Causal Machine Learning - Why it Works

Turing Masterclass

Oliver Dukes
Ghent University, Belgium

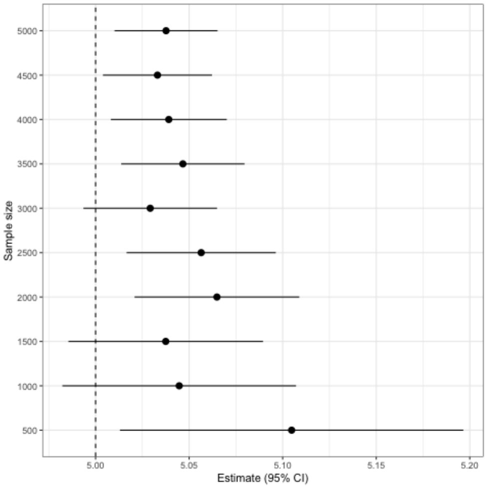
March 3, 2020

Plan

Bias

- This bias of an estimator ψ_n is defined as $E(\psi_n - \psi_0)$.
- Bias may not be problematic if it disappears as $n \to \infty$.
- However,
  for large-sample hypothesis tests and confidence intervals
  to have their correct size/coverage,
  we need bias$(\psi_n) \to 0$ sufficiently quickly as n increases.
- Otherwise, the confidence intervals well tend to shrink around the wrong value.

How fast does bias need to disappear? (1)



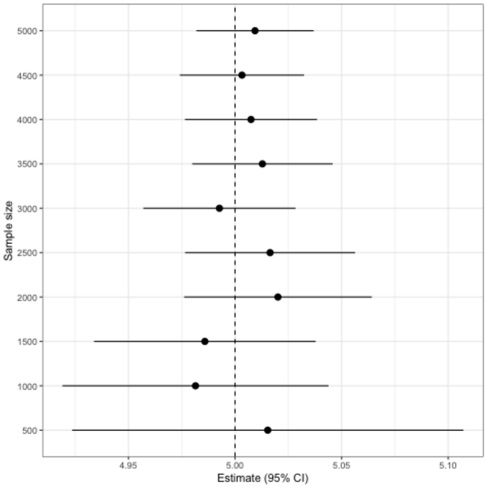
How fast does bias need to disappear? (2)

- For inference, we need the bias to be of lower magnitude than the standard error.
- Key condition: if

$$\sqrt{n}\text{bias}(\psi_n) = \sqrt{n}E(\psi_n - \psi_0) \to 0$$

  then we say that ψ_n has 'small bias'.
- In other words,
  $E(\psi_n - \psi_0)$ must shrink to zero faster than $\sqrt{n} \to \infty$.

How fast does bias need to disappear? (3)



Example: estimation of the counterfactual mean

- Let's consider a plug-in estimator of $E(Y^1)$:

$$\psi_n = \frac{1}{n}\sum_{i=1}^{n} \bar{Q}_n(W_i)$$

with $\bar{Q}_n(w)$ an estimator of $Q_0(w) = E(Y|A=1, W=w)$.

- Then for the bias, it is immediate that

$$\sqrt{n}\text{bias}(\psi_n) = \sqrt{n}E\{\bar{Q}_n(W_i) - Q_0(W_i)\}$$

- The bias of ψ_n is determined by the performance of the predictions.
- Throughout, I assume all machine learning estimators converge to the truth.

Plug-in bias: parametric methods

- Suppose the predictions are obtained via maximum likelihood estimation of a (correct) parametric model e.g.

$$\frac{1}{n}\sum_{i=1}^{n}\bar{Q}_n(W_i)=\frac{1}{n}\sum_{i=1}^{n}\text{expit}(\hat{\beta}'W_i).$$

- In that case,

$$\text{bias}(\psi_n)\propto\frac{1}{n}.$$

- Therefore $\sqrt{n}\text{bias}(\psi_n)\propto\sqrt{n}/n=1/\sqrt{n}$, which shrinks to zero as $n\rightarrow\infty$.
- The small bias condition is therefore satisfied....
- ....but this relies on a 'pre-specified' correct model.

Plug-in bias: machine learning

- The situation is very different for machine learning methods.
- The behavior of $\bar{Q}_n(W_i)$ - and therefore bias(ψ_n) - depends on tuning parameter(s).
- Standard choices of tuning parameters are designed to balance

$$\text{bias}^2\{\bar{Q}_n(w)\} \quad \text{and} \quad \text{var}\{\bar{Q}_n(w)\}$$

  to minimise mean-squared error of predictions.
- As we will see, this trade-off is sub-optimal for shrinking bias(ψ_n) to zero as quickly as possible.

