Seminar Data Science for Economics

MSc. Economics program

Madina Kurmangaliyeva
m.kurmangaliyeva@uvt.nl

Spring 2020

Tilburg University

Why does ML fail for inference even when there are no confounders?

What if no confounders?

Remember when we discussed Double Selection, we saw that ML fails to do proper inference because it omits important confounders.

But what if there are no confounders?

What if we work with a perfect Randomized Control Trial?

Naive ML?

Can we use ML now to simply learn $\hat{y}(D = 1, X = x)$ and $\hat{y}(D = 0, X = x)$ and use the difference between the two as our estimate for ATE?

In this lecture, you see that the answer is still "no". But there is a way to fix it.

Reason 1 of why ML fails even under RCT: Naive ML is biased

Naive Decision Trees Bias: simple example

What happens if we use Decision Trees naively?

Suppose we want to learn $\mu(d)$ and the data sample $\mathcal{S}$ looks this way:

	y	d
1	y_1	L
2	y_2	R
3	y_3	R
...	...	...
n	y_n	L

In other words, $x_i \in \{L, R\}$ and y_i is continuous variable.

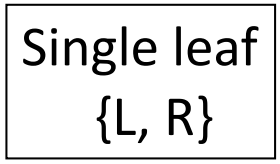
Unbiased estimator for treatment effects

Notice that the difference in sample means $\bar{Y}_L - \bar{Y}_R$ is an unbiased estimator of treatment effect in population $\mu(d = L) - \mu(d = R)$ thanks to unconfoundedness (RCT):

$$ATE = E(\bar{Y}_L - \bar{Y}_R) = \mu(d = L) - \mu(d = R)$$

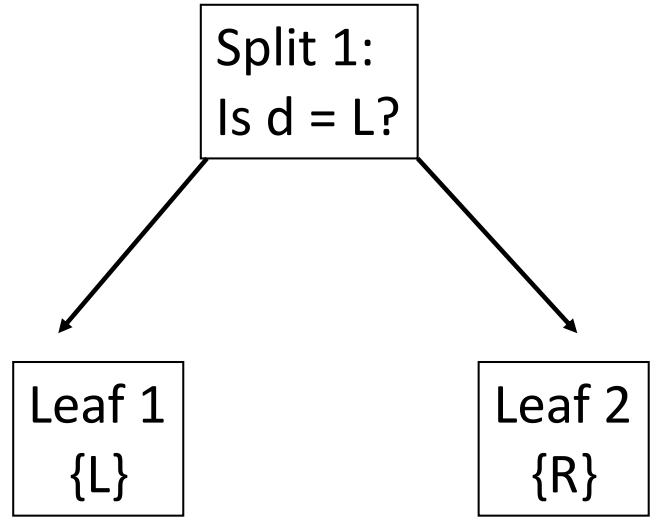
but the Naive Decision Tree estimator will NOT GIVE an unbiased estimator for $\mu(d = L) - \mu(d = R)$.

Potential tree 1



Tree does NOT find difference in outcomes by the value of d

Potential tree 2



Tree finds difference in outcomes by the value of d

Simple tree

For a decision tree, there are just two ways to partition the X-space.

$$\pi(\mathcal{S}) = \begin{cases} \{\{L, R\}\}, & \text{if } |\bar{Y}_L - \bar{Y}_R| \leq c, \\ \{\{L\}, \{R\}\}, & \text{if } |\bar{Y}_L - \bar{Y}_R| > c \end{cases}$$

given the sample $\mathcal{S}$ and some positive c.

Naive decision tree bias

In case the decision tree finds the difference between group L and group R:

$$0 < E(\bar{Y}_L - \bar{Y}_R | \bar{Y}_L - \bar{Y}_R > c) \neq E(\bar{Y}_L - \bar{Y}_R)$$

In case the decision tree does not find any difference:

$$0 = E(\bar{Y}_L - \bar{Y}_R | \bar{Y}_L - \bar{Y}_R \leq c) \neq E(\bar{Y}_L - \bar{Y}_R)$$

Naive ML estimator for treatment effects is BIASED, even if there are no confounders (pure RCT).

