Heterogeneous treatment effects

Magnus Nielsen, SODAS, UCPH

Agenda

- Potential outcomes
- Matching
- Causal tree
- Causal forest
- Generalized random forest
- When and how to estimate
- Using CATE's

Goal: Recipes

Estimation of conditional average treatment effects

- Done using causal forests in either R or Python
- Assumes selection on observables
- Can also use instruments in instrumental forests

How to use conditional average treatment effects

Goal: Understanding

Try to get an intuitive understanding of what the methods do

- As requested: No focus on maths
  - Papers are quite technical

Main contribution of each paper in the ‘generalized random forest’ series

- Causal tree
- Causal forest
- Generalized random forest

Potential outcomes

The Rubin Causal Model

Denote T_i as the treatment variable

- $T_i = 1$ corresponds to unit i being treated, $T_i = 0$ is not treated

Define the potential outcomes

$$Y_i = \begin{cases} Y_i(1), & T_i = 1 \\ Y_i(0), & T_i = 0 \end{cases}$$

A minor problem

The observed outcome Y_i can be written in terms of potential outcomes:

$$Y_i = Y_i(0) + [Y_i(1) - Y_i(0)] \cdot T_i$$

$Y_i(1) - Y_i(0)$ is the *causal* effect of T_i on Y_i

We never observe the same individual i in both states

- Known as the **fundamental problem of causal inference**

Average treatment effect

We need some way of estimating the state we do not observe (the ***counterfactual***)

- Utilize that we observe both treated and untreated individuals

Perhaps we can do a naive comparison by treatment status?

$$\tau = E[Y_i | T_i = 1] - E[Y_i | T_i = 0]$$

Decomposing the average treatment effect

Utilizing that

$$Y_i = Y_i(0) + [Y_i(1) - Y_i(0)] \cdot T_i$$

We get the following

$$\begin{aligned} E[Y_i|T_i = 1] - E[Y_i|T_i = 0] = & E[Y_i(1)|T_i = 1] - E[Y_i(0)|T_i = 1] + \\ & E[Y_i(0)|T_i = 1] - E[Y_i(0)|T_i = 0] \end{aligned}$$

Possible bias

The average *causal* effect of T_i on Y

$$E[Y_i(1)|T_i = 1] - E[Y_i(0)|T_i = 1] = E[Y_i(1) - Y_i(0)|T_i = 1]$$

Difference in average $Y_i(0)$ between the two groups

$$E[Y_i(0)|T_i = 1] - E[Y_i(0)|T_i = 0]$$

Often referred to as ***selection bias***

- Likely to be different from 0 when individuals are allowed to self-select into treatment

Randomization fixes everything

Random assignment implies T_i is independent of potential outcomes

$$E[Y_i(0)|T_i = 1] = E[Y_i(0)|T_i = 0]$$

Intuition: non-treated individuals can be used as counterfactuals for treated

- *What would have happened to individual i had they not received the treatment*?
- Overcomes the fundamental problem of causal inference

Not always feasible

If randomization by us is not feasible, we must rely on nature:

- *Quasi-experiments*: Randomization happens by “accident”

Today we will consider

- Matching

Matching

Selection on observables

Construct counterfactual potential treated and control units

- We *match* observations across treatment and control based on similarity

Why: Matching controls for the covariates used

- Excludes (observable) confounders

An alternative to matching is regression analysis, which is basically the same (Angrist &

k-nearest neighbor matching

For a given characteristic x, find k nearest treated (S_1) and untreated (S_0) observations

We can then estimate the conditional average treatment effect (CATE) using the following estimator

$$\tau(x) = \frac{1}{k} \sum_{i \in S_1(x)} Y_i - \frac{1}{k} \sum_{i \in S_0(x)} Y_i$$

Nearest could defined by distance in covariates or in propensities

Why aggregate?

When performing matching, it isn't necessary to aggregate up to an average treatment effect

We can instead just stop when we have estimated the CATE

- Treatment effect for given characteristics x

$$\tau(x) = E[Y_i(1) - Y_i(0) | X = x]$$

Question

Do any of the previously studied supervised models create ‘neighborhoods’? If yes, which?

Causal trees

Trees as matching

Trees inherently create partitions

- We partition to reduce impurity within leafs
  - Gini, MSE, etc.

One big problem: We’re matching on outcomes

- Why is this a big problem?

Spurious extreme values

Spurious extreme values of Y_i are going to be matched with other spurious extreme values

What does this mean?

- Confidence intervals are no longer valid!

How to fix?

We utilize sample splitting

An observation can be used for either

- Creating neighborhoods
- Estimation of within-leaf treatment effect

This is called being an *honest* tree (versus adaptive), and is proposed by Athey & Imbens (2016), *Recursive partitioning for heterogeneous causal effects*

- Causal trees

Question

What is the main drawback of honest estimation?

Modified splitting criterion

Split to identify heterogeneous treatment effects

- But unbiased estimates result in higher variance

Modify criterion in anticipation of this

- Reward finding heterogeneous effects, penalize high variance

It works!

