{|\{i : X_i \in C\}|} \sum_{\{i:X_i \in C\}} \xi^T A_P^{-1} \psi_{\hat{\theta}_P, \hat{\nu}_P}(O_i)$$

where A_P is any consistent estimator of $\nabla \mathbb{E}[\psi_{\hat{\theta}_P, \hat{\nu}_P}(O_i) | X_i \in P]$, and ξ is a vector that picks out the θ-coordinate from (θ, ν)

The Gradient Tree Algorithm

- **Labeling step**

  we compute $\hat{\theta}_P, \hat{\nu}_P$, and A_P^{-1} and get pseudo-outcomes

  $$\rho_i = -\xi^T A_P^{-1} \psi_{\hat{\theta}_P, \hat{\nu}_P}(O_i)$$

- **Regression step**

  we split P into C_1 and C_2 such as to maximize

  $$\tilde{\Delta}(C_1, C_2) = \sum_{j=1}^{2} \frac{1}{|\{i : X_i \in C_j\}|} \left(\sum_{\{i: X_i \in C_j\}} \rho_i \right)^2$$

The Gradient Tree Algorithm

$$
\begin{aligned}
\tilde{\Delta}(C_1, C_2) &= \sum_{j=1}^{2} \frac{1}{|\{i : X_i \in C_j\}|} \left(\sum_{\{i: X_i \in C_j\}} \rho_i \right)^2 \\
&= \sum_{j=1}^{2} \frac{1}{n_{C_j}} \left(n_{C_j} (\tilde{\theta}_{C_j} - \hat{\theta}_P) \right)^2 \\
&= \sum_{j=1}^{2} n_{C_j} \left((\tilde{\theta}_{C_j} - \hat{\theta}_P) \right)^2 \\
&= n_{C_1} (\tilde{\theta}_{C_1} - \hat{\theta}_P)^2 + n_{C_2} (\tilde{\theta}_{C_2} - \hat{\theta}_P)^2 \\
&\approx n_{C_1} n_{C_2} / n_P^2 (\tilde{\theta}_{C_1} - \tilde{\theta}_{C_2})^2
\end{aligned}
$$

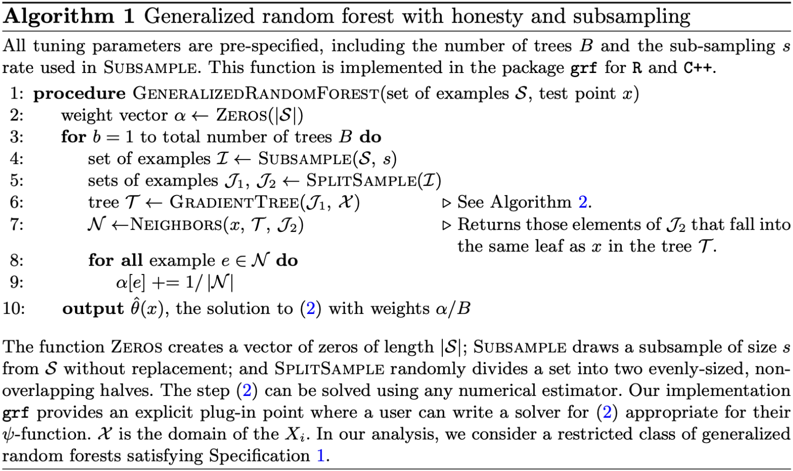
The Gradient Tree Algorithm

Proposition 2

If $|A_p - \nabla\mathbb{E}[\psi_{\hat{\theta}_P,\hat{\nu}_P}(O_i)|X_i \in P]| \xrightarrow{p} 0$, i.e., A_p is consistent, then $\Delta(C_1, C_2) = \tilde{\Delta}(C_1, C_2) + o_p(max\{r^2, 1/n_{C_1}, 1/n_{C_2}\}$

$\Longrightarrow$ It is possible to evaluate all possible split points along a given feature with only a single pass over the data in the parent node!