Why? Because we used the same data to decide how to partition into leaves and to estimate mean values within each leaf.

General setup

A general setup

Setting:

RCT. e.g. a company randomly charges either a default low price or an experimental high price for a product.

The customers that receive the high price are called “treated” ($D = 1$), while those who see the default price are called “control group” ($D = 0$).

A general setup (cont'd)

The company observes:

- the treatment status of each customer d_i
- how much profit each customer brings to the company (i.e., price by quantity demanded) $y_i \geq 0$
- a set of characteristics for each customer x_i (e.g., gender, age, home address, etc.)

What we want

We want to have:

- an unbiased estimate of $\mu(D = 1, X = x)$ and $\mu(D = 0, X = x)$
- a data-driven partition of X-space to capture heterogenous treatment effects
- standard errors for the TE estimator for each partition

Setting

Random Assignment of D, i.e., no confounders

Want to predict customer i's treatment effect:

$Y_i^{D=0}$ – profits from customer i at low price $Y_i^{D=1}$ – profits from customer i at high price

$\tau_i = Y_i^{D=1} - Y_i^{D=0}$ – but we never observe that because we can measure the reaction of a customer to either a high price or a low price. We cannot offer high and low price simultaneously.

Reason 2 why ML fails fails even under RCT: the core reason.

Fundamental problem of causal inference

In other words, we **can observe**:

$$\underbrace{Y_i^{D=1}(D=1)}_{\text{How } i \text{ reacts to high price when price is high}} \quad \text{or} \quad \underbrace{Y_i^{D=0}(D=0)}_{\text{How } i \text{ reacts to low price when price is low}}$$

but we **can NOT observe**:

$$\underbrace{Y_i^{D=0}(D=1)}_{\text{How } i \text{ reacts to low price when price is high}} \quad \text{or} \quad \underbrace{Y_i^{D=1}(D=0)}_{\text{How } i \text{ reacts to high price when price is low}}$$

If only we had known the counterfactuals...

If we had been able to observe the counterfactuals directly, then the data would have looked as:

	$y^{D=1}$	$y^{D=0}$	τ_i	d_i	x_i
1	$y_1^{D=1}$	$y_1^{D=0}$	τ_1	0	x_1
2	$y_2^{D=1}$	$y_2^{D=0}$	τ_2	1	x_2
3	$y_3^{D=1}$	$y_3^{D=0}$	τ_3	0	x_3
...	...	...	...	...	...
n	$y_n^{D=1}$	$y_n^{D=0}$	τ_n	1	x_n

Then, we would have been able to simply plug τ_i as the target variable and used directly cross-validated ML.

Reality

In reality, all we see is missing data

	$y^{D=1}$	$y^{D=0}$	τ_i	d_i	x_i
1	?	$y_1^{D=0}$	?	0	x_1
2	$y_2^{D=1}$	?	?	1	x_2
3	?	$y_3^{D=0}$	?	0	x_3
...	...	...	...	...	...
n	$y_n^{D=1}$	?	?	1	x_n

So we cannot use ML, we do not observe τ_i!

What to do?

But we CAN modify the classical Decision Tree objective function and estimation procedure for inference to get Causal Trees.

Causal Trees (Athey and Imbens, 2016)

Setup: CEF

Define a Conditional Expectation Function of the quantity for an individual with the set of characteristics:

$$\mu(d, x) = E(Y|D = d, X = x) \tag{1}$$

Then, treatment effect conditional on observables is:

$$\tau(x) = \mu(1, x) - \mu(0, x) \tag{2}$$

What we want

We want to have:

- an unbiased estimate of $\mu(D = 1, X = x)$ and $\mu(D = 0, X = x)$
- a data-driven partition of X-space to capture heterogenous treatment effects
- but partitions should be not too small to keep standard errors for the TE estimator narrow
- and yes, we do want standard errors for the TE estimator

Preview

Causal trees solve the bias problem by adopting Honest Splitting:

- uses an independent sample to estimate leaf means
- modifies the splitting and cross-validation objective functions to generate unbiased estimates, but
- accounting for the fact that more splits help capturing heterogeneous TE, but more lead to less precise TE estimators due to smaller leaves