Table 1. Simulation study

$N^{tr} = N^{est}$ Estimator	Design 1 500	Design 1 1,000	Design 2 500	Design 2 1,000	Design 3 500	Design 3 1,000
			No. of leaves			
TOT	2.9	3.2	*2.9*	3.5	3.6	5.4
F-A	6.1	13.1	6.3	13.0	6.2	13.0
TS-A	4.0	5.4	3.4	5.1	3.4	6.6
CT-A	4.0	5.5	3.2	3.7	3.5	5.4
F-H	6.0	12.9	6.3	13.0	6.3	13.1
TS-H	4.3	7.8	5.6	11.4	5.9	12.4
CT-H	4.2	7.6	5.6	11.4	6.1	12.5
			Infeasible MSE divided by infeasible MSE for CT-H*			
TOT-H	1.554	1.938	1.089	1.069	1.081	1.042
F-H	1.790	1.427	1.983	2.709	1.502	2.085
TS-H	0.971	0.963	1.183	1.145	1.178	1.338
			Ratio of infeasible MSE: Adaptive to honest†			
TOT-A/TOT-H		1.021		0.754		0.717
F-A/F-H		0.491		0.985		0.993
T-A/T-H		0.935		0.841		0.918
CT-A/CT-H		0.929		0.851		0.785
			Coverage of 90% confidence intervals – adaptive			
TOT-A	0.82	0.85	0.78	0.81	0.69	0.74
F-A	0.89	0.89	0.83	0.84	0.82	0.82
TS-A	0.84	0.84	0.78	0.82	0.75	0.75
CT-A	0.83	0.84	0.78	0.82	0.76	0.79
			Coverage of 90% confidence intervals – honest			
TOT-H	0.90	0.90	0.90	0.89	0.89	0.90
F-H	0.90	0.90	0.90	0.90	0.90	0.90
TS-H	0.90	0.90	0.91	0.91	0.89	0.90
CT-H	0.89	0.90	0.90	0.90	0.89	0.90

*$\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est}, \pi^{\mathrm{Estimator}}(\mathcal{S}^{tr}))/\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est}, \pi^{\mathrm{CT-H}}(\mathcal{S}^{tr}))$.

†$\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est} \cup \mathcal{S}^{tr}, \pi^{\mathrm{Estimator-A}}(\mathcal{S}^{est} \cup \mathcal{S}^{tr}))/\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est}, \pi^{\mathrm{Estimator-H}}(\mathcal{S}^{tr}))$.

Source: Athey & Imbens (2016)

Question

How can one increase performance of trees for a fixed sample?

Causal forest

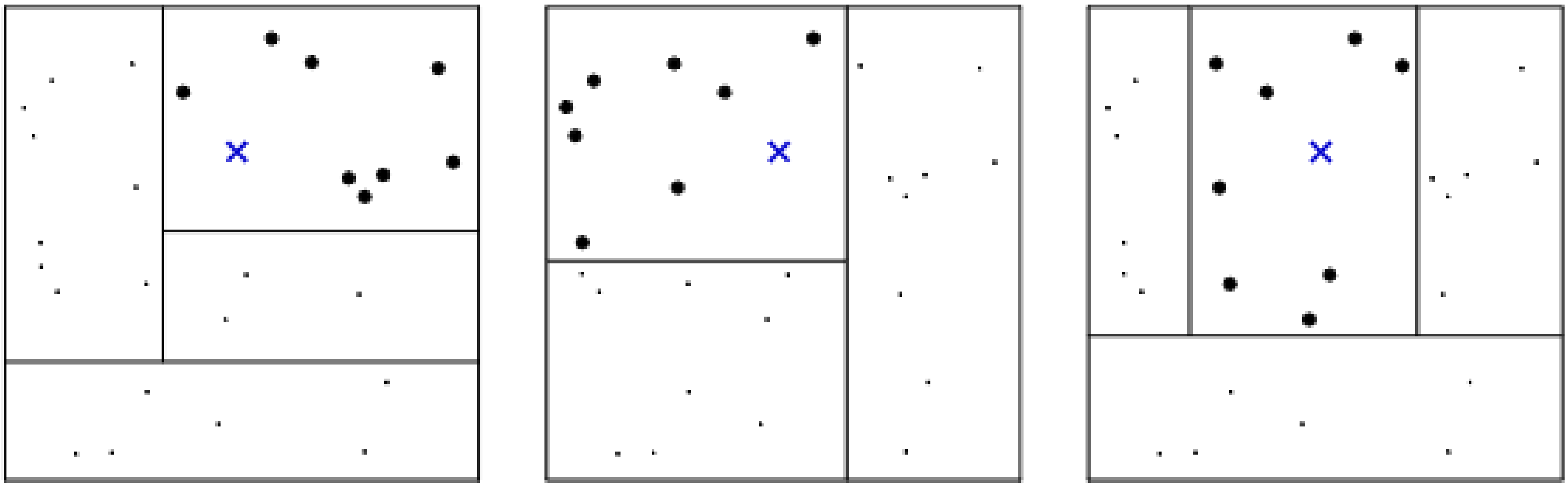
Growing a forest

Wager & Athey (2018), *Estimation and inference of heterogeneous treatment effects using random forests*, propose the causal forest, which is an ensemble of causal trees

- An ensemble of average trees often performs better than a single highly optimized tree, see Breiman (2001)

Reduces variance and creates less sharp boundaries

Many partitions



Source: Athey et al. (2019)

An average CATE

For each tree b, calculate the CATE of the observation as in the causal tree (eq. 5 in paper), denoted $\hat{\tau}_b(x)$

For ensemble of B trees, CATE estimator is then

$$\hat{\tau}(x) = B^{-1} \sum_{b=1}^{B} \hat{\tau}_b(x)$$

Asymptotic inference!