Example: kernel regression (1)

- For a scalar continuous W, $\bar{Q}_n(w)$ is the Nadaraya-Watson estimator with bandwidth parameter h_n.
- One can show that

$$\text{bias}(\psi_n) \propto h_n^2 + \frac{1}{nh_n}.$$

- The 'optimal' choice of h_n for prediction purposes is $h \propto n^{-1/5}$, as then

$$\text{bias}^2\{\bar{Q}_n(w)\} \propto \frac{1}{n^{4/5}} \quad \text{and} \quad \text{var}\{\bar{Q}_n(w)\} \propto \frac{1}{n^{4/5}}.$$

Example: kernel regression (2)

- For this choice of h_n,

$$\text{bias}\{\bar{Q}_n(w)\} \propto \frac{1}{n^{2/5}}, \qquad \text{bias}(\psi_n) \propto \frac{1}{n^{2/5}}$$

  and

$$\sqrt{n}\text{bias}(\psi_n) \propto \frac{\sqrt{n}}{n^{2/5}} = n^{1/10} \to \infty.$$

  The small bias condition fails to hold.
- Our bandwidth choice made $\text{bias}\{\bar{Q}_n(w)\}$ too large..

Undersmoothing

- Is it possible to choose h_n to directly shrink $\text{bias}(\psi_n)$ instead?
- In theory, yes. This is known as undersmoothing.
- Let us choose a smaller bandwidth, say $h_n \propto n^{-1/3}$. Then

$$\text{bias}\{\bar{Q}_n(w)\} \propto \frac{1}{n^{2/3}}, \qquad \text{bias}(\psi_n) \propto \frac{1}{n^{2/3}}$$

  and

$$\sqrt{n}\text{bias}(\psi_n) \propto \frac{\sqrt{n}}{n^{2/3}} = \frac{1}{n^{1/6}} \to 0.$$

- Choosing a sensible bandwidth is difficult in practice.
- Undersmoothing not generally feasible with > 1 covariate.

Plan

Causal machine learning - bias

Remember the AIPW estimator of $E(Y^1)$ from earlier:

$$\psi_n = \frac{1}{n}\sum_{i=1}^{n}\frac{A_i}{g_n(W_i)}\{Y_i - \bar{Q}_n(W_i)\} + \bar{Q}_n(W_i).$$

Then one can show that

$$
\begin{aligned}
&\sqrt{n}E(\psi_n - \psi_0) \\
&= \sqrt{n}E\left[g_0(W_i)\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{\frac{1}{g_0(W_i)} - \frac{1}{g_n(W_i)}\right\}\right] \\
&+ \sqrt{n}E\left[\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{1 - \frac{A}{g_0(W_i)}\right\}\right] \\
&+ \sqrt{n}E\left[\left\{\frac{1}{g_n(W_i)} - \frac{1}{g_0(W_i)}\right\}A_i\{Y_i - Q_0(W_i)\}\right]
\end{aligned}
$$

Causal machine learning - plug-in bias

- The bias term

$$\sqrt{n}E\left[g_0(W_i)\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{\frac{1}{g_0(W_i)} - \frac{1}{g_n(W_i)}\right\}\right]$$

depends on the product of errors, which $\rightarrow 0$ as fast or faster than either error alone.

- Even if

$$\sqrt{n}E\{\bar{Q}_n(W_i) - Q_0(W_i)\} \quad \text{and} \quad \sqrt{n}E\left\{\frac{1}{g_0(W_i)} - \frac{1}{g_n(W_i)}\right\}$$

both diverge to infinity, it's still possible that $\sqrt{n}\text{bias}(\psi_n) \rightarrow 0$!

Small bias property - example

- Let both $g_n(W_i)$ and $\bar{Q}_n(W_i)$ be kernel regression estimators.
- The optimal prediction bandwidth $h_n \propto n^{-1/5}$ is chosen, so

$$\text{bias}\{\bar{Q}_n(w)\}\frac{1}{n^{2/5}} \quad \text{and} \quad \text{bias}\{g_n(w)\} \propto \frac{1}{n^{2/5}}.$$

- One can show that

$$\sqrt{n}E\left[g_0(W_i)\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{\frac{1}{g_0(W_i)} - \frac{1}{g_n(W_i)}\right\}\right]$$
$$\propto \sqrt{n} \times \frac{1}{n^{2/5}} \times \frac{1}{n^{2/5}} = \frac{1}{n^{3/10}} \to 0.$$

- Can trade fast convergence of $\bar{Q}_n(W_i)$ with slower convergence of $g_n(W_i)$ (and vice versa).