Transitioning from classical Decision Trees to Causal Trees

The causal tree procedure consists of several building blocks (i.e., parts of Decision Tree procedure which we change for our “causal” needs)

Building block 1. Adding the estimation sample

For growing a causal tree, wou need to split data into two samples *at random*:

1. Training sample $\mathcal{S}^{tr}$ – use to construct a decision tree
2. Estimation sample $\mathcal{S}^{est}$ – a new element in the DT formula, we use this sample to estimate mean values for each leaf for a given decision tree that has been fit using training sample

The estimation sample is needed to avoid bias in the estimation of means μ.

(Plus, you may need a test sample.)

Building block 2. Honest Target

Remember for normal DTs we minimize the sum of squared residuals (RSS) at each split

For causal trees, we modify the objective function with two adjustements (in red):

$$MSE_{\mu}(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr}) \equiv \frac{1}{\#(S^{te})} \sum_{i \in \mathcal{S}^{te}} \left\{ \left(Y_i - \hat{\mu}(X_i; \underbrace{\mathcal{S}^{est}}_{\text{instead of } \mathcal{S}^{tr}}, \Pi^{tr}) \right)^2 \underbrace{- Y_i^2}_{\text{Normalization}} \right\}$$

where Π^{tr} is the partition of X based on training data

Building block 3. Honest Splitting

Honest Splitting uses Honest Target and explicitly treats $\mathcal{S}^{est}$ as a random variable

Hence, we have to work with minimizing Honest Target **in expectation**: $EMSE_{\mu}(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr})$

Building block 3. Honest Splitting (cont'd)

Ok, but how do we minimize $EMSE_{\mu}(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr})$ in practice without even touching the estimation sample?

(remember that we must decide how to split the data only based on the training sample)

Building block 3. Honest Splitting formula

Athey and Imbens (2016) show that the following formula is an unbiased estimator for $EMSE_\mu(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr})$ assuming that the leaf shares are approximately equal in the estimation and training samples:

$$\widehat{EMSE}_\mu(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr}) \equiv \underbrace{-\frac{1}{N^{tr}} \sum_{i \in \mathcal{S}^{tr}} \hat{\mu}^2(X_i; \mathcal{S}^{tr}, \Pi^{tr})}_{\text{Same as in classical DT objective}}$$

$$+ \underbrace{\left(\frac{1}{N^{tr}} + \frac{1}{N^{est}}\right) \sum_{l \in \Pi^{tr}} \overbrace{S^2_{\mathcal{S}^{tr}}(l(x; \Pi^{tr}))}^{\text{Variance within a leaf}}}_{\approx var(\hat{\mu});\ \text{New term thanks to Honest Target}}$$

Notice that the in-sample splitting criteria needs only the training sample and the number of observations in the estimation sample.

Building block 3. Honest Splitting formula explained

The last term in the formula accounts for the variance of the estimator of the mean. (remember that $var(\hat{\mu}_x) = \frac{var(x)}{N}$)

Creating finer partitions would allow to:

1. capture more heterogeneity in the data, i.e., the $\hat{\mu}^2$ term [Same as in normal DT]
2. but also it will increase the variance of the estimator in each leaf because of a smaller sample size [this is a new term] $\Longrightarrow$Therefore, the use of Honest Target penalizes small-sized leaves

Building block 3. Honest Splitting formula explained (cont'd)

Notice, that if for classical DT without stopping criteria, the tree will grow so big such that every observation is in its own leaf,

With Honest Target, it will in general stop earlier on its own, since small leaves result in higher variance of the mean estimator.

Building block 4. Honest cross-validation

Even if $\widehat{EMSE}_{\mu}(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr})$ is unbiased for a single split,

it is not when we use it for recursive partitioning (making splits again and again).

Solution: Cross-validation by penalizing tree complexity.