As long as trees are honest, we can perform asymptotic inference

- Can also deliver confidence intervals for regression forests

Two ways of achieving honesty, double-sampling (as in causal tree) or propensity trees

- Double-sampling better at heterogeneous treatment functions
- Propensity trees better at unconfounding

Reconciled in generalized random forest

Performance compared to k-NN

	Mean-squared error			Coverage		
d	CF	10-NN	100-NN	CF	10-NN	100-NN
2	0.02 (0)	0.21 (0)	0.09 (0)	0.95 (0)	0.93 (0)	0.62 (1)
5	0.02 (0)	0.24 (0)	0.12 (0)	0.94 (1)	0.92 (0)	0.52 (1)
10	0.02 (0)	0.28 (0)	0.12 (0)	0.94 (1)	0.91 (0)	0.51 (1)
15	0.02 (0)	0.31 (0)	0.13 (0)	0.91 (1)	0.90 (0)	0.48 (1)
20	0.02 (0)	0.32 (0)	0.13 (0)	0.88 (1)	0.89 (0)	0.49 (1)
30	0.02 (0)	0.33 (0)	0.13 (0)	0.85 (1)	0.89 (0)	0.48 (1)

Source: Wager & Athey (2018)

Coverage until $d = 10$, performance degrades after

- Lower MSE and better coverage than k-NN

Generalized random forest

Reframing

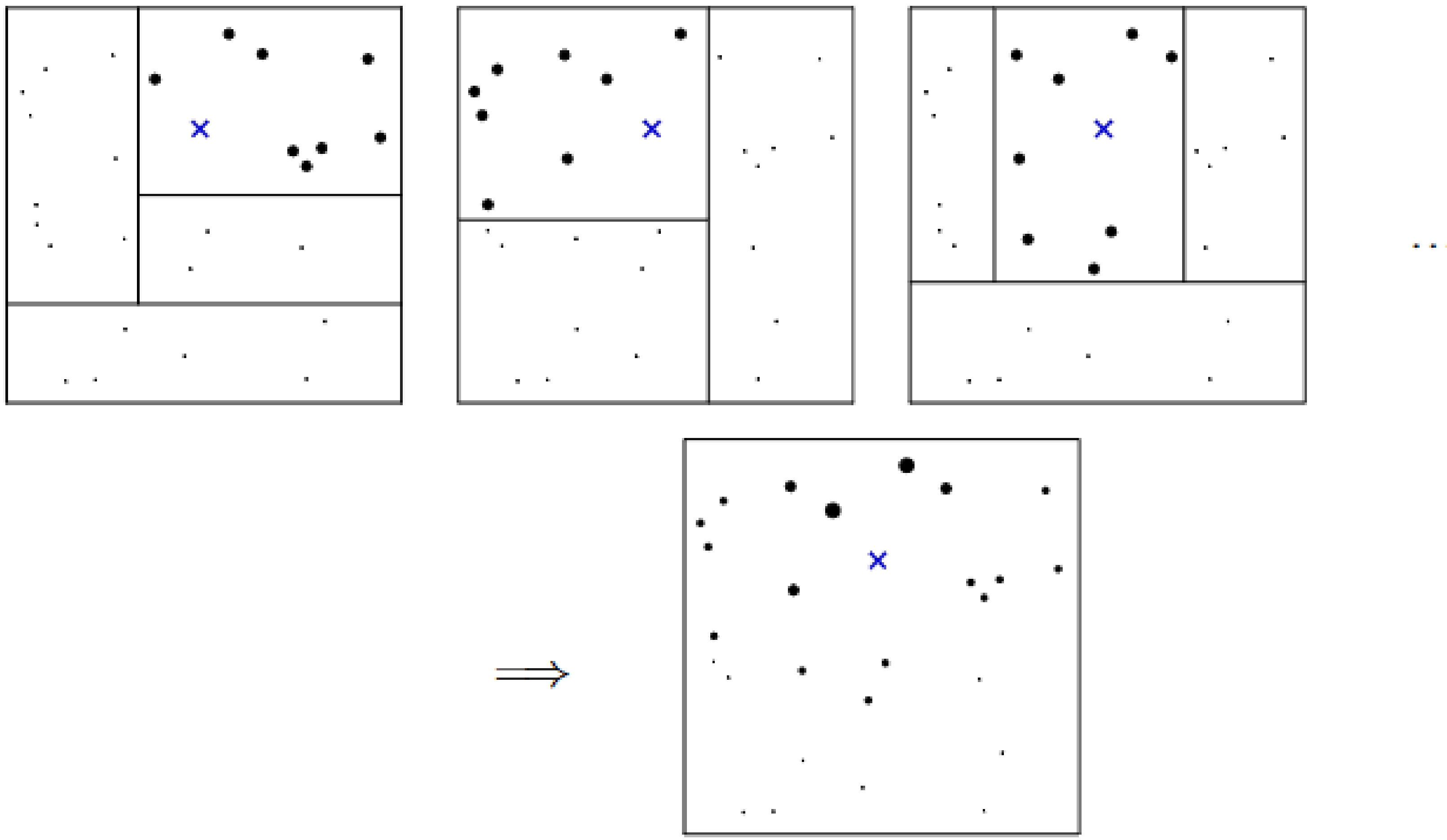
Trees create neighborhoods with CATE's

- Causal forest CATE was an average over these within the forest

Athey et al. (2019), *Generalized random forests*, reframe it as creating a weighting function usable in maximum likelihood estimation

