Treatment effect estimation

J.Josse, N. Acharki, B. Neal, S. Wager, J. Peters

Refresher on Treatment Effect Estimation

Framework

- n iid samples $(X_i, T_i, Y_i(0), Y_i(1)) \in \mathbb{R}^d \times \{0,1\} \times \mathbb{R} \times \mathbb{R}$
- Note $Y_i = Y_i(T_i)$, the observed data is: (Y_i, X_i, T_i)
- One has $Y_i = T_i Y_i(1) + (1 - T_i) Y_i(0)$

Individual causal effect of the treatment: $\Delta_i = Y_i(1) - Y_i(0)$
Problem: Δ_i never observed (only observe one outcome/indiv).

Refresher on Treatment Effect Estimation

X_1	X_2	X_3	T	Y(0)	Y(1)
5	1	F	1	NaN	10
-1	2	M	1	NaN	5
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\cdots$	$\cdots$	$\cdots$	0	6	NaN
$\cdots$	$\cdots$	$\cdots$	1	NaN	8

Refresher on Treatment Effect Estimation

Framework

- n iid samples $(X_i, T_i, Y_i(0), Y_i(1)) \in \mathbb{R}^d \times \{0,1\} \times \mathbb{R} \times \mathbb{R}$
- Note $Y_i = Y_i(T_i)$, the observed data is: (Y_i, X_i, T_i)
- One has $Y_i = T_i Y_i(1) + (1 - T_i) Y_i(0)$

Definitions

- Individual causal effect of the treatment: $\Delta_i = Y_i(1) - Y_i(0)$
- Average treatment effect (ATE) $\tau = \mathbb{E}[\Delta_i] = \mathbb{E}[Y_i(1) - Y_i(0)]$
- Propensity score $e(x) = \mathbb{P}(T_i = 1 | X_i = x)$
- Conditional response surface $\mu_t(x) = \mathbb{E}[Y_i(t) | X_i = x]$
- Variance (same for t) $\sigma(x) = V[Y_i(t) | X_i = x]$
- Response surface $m(x) = \mathbb{E}[Y_i | X_i = x]$

Refresher on Treatment Effect Estimation

Assumptions

1. Unconfoundedness : $\{Y_i(0), Y_i(1)\} \perp\!\!\!\perp T_i | X_i$
2. Overlap : $\eta < e(x) < 1 - \eta$ for some $\eta > 0$

Estimation and identification

- IPW, regression adjustement, matching

Refresher on Treatment Effect Estimation

IPW estimator

$$\widehat{\tau}_{IPW} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{T_i Y_i}{\widehat{e}(X_i)} - \frac{(1-T_i)Y_i}{1-\widehat{e}(X_i)}\right)$$

- Matching between similar individuals
- Consistent estimator of τ as long as $\widehat{e}$ is consistent

Difference in conditional means estimator

$$\widehat{\tau}_{OLS} = \frac{1}{n}\sum_{i=1}^{n}(\widehat{\mu}_1(x) - \widehat{\mu}_0(x))$$

Consistent estimator of τ as long as $\widehat{\mu}_t(x)$ are consistent

Refresher on Treatment Effect Estimation

IPW estimator

$$\widehat{\tau}_{IPW} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{T_i Y_i}{\widehat{e}(X_i)} - \frac{(1-T_i)Y_i}{1-\widehat{e}(X_i)}\right)$$

Propensity score: model treatment assignment as function of covariates, ignore outcome model

Difference in conditional means estimator

$$\widehat{\tau}_{OLS} = \frac{1}{n}\sum_{i=1}^{n}(\widehat{\mu}_1(x) - \widehat{\mu}_0(x))$$

Covariate adjustment: model outcome as function covariates, ignore treatment model

Refresher on Treatment Effect Estimation

Why double robust estimation?