Building a Forest with Theoretical Guarantees



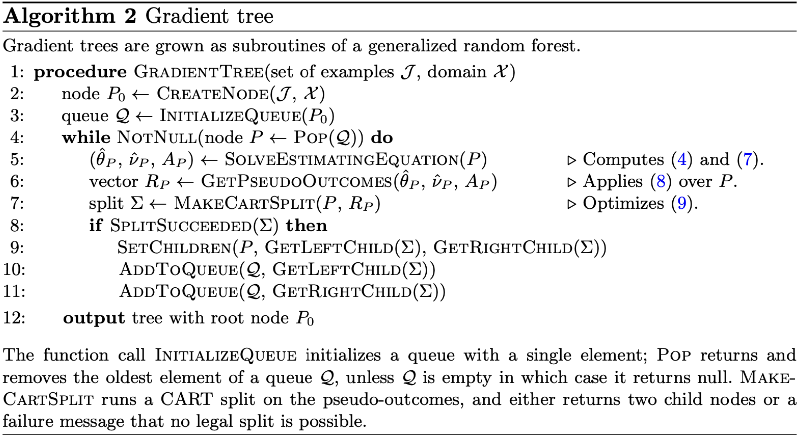
Algorithm 1 Generalized random forest with honesty and subsampling

All tuning parameters are pre-specified, including the number of trees B and the sub-sampling s rate used in SUBSAMPLE. This function is implemented in the package `grf` for `R` and `C++`.

```
procedure GENERALIZEDRANDOMFOREST(set of examples $\mathcal{S}$, test point $x$)
    weight vector $\alpha \leftarrow$ ZEROS($|\mathcal{S}|$)
    for $b = 1$ to total number of trees $B$ do
        set of examples $\mathcal{I} \leftarrow$ SUBSAMPLE($\mathcal{S}$, $s$)
        sets of examples $\mathcal{J}_1$, $\mathcal{J}_2 \leftarrow$ SPLITSAMPLE($\mathcal{I}$)
        tree $\mathcal{T} \leftarrow$ GRADIENTTREE($\mathcal{J}_1$, $\mathcal{X}$)        ▷ See Algorithm 2.
        $\mathcal{N} \leftarrow$NEIGHBORS($x$, $\mathcal{T}$, $\mathcal{J}_2$)        ▷ Returns those elements of $\mathcal{J}_2$ that fall into
                                                     the same leaf as $x$ in the tree $\mathcal{T}$.
        for all example $e \in \mathcal{N}$ do
            $\alpha[e]$ += $1/|\mathcal{N}|$
    output $\hat{\theta}(x)$, the solution to (2) with weights $\alpha/B$
```

The function ZEROS creates a vector of zeros of length $|\mathcal{S}|$; SUBSAMPLE draws a subsample of size s from $\mathcal{S}$ without replacement; and SPLITSAMPLE randomly divides a set into two evenly-sized, non-overlapping halves. The step (2) can be solved using any numerical estimator. Our implementation `grf` provides an explicit plug-in point where a user can write a solver for (2) appropriate for their ψ-function. $\mathcal{X}$ is the domain of the X_i. In our analysis, we consider a restricted class of generalized random forests satisfying Specification 1.

Building a Forest with Theoretical Guarantees



Algorithm 2 Gradient tree

Gradient trees are grown as subroutines of a generalized random forest.

```
procedure GradientTree(set of examples $\mathcal{J}$, domain $\mathcal{X}$)
    node $P_0 \leftarrow$ CreateNode($\mathcal{J}$, $\mathcal{X}$)
    queue $\mathcal{Q} \leftarrow$ InitializeQueue($P_0$)
    while NotNull(node $P \leftarrow$ Pop($\mathcal{Q}$)) do
        $(\hat{\theta}_P, \hat{\nu}_P, A_P) \leftarrow$ SolveEstimatingEquation($P$)        ▷ Computes (4) and (7).
        vector $R_P \leftarrow$ GetPseudoOutcomes($\hat{\theta}_P, \hat{\nu}_P, A_P$)        ▷ Applies (8) over $P$.
        split $\Sigma \leftarrow$ MakeCartSplit($P$, $R_P$)        ▷ Optimizes (9).
        if SplitSucceeded($\Sigma$) then
            SetChildren($P$, GetLeftChild($\Sigma$), GetRightChild($\Sigma$))
            AddToQueue($\mathcal{Q}$, GetLeftChild($\Sigma$))
            AddToQueue($\mathcal{Q}$, GetRightChild($\Sigma$))
    output tree with root node $P_0$
```

The function call InitializeQueue initializes a queue with a single element; Pop returns and removes the oldest element of a queue $\mathcal{Q}$, unless $\mathcal{Q}$ is empty in which case it returns null. MakeCartSplit runs a CART split on the pseudo-outcomes, and either returns two child nodes or a failure message that no legal split is possible.