- Trees create weights based on how often observations are in the same leaf

Adaptive nearest neighbor estimation



Source: Athey et al. (2019)

Curse of dimensionality

Previously used weights based on similarity but had strong issues with the curse of dimensionality

Generalized random forests use data-driven heterogeneity to lessen this

- The most important dimensions are *‘discovered’*

If really high dimensional, consider double machine learning (next session)

Many different possibilities

By rephrasing into moment conditions, multiple possibilities arise

- Selection on observables/randomization (causal forest)
- Instrumental variables (instrumental forest)
- Quantile regression (quantile forest)

Note that causal forests can refer to both causal forests in Wager & Athey (2018) and in Athey et al. (2019)

- Packages available implement the generalized random forest version (`econml` and `grf`)

Some slight changes

Compared to the causal forest in Wager & Athey (2018), a couple of other things are changed:

- A more efficient gradient based loss (sec. 2.2)
- Centering outcome and treatment before creating forests (sec. 6.1.1)
- Bootstrapped confidence intervals (sec. 4)
  - Supports cluster-based sampling, although there is no norm w.r.t. treatment heterogeneity and clustering yet, see e.g. discussion in Athey & Wager (2019)

Causal forests versus 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

Source: Athey et al. (2019)

Assumptions

Most critical assumptions are the “regular” assumptions:

- Causal forest: Selection on observables and overlap
- Instrumental forest: Relevance and exclusion

Test these as you usually would (if possible)

There are some additional technical assumptions (sec. 3)

When and how to estimate

Heterogeneity in treatment effects

Two approaches

- Data driven heterogeneity
  - Using non-parametric models such as causal forests
- A priori heterogeneity / theory
  - Using (semi) parametric models and interactions, e.g. OLS

When to choose which?

- Use data driven heterogeneity when
  - Aim is to use CATE’s for policy where you want to maximize impact
  - You have no prior or suspect non-linear heterogeneity
- Use a priori heterogeneity when
  - You have a specific theory you want to test, i.e. specific subgroups are adversely affected
  - Sample sizes are not powerful enough to utilize non-parametric methods

We go where the packages are

Best of both worlds?

Exercises will cover both R and Python

- I will write R code as text
- The grf documentation has a lot of useful tips and examples specifically for causal forests
- It is a very user friendly package

I do not expect you to learn R for this one thing, but wanted to supply some code

Out of bag or out of sample

When performing causal inference, we need to retain honesty

Either split the data or use out of bag predictions

- Splitting data: Fit model in one part and make inference in other part
- Out of bag: Utilize only trees in which the observation was not used to create partitions
  - Not possible for double machine learning

Under either scenario, causal inference is valid

Dimensionality

Causal forests perform best for relatively low-dimensional problems

- Consider encoding categorical variables as a single ordinal variable if possible
  - Trees make no assumptions in regards to linearity
- Consider keeping only the most relevant variables

Consider using a double machine learning variant if you have many covariates

Hyperparameters

The grf algorithm reference has some recommendations, amongst others:

- For small samples, increase the amount of data samples used for obtaining splits (`honesty.fraction`)
- For large samples, tune the trees to create shallower trees
- For tight CI's, increase amount of trees

`grf` implements an option which tunes hyperparameters called `tune.parameters`

The documentation is great

The packages really try to make causal inference more accessible

- The documentation is really good!
  - Look at the `grf` tutorials, top centre
  - Look at the `econml` user guide

Some examples

Athey and Wager have multiple examples where they implement models and describe their considerations

- Athey & Wager (2019) is an application of causal forests
  - What do they consider when implementing causal forests?
- Wager & Athey (2018) also has some examples intertwined between the math
  - Quantile, instrumental, causal forest

Using CATE’s

What to do after estimation

Many different things to do after estimating CATE's, broadly categorized:

- Testing whether heterogeneity exists
  - Median test, RATE & TOC
- Examining and using heterogeneity
  - RATE & TOC, feature importance, explainability, policy

Median split test

A simple naive test: Split data based on median CATE

- Calculate ATE within each group
- Test whether there's a significant difference

See evaluating a causal forest fit for an example

Note: When calculating ATE's, use built-in functionality to calculate doubly-robust ATE's