- The two estimators $\widehat{\tau}_{IPW}$ and $\widehat{\tau}_{OLS}$ are sensitive to model misspecification
- Doubly robust: combine these estimators and create an estimator which is consistent if at least one of the models is well-specified
- We shall give two examples of doubly robust estimators
  - IPW with covariate balancing propensity score (CBPS)
  - Augmented IPW

Doubly robust estimation

Assumptions

Linear-logistic model:

1. $e(x) = \mathbb{P}(T_i = 1 | X_i = x) = \frac{1}{1+^{x^T \alpha}}$
2. $Y_i(t) = \mu_t(X_i) + \varepsilon_i(t)$ with $\mu_t(x) = x^T \beta(t)$ for $t \in \{0, 1\}$

The parameters of the model are α, $\beta(0)$ and $\beta(1)$.

ATE estimation

- Estimator of ATE of the form

$$\widehat{\tau} = \frac{1}{n} \sum_i \left(\widehat{\gamma}_1(X_i) T_i Y_i - \widehat{\gamma}_0(X_i)(1 - T_i) Y_i \right)$$

- Define relevant weights $\widehat{\gamma}_1(\cdot)$ and $\widehat{\gamma}_0(\cdot)$ using the explicit expression of the propensity score?

Doubly robust estimation

$$
\begin{aligned}
\widehat{\tau} &= \frac{1}{n}\sum_i \left(\widehat{\gamma}_1(X_i)T_iY_i - \widehat{\gamma}_0(X_i)(1-T_i)Y_i\right) \\
&= \frac{1}{n}\sum_i \left(\widehat{\gamma}_1(X_i)T_iY_i(1) - \widehat{\gamma}_0(X_i)(1-T_i)Y_i(0)\right) \\
&= \frac{1}{n}\sum_i \left(\widehat{\gamma}_1(X_i)T_i(X_i^T\beta_1+\varepsilon_i(1)) - \widehat{\gamma}_0(X_i)(1-T_i)(X_i^T\beta_0+\varepsilon_i(0))\right) \\
&= \frac{1}{n}\sum_i \left(\widehat{\gamma}_1(X_i)T_i(X_i^T\beta_1+\varepsilon_i(1))\right) + \overline{X}^T\beta_1 - \overline{X}^T\beta_1 \\
&\quad -\frac{1}{n}\sum_i \left(\widehat{\gamma}_0(X_i)(1-T_i)(X_i^T\beta_0+\varepsilon_i(0))\right) + \overline{X}^T\beta_0 - \overline{X}^T\beta_0
\end{aligned}
$$

Doubly robust estimation

- We obtain

$$\begin{aligned}\widehat{\tau} &= \overline{X}^T(\beta_1-\beta_0)+\left(\frac{1}{n}\sum_i \widehat{\gamma}_1(X_i)T_iX_i-\overline{X}\right)^T\beta_1 \\ &\quad -\left(\frac{1}{n}\sum_i \widehat{\gamma}_0(X_i)T_iX_i-\overline{X}\right)^T\beta_0+\text{ other terms}\end{aligned}$$

- Aim : find the value of the α parameter involved in the definition of the PS, such that

$$\widehat{\gamma}_0=1/(1-e(X_i)),\, \widehat{\gamma}_1=1/e(X_i)$$

cancels the two last terms of the sum.

Doubly robust estimation

- It is such the case if $\frac{1}{n}\sum_i \widehat{\gamma}_1(X_i)T_iX_i - \overline{X} = 0$
- In this case

$$\widehat{\tau}_{CBPS} = \overline{X}^T(\beta_1 - \beta_0) + \frac{1}{n}\sum_i \left(\frac{T_i(Y_i - \mu_1(X_i))}{e(X_i)} - \frac{(1-T_i)(Y_i - \mu_0(X_i))}{1 - e(X_i)}\right)$$

  with $e(\cdot)$ defined by the Logistic model of parameter α
- The resulting propensity estimate a covariate balancing propensity score (CBPS)
- The name emphasize the fact that $\alpha(1)$ achieves moment balance between the features X_i in full sample and the weighted features X_i in the treated sample

Doubly robust estimation

Properties

1. Under linear-logistic models, τ_{CBPS} has "best" asymptotic variance
2. The estimator remains consistent in either one of the following cases:
   - Outcome model is linear but propensity score $e(x)$ is not logistic.
   - Propensity score $e(x)$ is logistic but outcome model is not linear. Note that the asymptotic variance might be different in these cases.