Asymptotic Analysis

To develop our asymptotic analysis, we define

$$M_{\theta,\nu}(x) := \mathbb{E}[\psi_{\theta,\nu}(O)|X = x]$$

$$V(x) := V_{\theta(x),\nu(x)}(x) := \frac{\partial}{\partial(\theta,\nu)} M_{\theta,\nu}|_{\theta(x),\nu(x)}$$

$$\rho_i^*(x) := -\xi^T V(x)^{-1} \psi_{\theta(x),\nu(x)}(O_i)$$

$$\tilde{\theta}^* := \theta(x) + \sum_{i=1}^{n} \alpha_i(x) \rho_i^*(x) = \frac{1}{B} \sum_{b=1}^{B} \tilde{\theta}_b^*(x)$$

$$\tilde{\theta}_b^*(x) = \sum_{i=1}^{n} \alpha_{ib}(x)(\theta(x) + \rho_i^*(x))$$

Asymptotic Analysis

Theorem 3

Given assumptions, $(\hat{\theta}(x), \hat{\nu}(x)) \xrightarrow{p} (\theta(x), \nu(x))$

Lemma 4

Suppose $\hat{\theta}(x) \xrightarrow{p} \theta(x)$. Then

$$\sqrt{\frac{n}{s}}\left(\tilde{\theta}^{*}(x) - \hat{\theta}(x)\right) = \mathcal{O}\left(max\left\{s^{-\frac{\pi}{2}\frac{log\left((1-\omega)^{-1}\right)}{log(\omega^{-1})}}, \left(\frac{s}{n}\right)^{\frac{1}{6}}\right\}\right)$$

Asymptotic Analysis

Theorem 5 (CLT)

Suppose $Var[\rho_i^*(x)|X_i = x] > 0$. Then there is a sequence $\sigma_n(x)$

such that $(\hat{\theta}_n(x) - \theta(x))/\sigma_n(x) \Rightarrow \mathcal{N}(0, 1)$ and

$\sigma_n^2(x) = polylog(n/s)^{-1} s/n$

Sketch of Proof

Assumptions

- (A.1 Lipschitz x-signal) For fixed values of (θ, ν), we assume that $M_{\theta,\nu}(x)$ is Lipschitz continuous in x
- (A.5 Existence of solutions) We assume that $\forall (\alpha_i)_i$ with $\sum \alpha_i = 1$, minimizer $(\hat{\theta}, \hat{\nu})$ of $\sum_{i=1}^{n} \alpha_i(x)\psi_{\theta,\nu}(O_i)$ at least satisfies $\|\sum_{i=1}^{n} \alpha_i \psi_{\hat{\theta},\hat{\nu}}(O_i)\|_2 \leq C \cdot max\{\alpha_i\}$
- (A.6 Convexity) $\psi_{\theta,\nu}(O_i)$ is a negative subgradient of a convex function, $M_{\theta,\nu}(X_i)$ is the negative gradient of a strongly convex function

Sketch of Proof

1. As $n \to \infty$, leaf gets smaller, hence $\|X_i - x\|_2$ gets smaller
2. Thanks to A.1, $\|\psi_{\theta(x),\nu(x)}(O_i)\| \xrightarrow{p} 0$
3. Thanks to A.5, $\|\psi_{\hat{\theta}(x),\hat{\nu}(x)}(O_i)\| \xrightarrow{p} 0$
4. Thanks to A.6, $\|\psi_{\theta(x),\nu(x)}(O_i)\|, \|\psi_{\hat{\theta}(x),\hat{\nu}(x)}(O_i)\| \xrightarrow{p} 0$ imply

   $\theta - \hat{\theta} \xrightarrow{p} 0, \nu - \hat{\nu} \xrightarrow{p} 0$

Sketch of Proof

Assumptions

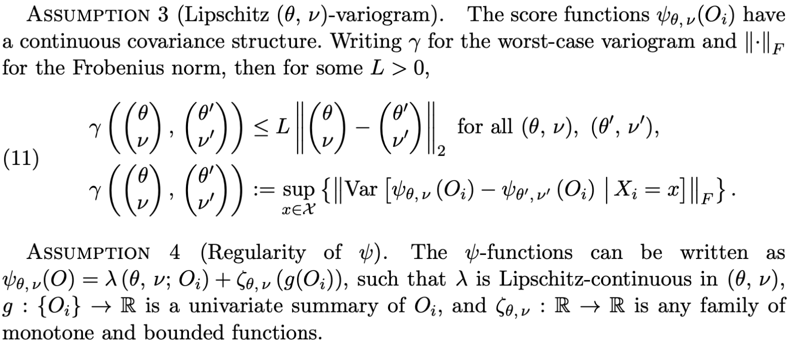
- (A.2 Smooth identification) When x is fixed, we assume that the M-function is twice continuously differentiable in (θ, ν) with a uniformly bounded second derivative, and $V(x) = V_{\theta(x),\nu(x)}(x)$ is invertible for all $x \in \mathcal{X}$