- `average_treatment_effect` function in R
- `ate` method in Python

A less naive approach

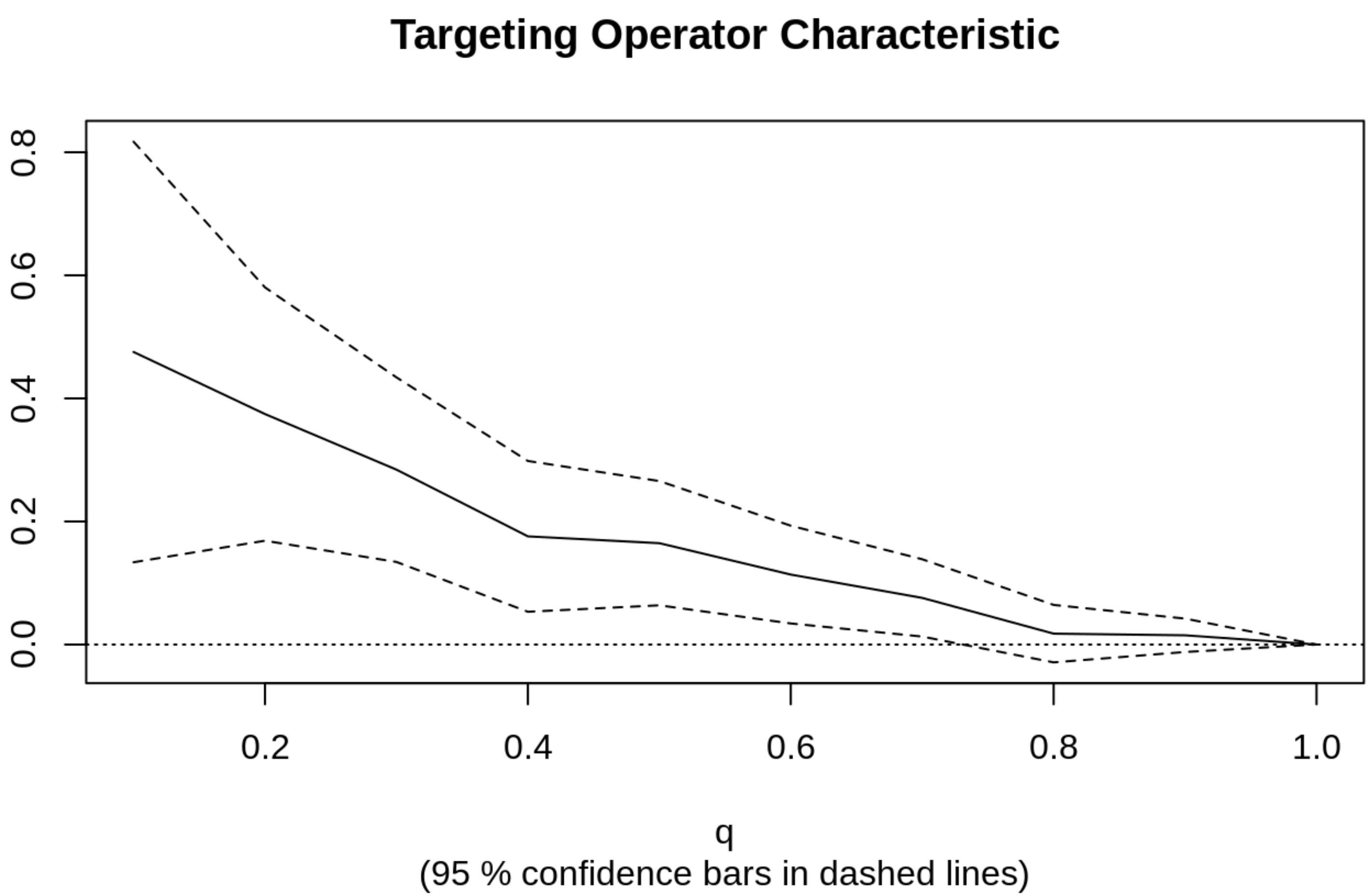
Alternatively, consider the Rank-Weighted Average Treatment Effect (RATE) introduced by Yadlowsky et al. (2021)

Procedure is as follows:

- Rank CATE's according to some rule
  - Size of CATE, a covariate or predicted risk
- For each percentile:
  - Estimate difference between ATE for people above that percentile and overall ATE

This creates the Targeting Operator Characteristic (TOC) curve

Targeting Operator Characteristic



Source: grf documentation, 2023

Using the TOC

Usage:

- Inspect the TOC curve to assess heterogeneity and select optimal cutoff
  - When does treating more people become unfeasible?
  - See also the policy learning methods
- Calculate the area under the TOC (AUTOC) and test whether it is different from zero

Not just testing for heterogeneity

If AUTOC is not different from zero, there are two possible explanations:

- There are no heterogeneous treatment effects
- Prioritization rule was not efficient at prioritizing treatment

All implemented in R in `rank_average_treatment`, see here

What variables drive heterogeneity

How does one interpret a fully non-parametric CATE estimation?

- We have assumed no structure, so we don’t know:
  - What drives heterogeneity
  - Which way it drives heterogeneity

Sadly, the packages are not very consistent in what is offered

- I’ll cover what’s available

Split based methods

A simple way to asses what drives heterogeneity is to look at splits in the trees

- How often does the model split on a feature weighted by depth
  - Implemented in `variable_importances` function in R
- How often does the model split on a feature weighted by depth and amount of heterogeneity created
  - Implemented in `feature_importances` method in Python

Question

What do we know about bias and explainability methods that use splits to calculate feature importance?