See : Imai, Kosuke, and Marc Ratkovic. Covariate balancing propensity score. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 76(1), 2014

Doubly robust estimation

Practical examples using the following package
See the website
`https://import-balance.org/docs/docs/overview/`

Doubly robust estimation

AIPW estimator

$$
\begin{aligned}
\widehat{\tau}_{IPW} \quad = \quad & \frac{1}{N}\sum\left(\frac{T_i(Y_i-\widehat{\mu}_1(X_i))}{\widehat{\mathbb{P}}(X_i)}+\widehat{\mu}_1(X_i)\right) \\
& -\frac{1}{N}\sum\left(\frac{(1-T_i)(Y_i-\widehat{\mu}_0(X_i))}{(1-\widehat{\mathbb{P}}(X_i))}+\widehat{\mu}_0(X_i)\right)
\end{aligned}
$$

Doubly robust estimation

- Possibility to use any (machine learning) procedure such as random forests, deep nets, etc. to estimate $\widehat{e}(\cdot)$ and $\widehat{\mu}_t(\cdot)$.
- Let machine learning focus on what it's good at (accurate predictions), and then uses its outputs for efficient treatment effect estimation.
- τ is a causal parameter, i.e. property we wish to know about a population. It is not a parameter of a model

Doubly robust estimation

Properties

The estimator $\widehat{\tau}_{IPW}$ is consistent if either the $\widehat{\mu}_t(\cdot)$ are consistent or $\widehat{e}(\cdot)$ is consistent

An example with Python

Heterogeneous Treatment Effect

- The treatment may have no effect in average but may differ significantly according to the indiviudals
- Estimate this causal effect taking into account the caracteristics of individuals?
- Heterogenous treatment effect vs avrage treatment effect

Heterogeneous Treatment Effect

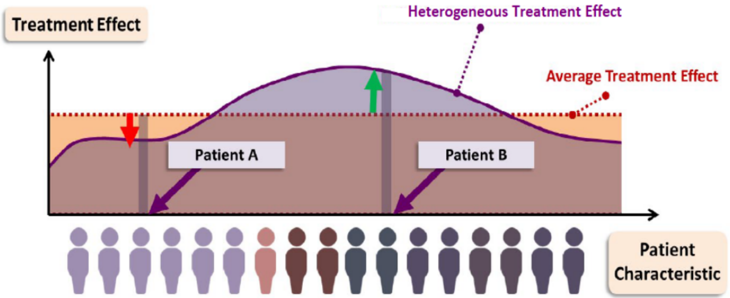


Illustration of the difference between the Average Treatment Effect and Individualized Treatment Effects (Bica et al. 2021)

Heterogeneous Treatment Effect

Some examples

- Average effect of a drug is 0, but positive for men and negative for women.
- Police body cameras cause a decline in the use of force by officers in large police departments, but have no effect for officers in small police departments
- Impact of Google ranking, depends on your profile (search Michael Jordan)

Heterogeneous Treatment Effect

Definition

For a given vector of covariates x, we define the Conditional Average Treatment Effect (CATE) function by

$$\tau(x) = \mathbb{E}[Y(1) - Y(0)|X = x]$$

Estimation of CATE?

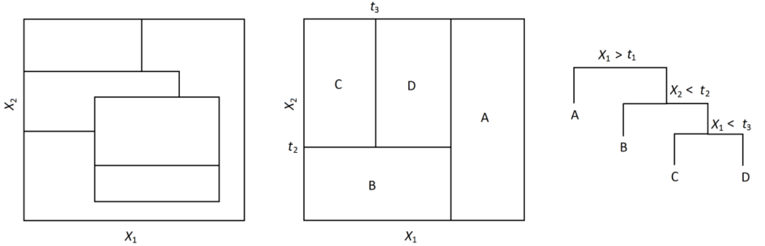
Different possible methods

- Incorporate Machine-Learning through modified models. An example : Causal Forest
- Free model approaches known as meta-learners : do not require to specify a ML method

CART (Breiman 1984)

- Target $\mathbb{E}[Y|X]$. Built recursively a tree by splitting the current cell into two children
- Find the feature j^* , the threshold z^* which minimises the loss $\mathcal{L}(j, z)$ where

$$\mathcal{L}(j, z) := \mathbb{E}\left[(Y - \mathbb{E}[Y|X_j \leq z])^2 \cdot 1_{X_j \leq z} + (Y - \mathbb{E}[Y|X_j > z])^2 \cdot 1_{X_j > z}\right]$$