Building block 5. Modifying the Honest Splitting for Treatment effects

Constructing our estimator for the treatment effect as a difference at predicted values of y for treated vs. controls using Honest Target:

$$\hat{\tau}(x; \Pi^{tr}) \equiv \hat{\mu}(d = 1, x; \mathcal{S}, \Pi^{tr}) - \hat{\mu}(d = 1, x; \mathcal{S}, \Pi^{tr}) \tag{3}$$

Finally, the Causal Tree objective function

Combining building blocks 1 to 5, the in-sample splitting criteria for Causal Trees is as follows (differences with the formula for μ in red):

$$\widehat{EMSE}_{\tau}(\mathcal{S}^{te}, \mathcal{S}^{est}, \Pi^{tr}) \equiv \underbrace{-\frac{1}{N^{tr}} \sum_{i \in \mathcal{S}^{tr}} \hat{\tau}^2(X_i; \mathcal{S}^{tr}, \Pi^{tr})}_{\text{rewards high heterogeneity}}$$

$$+ \underbrace{\left(\frac{1}{N^{tr}} + \frac{1}{N^{est}}\right) \sum_{l \in \Pi^{tr}} \left(\frac{S^2_{\mathcal{S}^{tr}_{treat}}(l)}{p} + \frac{S^2_{\mathcal{S}^{tr}_{control}}(l)}{1-p} \right)}_{\text{Penalizes splits leading to small leafs}}$$

where p is the probability of treatment, $\mathcal{S}^{tr}_{treat}(l)$ is a sample of treated observations within a leaf.

How to grow a Causal Tree (all steps)

1. Use training sample to grow trees using recursive tree splitting by minimizing the in-sample splitting criteria on the previous slide and penalizing complexity at different levels of λ
2. Cross-validate to find the best λ and get the corresponding tree
3. Use estimation sample to find unbiased estimates of $\hat{\tau}(l)$ for each leaf

The Causal Tree objective function explained

The criteria for Causal Trees:

- rewards a partition for finding strong heterogeneity in treatment effects
- penalizes a partition for creating variance in leaf estimates

Splitting not just for Heterogeneous Effects

Note that the Causal tree may want to split the sample into two groups even if the treatment effects are the same in those groups. Why?

Answer: it does so to decrease the variance of the treatment effect estimator when covariates affect the mean outcome.

In other words, some partitions in a Causal Tree exist NOT to capture heterogeneous treatment effects but to capture heterogeneous outcomes, narrowing down the standard errors of the TE estimators in those partitions.

Causal Forest

Causal Forest is the Random Forest but using Causal trees.

You can use `causal_forest()` function from `grf` package in R. (See Tutorial)

Summary: RT vs CT

	Regression Tree	Causal Tree
Splitting rule	min in-sample RSS	min in-sample Honest Target
Predictions based on	training sample	separate estimation sample
Segments X for heterogeneity	in outcomes	in treatment effects
Segments X to decrease $S^2(l)$	no	yes
If no stopping criteria	# of leaves = N	keeps leaves bigger to decrease $S^2(l)$

Summary: Data science vs Econometrics

	Data Science	E'metrics
Task	Prediction	Inference
Measure of success	Out-of-sample MSE	Estimator's st. error
Asymptotic theory	Not needed	Essential
(Cross-)validation	Essential	Not used
Data	High-dimensional	Low-dimensional
Variables selection	Data-driven	Theory-driven
Assumptions	Stable environment	Random assignment

In this course we saw how to use Data Science (prediction) techniques for causal inference.

This course summary

In this course, you learned new ML techniques:

- NN
- Decision Trees
- Lasso, Ridge
- Bootstrapping
- Cross-validation (because of Bias-Variance trade-off)
- Boosting
- Bagging

This course summary (cont'd)

But also why you cannot use ML naively for causal inference.

And how you should use ML for causal inference to bring data-driven approach to:

1. automatically search for instrumental variables (with Jan)
2. eliminate confounders using DS or DML:
    - Miracle! **effortless search for the functional form**
    - mostly useful for observational studies...
    - if you work for government agencies, think-tanks, academia
3. automatically find heterogeneity in treatment effects using Causal Forests
    - Miracle! **effortless segmentation of individuals** by how different they are in their reaction to treatment
    - mostly useful for A/B testing in a lot of business applications ...
    - if you work for private companies