SHAP values

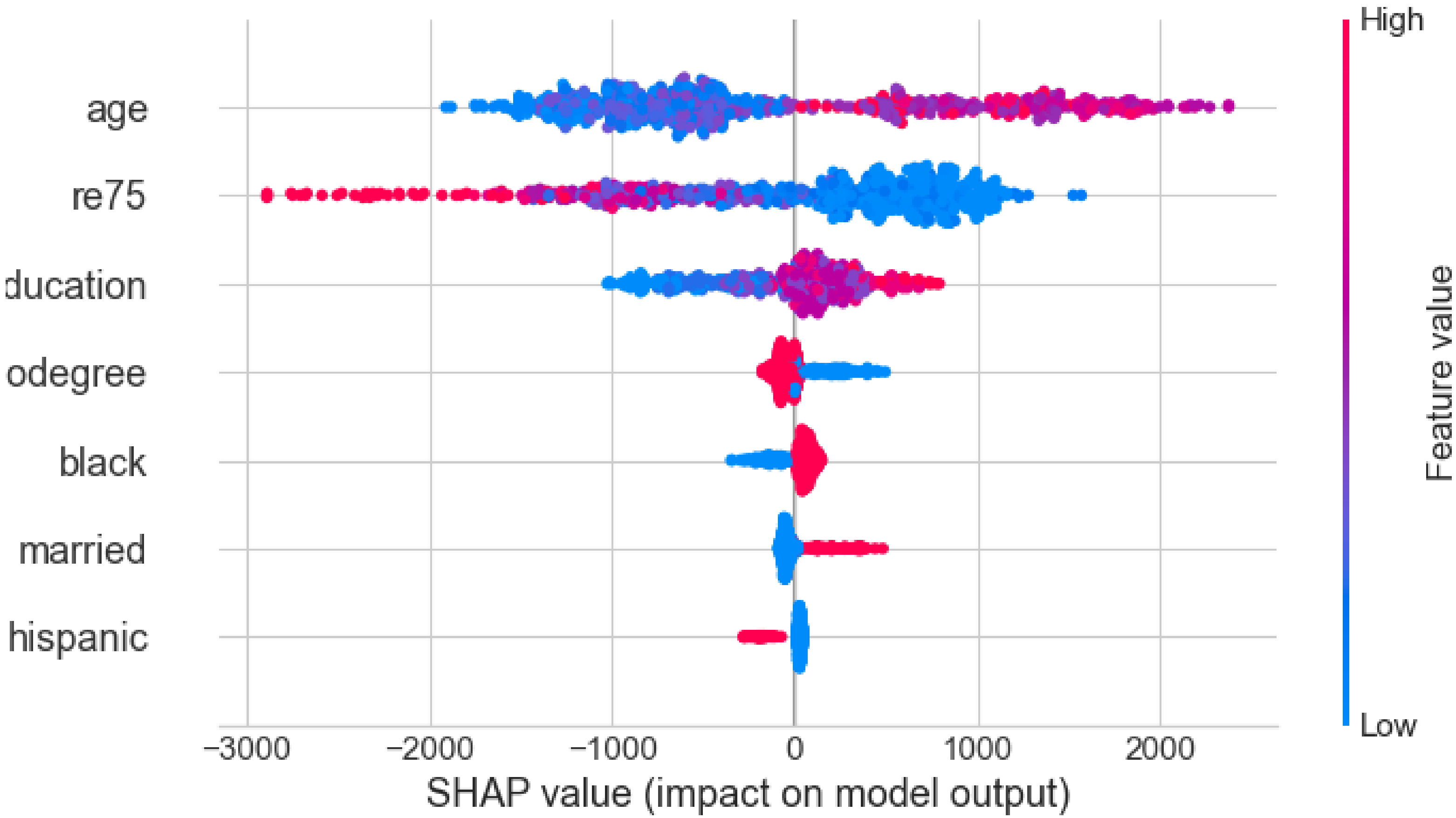
A neat way to explain heterogeneity is to use SHAP values!

- Same interpretation as in session 6
- Bar plots, summary plots etc.

To my knowledge only available in `econml`

- Models have `shap_values()` method

SHAP explanation example



Source: Me (2023)