CART (Breiman 1984)

- Pick a split to maximize the weighted difference $n_L n_R(\overline{Y}_L - \overline{Y}_R)^2$ with

$$\overline{Y}_L = \frac{1}{n_L}\sum_{i \in L} Y_i,\ \overline{Y}_R = \frac{1}{n_R}\sum_{i \in R} Y_i$$

- The tree tries to split such as the difference in average is as big as possible and the number of sample is important in each cell.
- Thereafter predict in L with $\frac{1}{n_L}\sum_{i \in L} Y_i$
- The idea is that when you find a localized part of the feature space when the target $\mathbb{E}[Y|X]$ is constant, estimate by an average of Y

Causal Tree (Athey and Imbens 2016)

- Target $\tau(x) = \mathbb{E}[Y_i(1) - Y_i(0)|Xi = x]$
- Similar idea operate via recursive partitioning: within each leaf, estimate treatment effect (not the mean).
- Split by maximizing $n_L n_R(\widehat{\tau}_L - \widehat{\tau}_R)^2$, with

$$\widehat{\tau}_L = \frac{1}{n_L^{(1)}} \sum_{i \in L, T_i = 1} Y_i - \frac{1}{n_L^{(0)}} \sum_{i \in L, T_i = 0} Y_i$$

and

$$\widehat{\tau}_R = \frac{1}{n_R^{(1)}} \sum_{i \in R, T_i = 1} Y_i - \frac{1}{n_R^{(0)}} \sum_{i \in R, T_i = 0} Y_i$$

- Predict in each leaf with the formula above

Causal Tree (Athey and Imbens 2016)

- The idea is that you find a localized part of the feature space where the treatment effect is constant and you estimate with a constant treatment effect estimator.
- Advantages: Interpretable $\widehat{\tau}(x)$, target CATE
- Drawbacks: justified in RCT (use difference in means), propensity score may vary within leaves

`Python implementation :`

`https://pypi.org/project/causal-tree-learn/`

Causal Forest (Wager and Athey, 2018), (Lechner, 2018)

- Random forests (Breiman, 2001) : prediction is an average of predictions made by individual trees.
- Athey, Wager (2018): an adaptive kernel method

$$\widehat{\mu}(x) = \sum \alpha_i(x) Y_i$$

where

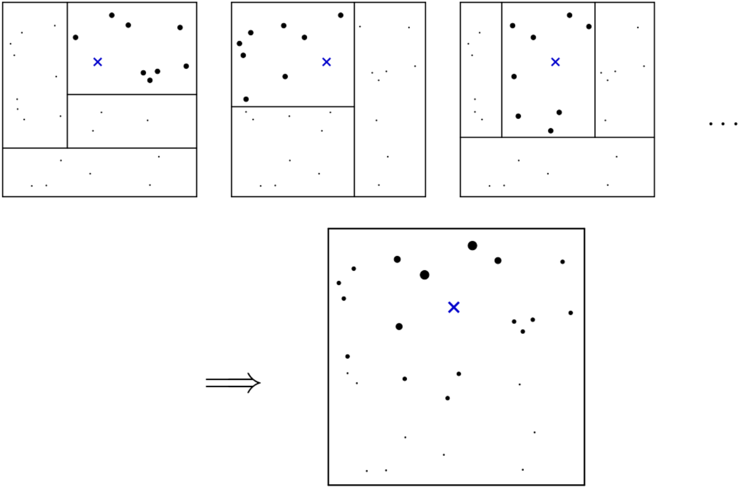
$$\alpha_i(x) = \frac{1}{B} \sum_b \frac{1_{\{X_i \in L_b, i \in b\}}}{|\{i : X_i \in L_b\}|}$$

with B the total number of trees in the forest, Lb (x) is the leaf where x falls into in tree b.
- An interesting library : `EconML`

Causal Forest (Wager and Athey, 2018), (Lechner, 2018)

- The weight $\alpha_i(x)$ can be seen as a data-adaptive kernel that measures how often the i-th training example falls in the same leaf as the test point x.
- This is a local average of all the Y_i corresponding to an X_i falling in the same leaf than x

Causal Forest (Wager and Athey, 2018), (Lechner, 2018)



Meta Learners

- Possible framework to tackle the estimation of the CATE ? Meta-learners
- Initially introduced and discussed by Künzel et al. [2019].
- Meta-learners derive consistent estimation of heterogneous treatment effects
- Valid in both Randomized Controlled Trials (RCT) and Observational studies.

