EC 709: (Automated) Double/Debiased Machine Learning of Causal Effects

Liang Zhong [1]

Boston University

samzl@bu.edu

December 2023

[1]Reference: Christian Hansen's lecture note in Northwestern University Causal Inference Workshop

Overview

Table of Contents

Motivation

- ML/AI methods: dealing with big data, but are designed for forecasting

⇒ Highly adaptive because of the high-dimensional settings, causing:

1. Hard to know if overfitting is "sufficiently" controlled in high-dimensional settings with highly adaptive processes
   - Overfitting Bias ⇒ spurious associations
2. Pretesting Issue: inference as if you came with a model selection first step - e.g. Leeb and Potscher (2008)
   - doesn't come to the data with what is effectively a pre-specified low-dimensional model as in traditional parametric, non-parametric, and sieve approaches

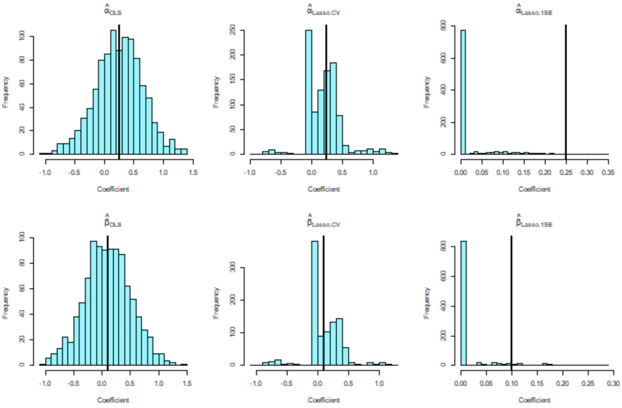
- Naive application may be highly misleading when conducting causal inference!

Toy Simulation Illustration

- Let's look at trying to learn parameters in a linear model with two regressors:
- $Y = \alpha D + \beta X + \epsilon$ with $n = 100$ observations
  - $D = \gamma X + \nu$
  - $X = \eta$
- $\alpha = 0.25, \beta = 0.1, \gamma = 1, \sigma_\epsilon = \sigma_\eta = 1, \sigma_\nu = 0.25$
- Should obviously just regress Y on (D, X)
- However, in practice we don't know model is linear and depends on only two variables
  - Suppose you did Lasso instead
  - Consider two cross-validated tuning parameter choices (CV-min and the so-called "1SE" rule)

Toy Simulation Illustration: Results

Based on 1000 simulation replications, we obtain



Highly non-standard distributions for Lasso estimators

Toy Simulation Illustration: Results

Based on 1000 simulation replications, we obtain

	$\widehat{\alpha}_{OLS}$	$\widehat{\beta}_{OLS}$	$\widehat{\alpha}_{Lasso:CV}$	$\widehat{\beta}_{Lasso:CV}$	$\widehat{\alpha}_{Lasso:1SE}$	$\widehat{\beta}_{1SE}$
Mean	0.250	0.100	0.211	0.126	0.027	0.018
Std. Dev.	0.401	0.409	0.282	0.283	0.059	0.047
Fraction 0	0.000	0.000	0.249	0.380	0.770	0.832

- Recall $\alpha = 0.25, \beta = 0.1$, Visible biases for both Lasso variants
- Could make things look much worse by adding more variables
- Don't want to make too much of this specific toy example but illustrates difficulties

Generalizable Point: Regularized/adaptive procedures make inference hard!

Table of Contents

Semiparametric Problem

- Consider inference about a target parameter α_0 in general semiparametric problem:
- target parameter is pre-specified, no p-hacking
  - not going to try to learn what we want to do inference about from the data
  - has a "scientific" question we are trying to answer with the data - not trying to find a question from the data
- **low-dimensional** target parameter with population value α_0: causal effect of some policy
- **high-dimensional** nuisance parameter with population value η_0: coefficients on other control variables
- Generally, α_0 is identified from moment condition:

$$E[\Phi(W, \alpha_0, \eta_0)] = 0$$

  - W is a random element; observe sample $\{W_i\}_{i=1}^n$ from distribution of W

Example: Partially Linear Model (PLM)

- $Y = D\alpha_0 + g_0(X) + \epsilon; E[\epsilon|D, X] = 0$
- $D = m_0(X) + U; E[U|X] = 0$
  - X are "confounders" - potentially related to both D and Y
- α_0 is the parameter of interest

  E.g. coefficient on $D = sex$ in a gender wage gap study

  E.g. coefficient on a policy variable D that is assumed exogenous after conditioning on X but not sure of functional form
- $g_0(X)$ is a nuisance function