Overfitting bias (1)

- However, we are still left with two extra bias terms:

$$\sqrt{n}E\left[\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{1 - \frac{A}{g_0(W_i)}\right\}\right] \quad (1)$$
$$+\sqrt{n}E\left[\left\{\frac{1}{g_n(W_i)} - \frac{1}{g_0(W_i)}\right\}A_i\{Y_i - Q_0(W_i)\}\right] \quad (2)$$

- Let's focus on (1).
- Suppose another analyst trained the ML algorithm for learning $Q_0(W_i)$ on a separate dataset.

Overfitting bias (2)

- We can ignore the randomness in $\bar{Q}_n$, by pretending the secondary data is fixed.
- Then

$$
\begin{aligned}
&\sqrt{n}E\left[\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{1 - \frac{A_i}{g_0(W_i)}\right\}\right] \\
&= \sqrt{n}E\left[\{\bar{Q}_n(W_i) - Q_0(W_i)\}E\left\{1 - \frac{A_i}{g_0(W_i)}\middle| W_i\right\}\right] = 0
\end{aligned}
$$

- Trick does not work if $\bar{Q}_n$ estimated from the original sample; $1 - A_i/g_0(W_i)$ is correlated with $\bar{Q}_n(W_i)$.
- Bias terms (1) and (2) arise by learning Q_0 and g_0 on the estimation sample for ψ_0: overfitting!

Plan

Sample-splitting (1)

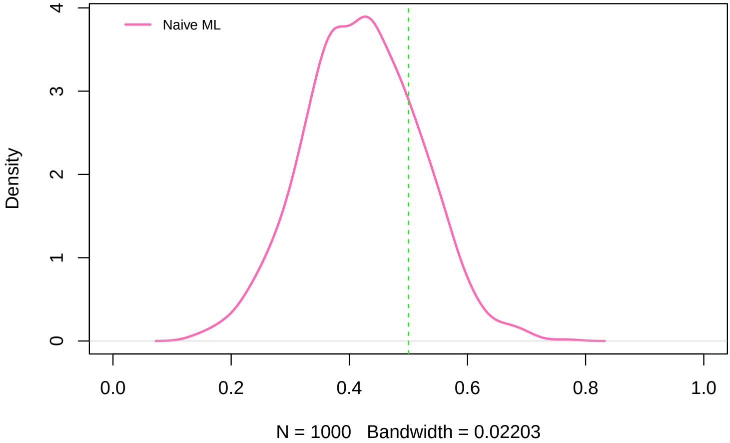
- This suggests that we can kill terms (1) and (2) via sample-splitting. (Bickel, 1982; Schick, 1986; van der Vaart, 1998)

Data (n)

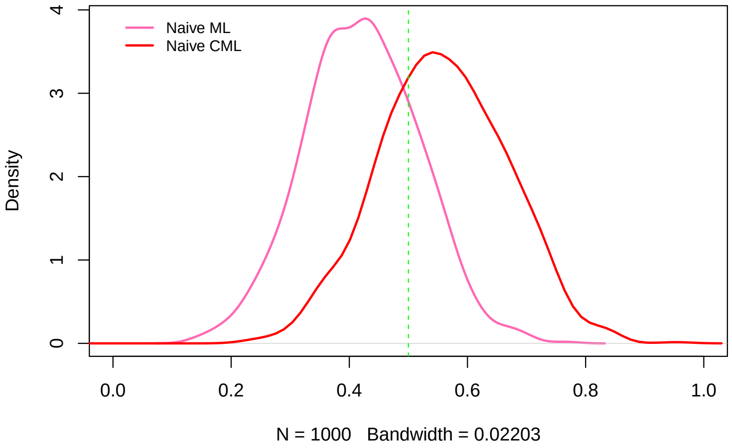
- **Training** ($\approx n/2$): Construct $\bar{Q}_n$ and g_n
- **Validation** ($\approx n/2$): Estimate ψ_n

- Let's see how this compares with a causal machine learning estimator without sample splitting....