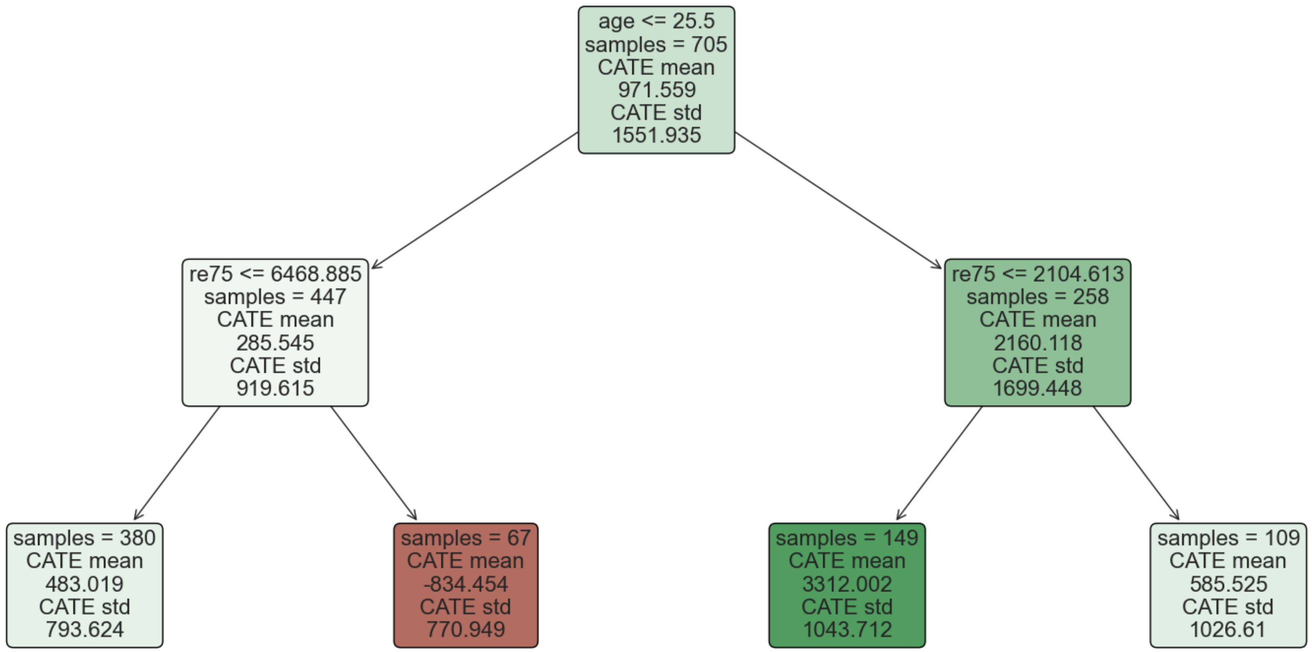
Predicting CATE's

Another possibility is to predict the CATE with an intrinsically interpretable model

- Use a shallow decision tree
  - Available in `econml.interpretation` as SingleTreeCateInterpreter
- Use a linear projection
  - Available in `grf` as best_linear_projection

One could easily train own models

CATE as a tree



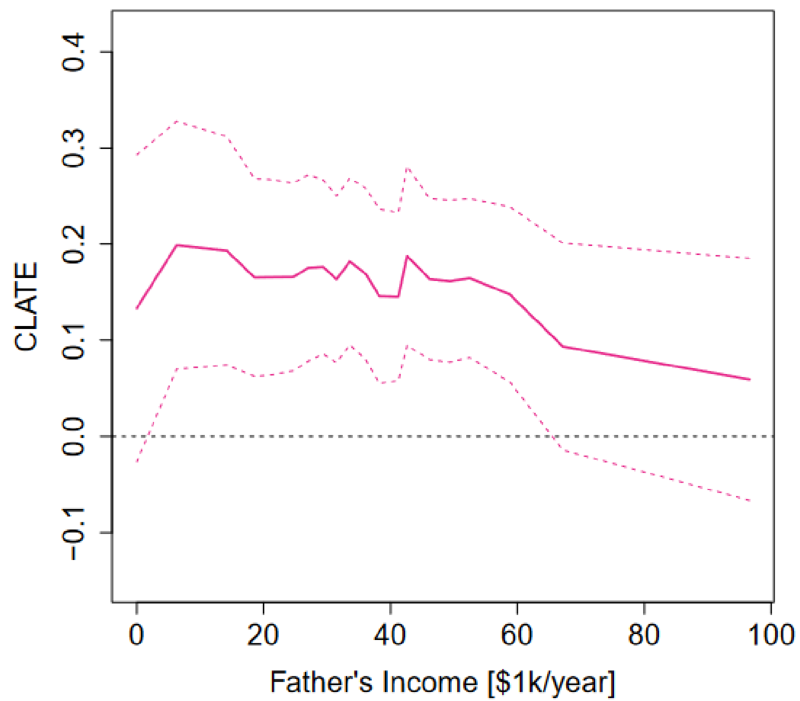
Source: Me (2023)