  E.g. want to understand the partial correlation between sex and log(wage) in wage example after partially out "job-relevant" characteristics X

  E.g. $g_0(X) = \beta' X$ in the linear model

PLM Moment Conditions

- Denote $l_0(X) = E[Y|X]$, many moment conditions available to learn α_0 in the PLM:

1. $E[(Y - D\alpha_0 - g_0(X))D] = 0$

   $\Leftarrow$ Regress of $Y - \hat{g}(X)$ on D; use regularized estimator $\hat{g}(\cdot)$
   - Nuisance function $\eta_0 = g_0(X)$, analogous to "regression adjustment"

PLM Moment Conditions

- Denote $l_0(X) = E[Y|X]$, many moment conditions available to learn α_0 in the PLM:

1. $E[(Y - D\alpha_0 - g_0(X))D] = 0$

   $\Leftarrow$ Regress of $Y - \hat{g}(X)$ on D; use regularized estimator $\hat{g}(\cdot)$
   - Nuisance function $\eta_0 = g_0(X)$, analogous to "regression adjustment"

2. $E[(Y - D\alpha_0)(D - m_0(X))] = 0$

   $\Leftarrow$ IV regression of Y onto D using $D - \hat{m}(X)$ as instrument; use regularized estimator $\hat{m}(\cdot)$
   - Nuisance function $\eta_0 = m_0(X)$, analogous to "propensity score adjustment"

PLM Moment Conditions

- Denote $l_0(X) = E[Y|X]$, many moment conditions available to learn α_0 in the PLM:

1. $E[(Y - D\alpha_0 - g_0(X))D] = 0$

   $\Leftarrow$ Regress of $Y - \hat{g}(X)$ on D; use regularized estimator $\hat{g}(\cdot)$
   - Nuisance function $\eta_0 = g_0(X)$, analogous to "regression adjustment"

2. $E[(Y - D\alpha_0)(D - m_0(X))] = 0$

   $\Leftarrow$ IV regression of Y onto D using $D - \hat{m}(X)$ as instrument; use regularized estimator $\hat{m}(\cdot)$
   - Nuisance function $\eta_0 = m_0(X)$, analogous to "propensity score adjustment"

3. $E[((Y - l_0(X)) - (D - m_0(X))\alpha_0)(D - m_0(X))] = 0$

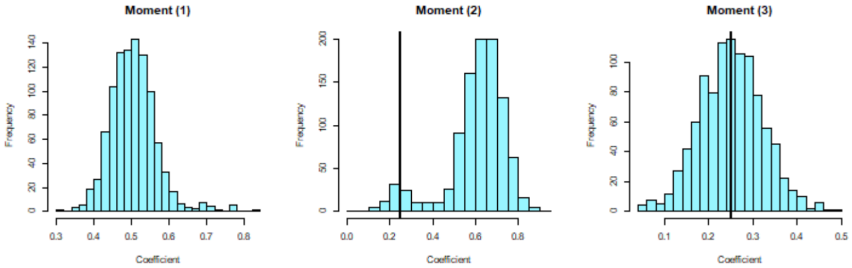
   $\Leftarrow$ Regress of $Y - \hat{l}_0(X)$ onto $D - \hat{m}(X)$, use regularized estimators $\hat{l}_0(\cdot)$ and $\hat{m}(\cdot)$
   - Nuisance function $\eta_0 = \{l_0(X), m_0(X)\}$, analogous to "double-robust" estimator

HDLM: Simulation Illustration

- Suppose the model is known to be linear: $g_0(X) = \beta' X$
- If $p < n$ (low-dimensional setting) $\Rightarrow$ No need to use regularization methods $\Rightarrow$ Three moments would produce identical estimators of α_0
- If $p \geq n$ (high-dimensional setting) $\Rightarrow$ α_0 is not identified without regularization $\Rightarrow$ Three moments behave differently:
- Consider $n = 200$ observations and $p = dim(X) = 200$ "controls"
    - $\alpha = 0.25$, $(X, \epsilon, U) \sim N(0, I_{p+2})$ i.i.d.
    - $g_0(X) = \beta' X$, $\beta = (1, 0.5, 0, ..., 0)$
    - $m_0(X) = \gamma' X$, $\gamma = (0.5, 1, 0, ..., 0)$
    - Don't know only the first two variables matter
- Again use Lasso for regularized estimators

HDLM: Simulation Illustration Results

Based on 1000 simulation replications, we obtain



- Recall $\alpha = 0.25$: Huge bias for results based on the first two moment conditions
- What's special about the third moment?