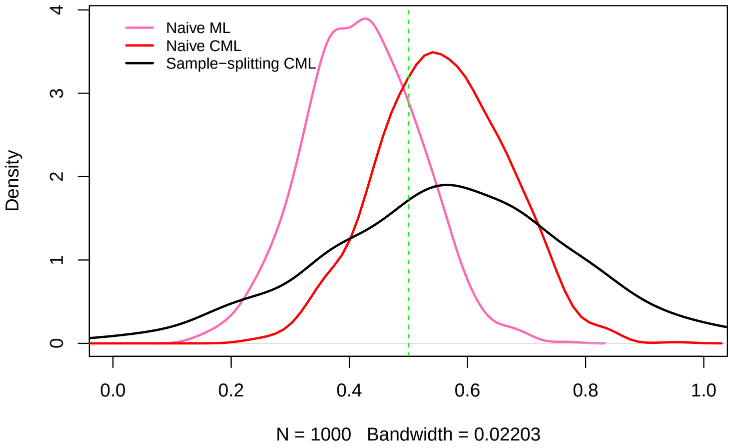
Sample-splitting (2)



Sample-splitting (3)

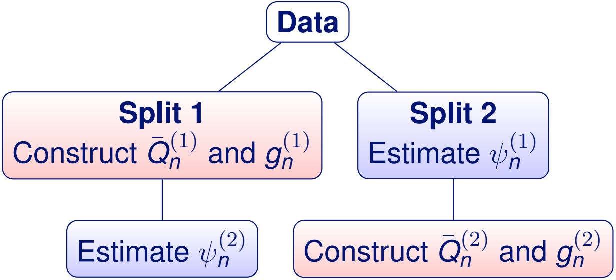


Sample-splitting (4)



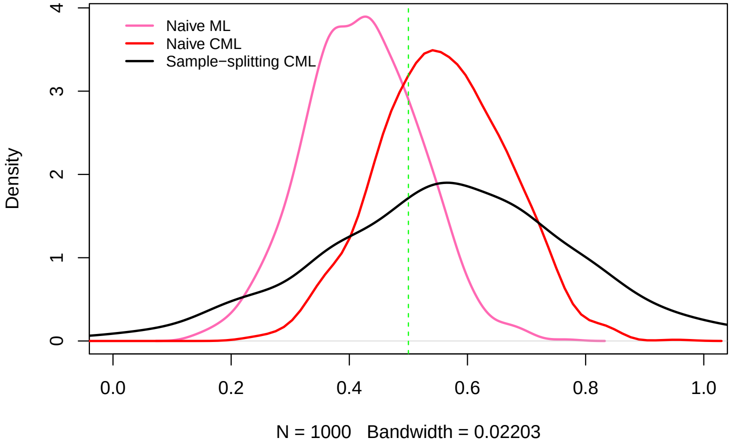
Cross-fitting (1)

- Naive sample splitting: big reduction in efficiency....
- Can be remedied through cross-validated TMLE/cross-fitting. (Zheng and van der Laan, 2011; Chernozhukov et al., 2018)

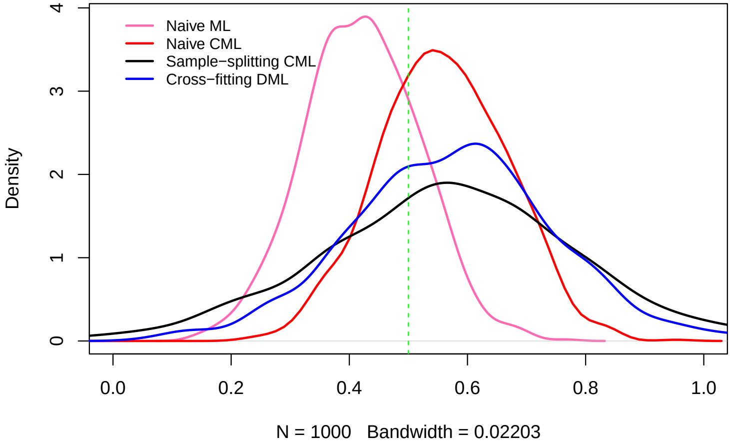


- We can then take the average of $\psi_n^{(1)}$ and $\psi_n^{(2)}$.
- Asymptotically, this regains full efficiency.

Cross-fitting (2)



Cross-fitting (3)



Implementation (1)

- R software for causal machine learning combined with splitting now available:
  - The ‘drtmle’ and ‘tmle3’ packages implement cross-validated TMLE.
  - ‘npcausal’ combines AIPW estimation with cross-fitting.