Counterfactual examples

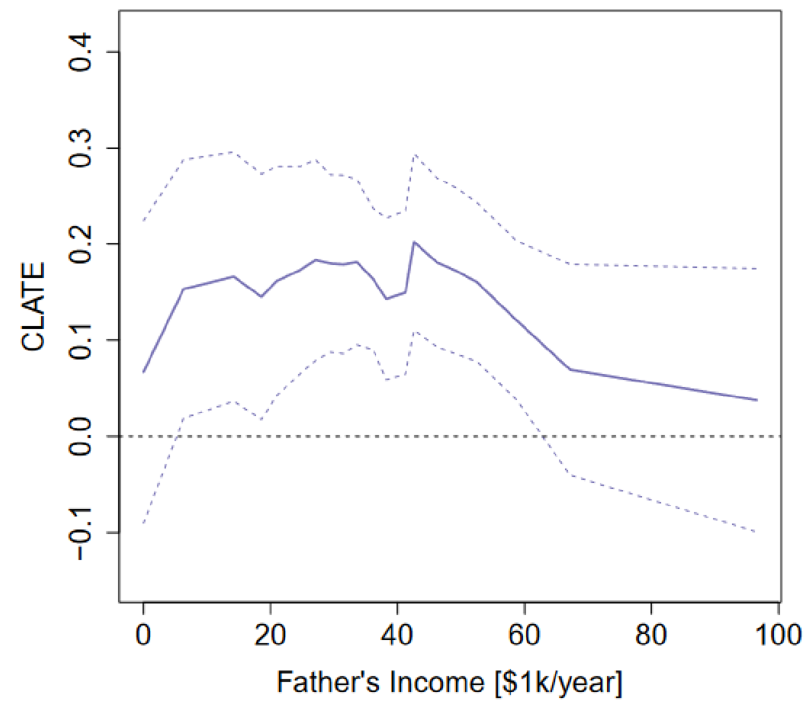
On the basis of what one has found, or priors, one can estimate counterfactual CATE's

- Interpreted much like partial dependence plots
  - How does CATE vary with covariate x
- Use fixed background covariates
  - Valid confidence intervals

Childbirth and labor force participation CLATE



Mother 18 years old at first birth



Mother 22 years old at first birth

Source: Athey et al. (2019)

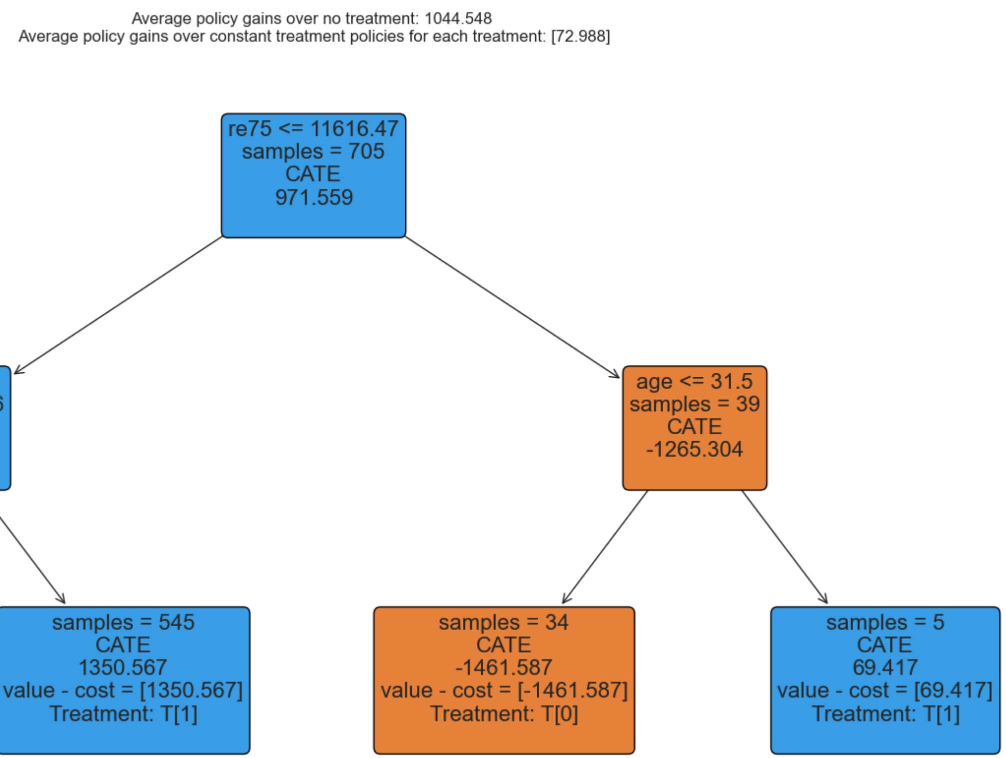
Creating policies

One can create policies based on the CATE's

- Use `policytree` in R, Sverdrup et al. (2020)
  - Doubly robust, see documentation
- Use `SingleTreePolicyInterpreter` in `econml.cate_interpreter` in Python
  - Not doubly robust, see documentation
  - `econml.policy` has trees and forests that are both doubly robust and not, see documentation

See e.g. Athey & Wager (2021)

Example treatment policy



Source: Me (2023)

References

Angrist, J. D., & Pischke, J. S. (2009). Mostly harmless econometrics: An empiricist's companion. Princeton university press.

Athey, S., & Imbens, G. (2016). Recursive partitioning for heterogeneous causal effects. Proceedings of the National Academy of Sciences, 113(27), 7353-7360.

Athey, S., Tibshirani, J., & Wager, S. (2019). Generalized random forests.

Athey, S., & Wager, S. (2019). Estimating treatment effects with causal forests: An application. Observational Studies, 5(2), 37-51.

Athey, S., & Wager, S. (2021). Policy learning with observational data. Econometrica, 89(1), 133-161.

Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

Sverdrup, E., Kanodia, A., Zhou, Z., Athey, S., & Wager, S. (2020). policytree: Policy learning via doubly robust empirical welfare maximization over trees. Journal of Open Source Software, 5(50), 2232.

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.

To the exercises!