Sketch of Proof

1. Thanks to A.2, Taylor expansion of M function can be written with remainder term H with $\|H\| \leq c\varepsilon_n^2/2$
2. As leaf gets smaller, the first order term in the Taylor expansion is asymptotically equivalent to $V(x)$
3. By simple algebra,

   $\|\hat{\theta}(x) - (\theta(x) - V(x)^{-1}\Psi(\theta(x), \nu(x)))\|_2 \xrightarrow{p} 0$

Sketch of Proof

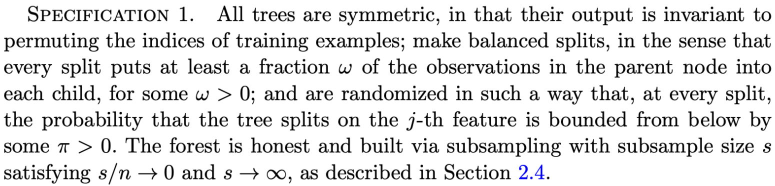


Assumption 3 (Lipschitz $(\theta,\, \nu)$-variogram). The score functions $\psi_{\theta,\, \nu}(O_i)$ have a continuous covariance structure. Writing γ for the worst-case variogram and $\|\cdot\|_F$ for the Frobenius norm, then for some $L > 0$,

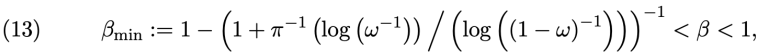
$$\begin{aligned}
&\gamma\left(\begin{pmatrix}\theta\\ \nu\end{pmatrix}, \begin{pmatrix}\theta'\\ \nu'\end{pmatrix}\right) \leq L\left\|\begin{pmatrix}\theta\\ \nu\end{pmatrix} - \begin{pmatrix}\theta'\\ \nu'\end{pmatrix}\right\|_2 \text{ for all } (\theta,\, \nu),\ (\theta',\, \nu'), \\
&\gamma\left(\begin{pmatrix}\theta\\ \nu\end{pmatrix}, \begin{pmatrix}\theta'\\ \nu'\end{pmatrix}\right) := \sup_{x\in\mathcal{X}}\left\{\left\|\operatorname{Var}\left[\psi_{\theta,\, \nu}\left(O_i\right) - \psi_{\theta',\, \nu'}\left(O_i\right) \,\middle|\, X_i = x\right]\right\|_F\right\}.
\end{aligned} \tag{11}$$

Assumption 4 (Regularity of ψ). The ψ-functions can be written as $\psi_{\theta,\, \nu}(O) = \lambda\left(\theta,\, \nu;\, O_i\right) + \zeta_{\theta,\, \nu}\left(g(O_i)\right)$, such that λ is Lipschitz-continuous in $(\theta,\, \nu)$, $g : \{O_i\} \to \mathbb{R}$ is a univariate summary of O_i, and $\zeta_{\theta,\, \nu} : \mathbb{R} \to \mathbb{R}$ is any family of monotone and bounded functions.

Sketch of Proof



Specification 1. All trees are symmetric, in that their output is invariant to permuting the indices of training examples; make balanced splits, in the sense that every split puts at least a fraction ω of the observations in the parent node into each child, for some $\omega > 0$; and are randomized in such a way that, at every split, the probability that the tree splits on the j-th feature is bounded from below by some $\pi > 0$. The forest is honest and built via subsampling with subsample size s satisfying $s/n \to 0$ and $s \to \infty$, as described in Section 2.4.



$$\beta_{\min} := 1 - \left(1 + \pi^{-1}\left(\log\left(\omega^{-1}\right)\right) \Big/ \left(\log\left(\left(1-\omega\right)^{-1}\right)\right)\right)^{-1} < \beta < 1, \tag{13}$$