Orthogonal Estimating Equations

Definition 1 (Neyman Orthogonality)

A moment condition for identifying α_0 in the presence of nuisance functions with true values η_0 is **Neyman orthogonal** if it satisfies $\partial_\eta E[\Phi(W, \alpha_0, \eta)]|_{\eta=\eta_0} = 0$, where ∂_η is the Gateaux derivative operator with respect to η

- intuitively - captures notion that moment condition is not violated by small perturbations of the nuisance functions around their true values
- don't have true values of nuisance parameters in real data
- allows for selection/estimation mistakes in learning nuisance parameters
- The key difference between moment condition (3) and moment conditions (1)-(2) is that (3) satisfies the orthogonality property

▸ Formal Proof from Christian Hansen

Comments on Neyman orthogonality

- Take another look at $E[((Y - l_0(X)) - (D - m_0(X))\alpha_0)(D - m_0(X))] = 0$:
  - Neyman orthogonality: Similar idea as FWL partialling out

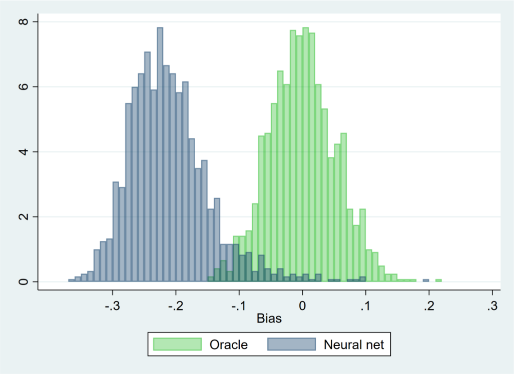
  $\Rightarrow$ eliminates the first order biases arising from the replacement of with a ML estimator
- Still require High-Quality Machine Learning Estimators
  - The nuisance parameters are estimated with high-quality (fast-enough converging) machine learning methods.
  - If the estimators are biased, would still lead to unreliable inference
- Now, has Neyman orthogonality solves all the problem in practice?

A More Complex Simulation

- $Y = D\alpha_0 + g_0(X) + \epsilon; D = m_0(X) + U$
- $(\epsilon, u) \sim N(0,1) i.i.d.$ and $(\epsilon, u) \perp X$
- $X \sim N(0, S_X)$ with $[S_X]_{i,j} = 0.5^{|i-j|}$
- Consider $n = 1000$ observations and $p = dim(X) = 50$ "controls"
- $\alpha_0 = 0.5$, $g_0(X) = m_0(X) = 1(X_1 > 0.3)1(X_2 > 0)1(X_3 > -1)$
- Nuisance functions estimated using a fully connected DNN with 2 hidden layers of 20 neurons each
- Use Neyman-orthogonal moment function

Overfitting: Simulation Illustration Results

Based on 1000 simulation replications, we obtain



- Using orthogonal moment, but large bias $\leftarrow$ Overfitting Bias
- How to handle it in practice?

Sample-Splitting - aka Cross-fitting

- Starting from Neyman-orthogonal moment condition for identifying α_0:

$$E[\Phi(W, \alpha_0, \eta_0)] = 0$$

- Goal: Keep estimation of nuisance functions "independent" of data used to estimate α_0
- To avoid the biases arising from overfitting, a form of sample splitting is used at the stage of producing the estimator of the main parameter
  - Use only part of the sample for estimation to avoid over fitting
  - However, naively drop your observations would seriously affect your efficiency

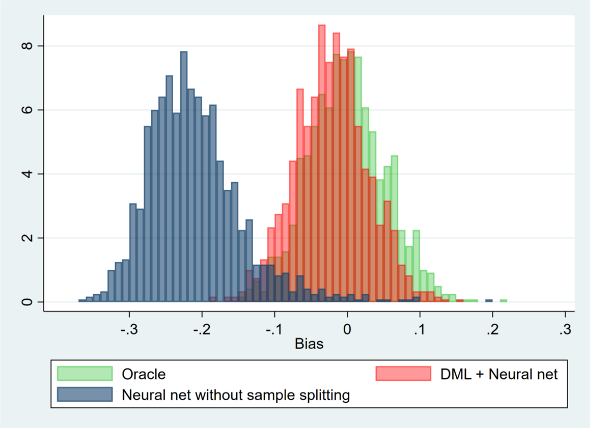
Inference Algorithm

1. Take a K-fold partition $(I_k)_{k=1}^{K}$ of observation indices $[n] = 1, ..., n$ such that the size of each fold I_k is (approximately) $N = n/K$
2. For each $k \in [K] = 1, ..., K$ construct an estimator $\hat{\eta}_k$, where $x \to \hat{\eta}_k$ depends only on the subset of data $(W_i)_{i \in I_k}$
3. Obtain estimate of the parameter of interest, $\hat{\alpha}$ as solution to $\frac{1}{n} \sum_{k=1}^{K} \sum_{i \in I_k} \Phi(W_i; \hat{\alpha}, \hat{\eta}_k(W_i)) = 0$
4. Obtain standard error in the usual way ignoring estimation of $\hat{\alpha}$