Implementation (2)

```
> n <- 1000
> w1 <- rbinom(n, size=1, prob=0.5)
> w2 <- rbinom(n, size=1, prob=0.65)
> w3 <- round(runif(n, min=0, max=4), digits=3)
> w4 <- round(runif(n, min=0, max=5), digits=3)
> A  <- rbinom(n, size=1,
   prob= plogis(-0.4 + 0.2*w2 + 0.15*w3 + 0.2*w4 + 0.15*w2*w4))
> Y  <- rbinom(n, size=1,
   prob= plogis(-1 + A -0.1*w1 + 0.3*w2 + 0.25*w3 + 0.2*w4
+ 0.15*w2*w4))
```

True effect size is 0.198.

(https://migariane.github.io/TMLE.nb.html)

Implementation (3)

```
> library(SuperLearner)
> library(earth)
> library(gam)
> library(ranger)
> library(rpart)
#Specify SuperLearner libraries
> SL.library <- c("SL.glm","SL.glm.interaction","SL.ranger")
#Data frame with baseline covariates
> W<- as.data.frame(cbind(w1,w2,w3,w4))
```

Implementation (4)

```
#AIPW with 2 splits
aipw_cf<-ate(y=Y, a=A, x=W, nsplits=2, sl.lib=SL.library)
#CV-TMLE WITH 2 splits
cvtmle<-drtmle(Y=Y,A=A,W=W,a_0 = c(1, 0),family = binomial(),
stratify = TRUE,
  SL_Q = SL.library,SL_g = SL.library,SL_Qr = "SL.glm",
SL_gr = "SL.glm",cvFolds = 2)

> aipw_cf$res$est[3]
[1] 0.1772749

> cvtmle$tmle$est[1]-cvtmle$tmle$est[2]
[1] 0.1723392
```

Plan

Recap so far

- The bias of naive plug-in estimators shrinks too slowly for statistical inference.
- For the causal machine learning estimators:
  - The plug-in bias shrinks faster than that of the “plugged-in" machine learning estimator(s).
  - The overfitting bias is removed via cross-fitting.
- For $E(Y^1)$, so long as

$$\sqrt{n}E\left[g_0(W_i)\{\bar{Q}_n(W_i) - Q_0(W_i)\}\left\{\frac{1}{g_0(W_i)} - \frac{1}{g_n(W_i)}\right\}\right] \to 0$$

we are in a good position to obtain tests/confidence intervals!

Statistical inference

- How do we calculate standard errors for ψ_n?
- In particular, how do we take into account the uncertainty in $\bar{Q}_n(W)$ and $g_n(W)$?
- It turns out that a consistent variance estimator is readily obtained
  as 1 over *n* times the sample variance of

$$\frac{I(A_i = 1)}{g_n(W_i)} \{ Y_i - \bar{Q}_n(W_i) \} + \bar{Q}_n(W_i),$$

despite the uncertainty in the data-adaptive estimators being unknown.

Why is this valid?

- In addition to removing overfitting bias, cross-fitting helps kill the contribution of $\bar{Q}_n(W)$ and $g_n(W)$ to the variance of ψ_n.
- Therefore ψ_n has the same asymptotic variance regardless of whether $Q_0(W)$ and $g_0(W)$ are estimated or known!
- When functional forms of $Q_0(W)$ and $g_0(W)$ are unknown, no (regular) estimator can have lower variance than this: $\rightarrow \psi_n$ is asymptotically efficient!

Using software

```
#AIPW 95% confidence interval
> aipw_cf$res$ci.ll[3]
[1] 0.1130162
> aipw_cf$res$ci.ul[3]
[1] 0.2415335

#CVTMLE 95% confidence interval
est.cvtmle<-cvtmle_5$tmle$est[1]-cvtmle$tmle$est[2]
var.cvtmle<-cvtmle$tmle$cov[1,1]+cvtmle$tmle$cov[2,2]
-2*cvtmle$tmle$cov[1,2]
se.cvtmle<-sqrt(var.cvtmle)
> est.cvtmle-1.96*se.cvtmle
[1] 0.1053619
> est.cvtmle+1.96*se.cvtmle
[1] 0.232179
```

A few caveats....

- It may be that that one/both machine learning estimators does not converge fast enough to eliminate the bias terms.
  - Standard inference is not possible, and one may to resort to higher-order influence functions/TMLE.

    (Robins et al., 2008; Carone et al., 2014)
- Cross-fitting may sometimes work worse than not splitting at all....
- When treated and untreated subjects have limited overlap, the variance of ψ_n may still be too large for meaningful causal inference.