Meta Learners

Definition

A Meta-learner is a statistical framework that models and estimates the CATE model such that

$$\tau(x) = \mathbb{E}[Y(1) - Y(0)|X = x]$$

- The advantage of meta-learners is that they do not require a specific Machine Learning method.
- They can support any supervised regression parametric or nonparametric method (e.g. random forest, gradient boosting methods).
- These methods are called base-learners when applied to a meta-learner

Meta Learners

All meta-learners fall in a taxonomy of CATE's estimators given by Knaus et al. [2020b], Curth and van der Schaar [2021b].

- direct plug-in (one step) meta-learners (T- and S-learners)
- pseudo-outcome (two-step) meta-learners (X-, M- and DR-learners)
- Neyman-Orthogonality based learners (R-learner)

Meta Learners

T-learner

- From the definition of CATE , the first meta-learner to be considered is the T-learner
- This meta-learner builds a CATE estimator using two models
  - Regress separately $Y(0)$ and $Y(1)$ on the covariates to build estimators $\widehat{\mu}_0$ (resp $\widehat{\mu}_1$) of $\mathbb{E}[Y(0)|X=x]$ (resp. $\mathbb{E}[Y(1)|X=x]$)
  - Estimate the CATE as the difference between these two estimators

Meta Learners

S-learner

- The second meta-learner to be defined is the S-learner where S refers to single
- Based on the identifiability of the counterfactual response, namely under suitable assumptions

$$\tau(x) = \mathbb{E}[Y_{obs}|T = 1, X = x] - \mathbb{E}[Y_{obs}|T = 0, X = x]$$

- herefore, one can take the treatment T as a feature similar to all the other covariates and build as follows :
  - Regress Y on the treatment T and the covariates X by a single model
  - Estimate the CATE as $\widehat{\tau}_S := \widehat{\mu}(x, 1) - \widehat{\mu}(x, 0)$

Meta Learners

- The T-Learner and the S-Learner may not produce the same result as the regression procedure is different for each learner.
- Using the propensity score e, we may define additional meta-learning algorithms whose objective is to estimate the CATE more efficiently.

Meta Learners

X-learner [Künzel et al., 2019]

- X : refers to the cross-learning approach of the algorithm ,
- Introduced to overcome the problem of unbalancing groups, .
- Let us consider the two random variables $D(1) := Y(1) - \mu_0(X)$ and $D(0) := \mu_1(X) - Y(0)$
- One has

$$\begin{aligned}
\mathbb{E}[D(1)|X = x) &= \mathbb{E}[Y(1) - \mu_0(X)|X = x] \\
&= \mathbb{E}[Y(1) - \mathbb{E}[Y(0)|X]|X = x] \\
&= \mathbb{E}[Y(1)|X = x] - \mathbb{E}[Y(0)|X] \\
&= \tau(x)
\end{aligned}$$

- Same for $D(0)$

Meta Learners

X-learner [Künzel et al., 2019]

The X-Learner can be built from the sample as follows

- Similarly to T-Learner, regress $Y(j)$ on the covariates X to build estimators $\widehat{\mu}_j$ of $\mu_j(x) = \mathbb{E}(Y(j)|X = x)$
- Estimate the missing potential outcomes $d_i^{(0)} := y_{obs,i} - \widehat{\mu}_0(x_j)$ on S_0 (resp $d_i^{(1)} := \widehat{\mu}_1(x_j) - y_{obs,i}$ on S_1)
- Regress $D(1)$ and $D(0)$ on the covariates X by two models $\widehat{\tau}_1$ and $\widehat{\tau}_0$ using the subsets $(x_i, d_i^{(j)})$
- Estimate the CATE by a weighted average function g (e.g. propensity score e) of the estimated models