- Efficiency gains by using cross-fitting (swapping roles of samples for train / hold-out)
- Rerun the simulation above using cross-fitting with 5 folds

Overfitting: Simulation Illustration Results

Based on 1000 simulation replications, we obtain



Cross-fit results look much more palatable than full-sample results

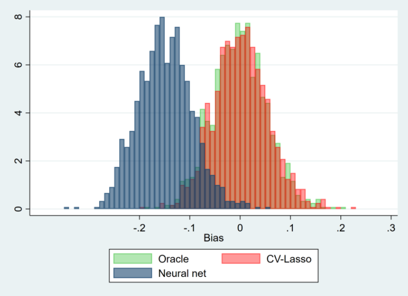
Practical issues of DML

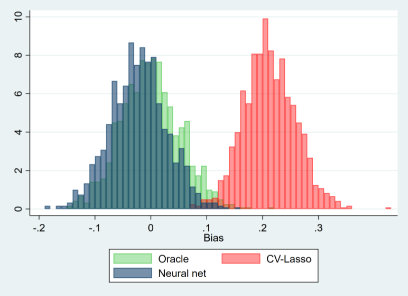
- Estimation involving Neyman orthogonal scores and cross-fitting is termed DML (Double/De-biased ML).
  - provides asymptotically normal inference under reasonably general conditions
  - formal results can be found in Chernozhukov et al. (2018)
  - Still a lot of uncertainty in practice

1. How to choose the ML model?

   $\Leftarrow$ Rarely know the right specification/model/learner
   - No learner performs best across all instances

Choice of learner matters





- Left: (true) linear model estimated with DML using lasso (CV) or neural net
- Right: (true) nonlinear model estimated with DML using lasso (CV) or neural net

⇒ Try several learners and using some form of stacking; e.g. van der Laan and Rose (2011), Ahrens et al. (2022)

Practical issues (cont.)

2. Randomization from the sample splits might impact the results
   - Set seeds
   - Try multiple sample splits and look at sensitivity of results
   - Report mean or median of the estimate across different sample splittings

3. Avoid the amount of flexibility using theoretical intuitions

   E.g. Might know demand is monotonic
   - Reduce model complexity and increase interpretability of your results

   E.g. Avoid propensity score estimates outside of $[0, 1]$

Some ML Packages

- About ML implementations:
  1. Scikit-learn - Pedregosa et al. (2011): nice Python library of ML tools
  2. pdslasso - Ahrens et al. (2018): inference after lasso in Stata
  3. pystacked - Ahrens et al. (2022): many ML tools in Stata, including stacking
  4. ddml - Ahrens et al. (2023): DML in Stata for canonical parameters
- About DML and its examples:
  1. DoubleML.org (R and Python)
  2. DDML (Stata)
  3. EconML (Python)
- Please help me complete the course evaluation; any feedback is highly appreciated!

Thank You!

Supplementary Slides

Formally verifying Neyman orthogonality in PLM

- Let $\varepsilon = (Y - \ell_0(X)) - (D - m_0(X))\alpha_0)$
- For any $\eta = (m, \ell)$ that are square integrable, the Gateaux derivative in the direction

$$\Delta = \eta - \eta_0 = (m - m_0, \ell - \ell_0)$$

is

$$\begin{aligned}
&\partial_\eta \mathrm{E}\psi(W; \alpha_0, \eta_0)[\Delta] \\
&\quad = -\mathrm{E}\Big[\varepsilon(m(X) - m_0(X))\Big] \\
&\quad\quad + \mathrm{E}\Big[\Big((m(X) - m_0(X))\alpha_0 - (\ell(X) - \ell_0(X))\Big)(D - m_0(X))\Big] \\
&\quad = 0
\end{aligned}$$

  - follows from law of iterated expectations since $\mathrm{E}[D - m_0(X)|X] = 0$ and $\mathrm{E}[\varepsilon|D, X] = 0$

