Seminar Data Science for Economics

MSc. Economics program

Madina Kurmangaliyeva
m.kurmangaliyeva@uvt.nl

Spring 2020

Tilburg University

Why using ML naively for inference fails?

Temptation to use ML for inference in observational studies

You might feel tempted to:

- train a Machine Learner to predict y from d (treatment 0/1) and Z,
- then get prediction at $d = 0$: $\hat{y}(d = 0, Z)$
- and the prediction at $d = 1$: $\hat{y}(d = 1, Z)$.
- to estimate the treatment effect naively as:

$$\beta^{Naive} = \hat{y}(d = 0, Z) - \hat{y}(d = 0, Z) \tag{1}$$

- **but you should never do that!**

Why?

… but ML is for prediction, it does not care about confounders

The reason we cannot do that is because of confounding variables:

- Confounding variables are usually **highly correlated with treatment**.
- And the machine learner will **tend to drop either the treatment or the confounder**, as both substitute each other in terms of prediction quality.
- Choosing only one variable among two (or more) substitutes is **because we penalize ML** for overfitting.
- So the ML ”tries hard” to find a way to predict well using the smallest number of variables.

Example: college effect

For example, you are interested in the effect of **college** degree on the mid-life **income** of individuals.

However, mid-life **income** depends a lot on the **parental income**: richer kids grow up and inherit wealth, while poorer kids do not inherit anything.

Moreover, **parental income** also affects the probability that the individual will go to **college** or not: richer kids can afford higher education, while poor kids cannot, or richer kids went to better schools thus guaranteeing to be admitted to college, etc.

Example: college effect (cont'd)

Hence, we will have two major groups: rich kids who attend college and poor kids who do not.

There will be some poor kids who attend college, and rich kids who do not, but those are not that big in comparison to the two major groups.

Example: naive ML for college effect

You use a Machine Learner on the data

- Your Machine Learner may decide to use **only** **parental income** to predict mid-life **income**, dropping **college** degree. Or vice versa.
- If it drops **college** degree, then your "estimate" of college degree effect will be zero, which is not true.
- If it drops **parental income**, then your estimate will be biased upwards: it will include the true effect of college degree PLUS the direct effect of the parental income (the inheritance).

Example Naive ML (bottomline)

- Your Machine Learner is “happy”, because it did its job of predicting mid-life income very well.
- Since its goal is prediction, it will actually most likely drop the confounder or the treatment, something we really do not want when we want to see the (causal) effect.
- When we care about prediction only, we do not care about the ingredients of the “black box”.
- Hence, we **should not use the model trained for prediction** to answer causal questions.

Formalized example Naive ML

To formalize, suppose that z causally affects d and causally affects y. And d affects y. (e.g., z = family income, d = college degree, y = mid-life income). And the true causal model is:

$$y = \beta d + \gamma z + e \tag{2}$$

$$d = \psi z + u \tag{3}$$

We can always rewrite the y equation by substituting d with z as:

$$y = (\beta\psi + \gamma)z + \epsilon_1 \text{ (where } \epsilon_1 = \beta u + e) \tag{4}$$

OR by substituting z with d

$$y = (\beta + 1/\psi)d + \epsilon_2 \text{ (where } \epsilon_2 = e - 1/\psi u) \tag{5}$$

Formalized example Naive ML (bottomline)

Hence, the ML algorithm can decide to use just one variable to predict y:

$$\hat{y}^{(1)}(d, z) = (\beta\psi + \gamma)z \tag{6}$$

OR

$$\hat{y}^{(2)}(d, z) = (\beta + 1/\psi)d \tag{7}$$

In the first case, your approach will estimate $\hat{y}^{(1)}(d = 1, z) - \hat{y}^{(1)}(d = 0, z) = 0$ (which is wrong)

In the second case, your approach will estimate $\hat{y}^{(2)}(d = 1, z) - \hat{y}^{(2)}(d = 0, z) = (\beta + 1/\psi)$ (which is also wrong)

What to do?

Avoid using naive ML estimators

Use Double Selection, Double Machine Learning, and Causal Trees instead