Confidence Intervals

$$Var[\tilde{\theta}^*(x)] = \xi^T V(x)^{-1} H_n(x;\theta(x),\nu(x))(V(x)^{-1})^T \xi$$

$$H_n(x;\theta,\nu) = Var[\sum_{i=1}^{n} \alpha_i(x)\psi_{\theta,\nu}(O_i)]$$

$$\hat{\sigma}_n^2 = \xi^T \hat{V}(x)^{-1} \hat{H}_n(x)(\hat{V}(x)^{-1})^T \xi$$

$\Longrightarrow$ presents results based on a variant of the bootstrap of little bags algorithm (Sexton and Laake (2009)

Application: Quantile regression

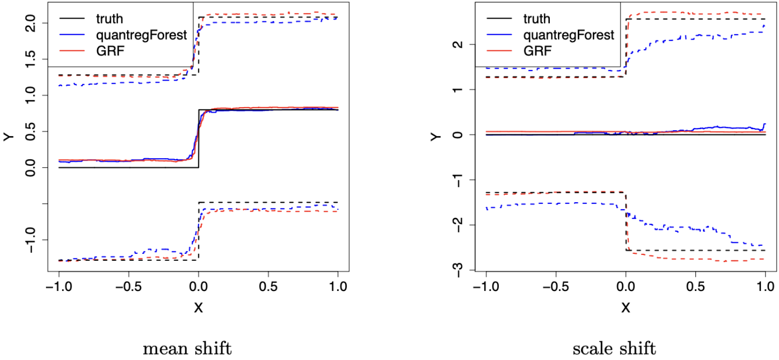
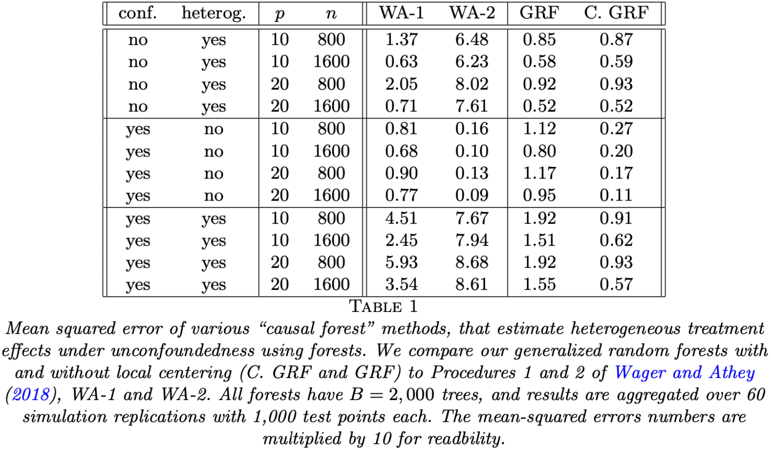


Figure: Comparison with Meinshausen (2006)

Application: Causal Forests



conf.	heterog.	p	n	WA-1	WA-2	GRF	C. GRF
no	yes	10	800	1.37	6.48	0.85	0.87
no	yes	10	1600	0.63	6.23	0.58	0.59
no	yes	20	800	2.05	8.02	0.92	0.93
no	yes	20	1600	0.71	7.61	0.52	0.52
yes	no	10	800	0.81	0.16	1.12	0.27
yes	no	10	1600	0.68	0.10	0.80	0.20
yes	no	20	800	0.90	0.13	1.17	0.17
yes	no	20	1600	0.77	0.09	0.95	0.11
yes	yes	10	800	4.51	7.67	1.92	0.91
yes	yes	10	1600	2.45	7.94	1.51	0.62
yes	yes	20	800	5.93	8.68	1.92	0.93
yes	yes	20	1600	3.54	8.61	1.55	0.57

TABLE 1

Mean squared error of various "causal forest" methods, that estimate heterogeneous treatment effects under unconfoundedness using forests. We compare our generalized random forests with and without local centering (C. GRF and GRF) to Procedures 1 and 2 of Wager and Athey (2018), WA-1 and WA-2. All forests have $B = 2{,}000$ trees, and results are aggregated over 60 simulation replications with 1,000 test points each. The mean-squared errors numbers are multiplied by 10 for readbility.

Figure: Comparison with Wager, Athey (2018)

Reference

- Susan Athey. Julie Tibshirani. Stefan Wager (2019). Generalized random forests. Ann. Statist. 47 (2) 1148 - 1178.
- Wager, S., Athey, S. (2018). Estimation and Inference of Heterogeneous Treatment Effects using Random Forests. Journal of the American Statistical Association, 113(523), 1228–1242.
- Meinshausen, N. (2006). Quantile Regression Forests. J. Mach. Learn. Res., 7, 983-999.