Coefficient estimator in PLM

Define

- $r_i^D = \widehat{m}(X_i) - m_0(X_i)$ and $r_i^Y = \widehat{\ell}(X_i) - \ell_0(X_i)$
- $\tilde{D}_i = D_i - \widehat{m}(X_i) = U_i - r_i^D$
- $\tilde{Y}_i = Y_i - \widehat{\ell}(X_i) = \varepsilon_i + \alpha_0 \tilde{D}_i + \alpha_0 r_i^D - r_i^Y$

In PLM, estimator of α_0 from (3.3) is

$$\widehat{\alpha} = \frac{\frac{1}{n}\sum_{i=1}^n \tilde{D}_i \tilde{Y}_i}{\frac{1}{n}\sum_{i=1}^n \tilde{D}_i^2}$$

which yields expansion

$$\begin{aligned}
\sqrt{n}(\widehat{\alpha} - \alpha_0) &= \frac{\frac{1}{\sqrt{n}}\sum_i U_i \varepsilon_i}{\frac{1}{n}\sum_i U_i^2} && (4.5)\\
&+ \frac{1}{\frac{1}{n}\sum_i U_i^2}\left(\alpha_0 \frac{1}{\sqrt{n}}\sum_i U_i r_i^D - \frac{1}{\sqrt{n}}\sum_i U_i r_i^Y - \frac{1}{\sqrt{n}}\sum_i \varepsilon_i r_i^D\right) && (4.6)\\
&+ \frac{1}{\frac{1}{n}\sum_i U_i^2}\left(-\alpha_0 \frac{1}{\sqrt{n}}\sum_i (r_i^D)^2 + \frac{1}{\sqrt{n}}\sum_i r_i^D r_i^Y\right) && (4.7)\\
&+ \text{higher order terms} && (4.8)
\end{aligned}$$

Expansion and Neyman Orthogonality

Expansion in (4.5)-(4.8)

- (4.5): $\frac{\frac{1}{\sqrt{n}}\sum_i U_i \varepsilon_i}{\frac{1}{n}\sum_i U_i^2}$
  - the usual term that leads to asymptotic normality
- (4.6): $\frac{1}{\frac{1}{n}\sum_i U_i^2}\left(\alpha_0 \frac{1}{\sqrt{n}}\sum_i U_i r_i^D - \frac{1}{\sqrt{n}}\sum_i U_i r_i^Y - \frac{1}{\sqrt{n}}\sum_i \varepsilon_i r_i^D\right)$
  - first order terms in expansion
  - compare (4.6) to the derivative on slide (15)
  - trivially vanish asymptotically if
    1. estimation errors r_i^D and r_i^Y **are independent** of model errors U_i, ε_i
    2. $\widehat{m}$ and $\widehat{\ell}$ are consistent
  - otherwise, need technical work showing tight control of estimation errors

Expansion and Neyman Orthogonality

Expansion in (4.5)-(4.8) (cont)

- (4.7): $\frac{1}{\frac{1}{n}\sum_i U_i^2}\left(-\alpha_0 \frac{1}{\sqrt{n}}\sum_i (r_i^D)^2 + \frac{1}{\sqrt{n}}\sum_i r_i^D r_i^Y\right)$
  - $\frac{1}{\sqrt{n}}$ normalized sums of **non-mean-zero** quantities
  - approximately bounded by $\sqrt{n}n^{-2\varphi}$ where φ is an appropriate bound on convergence rates of estimators for $m_0(X)$ and $\ell_0(X)$
  - in high-dimensional/nonparametric settings $\sqrt{n}n^{-\varphi}$ will diverge because of slower than parametric convergence of high-dimensional/nonparametric estimators but can still have $\sqrt{n}n^{-2\varphi} \to 0$

Generalizable takeaway: Neyman orthogonality leads to asymptotic expansions where first order terms vanish so estimation errors in nuisance objects show up in products that can vanish even when scaled by $\sqrt{n}$.

- without Neyman orthogonality, nuisance function estimation errors show up at first order (as terms that behave like $\sqrt{n}n^{-\varphi}$ after normalization) = poor behavior of estimators

Is Neyman Orthogonality enough?

There's an important point "hidden" in the derivation:

Terms in (4.6) trivially vanish **if estimation errors r_i^D and r_i^Y are independent of model errors U_i, ε_i**

BUT, r_i^D and r_i^Y depend on **all** the U_j and ε_j for the observations used to estimate m_0 and ℓ_0

- in general, independence does not hold
- overfitting in particular is a problem as it means the estimated models are specialized to the (non-generalizable) features of the data - i.e. strongly related to the U and ε in our PLM example

Back